The use of image analysis techniques to characterise mid-ocean ridges from multibeam and sidescan sonar data.

Jane A. Keeton

The Use of Image Analysis Techniques to Characterise Mid-Ocean Ridges from Multibeam and Sidescan Sonar Data

A thesis presented for the degree of

Doctor of Philosophy

by

Jane A. Keeton

University of Durham
Department of Geological Sciences

February 1994

ABSTRACT

Two types of sonar data have been analyzed from the Mid-Atlantic Ridge: multibeam echosounder and high resolution sidescan sonar. Simrad EM12 multibeam echosounder surveys were conducted over the 45°N area of the Mid-Atlantic Ridge, and TOBI high resolution sidescan sonar data were collected on the Reykjanes Ridge and between 24-30°N on the Mid-Atlantic Ridge.

The Simrad EM12 multibeam echosounder collects co-registered backscatter and bathymetry data, and processing techniques have been developed for the backscatter data, culminating in a backscatter mosaic displayed in a number of combinations with the bathymetry data. This processing enabled a quantitative analysis of the results, based on acoustical principles. This revealed the EM12's capabilities of resolving backscatter variations according to seafloor type, both in terms of backscatter levels and dependencies on incidence angles. Backscatter strength relationships with seafloor slope distributions and incidence angles have been explained in terms of geological processes. The normalisation of the backscatter data yielded a tentative classification of seafloor lithologies.

TOBI high resolution sidescan sonar images of some axial zones of the Mid-Atlantic Ridge revealed complex bottom type variations, often more in terms of morphological characteristics rather than material differences. An investigation of texture analysis techniques led to the successful discrimination of these bottom types using a procedure based on fractal principles, thus providing a very useful mapping tool. TOBI imagery were also used for the application of the Hough transform for detecting seamounts, the distributions of which can provide important geological indicators. The technique was successful within small windows of data, but was deduced to have limited future prospects for this type of data.

ACKNOWLEDGEMENTS

Initially, I wish to thank Prof. Roger Searle for supervising the project for the last three years, and reviewing drafts of this thesis. I also thank Neil Mitchell (Durham), Laurie Linnett, Stuart Clarke (Heriot-Watt), Debbie Smith (WHOI), Erik Hammerstad (Simrad), Tim Le Bas (IOS), Geoff Wadge and David Mason (NUTIS) for helpful discussions and/or provision of software.

Funding for this research was provided by the Natural Environment Research Council (CASE studentship with Worldwide Ocean Surveying Ltd.).

Crucial to the project was my participation in four research cruises, so I acknowledge the principal scientists: Lindsay Parson, Joe Cann and Debbie Smith for inviting me along on the Ewing and Darwin, respectively; and Roger, Joe and Mary Fowler (Royal Holloway, University of London) for commissioning the surveys on the Ocean Surveyor. Eva Valsami (University of Leeds) is acknowledged for her geological analysis of the EM12 bathymetry data. Worldwide Ocean Surveying Ltd. are thanked for their collaboration in this work. I should also like to personally thank a few friends and colleagues I've been at sea with: Cherry Walker, Paul Field, Eddie McAllister, Sara Spencer, Lisa Redbourne, Chris MacLeod, and my seasick pills!

I was able to spend a month working at the Ocean Mapping Development Center, University of Rhode Island, for which I thank Bob Tyce, Scott Ferguson and Joyce Miller.

In Durham, Dave Stevenson is thanked for sorting out my many computing problems, and Carole, Lynne, Karen, Neville and Dave Asbery for helping out aswell. Cheers to my office chums over the years - Ruth, Miles, Aiyad, Roberto and Nilpf; house mates - Charlotte, Zoe, Kate and Sarah; the neighbours - Andy, Sue and Jon; and the rest of the bunch.

I wish to mention Dr. Cath Allen who taught me some of my first geophysics during my undergraduate years at the University of Lancaster, and enthusiastically encouraged me to do a PhD. Cath sadly died a few months after I left Lancaster, and I should like to remember her here for all the support she gave me.

Many thanks to Ivan for....well being Ivan really, and without whom I'd never have discovered the joys of drag racing and conodont palaeobiology!!

Finally, I'd like to express my sincere thanks to my parents - Moira and Roy Keeton, my sister Emma and my grandparents. They have supported and encouraged me through my long years of education and I dedicate this thesis to all of them.

DECLARATION

I declare that the work contained in this thesis has not been submitted elsewhere for any other degree or qualification, and is my own work, unless otherwise referenced.

Jane A. Keeton

University of Durham

February 1994

LIST OF CONTENTS

LIST OF FIGURES

CHAPTER 2: PREVIOUS WORK

CHAPTER 3: INSTRUMENTATION

CHAPTER 4: PROCESSING OF EM12 SONAR DATA FROM THE MID-ATLANTIC RIDGE (45°30'N)

CHAPTER 5: EM12 BACKSCATTER AND BATHYMETRY RELATIONSHIPS

CHAPTER 6: TEXTURE ANALYSIS OF TOBI DATA

CHAPTER 7 THE DETECTION OF SEAMOUNTS USING THE HOUGH TRANSFORM

LIST OF TABLES

LIST OF APPENDICES

CHAPTER 1
INTRODUCTION

The theme of this thesis is to examine quantitative methods for studying mid-ocean ridge sonar data. The two types of data available are Simrad EM12 multibeam backscatter and bathymetry, and TOBI deep-towed sidescan sonar; collected from the Reykjanes Ridge, 45°N area and Kane-Atlantis (24-30°N) section of the Mid-Atlantic Ridge. The quantitative analysis of these types of data can encompass many fields; broadly acoustics, statistics and object recognition.

1.1 BACKGROUND

Major advances in deep ocean mapping instrumentation have occurred in recent years. Both multibeam echosounders and sidescan sonars have been around for about 20 years, though within the last five years or so, we have seen the advent of multibeam echosounders capable of simultaneously collecting acoustic backscatter data (e.g. EM12), and a deep-towed high-precision sidescan sonar instrument (TOBI). Both these types of instruments provide important new information about the ocean floor.

Data are logged by such instruments in a relatively raw format, which then require extensive processing in order to produce images and maps of the seafloor. EM12 bathymetry and TOBI sidescan sonar data are routinely processed to a usable level, but a processing suite was required for the EM12 sidescan dataset.

In addition to traditional sidescan image displays and bathymetry contour maps, the development of visualisation techniques can greatly aid geological interpretations. Also, the relatively recent advent of geographical information systems (GIS) in many fields (such as both physical and human geography), and the fact that a number of geophysical datasets are usually collected over any one site of interest on the ocean floor, has led me to apply a commercial land-based software system for marine purposes. EM12 datasets and associated vector information are combined into such a system.

The quantitative analysis of any dataset is clearly superior to purely qualitative approaches. This can be difficult in the complicated field of geology, but since the marine datasets are now recorded digitally, quantitative characterisations provide important complementary information to traditional qualitative interpretations. Qualitative analyses are not necessarily repeatable as they can be dependent upon a particular geologist's interpretation; additionally, surveys conducted by different instruments, at different times and/or over different areas must be suitably related to each other.

Remote sensing of the seafloor and the subsequent classification of bottom materials and morphologies are much researched subjects by geologists, geophysicists, mathematicians, engineers, image processors and acousticians alike. The broad field of quantitative seafloor characterisation, using both bathymetric and backscatter types of information, is an important way to quantify sonar data of the seafloor, and thus far comprises statistical studies of bathymetry and sidescan data separately, and acoustical studies of combined datasets.

The data available for the research in this thesis are two TOBI high resolution sidescan sonar datasets from the Reykjanes Ridge and Kane-Atlantis section of the MAR; the former supported by Hydrosweep bathymetry and the latter by Sea Beam (collected on separate cruises), dredging and photography. An EM12 multibeam bathymetry and backscatter survey was conducted over the 45°N area of the MAR.

1.2 OBJECTIVES

The objectives of this thesis are to explore and develop processing techniques to enhance the visualisation and interpretation of sonar data; in particular to:

- investigate visualisation techniques and GIS data management principles for sonar data
- examine the EM12 bathymetry and backscatter data quantitatively (acoustical study)
- assess various texture analysis techniques for characterising TOBI data (statistical study)

- appraise usefulness of automatic detection of seamounts using the Hough transform (object recognition)

- develop processing procedures for data (primarily EM12 backscatter data)

1.3 GEOLOGICAL BACKGROUND

Mid-ocean ridges are the primary site for production of the Earth's crust. They encompass lengths of some 60 000 km along divergent plate boundaries over the ocean floors, forming some of the most geologically active regions of the planet. Various ridges possess differing characteristics, and studies of specific areas of the Mid-Atlantic Ridge (MAR) are currently subjects of considerable scientific interest and research.

The 45°N area of the MAR was one of the first areas to be surveyed in detail using various techniques. In fact, one of the first depth measuring traverses of the MAR occurred over the Gog and Magog volcanoes blocking the median valley at about 45°45'N, and it was thought at this stage that no rift valley existed; a feature which is in fact a dominant characteristic of slow-spreading ridges. The rocks in the 45°N area possess unusual geochemical characteristics suggesting a plume environment, though there is no further present-day evidence for such a feature. The rifting characteristics are very interesting in this area, and deep crustal seismic studies have indicated the lack of a magma chamber.

The MAR interacts with a mantle plume to form Iceland, and the Reykjanes Ridge is the submarine extension of these effects. As the MAR approaches Iceland, the ridge re-orientates to spread obliquely, and the rift valley transforms into an axial high. This axial rise type topography is not caused by an increase in spreading rate, though it resembles fast-spreading ridge type morphology, but is an effect caused by the nearby influence of the Icelandic plume. It is therefore an important area to study in terms of the causes of axial rift/rise topography, its oblique spreading nature, and other associated plume effects.

The Kane-Atlantis section of the MAR ridge is an 800 km long classic section of normal slow-spreading ridge and has been the site for detailed studies over the last few years, with research currently focussing in on detailed, high resolution

investigations at the segment scale. These studies provide a benchmark for all research work on slow-spreading and indeed all mid-ocean ridges.

The sonar data studied in the thesis were gathered from all three sites described above. Their analysis, adopting quantitative methods, should therefore yield important information to add to the geological knowledge of the areas.

1.4 DATA COLLECTION

1.4.1 Simrad EM12 bathymetry cruises

These are described in detail in chapter 4, section 4.4; a brief summary will be presented here. I participated in two short bathymetry surveys in December 1990 and March 1993, aboard the R.V. Ocean Surveyor, operated by the commercial company, Worldwide Ocean Surveying Ltd. (WOSL). These surveys encompassed a total of 4500 km^2 on the axis of the Mid-Atlantic Ridge (MAR) centred at 45°30'N; this area is described in chapter 4. The Simrad EM12 multibeam echosounder also records co-registered backscatter data, which shall be investigated in detail in this thesis. Prof. R.C. Searle (University of Durham, U.K.) and Prof. J.R. Cann (University of Leeds, U.K.) purchased the December 1990 data, while Prof. J.R. Cann and Dr. C.M.R. Fowler (Royal Holloway and Bedford New College, University of London, U.K.) commissioned the second survey. No other work has been formally published on these data at the time of writing.

1.4.2 TOBI high resolution sidescan sonar cruises

The first of these cruises took place in December 1990, aboard the R.V. Maurice Ewing, owned by Lamont Doherty Geological Observatory, which is a subsidiary of Columbia University, New York, U.S.A. (cruise reference - EW9008; P.I.s - Dr. L.M. Parson (IOSDL) & Prof. R.C. Searle). The survey investigated the structure of the Reykjanes Ridge, collecting 2000 km^2 of TOBI sidescan sonar data, and over 35 000 km^2 of Hydrosweep bathymetry, plus underway gravity and magnetics. Parson *et al.* (1993) and Murton and Parson (1993) present some of the preliminary results from this research cruise. Only a small amount of these data were analyzed in this thesis for the seamount detection technique trials in chapter 7.

The R.R.S. Charles Darwin CD65 cruise to the MAR (24-30°N) was led by Prof. J.R. Cann and Dr. D.K. Smith (Woods Hole Oceanographic Institute) in February 1992. Following earlier work by Smith and Cann (1990, 1992) analyzing the

seamount distribution on the MAR, we went back to conduct a comprehensive high resolution sidescan survey using TOBI over 8 out of the 18 spreading segments of the ridge between the Kane and Atlantis fracture zones (spreading at 12-14 mm per year). 14 out of the 23 science days comprised of TOBI surveying, while the rest of the time consisted of detailed rock dredging and camera work based on the TOBI observations. Cann *et al.* (1992) present a comprehensive cruise report, and Smith and Cann (1993) are the first to publish some of the data.

1.5 STRUCTURE OF THESIS

This thesis is divided into eight chapters. The first three present background information, four describe the major research topics, and a final one concludes and discusses the results of the research.

Chapter 2 is entitled 'Previous Work', and from the outline in section 1.1, this clearly comprises many fields. Chapter 2 is divided into two main sections: seafloor backscatter characterisation methods (section 2.1) and Mid-Atlantic Ridge geology (section 2.2).

Chapter 3 follows with a description of the sonar instrumentation used for surveying the deep ocean floor, beginning with a general overview of multibeam, sidescan and the newer hybrid systems (section 3.2), and then specifications of the EM12 multibeam echosounder (3.3) and TOBI high resolution sidescan sonar (3.4), which supplied the data for this thesis.

The acquisition, processing and visualisation of the EM12 backscatter and bathymetry data is covered in chapter 4. A brief resumé of the 45°N area is presented in section 4.2, an account of previous sonar processing work is given in section 4.3, and section 4.4 describes the data acquisition details. The necessary depth and navigation processing techniques are outlined in section 4.5, while the sidescan processing, including mosaicking and various image restoration techniques, is detailed in section 4.6. Section 4.7 describes how the EM12 bathymetry and backscatter data are incorporated into a GIS, section 4.8 provides an investigation into advanced visualisation techniques, while section 4.9 presents the results, together with a comparison of the EM12 backscatter with a GLORIA dataset of the same area.

Chapter 5 comprises what could be broadly defined as an acoustical study. After an assessment of previous research in the field (section 5.2) and definitions of the nature of EM12 backscatter measurements (section 5.3), an analytical method is introduced in section 5.4. This assesses both the backscatter values themselves and their angular responses over a selection of experimental sites. The analytical results are presented in section 5.5. A tentative attempt to classify single bottom materials is accomplished in section 5.6, by normalising backscatter measurements according to the angular response curve of fresh volcanics. Profiles are also constructed for specific deductions. Section 5.7 relates the results to those from other studies, followed by discussions (5.8) and conclusions (5.9).

A statistical study is applied to TOBI sidescan datasets from the MAR, where backscatter intensities were not converted into backscatter strengths and a similar resolution co-registered dataset was not available. Two published methods that had previously been applied to quite different types of sidescan data are the Grey Level Co-Occurrence Matrices (GLCM) and fractal methods. Section 6.2 introduces the data, section 6.3 describes standard classification algorithms, and section 6.4 provides an analytical outline. The GLCM method assessment is detailed in section 6.5 and the fractal method is thoroughly investigated in section 6.6, yielding useful geological results. Section 6.7 compares the effectiveness of the two methods, and conclusions are drawn in section 6.8.

An appraisal of the usefulness of the Hough transform technique for automatically detecting seamounts is covered in chapter 7. This forms a possible object recognition methodology for characterising the sidescan data. Section 7.1 describes why the statistical analysis of seamount distributions is an important geological indicator, while section 7.2 summarises the concepts of the Hough transform. The method is applied to several portions of TOBI data (7.3), and sections 7.4 and 7.5 deduce the success and future potential uses of the technique.

Finally, chapter 8 draws the conclusions from the four research chapters together. Assessments are made of their success or otherwise, and future potential applications as tools for geologically interpreting mid-ocean ridge sonar data are suggested.

CHAPTER 2
PREVIOUS WORK

2.1 SEAFLOOR BACKSCATTER CHARACTERISATION METHODS

The remote classification of seafloor materials has been a subject of great interest for many years. Its applications include mineral assessments, military uses, sonar modelling of acoustic behaviour at sediment/water interfaces, and of course, geology. The backscatter intensities represented in sonar imagery are basically controlled through the interaction of three properties:

- angle of incidence
- microtopography or roughness
- physical properties of the medium

Section 5.2.1 describes these phenomena in detail.

Two types of data are examined in this thesis: backscatter strengths recorded by a multibeam echosounder (EM12), and high resolution sidescan sonar data (TOBI). Analyses of acoustic backscatter measurements from multibeam sonars are relatively recent; these backscatter strengths have been compensated for gain variations, source levels, and beam patterns. They also possess co-registered bathymetry measurements, so true three dimensional angles of incidence may be determined, enabling acoustical analyses. de Moustier and Matsumoto (1993) describe four basic approaches for analyzing multibeam backscatter data:

- probability density functions of the peak amplitudes of near-normal incidence returns
- statistics of instantaneous amplitude and phase samples
- shape of backscatter angular dependence function
- actual backscatter coefficients

Sidescan sonar data are traditionally interpreted by geologists, rather analogously to photo-interpretation, through the variation of tone in terms of texture (e.g. various

lava morphologies) and shape (typical shapes may be volcanoes, fault scarps etc.), supported by ground truth sampling and geological knowledge about, for example, tectonic settings, sedimentary facies relationships etc. Published quantitative methods for analysing sidescan data can be categorised as:

- conversion of intensities to backscatter strengths
- texture analysis
- spectral analysis
- object recognition

Those techniques concerned with studying true backscatter strength data can be broadly defined as acoustical studies, while those analysing the textural variations of backscatter intensities will be classed as statistical studies. The following two sections briefly describe some of the previous research performed in these fields; those methods investigated in this thesis will be dealt with in more detail in their relevant chapters, and are summarised in the objectives stated in section 1.2.

2.1.1 Acoustical studies

Section 5.2 of chapter 5 provides a detailed review of previous acoustical studies, concentrating on those with particular relevance to the subsequent analysis of the EM12 backscatter data. Here, a paragraph will be designated to describe each of the four potential methods for multibeam backscatter analysis presented above.

The relationship of probability density functions of the peak amplitudes of near-normal incidence backscatter returns with seafloor microroughness have been investigated by Stanton (1984), de Moustier (1986) and Talukdar and Tyce (1992). Stanton (1984) first presented the concept of combining probability density function (PDF) distributions with scattering theory. PDFs were fitted to near-normal incidence echo envelopes of 3.5 kHz backscatter data, and related to a generalized version of spherical wave Helmholtz-Kirchhoff based scattering theory (Clay and Medwin, 1974). This approach yielded two parameters describing the shape of the PDFs: rms roughness and correlation area. The correlation area is useful for distinguishing cases such as a seafloor composed of ripples and rocks, which may possess the same rms roughness. Effectively, a description of the small-scale vertical dimension of the seafloor is derived from the sidescan data, in addition to the traditional lateral information. The methods of Stanton (1984) were extended to multibeam (Sea Beam) datasets by de Moustier (1986) and Talukdar and Tyce

(1992). de Moustier (1986) applied Stanton's (1984) methodology, alongside traditional first-order statistical and angular dependence derivations, to characterise Sea Beam backscatter data collected over three distinct geological provinces: a manganese nodule field, a deeply sedimented region and a portion of the East Pacific Rise crest. Talukdar and Tyce (1992) identified variations in Rice and extremal PDFs between abyssal plain and mid-ocean ridge Sea Beam backscatter envelopes, as well as conducting an examination of backscatter angular dependences (see relevant paragraph in this section). The backscatter envelopes are not preserved for EM12 amplitude data, so investigation of this particular method is not possible.

Alexandrou *et al.* (1992) evaluated the statistics of instantaneous amplitude and phase samples by simulating reverberation PDF fits, based on non-Poisson scatter distributions. These fits enabled the generation of feature vectors reflecting the distinguishing characteristics of each scattering distribution. Alexandrou *et al.*'s (1992) theory was validated with Sea Beam data from sedimented areas; important information about the distribution of scattering centres on the seafloor was deduced. Since the receive acoustic backscatter envelopes are not retained for the EM12 data, this method may not be applied.

It has been widely established that the shape of backscatter angular dependence functions varies with bottom type (e.g. McKinney and Anderson, 1964; Wong and Chesterman, 1968; Urick, 1983; and Hughes Clarke, 1993). Jackson *et al.* (1986) derived a theoretical model for high frequency backscatter over the full 0-90° incident angular range, which successfully matched their experimental data. de Moustier and Alexandrou (1991) fitted Jackson *et al.*'s (1986) model to Sea Beam backscatter data collected over the same sedimented sites as described in Alexandrou *et al.* (1992). Talukdar and Tyce (1992) related Jackson *et al.*'s (1986) model to Fox and Hayes' (1985) bathymetric roughness descriptors. Dziak *et al.* (1993) followed a similar approach to Talukdar and Tyce (1992) for Sea Beam data collected on the Juan de Fuca Ridge, delineating volcanic zones and tectonically disrupted regions using Jackson *et al.*'s (1986) model for estimating small scale topographic roughness spectra, which one is unable to do with multibeam bathymetric measurements. Dziak *et al.* (1993) verified their results with SeaMARC I and II sidescan data, plus the detailed geological knowledge of the region. Since the analysis of the EM12 dataset in chapter 5 followed the angular approach, further background descriptions will be reserved for sections 5.2.2 and 5.2.3.

Finally in terms of acoustical studies, the variation of mean backscatter levels with bottom type provides obviously important information, as do other first-order statistics. Multibeam backscatter data currently available are not calibrated, so care must be exercised when comparing data collected from different systems and making analogies with calibrated data; the general relationships and contrasts are helpful though. The first-order backscatter statistics are often used in conjunction with other more advanced descriptors (e.g. de Moustier, 1986; Hughes Clarke, 1993; and chapter 5).

2.1.2 Statistical studies

Ideally, sidescan sonar backscatter intensities should be converted to backscatter strengths, providing system gains, beam pattern information etc. are known about the particular instrument providing the data. If a co-registered bathymetry dataset of comparative resolution is also available, then one can begin to apply some of the acoustical analysis methods described in section 2.1.1 in order to quantitatively characterise the data. For example, Mitchell and Somers (1989) computed backscatter strengths from the GLORIA 6.5 kHz sidescan sonar, combining it with independently collected Sea Beam bathymetry data. However unfortunately, this approach is difficult for datasets that are not collected by the same system, and anyway, for most sidescan sonar data collected over the ocean floors, this additional information is not available. A number of, what will be generalized as, 'statistical' approaches have been researched in an attempt to quantitatively analyze these types of data.

The second-order statistical approach of texture analysis is independent of unknown gain variations etc. Two principal textural approaches have been applied to sidescan data: the Grey Level Co-Occurrence Matrices method (Pace and Dyer, 1979; Reed and Hussong, 1989); and fractal techniques (Linnett *et al.*, 1991; 1993). Both these methods successfully discriminated the necessary bottom types in the data tested. Chapter 6 of this thesis describes the application of both these methods to TOBI high resolution sidescan sonar data from the Mid-Atlantic Ridge, in terms of discriminating material, microroughness and 'macro-'roughness (i.e. small scale morphological features resolved by TOBI) properties of the seafloor.

An alternative approach for the characterisation of seafloor materials utilises spectral analysis methods. These were first introduced by Reut *et al.* (1985), who derived power cepstrum information, leading to the derivation of two integral features, based on the intercept and slope of the power cepstrum integral within a

fixed time interval. The power cepstrum is the Fourier transform of the log of the the power spectrum

of the sampled amplitude trace. Reut *et al.* (1985) plotted up their 48 kHz sidescan data in the plane of the two feature descriptors, and yielded distinct discriminations for 6 bottom types (rock, sand, mud,...) in this feature space. Pace and Gao (1988) extended the success of these results by looking at feature descriptors derived from both power spectrum and cepstrum information. Tamsett (1993) mathematically modelled these power spectra using a low-pass Butterworth filter, and was able to reduce the description of the power spectrum for each bottom type to just 6 numbers. Tamsett (1993) achieved the best bottom classification results so far, using what he described as a more rigorous method than the somewhat arbitrary definitions of features by Pace and Gao (1988). However, Tamsett (1993) does concede that his methods are far more demanding computationally.

A final method for quantitatively characterising aspects of sidescan sonar data is in the field of object recognition. Object recognition is a widely-researched subject amongst the wealth of image processing literature, though the problem becomes difficult when the objects of interest are located on a background that is anything other than homogeneous; few applications have been accomplished with sidescan sonar data. Man-made objects on the seafloor are important articles requiring automatic detection. One object recognition method is described by Linnett *et al.* (1993), who after fractal texture segmentation of the background and histogram modifications to yield Gaussian responses, derived a probability image from the input data, based on Poisson distributions, where any significant deviations in probability corresponded with the bright/shadow nature of the objects of interest. A geological application is Little and Smith's (Fault scarp identification in side-scan sonar and bathymetry data from the Mid-Atlantic Ridge using 2-D wavelet analysis, submitted to *Marine Geophysical Researches,* 1993) automatic fault detection procedure using high resolution TOBI sidescan sonar data from the Mid-Atlantic Ridge (the same dataset as used in this thesis). They applied digital filters using wavelet theory to successfully locate the majority of brightly backscattering fault scarps identified by geological interpretation. Basically, the wavelet transform subdivides an image into a series of images which contain information on the location of features at specific spatial frequencies and orientations. The wavelet can be tuned according to the nature of features being sought. In this thesis, chapter 7 is devoted to an appraisal of an object detection technique: the automatic detection of seamounts from TOBI sidescan sonar data using the Hough transform.

2.2 MID-ATLANTIC RIDGE GEOLOGY

The multibeam and sidescan sonar data analysed in this thesis were collected on four research cruises to three sites on the axis of the Mid-Atlantic Ridge (MAR). Figure 2.1 illustrates the locations of these areas: the Reykjanes Ridge (~58-62°N); the MAR at 45°N; and the MAR between the Kane and Atlantis fracture zones (24-30°N). Section 1.4 describes the basic cruise details. This thesis does not intend to provide a geological analysis of these datasets, but an assessment of various methods for quantitatively analysing the sonar data, which in turn will provide further insights and ease to the geological analysis. This section will therefore concentrate on a short overview of the surficial morphological and lithological characteristics of the northern MAR in as far as it is relevant to the proceeding research of this thesis. No attempts shall be made to explain models of the geology, and no account will be made of petrological, geochemical and deep structural aspects of the region. Many review papers summarise the detailed aspects of mid-ocean ridge geology, e.g.: Macdonald (1986), Sempéré and Macdonald (1987), Searle (1992a) and Smith and Cann (1993).

The primary topographic characteristics of mid-ocean ridges are determined by the spreading rate. For example, the fast spreading East Pacific Rise (EPR), reaching a full spreading rate of 180 mm yr^{-1}, exhibits a more or less continuous axial high volcanic zone, some 0.5-2 km wide (Sempéré and Macdonald, 1987). This volcanic zone is approximately 100 m high, can extend for many tens of kilometres along strike (Lonsdale, 1977), and is composed of low relief lava flows, with sheetflows predominating over pillows at higher spreading rates (Searle, 1992a). In contrast, the slow spreading MAR, spreading at a full rate of some 25 mm yr^{-1} (Smith and Cann, 1993), comprises an axial rift valley 30-45 km wide (often referred to as the median valley) and 1-2 km deep (Macdonald, 1986). The volcanism is principally confined to axial volcanic ridges (AVRs), which can reach hundreds of metres in relief, 1-5 km in width and up to tens of kilometres in length (e.g. Sempéré *et al.*, 1990; 1993). These ridges are composed of coalesced individual small volcanoes (Smith and Cann, 1990; 1992).

Mid-ocean ridge spreading axes are divided into segments, which can be classified into a number of scales. Macdonald *et al.* (1988) define four classes of segmentation, the bounds of which can be broadly categorised into transform and non-transform offsets. Transform offsets comprise the major transform faults,

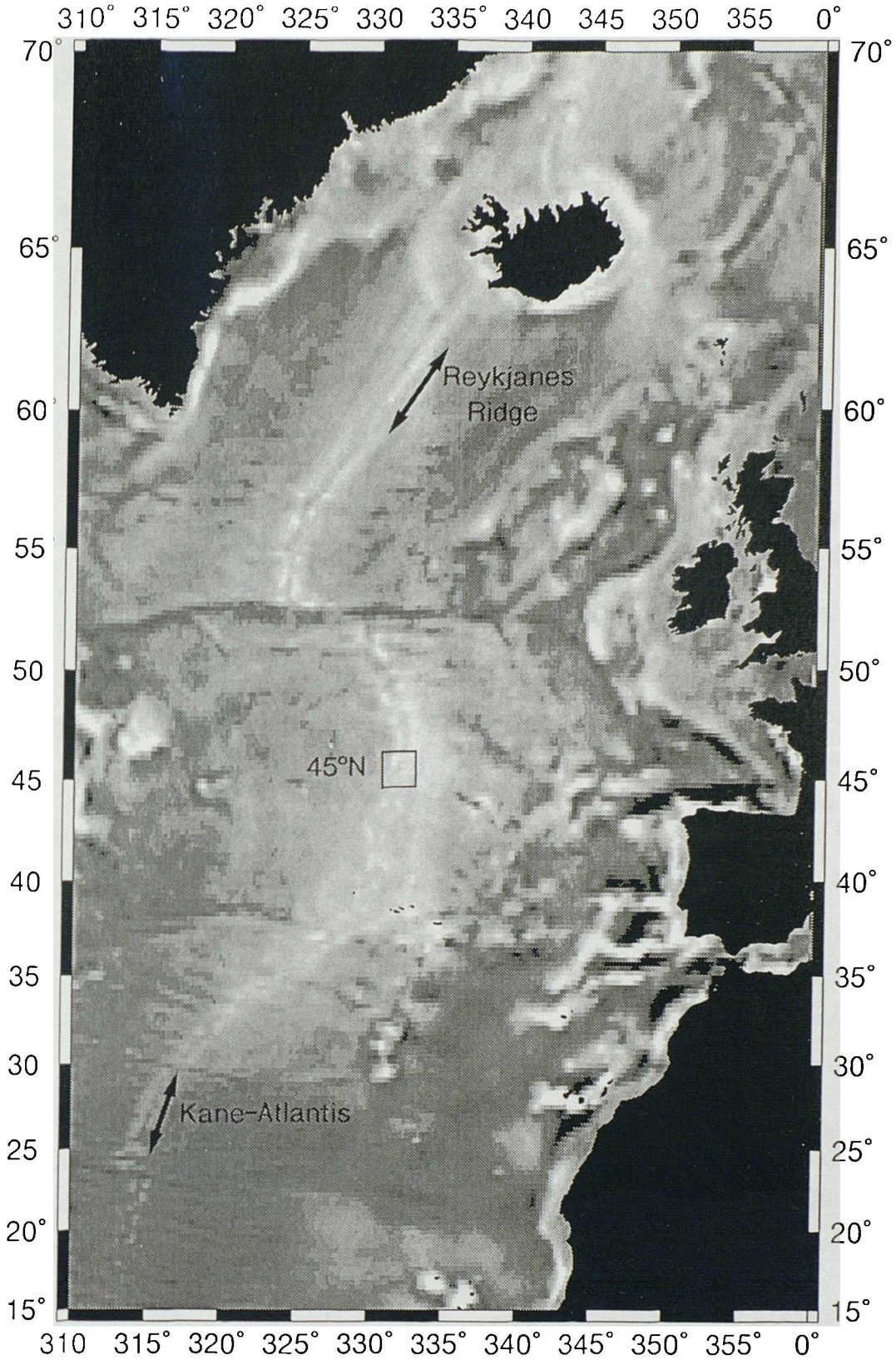


Figure 2.1 *Location map for survey areas, the background is the Seasat free air gravity anomaly data, which closely resembles the seafloor topography. The Mid-Atlantic Ridge is expressed as a gravity high (light grey).*

separated by along-strike distances of more than 100 km, while the non-transform offsets grade through along-strike length intervals of a few tens of kilometres, and normal displacements of between 0.5 and 5 km. The larger non-transform discontinuities are associated with off-axis discordant zone traces, consisting of a series of isolated basins, parallel to the spreading direction. Most of the axis of the northern MAR comprises a series of discrete spreading segments, tens of kilometres in length (Sempéré *et al.*, 1990; 1993), each possessing different volcanic and tectonic styles (Karson *et al.*, 1987). A regular variation in the along axis depth profile demonstrates that the segments are shallow at their centres, and deep towards their offset margins; complementary studies by Lin *et al.* (1990) suggest focussed magmatic accretion at the centres of the segments from mantle Bouguer anomalies.

Within each spreading segment, an axial volcanic ridge (AVR) can usually be identified, aligned normal to the spreading direction regardless of local ridge orientations (even at the 36° oblique-spreading Reykjanes Ridge: Searle and Laughton, 1981; Parson *et al.*, 1993). These AVRs form the major sites of crustal production (e.g. Ballard and van Andel, 1977). From submersible observations, Ballard and van Andel (1977) also identified AVR-like morphologies within the bounding walls of the median valley, which they inferred to be fossil ridges (chapter 6, section 6.6.6b). This confirms that crustal production in the form of AVRs has been prevalent for at least the last million years (Smith and Cann, 1993).

At the MAR between 24 and 30°N, Smith and Cann (1990; 1992) identified abundant small near-circular volcanoes from Sea Beam multibeam bathymetry data on the median valley floor. Some were isolated constructs, others formed linear trends, but the majority were confined to the AVR region. Smith and Cann (1993) describe the various forms of volcanic constructs imaged with the TOBI 30 kHz deep-towed sidescan sonar (data used in this thesis), culminating in a median valley floor composed of a mixture of individual flat-topped seamounts, piled up small volcanoes, hummocky and smooth lava flows. The proportion of each of these morphologies appears to vary between spreading segments (chapter 6, section 6.6.6a). Most of the AVRs at the MAR are composed of pillow basalts (as observed in the FAMOUS area by Ballard and van Andel, 1977), with a diameter of approximately 1 m. Some flatter basaltic sheetflows have also been identified.

The sedimentation rate of the MAR at 35°N is 1.9 m Myr^{-1} (Balsam, 1988), suggesting sediment thicknesses of about 8 m at the edge of the inner valley floor. Submersible observations by Ballard and van Andel (1977) and Deep-Tow

photographs from Luyendyk and Macdonald (1977) yielded a light dusting of sediments (~10-15 cm thick) on axial volcanic ridges, and more than a metre of sediments further off-axis but still within the median valley, with significant amounts gathering in topographic pockets.

Ballard and van Andel (1977), from submersible observations, commented how the flanks of the AVRs are often fissured and faulted; these faults striking normal to the spreading direction and usually oriented parallel to the ridge axis. This 'zone of fissuring' is about 1-3 km wide at all spreading rates (e.g. Ballard and van Andel, 1977; Lonsdale, 1977). Ballard and van Andel (1977) suggested that this zone was the initial source of the major median valley bounding faults. Searle (1992a) describes a 'fault zone' bordering the fissure zone, starting at a distance of 0.5-2.5 km from the spreading axis. The degree and throw of faulting increases considerably here, with an approximate width for the series of normal fault blocks of 1-2 km, and many tens of kilometres long and up to several hundreds of metres high. The first of these major faults marks the edge of the median valley inner floor. GLORIA and multibeam data suggest single scarps, though higher resolution studies (e.g. Kong *et al.*, 1988), observed multiple series of closely spaced smaller scale scarps. Either side of the median valley are the rift or crestal mountains, which between 24° and 30°N, rise 400-3000 m above the valley floor, and consist of axis parallel ridges, continuous along strike for up to 60 km (Sempéré *et al.*, 1993). It has been postulated (Searle, 1992a, cites the relevant publications) that some of the tectonic compensation for the inward dipping fault scarps is accommodated by their outward rotation. Shaw (1992) analysed the distribution of faults for the Kane-Atlantis section of the MAR.

Fault scarps are generally composed of weathered basalt, and possibly gabbroic, doleritic or even ultra-mafic rocks, and often exhibit considerable ramps of talus at their bases. Tucholke (1992) identified a massive rockslide from bathymetry data from the Kane-Atlantis section of the MAR, and Cann *et al.* (1992) confirmed further evidence of extensive landslides in the vicinity from TOBI high resolution sidescan sonar data. Associated dredging yielded uniformly sized, weathered basalt blocks for Tucholke's (1992) slide, while another also supplied meta-basalts, serpentinites and gabbros.

The Reykjanes Ridge comprises over 1000 km of the MAR, extending from the southwest shore of Iceland (~64°N) to the Bight fracture zone (57°N). The ridge is obliquely spreading; oriented at 036°, but spreading at 099° with a half spreading rate of 12 mm per year (Talwani *et al.*, 1971; Vogt and Avery, 1974). The gross

morphology of the Reykjanes Ridge is influenced by the Icelandic plume, and is expressed as an axial high in the north, and exhibits a rudimentary median valley to the south (Talwani *et al.*, 1971).

The 45°N area of the MAR possesses classical slow-spreading ridge morphology, with comparable segmentation patterns to those observed further south by Sempéré *et al.* (1993). The rifting characteristics are well-matched either side of the axis and fossilised axial volcanic ridges can be identified off-axis. Aumento *et al.* (1971), Searle *et al.* (1978), and Laughton and Searle (1979) present much of the previous work in the area.

CHAPTER 3
INSTRUMENTATION

3.1 INTRODUCTION

The seabed has been surveyed by sound since the 1930s, when acoustic echosounder methods replaced lead lines. The next major advancement occurred in the 1970s when multibeam echosounders (Glenn, 1970) and long range sidescan sonar devices (Laughton, 1981) were introduced. Seabed mapping includes both geometrical and physical mapping: geometrical involves the determination of the topography, or bathymetry, of the seabed; while physical can refer to the measurement of material properties, characteristic roughness and textural classification of the sea bottom types. The relevant respective measuring tools investigated in this thesis are the multibeam echosounders and sidescan sonars, including one of the recently hybrid systems, capable of simultaneously collecting both types of data.

In this chapter, a general technical description of the concepts of the devices shall be given, followed by an overview of the various systems used for geological mapping purposes. A detailed technical report will then be presented of the two systems from which the data were collected for analysis in this thesis: the Simrad EM12 multibeam echosounder and the TOBI high resolution sidescan sonar.

3.2 TECHNICAL OVERVIEW

Many texts are available which review the design and performance of various seafloor mapping systems. Tyce (1987) reviews some of the widely used multibeam and sidescan sonar systems, de Moustier (1988) presents a state of the art assessment of multibeam echosounding systems, de Moustier and Kleinrock (1986) introduce some of the commonly found artifacts in multibeam bathymetry data, while Davis *et al.* (1987) and Kleinrock *et al.* (1992) examine and appraise the types and applications of data collected by some of the most widely used swath imaging and bathymetric mapping tools.

3.2.1 Multibeam echosounders

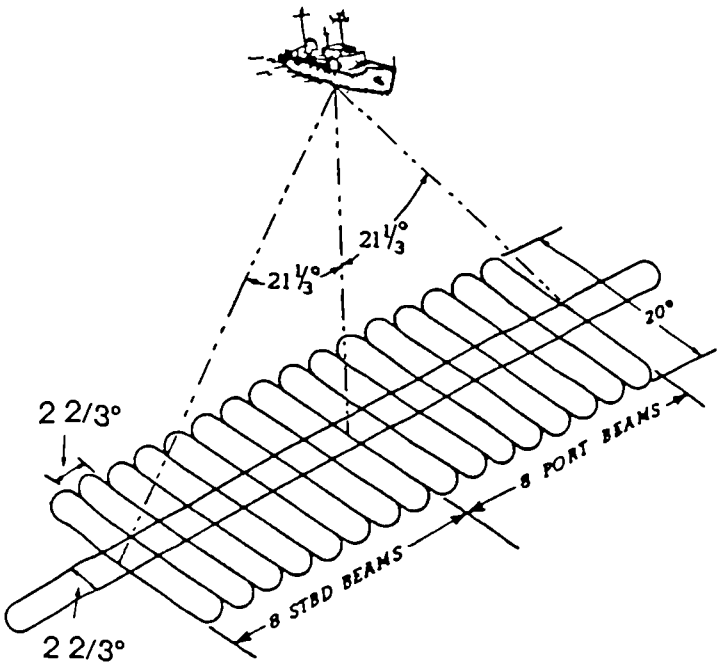


Figure 3.1 *Geometry of Sea Beam multibeam echosounder (from de Moustier and Kleinrock, 1986).*

Figure 3.1 presents the geometry of a multibeam echosounder (Sea Beam), taken from de Moustier and Kleinrock (1986). The transmission and reception arrays are mounted on the hull of the ship at right angles to each other, with each array producing a beam which is narrow in the direction perpendicular to its short axis. That is, a transmission beam is formed which is very narrow along-ship and very wide across-ship; while conversely, a reception beam is formed very narrow across-ship and very wide along-ship. If the phase of the returned signal is altered, the reception beam can be steered athwartships at specific intervals, between ±45° of incidence for the EM12. The acoustic energy received at the ship comes from the intersection of the transmit and receive beam patterns, forming 81 beams for the EM12. The actual beam footprints are not the regular rectangles inferred in figure 3.1, but are ellipses whose areas increase away from vertical incidence. Each bathymetry sample measured is in fact a convolution of the seafloor topography with the footprint of the beam function.

The received energy for each of the formed beams is filtered, rectified and amplified before being transferred to the multibeam computer. Figure 3.2, from de Moustier and Kleinrock (1986), presents a typical output for the 16 beams of Sea

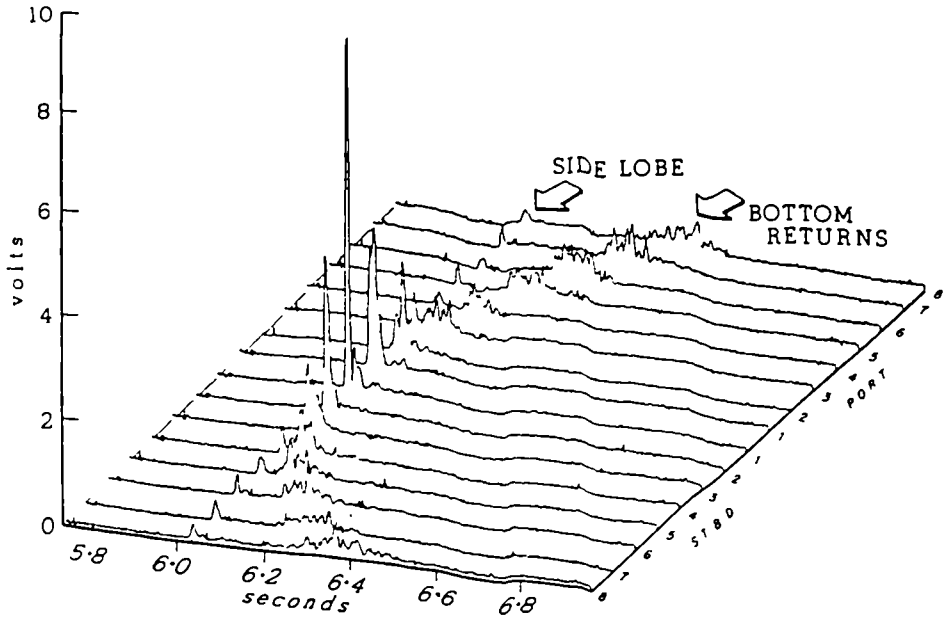


Figure 3.2 *Acoustic backscatter envelopes from Sea Beam for each of the 16 receive beams (from de Moustier and Kleinrock, 1986).*

Beam. The hyperbola formed by the peaks of the return signals indicates a flat portion of seabed. These waveforms are digitised by the multibeam computer, concurrently performing gain corrections, roll compensations and echo detections. The slant range for each beam is computed from the centre of mass of the detected signal amplitudes and multiplying by the speed of sound in water/2. Knowing the beam angle, the depth and across-track horizontal distance can be determined through simple trigonometry (if refraction is ignored).

The accuracy of measurements achieved by a multibeam echosounder are dependent upon the determination of the velocity of sound through the seawater, since variations will introduce refraction effects on oblique beams, limiting the maximum range (swath width) achievable with a system. The velocity is a function of temperature, salinity and pressure of the seawater. Failure to compensate for sound velocity variations will produce errors in the determination of angles of arrival, yielding errors in depth and across-track distances, which increase with increasing incident beam angles. de Moustier (1988) presents the appropriate sound velocity correction equations required to compensate for inaccurate initial sound velocity profiles, based on Snell's Law. Sound velocity probes are usually deployed for a kilometre or so down at regular intervals throughout a survey.

de Moustier (1988) also describes how noise can affect multibeam echosounders. For the specific frequency of a system, the beamwidths dictate the size of the transducer arrays, which in turn influence whereabouts on the ship's hull they may be mounted. Optimally, this should be in a position least affected by air bubble masking and machinery noise. de Moustier (1988) states that he noticed no increase in noise level for the Sea Beam system on board the R.V. Thomas Washington when operating the ship's retractable bow thrusters in any direction at maximum speed, though bubble masking during the reception cycle in sea state 4, raised the noise level by 20-30 dB. de Moustier (1988) therefore concluded that the performance of most multibeam systems is hindered beyond sea state 4, particularly when heading into the sea.

de Moustier (1986) preserved the signal amplitude samples for each beam (as presented in figure 3.2) to generate sidescan images as a function of time. Talukdar and Tyce (1990) geometrically corrected the Sea Beam sidescan images using the bathymetry information (chapter 4, section 4.3 describes this processing further).

3.2.1a Characteristics of various multibeam echosounders Table 3.I presents some basic characteristics of the non-military multibeam echosounders that have commonly been used for geological mapping purposes. Some of these swath widths vary according to water depths, or maybe additional software options; the maximum widths are presented.

Manufacturer	Instrument	Frequency /kHz	Swath width (× water depth)	No. of beams	Max. water depth	backscatter ?
Sea Beam	Sea Beam	12	0.75	16	> 11 km	
	Hydrochart	36	2.5	17	1.5 km	
	Sea Beam 2000	12	3.4	121	> 11 km	✓
Simrad	EM100	95	2	32	700 m	✓
	EM1000	95	7.4	60	1 km	✓
	EM12	13	2 (& 3.5 option)	81	> 11 km	✓
	EM12 dual	13	7.4	162	> 11 km	✓
Krupp Atlas	Hydrosweep	15	2	59	> 7 km	✓?

Table 3.I - Multibeam echosounder characteristics

de Moustier (1988), Hughes Clarke *et al.* (1993), Simrad (1992) and Gutberlet and Schenke (1989) provided the data presented, and also describe many more assets of the systems, including beam widths, resolution capabilities etc., as the echosounders all possess their own unique characteristics. A detailed account of a Simrad system

(EM12) is given in section 3.3, and the backscatter capabilities of the new multibeam systems shall be addressed in section 3.2.3a.

3.2.1b Bathymetric artifacts de Moustier and Kleinrock (1986) describe some bathymetric artifacts commonly observed in Sea Beam data, due to errors in echo detection and processing. They cite three common causes of artifacts: (1) side lobe interference; (2) external sound source interference; and (3) 'omega' effects and data gaps. The side lobe interference is known as the 'tunnelling' effect and is caused by the amplitude of the side lobes of the beam signal being larger than those of the bottom returns (figure 3.2 illustrates side lobe signals, which are lower here than the bottom returns). This renders a trough-like bathymetric morphology on a flat seafloor. External sound sources, like other single beam echosounders, seismic sources and bottom transponders, can interfere in various ways with the received signals. 'Omega' effects are data gaps caused by the failure of bottom tracking at sudden changes of slope along-track, combined with side lobe effects. Unfortunately, many other geophysical surveying techniques (e.g. gravity and magnetics) require track lines perpendicular to the tectonic fabric (highly fluctuating at mid-ocean ridges), which exacerbates this problem. de Moustier and Kleinrock (1986) stress the importance of a clear understanding of the capabilities and limitations of multibeam systems prior to interpret ing the geology.

3.2.1c Comparison of data collected by various mapping tools Davis *et al.* (1987) and Kleinrock *et al.* (1992) assess the geological surveying characteristics of various well-known seafloor mapping tools. Davis *et al.* (1987) present various datasets collected over the central Juan de Fuca Ridge. Increasing resolution of bathymetric mapping techniques was demonstrated by successive contour plots of single beam echosounder, SeaMARC II and Sea Beam bathymetry, though spatial coverage per swath was greater for SeaMARC II than Sea Beam.

Kleinrock *et al.* (1992) compared Sea Beam, SeaMARC II and GLORIA data collected over the same region of the Galapagos propagating rift systems, with smaller areas within this zone mapped by Deep-Tow sidescan, camera and the Alvin submersible. System capabilities were determined according to the devices' ability to detect and measure various geological features. Kleinrock *et al.* (1992) present a table documenting the detection characteristics of the various mapping tools. Goff and Kleinrock (1991) complement these results with a statistical analysis of topographic profiles measured by different systems, noting at which resolution and slope angles data begins to degrade. Kleinrock *et al.* (1992) conclude that the

theoretical resolution of these mapping instruments generally overestimates the detail actually detectable by something like a factor of two; they state that in order to extract further information, such as elongation, relief, shape etc., a reduction in resolving capacity of a factor of 10 is attained.

3.2.2 Sidescan sonar systems

The principle of sidescan sonar systems is conceptually very simple. Sidescan sonars, such as GLORIA II (Somers *et al.*, 1978), comprise two linear arrays of transducers, used for both transmission and reception, mounted either side of the instrument, scanning out to each side of the vehicle's track (figure 3.3 summarises the geometry).

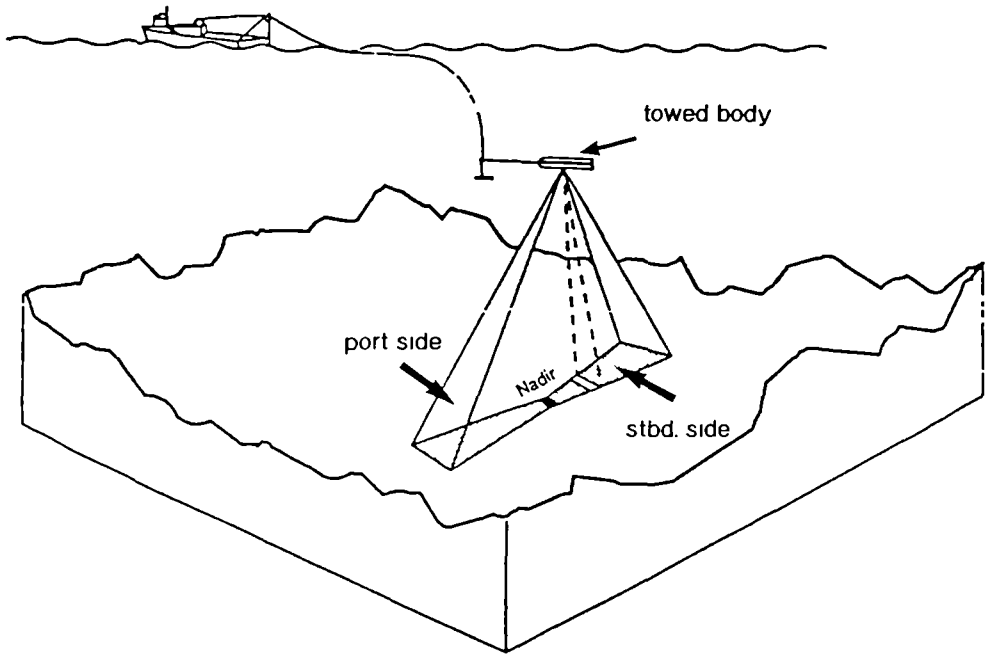


Figure 3.3 *Sidescan sonar operational geometry (from Johnson and Helferty, 1990).*

For GLORIA II, the transducer arrays are directed 20° below horizontal, and as with all side looking sonars, are very narrow along-track (2.7° for GLORIA II) and very wide across-track (Searle *et al.*, 1990, display the beam pattern). Resolution is primarily governed by the acoustic beamwidth and the length of the transmitted pulse, except in frequency modulated pulse systems such as GLORIA II, where the resolution is determined by the bandwidth of the frequency modulation. This resolution can be up to ten times poorer along- than across-track. In order to

produce the required beam shape, the transducer arrays must be long in the horizontal axis and short in the vertical.

Sidescan sonars are generally towed, rather than hull-mounted, for three main reasons (Tyce, 1987): (1) to minimize processing complications caused by the pitch and roll of surface ships; (2) to minimize the influence of the refraction of sound near the sea surface; and (3) to increase the resolution by enabling the instrument to be towed near to the seafloor. The range of a sonar is controlled by the degree of sound absorption in the sea water, which in turn is directly proportional to the frequency squared. Swath widths are generally about ten times the vehicle's altitude.

Johnson and Helferty (1990) describe the sidescan process, beginning with the transducers transmitting an acoustic pulse down to the seafloor. The transducers are then set in receiving mode. The first bottom return activates a clock, thus producing a voltage time series of subsequent acoustic returns (figure 3.4).

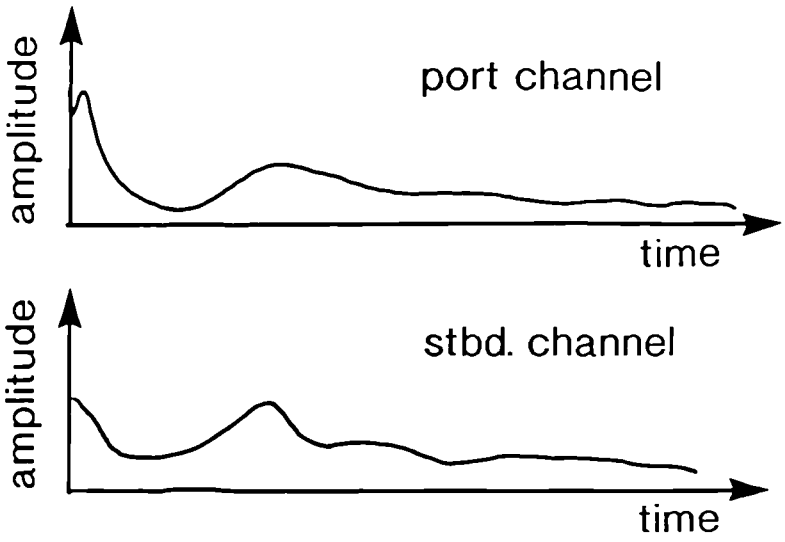


Figure 3.4 *Examples of typical sidescan amplitude traces.*

This series is digitised by dividing it into time slices, which are narrow for early returns and wider for later returns from greater slant ranges. This integrated value represents the amount of acoustic signal backscattered from the seafloor within the particular time interval. After pre-amplification, a time-varying gain (TVG) is applied to counteract spreading losses and attenuation in the water column. The amplitude time series can then be converted into across-track horizontal distances using a slant-range to ground-range projection, either by assuming a flat seafloor and using the vehicle altitude data, which can lead to number of distortions,

or better to project using a co-registered bathymetry grid, much in the same manner as Naraghi *et al.* (1983) did for satellite radar imagery (see processing section 4.3).

3.2.2a Characteristics of various sidescan sonars Table 3.II presents the major characteristics of some of the commonly used sidescan sonars.

Instrument	Frequency /kHz	Towing depth (D) or towing altitude(A)	across-track res. /m	along-track res. /m	Survey speed /kts	bathymetry ?
GLORIA II	6.5	30-60 m (D)	30	220	10	✓
SeaMARC I	30	25-1250 m (A)	0.5-2.5	0.7-37	2	
SeaMARC II	12	50-100 (D)	5	175	8	✓
Deep-Tow	110	10-150 m (A)	0.1-0.3	0.1-2	1-2	
TOBI	30	~400 m (A)	2-7	4-42	2	

Table 3.II - Sidescan sonar characteristics

Somers *et al.* (1978), Tyce (1987) and Flewellen *et al.* (1993) provided this information. As explained above, frequencies, tow depths, survey speeds and resolutions are all interlinked. Reviewing the performance of various sidescan systems, Davis *et al.* (1987) confirm that although the resolving power of typical sonars increases through the GLORIA II, SeaMARC II, TOBI, SeaMARC I and Deep-Tow instruments; the daily spatial coverage decreases concurrently. Tyce (1987) reviews the designs of the most widely used sidescan systems.

3.2.2b Sidescan artifacts The commonly identified artifacts associated with sidescan sonar data have been addressed through extensive post-processing procedures. Therefore, the description of the artifacts will be described alongside their associated techniques for suppression in chapter 4, section 4.3.

3.2.3 Hybrid systems

3.2.3a Backscatter measurements with multibeam echosounders Referring back to the 16 acoustic waveforms measured with Sea Beam in figure 3.2, de Moustier and Matsumoto (1993) claim that very useful quantitative information describing the waveforms can be derived in a number of ways. These are outlined in chapter 2, section 2.1.

In order to determine backscatter measurements using a multibeam echosounder, a number of steps must first be taken. de Moustier and Alexandrou (1991) describe how sidelobe interference (figure 3.2) must first be removed. The centroid of the echo received by each beam is determined and is assumed to coincide with the

maximum response axis of the beam. The beam pattern function and area insonified can therefore be used to normalise the power for each return. Roll and refraction corrections are required for Sea Beam, though roll compensation forms an integral part of the beam formation process for the Simrad systems.

de Moustier and Matsumoto (1993) present acoustic backscatter responses plotted against incidence angle from Sea Beam data collected over the uniformly sedimented Magellan Rise (7°N, 177°W). These were: (a) 'raw' echo powers; (b) after an across-track bottom slope correction; and (c) following a combined bottom slope and area insonified correction. They conclude from the difference between curves (a) and (b) that a flat bottom assumption is inadequate even for this low relief region, and they have not yet considered the effects of along-track slopes. Curve (c) reinforces the need to account for the larger footprint size of the outer beams compared with the inner. de Moustier and Matsumoto (1993) suggest normalising all backscatter measurements according to curve (c) to produce maps of backscatter coefficients. However, Hughes Clarke (1993) and many others demonstrate that varying bottom types exhibit different angular responses, so instead of normalisation, these may provide a bottom type characterisation. Chapter 5 of this thesis investigates both these phenomena over the expected spatially and lithologically heterogeneous bottom materials encountered at a section of mid-ocean ridge.

The multibeam systems capable of the real-time recording of backscatter measurements have only been introduced within the last 5 years; the leading manufacturers of instruments used within the geological community are Sea Beam and Simrad. The design of Sea Beam 2000 is described by Talukdar *et al.* (1992) and the major components are presented in table 3.I, the details of the Simrad EM12 system shall be described in section 3.3, and Simrad's shallow-water (EM100, EM1000) and wider swath width (EM12 dual) systems operate under similar principles.

3.2.3b Backscatter measurements with bathymetric sidescan sonars

de Moustier and Matsumoto (1993) present the operational principles of bathymetric sidescan sonars. Since these systems are mounted on towed bodies, they may operate at any frequency for full ocean depth, provided there is sufficient length of cable and enough bandwidth to accommodate the data transmission rate. As has been stressed before, and de Moustier and Matsumoto (1993) reiterate, the gathering of co-registered bathymetry and sidescan data is very important for the conversion of

a qualitative acoustic image into a quantitative geographic map of backscattering coefficients.

de Moustier (1988) describes how sidescan sonars measure bathymetry using interferometry. The interferometric effect is used for reception beams only, and is possible because two transducer arrays are mounted parallel to each other, and at a specified distance apart. The angular resolution is obtained by relating the phase difference of the signals arriving at the two arrays, to the angle of arrival of these signals (much as described for EM12 interferometry in section 3.3).

3.3 THE SIMRAD EM12 MULTIBEAM ECHOSOUNDER

3.3.1 Construction

Figure 3.5 presents the main system units of the EM12 (from Simrad, 1992). The two transducer arrays are mounted in a cross-shaped configuration, with one array used for transmission and the other for reception. These transducers are set into a blister arrangement on the hull of the ship. The receiver array is connected to a preamplifier unit and both transducer arrays are linked to the transceiver unit. This unit comprises power amplifiers, the data acquisition boards and digital signal processors for beamforming, filtering and system control. The transceiver unit also has built in interfaces for gyrocompass and heave/roll/pitch sensor. On the R.V. Ocean Surveyor, the preamplifier and transceiver units are located in the hold of the ship.

The bottom detector unit is sited in the ship's laboratory and controls the transceiver unit, setting parameters such as operation mode, coverage sector, depth range etc. The operator unit is mounted in a separate display console, presenting the bathymetric data, ping by ping. Position and clock data may be received on separate serial lines. An optical disk drive and colour plotter recorder are also connected to this unit. A sound velocity profile is taken separately, using sound velocity probes (Applied Microsystems Ltd. STD-12 and Navitronic SVP-1); this information is read into the operator unit, and accounts for refraction effects through the water column. Additionally, continuous surface sound velocity probe measurements control the beam directions. The quality assurance unit comprises two colour VDUs; one is typically used for a real-time geographical bathymetry plot covering the survey area, while the other displays partly-processed backscatter data.

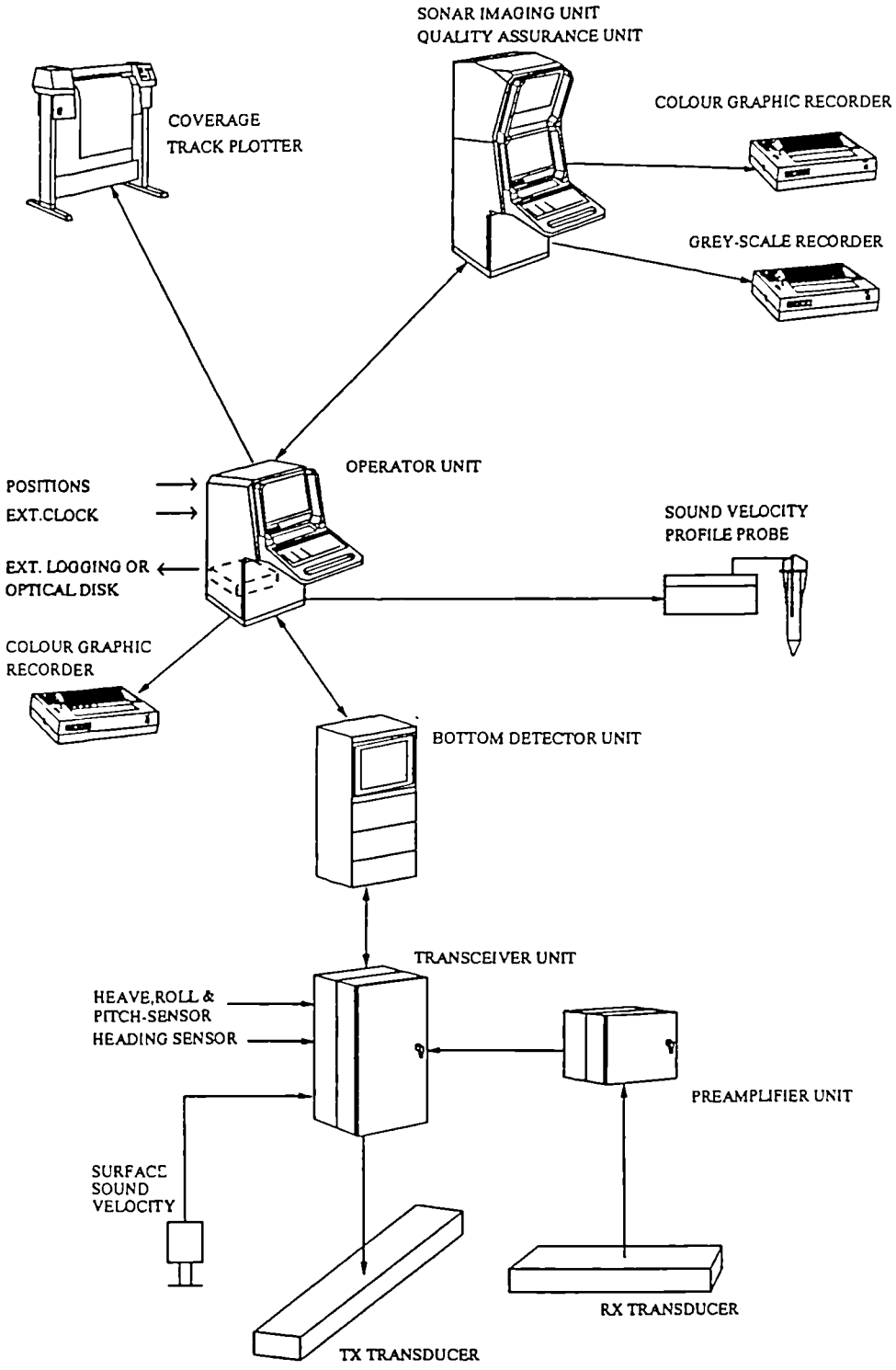


Figure 3.5 *Constructional architecture of Simrad EM12 multibeam echosounder (from Simrad, 1992).*

3.3.2 Theory of operation

The Simrad EM12 is a member of the new generation of multibeam echosounders, providing co-registered bathymetry and sidescan data. The single system EM12 operates at 13 kHz, with full ocean depth capability. The transducer arrays are mounted perpendicular to each other and the transmit beam is roll and pitch stabilized so it always points straight downwards. In the version used on the R.V. Ocean Surveyor, the transmit beam is 1.8° wide along-track and 90° across-track (though a dual system is available, incorporating two sets of transducer arrays, yielding an angular sector of 150°). Two transmit pulse lengths are implemented: 2 ms for shallow waters (<1.6 km), and 10 ms for deep. The reception along-track beamwidth is 18° and across-track is 3.5°, with 81 beams at a beam spacing of 1.1°. Roll stabilization is performed on the receiving beams during beamforming. The range sampling interval is 2.4 metres for the deep operational mode. Hammerstad *et al.* (1991) provide a detailed description of the system design.

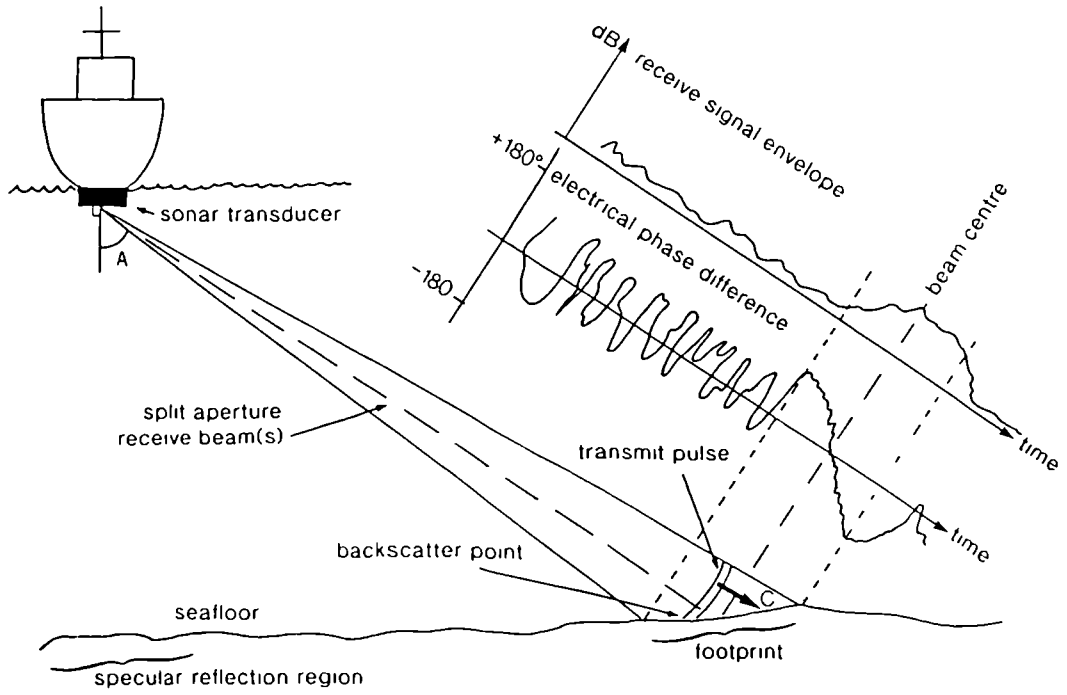


Figure 3.6 Operational principles of Simrad EM12 multibeam echosounder (adapted from Pøhner and Lunde, 1988)

One novel feature of the EM12 is the combined use of amplitude and phase information of the returned acoustic signal for bottom detection purposes (refer to figure 3.6). Conventionally, the amplitude detection method relies on the amplitude

of the energy received during the time period when the transmit pulse hits the seafloor inside the footprint of the beam; this is represented by the receive signal envelope (as in figure 3.2, showing Sea Beam receive envelopes). For specular reflections at near-nadir beams, this receive signal has a well defined leading edge which can be used to measure the travel time in the water column. However at far range, the receive signal envelope becomes stretched and a precise travel time estimate is difficult. The EM12 uses two separate, but parallel, transducer sections to receive each beam; the electrical phase difference is measured between these two sections, tracing the angle of the backscatter point as it traverses the beam. The phase difference between the split receive beams is zero at the centre of the full beam. The viability of this phase detection method collapses at normal to near-normal incidence, where the amplitude detection method is employed.

From the principle of interferometry, once the electrical phase angle (φ) has been measured between the two split receive beams, the geometric angle (A) can be determined (refer to figure 3.6). Pøhner and Lunde (1990) derived this from first principles, leading to:

$$\varphi = \frac{2\pi\delta p}{\lambda} \sin A$$

where δp is the distance between the two sections of the transducer face receiving the beam and λ is the acoustic wavelength of the signal. The accuracy of the measurement of A allows the backscattered energy for each beam sample to be placed in its exact horizontal position on the seabed, thus producing a geometrically correct sidescan image, directly correlated with the bathymetry information.

Hammerstad *et al.* (1991) estimated a standard deviation of depth measurements to be around 0.1% of water depth during test trials of the EM12 system on the R.V. Ocean Surveyor. Some of this testing was performed in similar water depths as encountered at mid-ocean ridges, producing an error of around 3 m in 3 km of water. The precise nature of the recorded backscatter measurements will be described in chapter 5, section 5.3, where the quantitative analysis of the data is explored. Processing steps, for both bathymetry and sidescan data are described in chapter 4; the data formats, in the form of telegrams, are presented in appendix 1.

3.4 THE TOBI HIGH RESOLUTION SIDESCAN SONAR

3.4.1 System design

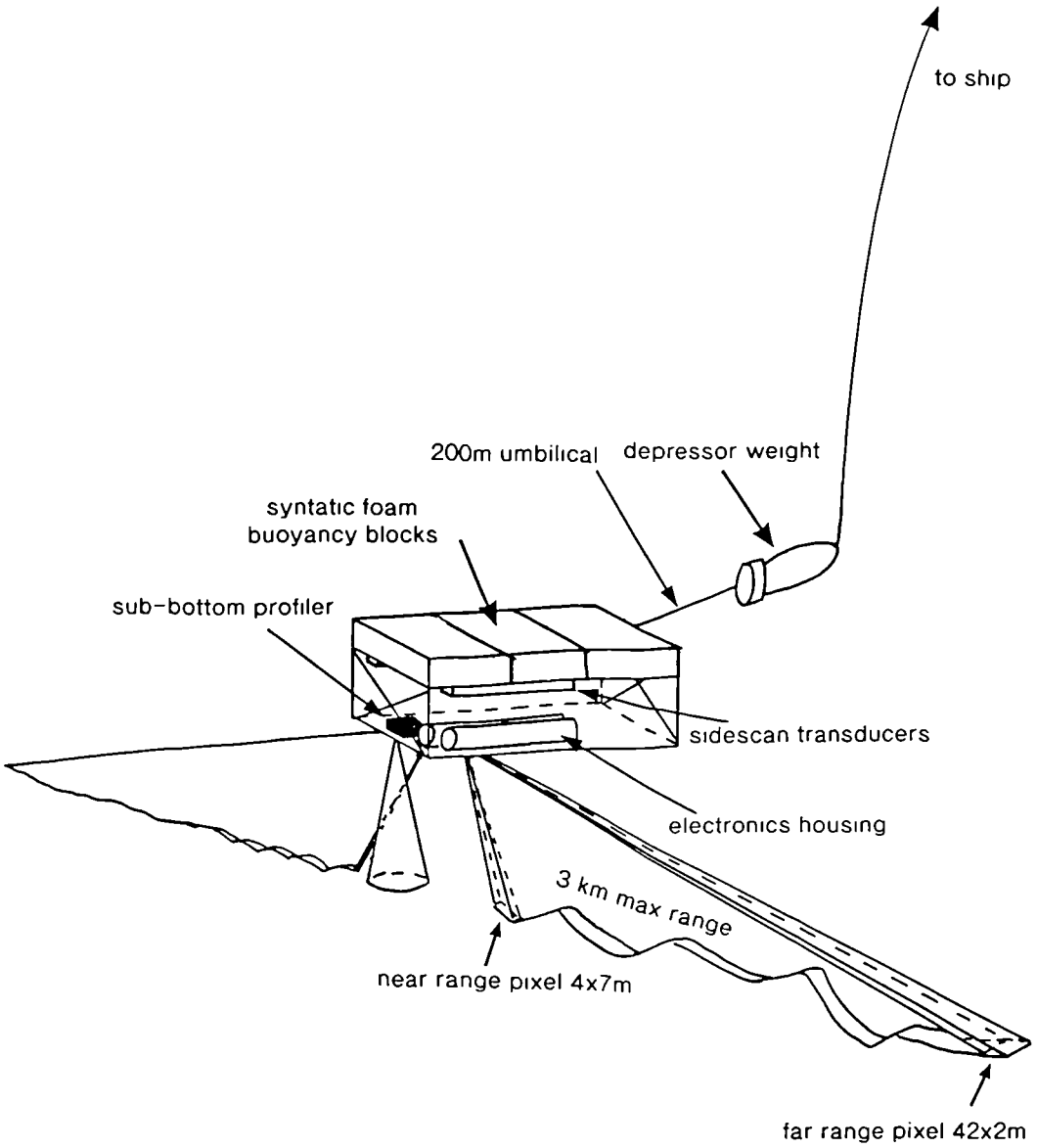


Figure 3.7 Design of TOBI high resolution sidescan sonar (adapted from information in Flewellen et al., 1993).

The TOBI (towed ocean bottom instrument), designed at the Institute of Oceanographic Sciences Deacon Laboratory (IOSDL), U.K., was first launched in

February 1990, and has since been employed extensively for high resolution sidescan surveying and other types of data gathering. The TOBI sidescan instrument operates at 30 kHz and the vehicle is towed at an altitude of approximately 400 m above the seafloor, producing a swath width of 6 km. Flewellen *et al.* (1993) describe the instrument in detail. The resolution, determined by the acoustic beam width of 0.8°×45° (acoustic aperture is 3×0.06 m), combined with a pulse length of 2.8 ms, yields an acoustic footprint varying from 4×7 m close to nadir to about 42×2 m at far range. A time-varied gain (TVG) of $40\log R + 2\alpha R$ (R is range, and α attenuation coefficient of water - 3.65 dB/km at 30 kHz) is applied to the receive signals to compensate for through-water transmission losses.

In addition to the sidescan capability, TOBI houses a 7.5 kHz sub-bottom profiler, which is able to achieve a penetration depth of about 60 m into sediments, a tri-axial fluxgate magnetometer, thermistor, photo-transmissometer and vehicle status sensors (pitch, roll, heading and hydrostatic pressure) (Murton *et al.*, 1992). There is room, both physically and to a limited extent in terms of bandwidth, to include further sensors. The vehicle is 4.5 m long and weighs approximately 2 tonnes in air, made slightly positively buoyant in water using syntatic foam blocks. The vehicle is towed in a two-body configuration, decoupled from the ship's movements by a 600 kg depressor weight. Figure 3.7 displays the mechanical design of the TOBI vehicle and its towing configuration.

The electronic configuration of TOBI, both on the vehicle itself, and within the ship board processing suite, is described in detail by Flewellen *et al.* (1993). The next important step in the TOBI data gathering cycle, from the analyst's perspective, is the nature of the data logging (Le Bas, 1993, describes this; appendix 2 presents the format of each ping). Data is logged ping-by-ping on read-write magneto-optical disks. Each ping format comprises header information including roll, pitch, heading, pressure and altitude measurements; the three magnetometer readings, temperature and transmissometer; then 4000 samples each of port then starboard sidescan imagery (2 bytes per pixel), followed by 4000 (16 bit) sub-bottom profiler samples. This amounts to over 24 kbytes of information per ping, 65% of which is sidescan data. A ping is emitted every 4 s, yielding over 22 Mb of data per hour.

3.4.2 TOBI processing

The large TOBI datasets are unwieldy for the current computer processing power available. For a geometric balance between footprint size across- and along-track, and anamorphic corrections along-track, the data are decimated[1] by a factor of

[1] by sub-sampling every eighth pixel

8 across the swath. For the data analysed in this thesis, the program *tobi_read.f* (appendix 3) was written to process the Reykjanes Ridge imagery, while routines written by Scott Garland (described in Cann *et al.*, 1992) processed the MAR (24-30°N) imagery. Since the original data were logged in MS-DOS format, the 2 bytes of sidescan imagery had to swapped when processed on a SUN workstation , but not when processed on a DEC machine, since the systems operate with different architectures. In both cases, the 16 bit integers containing each sidescan sample were linearly compressed to 8 bits.[1] This compression was effectively made only from 12 to 8 bits, since the analogue to digital conversion procedure in the TOBI electronics results in a 12 bit pixel value; therefore yielding no major loss in information for these analytical purposes. The data are then slant-range corrected, adopting a routine written by P. Slootweg at IOSDL, in both versions of software. Some distortion occurs near-nadir due to the oscillation of depths derived from the profiler record caused by mis-matched sampling rates of the sonar and altitude data; these values should ideally have been filtered first, but on these cruises were not.

Le Bas (1993) has described methods for the comprehensive post-processing of TOBI data. For this work, only the preliminary steps outlined above were performed, since the analyses (texture and Hough transform) presented in this thesis were not critically affected by the artifacts.

[1] over the full range of the data

CHAPTER 4
PROCESSING OF SIMRAD EM12 SONAR DATA FROM THE MID-ATLANTIC RIDGE (45°30'N)

4.1. INTRODUCTION

The Simrad EM12 multibeam echosounder collects both bathymetry and backscatter data (Hammerstad and Pøhner, 1990). Shipboard processing techniques are accessible for the depth information, but no facility was yet available for the presentation of the backscatter data. This chapter describes the processing procedures developed for the display and analysis of the logged backscatter data and also the incorporation of the data into existing image processing packages, whilst retaining easily portable formats. The aim of the processing procedure is twofold. Firstly, a straightforward mosaicked display of the sidescan data is desirable and, secondly, the backscatter and bathymetry data are required for later quantitative analyses (see chapter 5) and so must be processed into appropriate forms. The fact that the EM12 backscatter and bathymetry data are collected simultaneously by the same system, enables them to be precisely geographically registered. Other information such as ship track vectors, or independently collected gravity and magnetic data can also be incorporated into an easily accessible marine database. These data may now be combined for visualisation and interpretation. The creation of such a database, utilising a commercial, 'off-the-shelf' land-based image processing package, may be considered as a type of marine geographic information system.

4.2 AREA

The axis of the Mid-Atlantic Ridge (MAR) has been surveyed extensively using single beam, multibeam and sidescan sonar equipment for the last thirty years or so. In fact, the 45°N area was the first area of the ridge to be thoroughly investigated, since it formed the basis for the Canadian Hudson Geotraverse surveys in the 1960s, where a 1° length of the median valley was examined from 45-46°N, extending 200 km west and 80 km east of the axis (Aumento *et al.*, 1971). Bathymetry, gravity and magnetic data were collected, with about a 5 km line spacing, enhanced by numerous dredge and core samples. Additionally, five seismic refraction and three reflection

experiments were accomplished. The results are summarised in Aumento *et al.* (1971), and specific descriptions of the median valley are given by Loncarevic *et al.* (1966). In 1975, these results were complemented by a GLORIA 6.5 kHz long range sidescan sonar survey (see Laughton and Searle, 1979; Searle *et al.*, 1978). Fowler (1978) presents a structural model for the MAR at 45°N based on seismic refraction data, deducing that no magma chamber was imaged. During Leg 49 of DSDP, Cann *et al.* (1978) identified E-MORB type rocks in the area, typical of a mantle plume environment, although no further evidence for this is suggested as the ridge appears to be a fairly typical piece of MAR, morphologically, and is of normal depth in this area.

4.3 PREVIOUS PROCESSING WORK

It is only very recently that sonar instruments capable of simultaneously collecting bathymetry and sidescan data have become available. de Moustier and Matsumoto (1993) and chapter 3 of this thesis describe the principles and potential uses of these systems, which include multibeam echosounders with backscatter recording capabilities (e.g. Talukdar *et al.*, 1992; Hammerstad *et al.*, 1991) and bathymetric sidescan sonars (e.g. Blackinton *et al.*, 1983). Malinverno *et al.* (1990) and others present processing methods for the SeaMARC II bathymetric sidescan sonar. However, details of the processing of the new generation multibeam data are only beginning to be published at the time of writing of this thesis.

Multibeam bathymetry data alone have been digitally processed for the last few years and extensive literature has been devoted to the processing of traditional sidescan datasets. In this section, the processing techniques of some of the independently collected data will be described, followed by an as up-to-date as possible account of the recent dual data processing systems. The latter should not necessarily be classed as 'previous work' to mine as we have all been working on these problems simultaneously, and I have found out about how other parties process their data during and after I developed the processing steps for this EM12 mid-ocean ridge dataset.

4.3.1 Multibeam bathymetry processing

Digital processing of multibeam bathymetric data has been undertaken for many years now, principally with the aim of producing bathymetric contour maps. Centres such as the Ocean Mapping Development Center, University of Rhode Island (URI),

Lamont Doherty Geological Observatory (LDGO) and Scripps Institution of Oceanography have all developed their own processing systems for these purposes. Slootweg (1978) and Ware *et al.* (1991) describe some of the filtering, statistical and gridding methods available for dealing with these tasks. Further processing advances dealing with the visualisation of bathymetric datasets have been accomplished by Edwards *et al.* (1984) and Miller (1991). Edwards *et al.* (1984) display shaded relief representations of bathymetric data, while Miller (1991) overviews some of the state of the art advances in three dimensional visualisations. More recent progress in this field has dealt with processing data from multibeam sonars with backscatter recording capabilities, so this will be dealt with in section 4.3.3.

4.3.2 Sidescan processing

Since 1980, when GLORIA II long range sidescan sonar data became available in a digital format (Searle *et al.*, 1990, describe the chronology of these advances), and also SeaMARC II data (e.g. Malinverno *et al.*, 1990) began to be collected, comprehensive digital image processing of sidescan data has become widely possible. Extensive GLORIA image processing was undertaken by Chavez (1986) and incorporated into the USGS Mini Image Processing System (MIPS), run within a VAX/VMS operating environment. Chavez (1986) describes all the standard first-order steps required to process a sidescan image. He categorised these into two main sections: (1) geometric corrections, and (2) radiometric corrections. Geometric corrections alter the position of pixels in an image, and sidescan geometric processing requirements include compensation for the water column offset, slant-range to ground-range projection and adjustment for aspect or anamorphic distortions due to across- and along-track sampling differences and vehicle velocity variations. Radiometric correction routines adjust the grey level of pixels within an image and for sidescan purposes are necessary to apply shading corrections across swaths due to power drop-off from near- to far-range, and for the removal of noise, such as speckle and striping. Once these stages have been accomplished, Chavez (1986) presents various methods for information extraction, including many standard image processing filtering tasks for structural enhancement, plus methods for radiometric contrast stretches etc. (refer to one of standard image processing texts for details, e.g. Pratt, 1978).

Searle *et al.* (1990) present an overview of GLORIA image processing to that date, extending Chavez' (1986) work and including details of research performed by Mitchell and Somers (1989) and Mitchell (1991). Mitchell (1991) improves geometric corrections of GLORIA data through combination with independently

collected Sea Beam bathymetry to reduce the effects of artifacts such as layover and foreshortening; while Mitchell and Somers (1989) and Mitchell (1991) present the techniques and benefits of computing GLORIA backscatter strengths. Searle *et al.* (1990) review revised methods for across-track shading corrections, mosaicking and combination with Sea Beam bathymetric data, together with visual composites of the two datasets.

Other sidescan image processing texts have included the work of Reed and Hussong (1989). They describe geometric and radiometric corrections developed for processing SeaMARC II sidescan data; these covered accurate slant-range to ground-range conversions using SeaMARC II's co-registered bathymetry information, a 'point migration' routine to reduce anomalous effects caused by the variable beam footprint size from nadir to maximum range, and across-track shading corrections similar to Chavez (1986). Malinverno *et al.* (1990) present processing routines developed for both SeaMARC I and II systems, involving initial demultiplexing, decimation and noise removal; and radiometric corrections, such as histogram equalisation methods, shading corrections and destriping (section 4.6.4). Further work also using the bathymetric capabilities of the shallow-towed SeaMARC II system will be considered in section 4.3.3.

More recent and specific sidescan processing work has been published by Searle (1992b), Mason *et al.* (1992), Cobra *et al.* (1992) and Cervenka and de Moustier (1993). Searle (1992b) presents a method for filling in the missing GLORIA nadir data using pseudo-sidescan imagery generated from HYDROSWEEP multibeam bathymetry data (the HYDROSWEEP swath width is fortunately approximately the same size as the GLORIA near-nadir region). Mason *et al.* (1992) introduce a novel 'deblurring' method applied to refine the along-track dimension of GLORIA imagery. Cobra *et al.* (1992) present methods for correcting geometric distortions of high resolution sonar imagery due to motion instabilities of the sonar towfish. They derived a mathematical model for these distortions and reconstructed the sampling pattern on the seabed. Cervenka and de Moustier (1993) have recently published some excellent techniques for the improved processing of SeaMARC II sidescan images, resulting in very impressive noise removal from a heavily degraded swath of data, using a Chebyshev approximation to identify noise in the image. They also describe similar histogram equalisation techniques for matching up the grey levels of swaths within a mosaic as those developed in this thesis (section 4.6.4).

4.3.3 New generation of co-registered bathymetry and sidescan processing

The co-registered bathymetry and sidescan instruments are divided into interferometric sidescan sonars with bathymetric facilities and multibeam echosounders capable of recording amplitude data (chapter 3, section 3.2.3). Literature regarding the processing of the former are predominantly concerned with data from the SeaMARC II system (e.g. Malinverno *et al.*, 1990; Reed and Tucholke, 1991; and Cervenka and de Moustier, 1993); however, since the demise of new SeaMARC II data (after the instrument was lost at sea), Davis *et al.* (1993) present an outline of a bathymetric and sidescan processing system developed for MR1 (which replaced SeaMARC II) imagery, and also for the SEAMAP towed mapping system data. The two widely publicised multibeam echosounders successfully recording amplitude data are the Simrad suite (EM100, 1000 and various EM12 systems) and Sea Beam 2000 (chapter 3). Published processing texts at the stage of writing (1993) are rare: Simrad processing, of principally shallow water data to date, has been extensively developed at the University of New Brunswick (Hughes Clarke *et al.*, 1993); Mitchell (Mapping attributes of multibeam sonar data, submitted to *Marine Geophysical Researches*, 1993) has developed a suite of routines to process multibeam bathymetry and backscatter data; and Keeton and Searle (1993) (and in this chapter) present methods to achieve similar ends. Talukdar and Tyce (1990) describe routines for processing Sea Beam amplitude data, though since the advent of the new Sea Beam 2000 system (Talukdar *et al.*, 1992), the relevant processing results are beginning to appear (e.g. Miller *et al.*, 1993; and Mayer *et al.*, 1993). Even while writing, I am beginning to hear about further work presented at conferences, but as yet unpublished, so this review of processing can only be considered to be complete regarding papers obtained by December 1993.

Regarding bathymetric sidescan sonar processing first, some specific sidescan imagery related steps from these data have been described in section 4.3.2, but details of procedures making specific use of the co-registered nature of the systems shall be presented here. Malinverno *et al.* (1990) were able to plot SeaMARC II imagery on a UTM projected geographic grid, with gaps interpolated according to inverse square distance weighting of the values at the vertexes of a surrounding polygon. Cervenka and de Moustier (1993) positioned their amplitude samples similarly. Reed and Tucholke (1991) used SeaMARC II bathymetry to accurately slant-range correct the co-registered sidescan imagery, though it was necessary for them to apply advanced processing skills initially to the heavily noise corrupted bathymetry data. Davis *et al.* (1993) overview their new UNIX-based processing

system, written both for acquisition and processing purposes, using fast, visual interfaces.

The multibeam bathymetric processing suite developed by Hughes Clarke *et al.* (1993), Ware *et al.*, (1991) and Wells *et al.* (1991) enables data investigation both in a line-oriented and geographic-oriented mode, with the bathymetry information always complemented by the corresponding backscatter data. Talukdar and Tyce (1990) describe the processing of sidescan images generated from a 'conventional' Sea Beam multibeam system equipped with special acquisition hardware. The geometric corrections comprise positioning the amplitude samples across-track by interpolating a cubic spline through the beam centres, and the along-track correction is made in the same manner as traditional sidescan systems. Talukdar and Tyce (1990) performed Chavez' (1986) column averaged shading and destriping procedures for the radiometric corrections. Mosaicking through the transformation of line and sample amplitude coordinates to geographic coordinates was accomplished, and overlapping regions were mapped as the average of the digital numbers (DNs). Very recently, Mayer *et al.* (1993) and Miller *et al.* (1993) have begun to present Sea Beam 2000 processing results: Mayer *et al.* (1993) present interactive fly-throughs of Sea Beam 2000 data, while Miller *et al.* (1993) show combined Sea Beam 2000 bathymetry and backscatter data from the East Pacific Rise.

4.3.4 EM12 multibeam processing

Simrad have generated their own processing systems for the EM12 data, though at the time of collection of these data, no backscatter capabilities were available. Midthassel *et al.* (1988) and Pøhner (1988) describe the processing facilities available on the R.V. Ocean Surveyor during the two 45°N MAR surveys studied in this thesis, which were used to process the bathymetry data to a degree (refer to section 4.4 and 4.5). The processing steps are fairly standard for the bathymetry data: noisy data can be filtered using surface fitting and rejecting depth data lying more than say 2 standard deviations from this surface; gridding is effectively nearest-neighbour, based on a spiral search principle.

4.4 DATA ACQUISITION

Two separate cruises aboard the R.V. Ocean Surveyor, operated by Worldwide Ocean Surveying Ltd., provided these data. The first, in December 1990, formed a

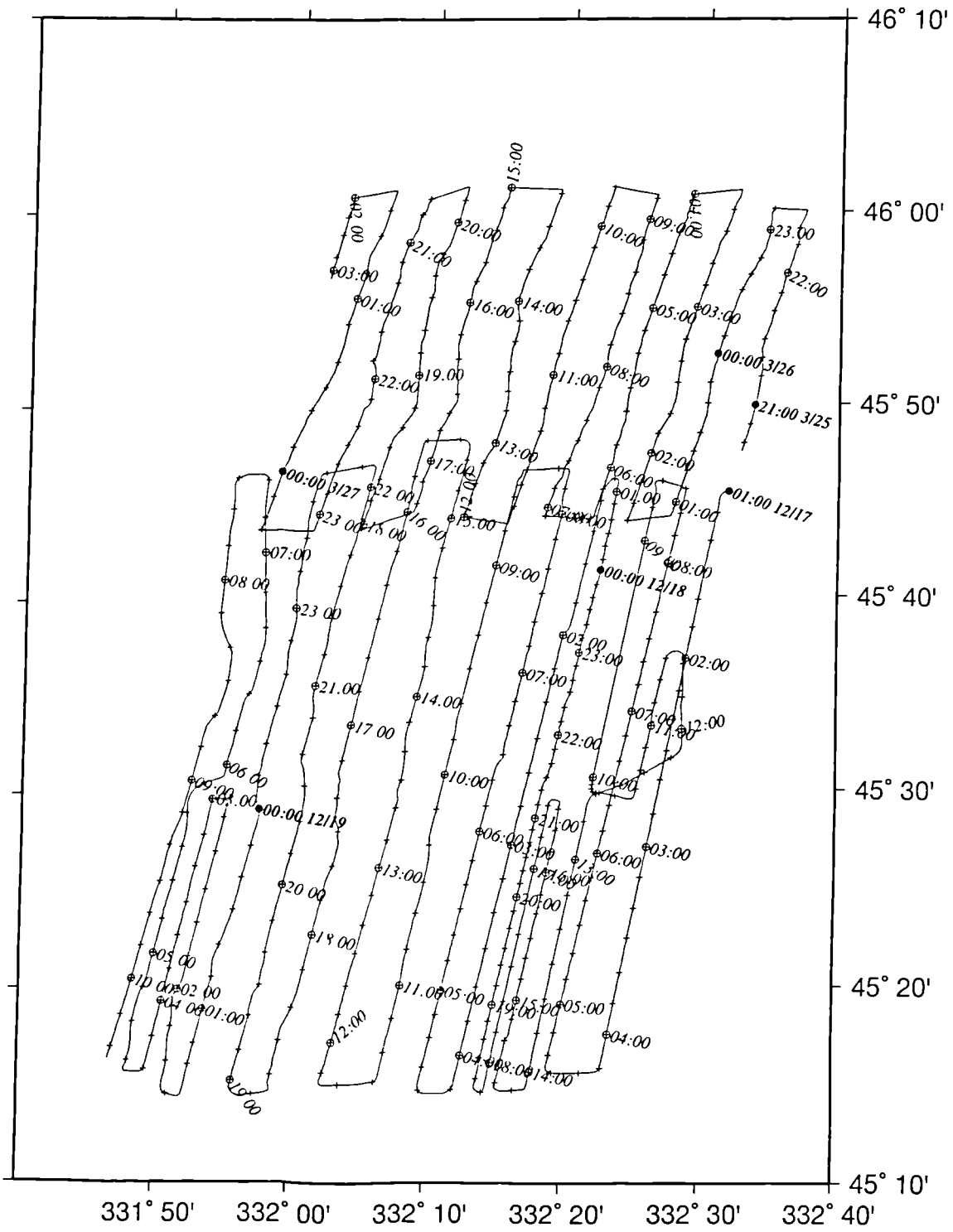


Figure 4.1 *Ship track plot for December 1990 (southern) and March 1993 (northern) surveys, with UTM projection. Tickmarks are positioned once every 10 minutes, with hourly and daily points labelled.*

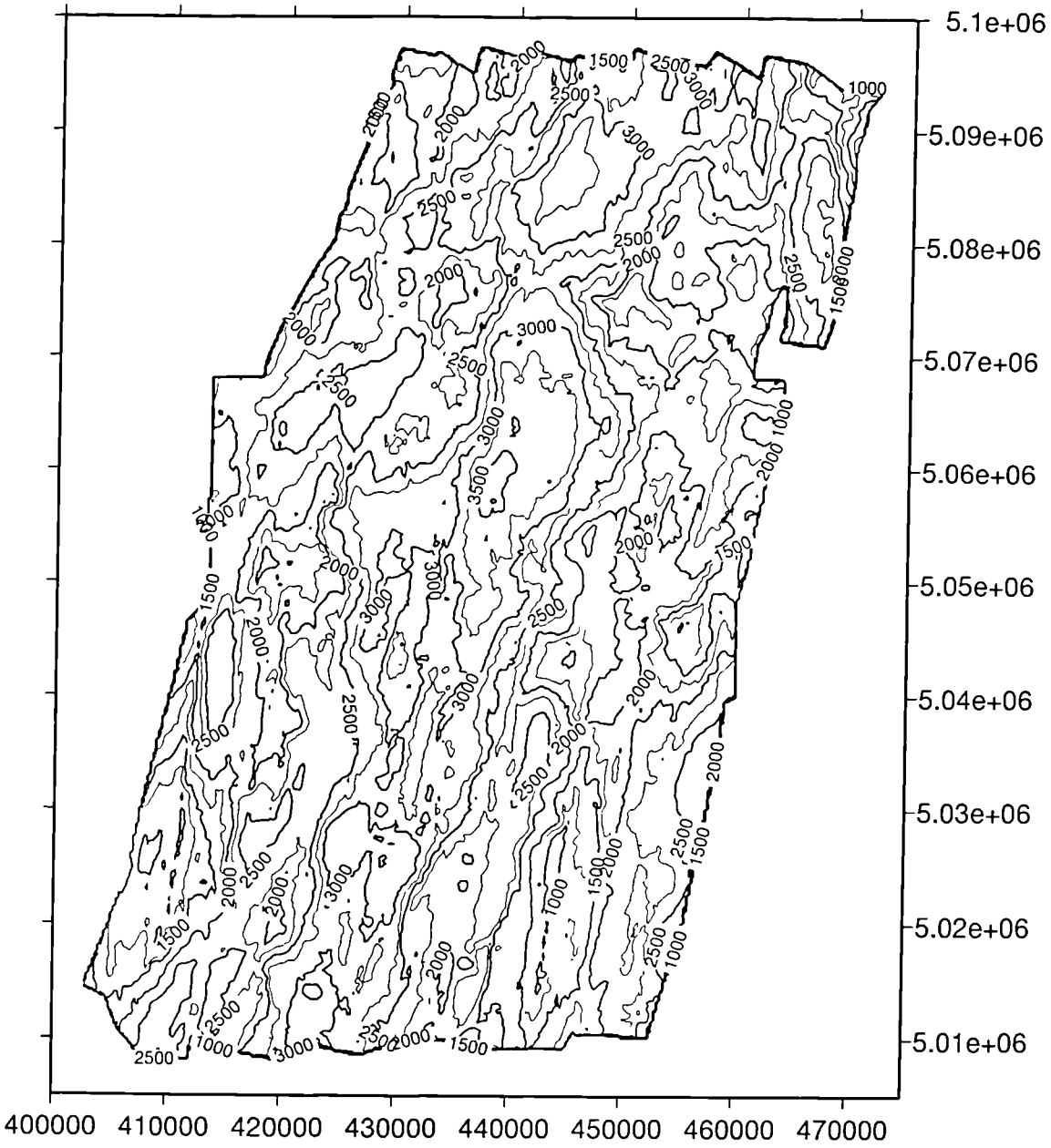


Figure 4.2 *Bathymetry plot for both surveys: contour interval of 250 m; UTM projection; scale of 1:625 000.*

half degree square, centred on the MAR axis at 45°30'N, encompassing some 3000 km² (figure 4.1). Surveying began in the north east portion of the area, with tracklines oriented 013° and parallel to the ridge axis. Here, we were experiencing a Force 8 gale, where the ship could only manage a speed of 4-5 knots. Conditions had improved for the western part of the survey, where the maximum ship speed of 10-11 knots was achieved. The second survey was undertaken in March 1993, extending the dataset northwards to 46°N, and to the same distance off-axis as the first survey. This added an extra 1500 km² to the dataset. The weather conditions were again marginal, ranging from Force 5 to 7, with a ship speed of approximately 8 knots, and tracklines oriented at 019°.

During both cruises, the EM12 operated with a pulse length of 10 ms and a ping rate of some 500 pings per hour. GPS navigation was almost continually available, with generally better than 23.5 hours per day coverage. The EM12 operating system archives data on WORM ('write-once, read-many-times') optical disks in the form of data 'telegrams', each one comprising a particular aspect of the data collected, e.g. depths, positions, sound velocities, backscatter etc. These were identified in their logging format by a flag, specific for each telegram type. Figure 4.1 illustrates the ship tracks from both surveys, while figure 4.2 presents a bathymetry contour plot of the combined datasets.

4.4.1 Depth and position data

From the December 1990 survey, the depth and position telegrams were archived together in files containing approximately one hour's worth of data, denoted, for example, as '*de181726.04*' where '*de*' refers to the month, '*18*' the date, '*17*' the hour and '*26.04*' the number of minutes at the time of creation of the file. The raw position telegrams comprise geographic and UTM coordinates at approximately one minute time intervals, while the raw depth telegrams consist of header information, such as gyro compass ship headings, and then the depth values located in metres across- and along-track from the ping origin. The bathymetry data were partly processed on board ship using the *IRAP* processing package, the principles of which are described in Pøhner (1988), although further post-cruise editing of the ungridded 'xyz' files was required due to excessive noise in the data (see Valsami, E., J.R. Cann, R.C. Searle, M. Ackers, and J.A. Keeton, Morphological characteristics of the Mid-Atlantic Ridge at 45°N: implications for magmatic and tectonic processes, manuscript in preparation, 1993). The 'xyz' files comprise each depth value located with its corresponding UTM easting and northing. The three forms of bathymetry data acquired were the raw depth telegrams; ungridded 'xyz' files, following combination

with navigation information held in position telegrams; and finally, a digital grid of the bathymetry generated from the edited 'xyz' files using the gridding routine *surface* in the *GMT* package (Smith and Wessel, 1990), retaining the *GMT grd* format. Figure 4.3 clarifies both the data types acquired in their raw telegram format and after shipboard (and post-cruise) processing, by WOSL personnel and E. Valsami (University of Leeds), respectively. The data are gridded at a 100 m interval, with UTM bounds: north west 400000E, 5100000N; south east 470000E, 5005000N. Further processing techniques developed for the depth and position data are described in section 4.5.

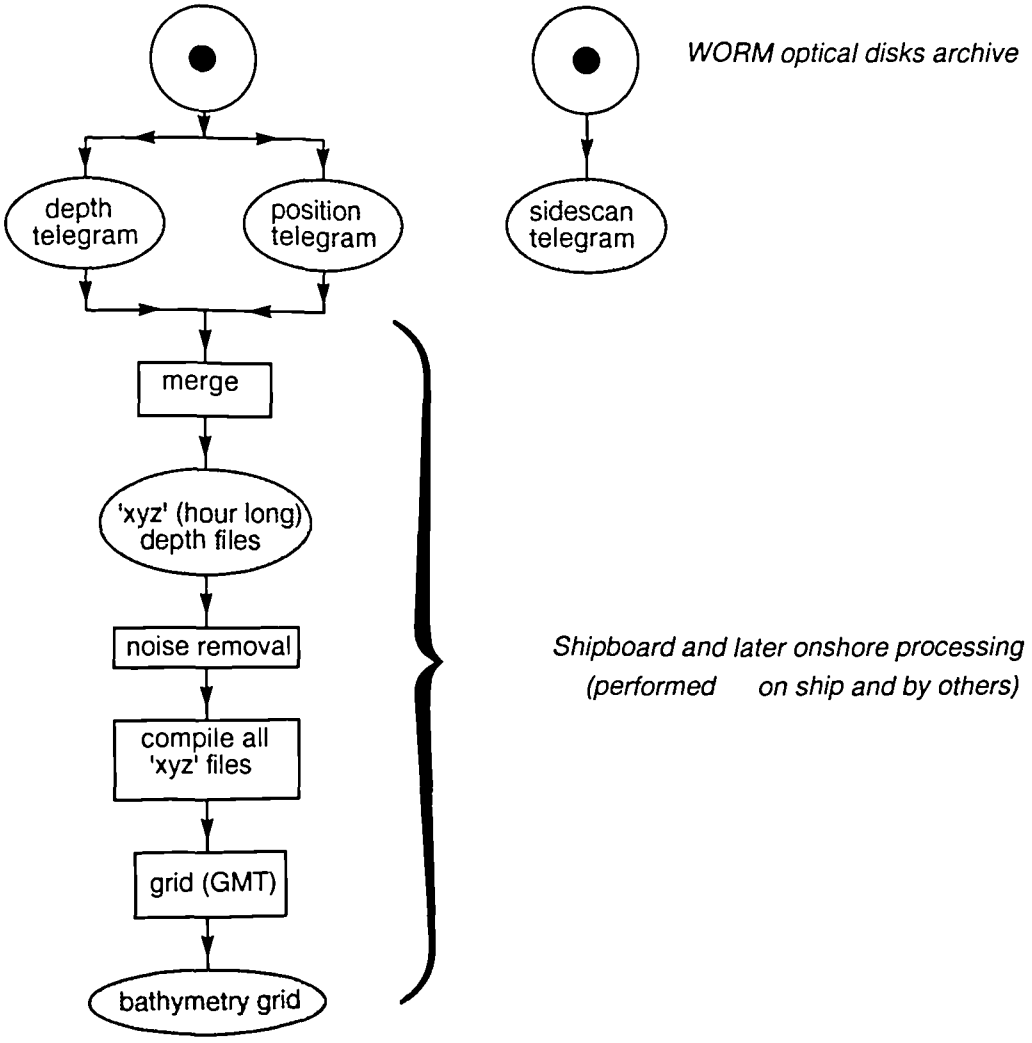


Figure 4.3 *Flowchart summarising processing steps performed before processing developed in this thesis, including form of data logged on optical disk.*

Similar data formats were acquired from the March 1993 survey, though all the different types of telegram were stored within one file on this occasion. Gridded and

'xyz' format files of the bathymetry were generated onboard ship using the *IRAP* software again.

The bathymetry grids from the two surveys were combined using *GMT* routines (Smith and Wessel, 1990). Figures 4.2 and 4.23 show how closely the navigation and depth measurements matched between the two surveys. The join is at UTM northing 468000.

4.4.2 Sidescan data

The EM12 sidescan data from the December 1990 survey were only available in a raw telegram format, with each telegram comprising approximately one hour's worth of data, with around 500 samples per ping and being about 1 Mbyte in size. Backscatter samples are measured in decibels and are located in metres across- and along-track from the ping origin. Section 4.6 describes the processing techniques developed for the sidescan data.

The sidescan telegrams should have been named in the form of *'ds181737.06'*, for example. This format is similar to that used for the files comprising the position and depth telegrams. However, the file names acquired from this survey comprised the two date characters replaced with the year, e.g. *'ds901737.06'*. These were renamed and processing ran smoothly, except on one occasion where files from two different days must have been created at the same time because the sidescan telegram clearly exhibited two separate dates for the data records. These data were separated into two new files, but the initial overwriting of some of the data when the data was archived by the EM12 operating system resulted in a large gap in the data in the north west portion of the survey area (see figure 4.17).

The form of logging the backscatter information changed significantly for the March 1993 survey. Instead of positioning every amplitude sample in metres relative to the ping origin, information about particular beams was stored in the telegram header, such as beam number, frequency, number of amplitude samples recorded within the beam and the number of the beam centre sample; then for each beam, an array contained the recorded amplitudes. Effectively, a sampled amplitude trace for each beam is now retained (similar to figure 3.2). No positioning information was stored in this telegram apart from the beam number. The sidescan telegrams were logged in the same file as the depth and position data for this survey.

4.5 PROCESSING OF DEPTH AND POSITION TELEGRAMS

In order to begin processing the sidescan telegrams, the relevant information from the position and depth telegrams is required in an easily accessible format. Additionally, for further quantitative analysis of the bathymetry and sidescan datasets, a tiled image format is preferred (i.e. one where the ship track is represented by a straight line and the across-track dimension is normal to it), so tiled output of both depth and beam angle data, the latter for true angle of incidence calculations, is also necessary.

4.5.1 December 1990 data

The programs *sim_nav.f* and *sim_depth.f* (appendices 4 and 5, respectively) were written to process the depth and position telegrams from the December 1990 survey. *sim_nav.f* dealt with the position telegrams, outputting ASCII representations of the navigation data. These were subsequently compiled into direct-access, day-long databases, after filtering to remove any navigational inaccuracies. The navigation data were also converted into formats suitable for incorporation into *GMT* (Wessel and Smith, 1990) and *EASIPACE* (PCI, 1988) processing packages. *sim_depth.f* processed the depth telegrams, reserving the header information which was then collated into daily databases. The depth data for each file were merged with the corresponding day's navigation database. The reference used for merging was to convert all times into the number of seconds since 0000Z on 1 January of the particular year in question. Each ping origin now possessed a UTM position, so a tile format file could be created. An 'xyz' ungridded tile file was therefore produced from *sim_depth.f*, where 'x' and 'y' refer to the position of each value in metres from the image origin, while 'z' respectively refers to depth in metres. Alternatively, additional subroutines can enable 'z' output as beam angle in degrees, for example. The orientation and subsequent gridding of such tiled images is described in section 4.6.1, utilising the same routine as the sidescan data, with gridded floating-point binary output. All files produced during the processing steps were archived on optical disks, and figure 4.4 illustrates the processing steps for the depth and position data.

4.5.2 March 1993 data

Since the logging formats had changed for the March 1993 survey, it was easier to adapt a suite of software that had been written since December 1990 by Mitchell (Mapping attributes of multibeam sonar data, submitted to *Marine Geophysical Researches*, 1993). For this application, Mitchell's bathymetry processing capability

was utilised and amended to output beam angle and therefore angle of incidence data (see chapter 5, section 5.3 for applications).

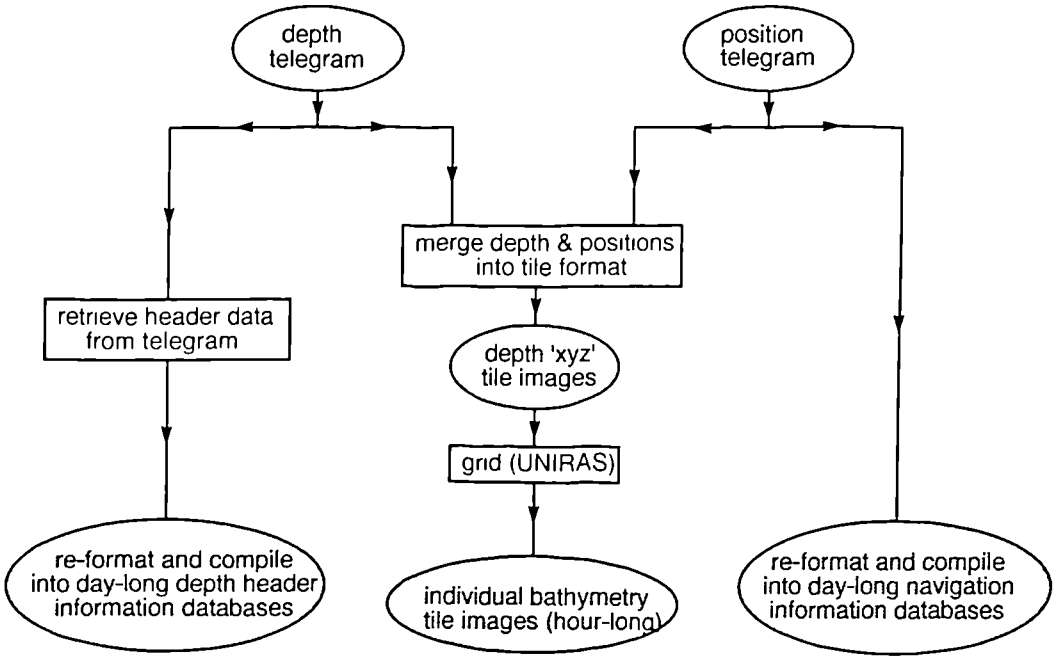


Figure 4.4 *Depth and position telegram processing flowchart.*

4.6 SIDESCAN DATA PROCESSING

4.6.1 Telegram to image conversion

4.6.1a December 1990 data The first stage of the sidescan processing of the December 1990 data was to convert each sidescan telegram into both a traditional sidescan tile image and an exact geometrically correct image. For both image types, the UTM coordinates and other header information were determined for each ping origin through merging with the navigation databases. This merging operation was again performed using the time format described in section 4.5. UTM coordinates were used partly for ease of communication between *IRAP* and my programs, and because the UTM projection provides an isotropic reference frame in rectangular coordinates. Figure 4.5 summarises the sidescan processing steps. The derivation of the empirical beam pattern is described in section 4.6.2.

The sidescan tile image begins with the first ping of the telegram located at the top of the image and the most recent ping at the bottom; correspondingly, port returns are presented on the right hand side and starboard on the left. The image

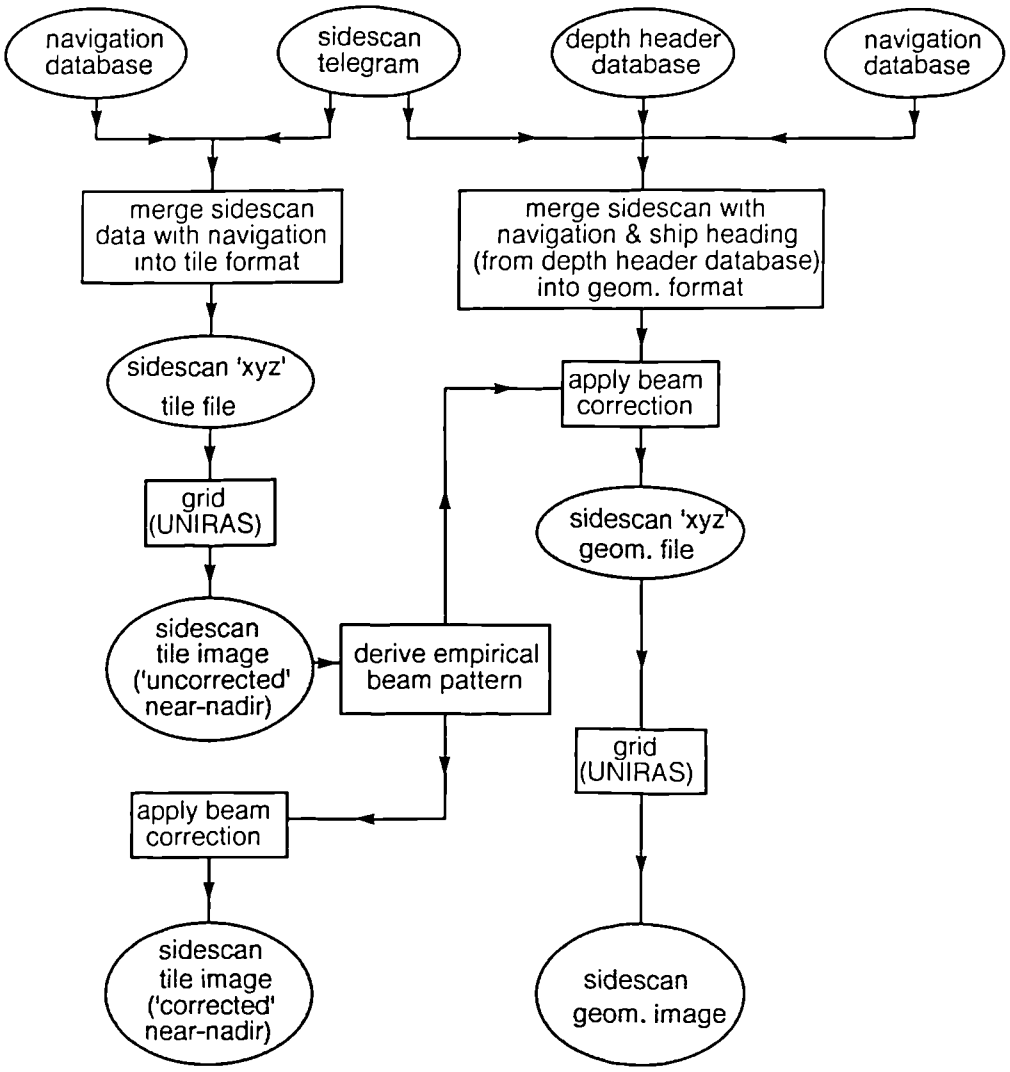


Figure 4.5 Sidescan telegram processing flowchart.

origin is located at the centre beam of the first ping and all subsequent backscatter samples of the image are measured in decibels and located in metres from this origin. Figure 4.6 illustrates the geometry of these tiles. The same principle was used for the depth tiles (section 4.5). The distance between pings is determined according to how far the ship has travelled, derived from the UTM coordinate difference between adjacent pings. This format is chosen particularly for ease in later quantitative analysis, where properties such as across-track profiles are important (see section 5.6.3, chapter 5). Figure 4.7 presents an example of such a sidescan tile image. The program *sscanread_tile.f* (appendix 6) converts the original sidescan telegram into an 'xyz' ungridded tile; header information is retained separately.

The geometrically correct images are generated for mosaicking purposes, using a similar approach to Malinverno *et al.* (1990). The UTM position (from the navigation database) and gyro compass heading (from the depth header database) for

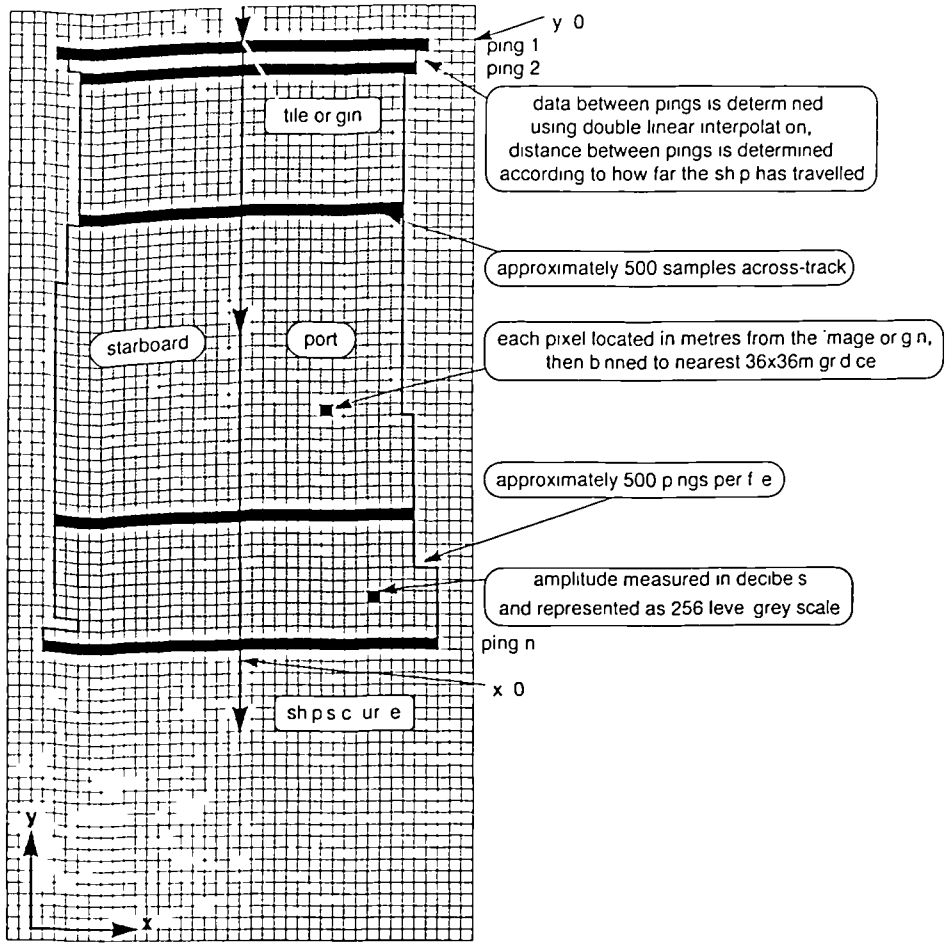


Figure 4.6 Geometry of tile format images.

each ping origin are used to interpolate the UTM coordinates for every backscatter sample across-track. The resulting ungridded 'xyz' files comprise backscatter values measured in decibels, corrected for the empirical beam pattern derived in section 4.6.2, and the corresponding UTM eastings and northings. Figure 4.8 illustrates the geometry and an example is illustrated in figure 4.9. The program *sscanread_geom.f* (appendix 7) performs these processing steps.

4.6.1b March 1993 data Due to the major logging format change of the sidescan telegrams, software written by Mitchell (Mapping attributes of multibeam sonar data, submitted to *Marine Geophysical Researches,* 1993) was adapted to output approximately all amplitude samples recorded with a UTM projection, in the same 'xyz' format tile and geometrically correct images. Mitchell positioned the amplitude samples by projecting the samples onto a straight line with an across-track slope determined from adjacent soundings. Since the angular sweep of the instrument is small, the across-track distance from nadir can simply be computed knowing the beam number, sample number within the beam, sampling interval in slant range, beam

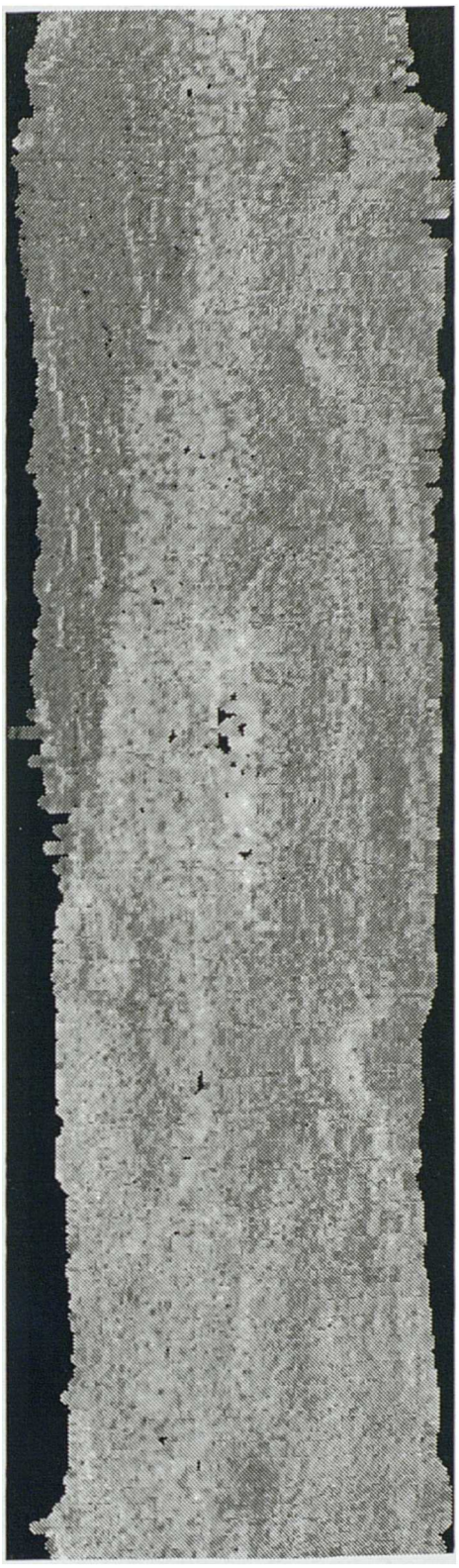


Figure 4.7 *Example of a tile format sidescan image, before empirical beam correction. Width ~6.3 km, length ~21.6 km; white corresponds with a backscatter strength of -0.5 dB, black to -64 dB. Note the linear fault scarp at the top left, and circular seamount towards the lower right limit.*

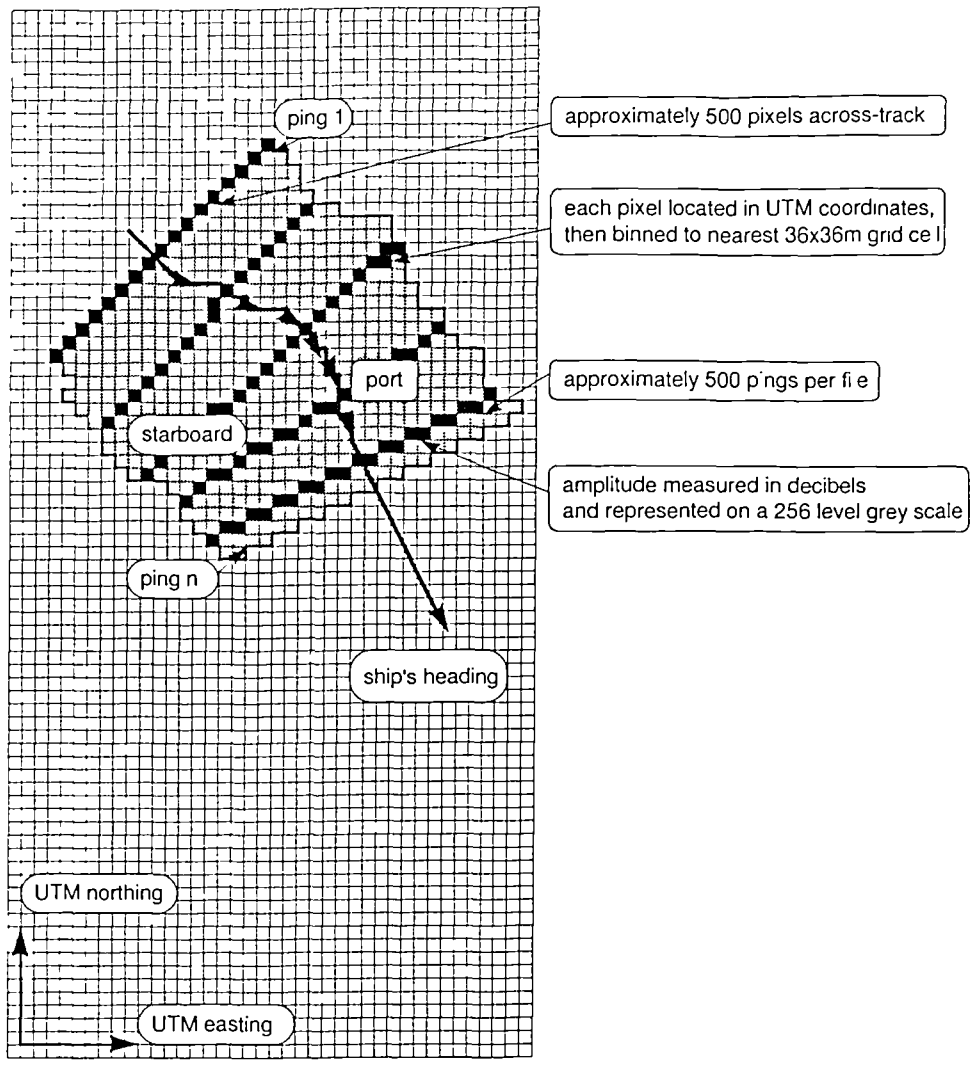


Figure 4.8 Geometry of geometric format images.

Figure 4.9 *Example of a geometric format sidescan image (same data as figure 4.7). Width ~11.2 km, length ~22.0 km.*

inclination angle and across-track slope. Full resolution sidescan data, in fact sampled to no more than 1024 samples per ping (which was generally the case), were output to 'xyz' files.

Following these principles, one tile and one geometric 'xyz' file are generated for every sidescan telegram, each approximately 3-4 Mbytes in size, since existing display packages, such as *UNIMAP*, *GMT* etc., require 'xyz' data to be held in ASCII floating point files. A day's worth of processed sidescan data would be of the order of 100 Mbytes in size due to this bulky format, so clearly the size of many days' worth of files becomes a problem. Therefore, the files are gridded at this stage, employing the gridding routine *'GINTP1'* from *UNIRAS* software, incorporated into *sim_grid.f* program (appendix 8). The 'xyz' data are binned to their closest grid nodes, and double linear interpolation is performed to fill in some of the gaps, according to a specified search radius. A 36 metre cell size is selected here, and a search radius of 72 metres. Floating point binary files were output. Figure 4.10 summarises the uses of the various file types.

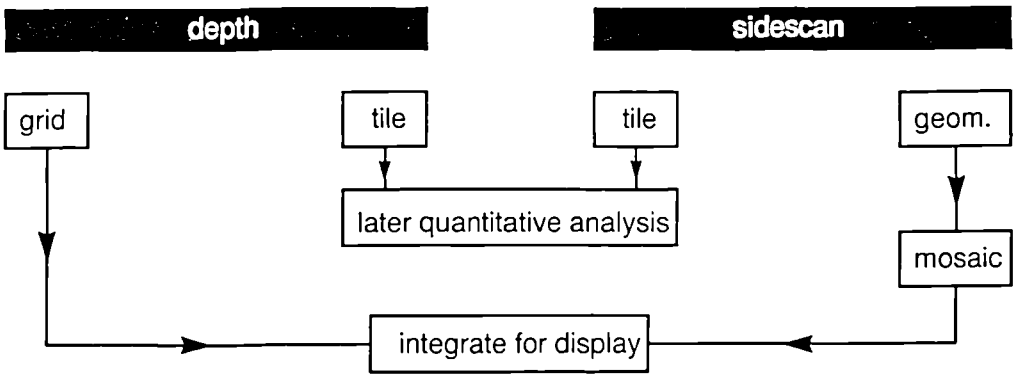


Figure 4.10 *Schematic representation of future applications for different depth and sidescan data formats.*

4.6.2 Derivation of Empirical Beam Pattern

The inner 20 degrees of near-nadir backscatter data are unreliable. Specular reflections near-nadir can provide important information about the scattering characteristics of the seafloor material (Chotiros, 1993); however, it is not known precisely what gain corrections the Simrad software has applied to these data, since this information is not logged, so the decision was made to eliminate these bright reflections. Although the sidescan telegrams provided by the EM12 operating system had already had a correction applied for beam effects, a bright stripe remains near-nadir in the December 1990 data (figure 4.7). An attempt has been made to reduce

this 'stripe', purely for aesthetic purposes, in order to reduce the visual distraction of the artifact from the actual data in the final mosaic. The remaining 'beam pattern' is therefore derived through the selection of average across-track profiles generated from six of the best quality sidescan tiles, representing 10% of the total number. A simple correction function, displayed in figure 4.11, is determined by combining these profiles, averaging and filtering.

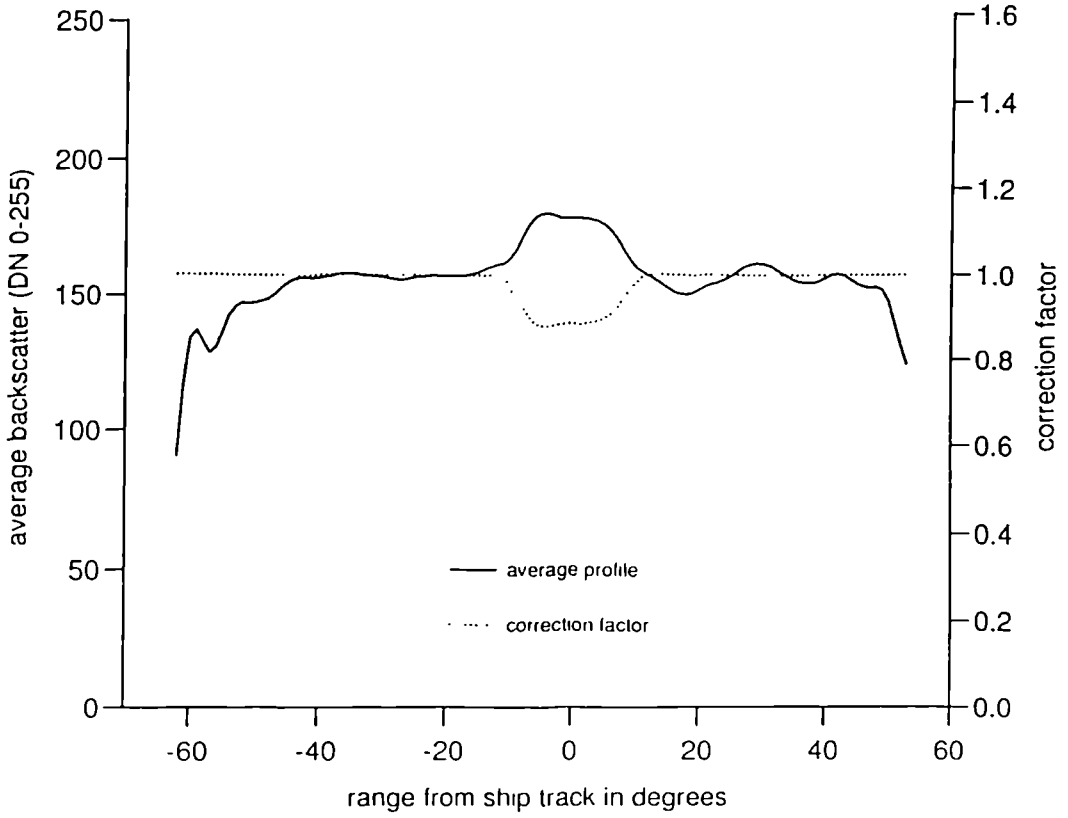


Figure 4.11 Average across-track backscatter profile, and corresponding correction function (for ± 10° from nadir only).

The near-nadir data from the March 1993 survey looks different from the December 1990 data because a revised EM12 operating system appears to have altered the time-varying gain (TVG) correction in this region. The near-nadir region here is characterised by a more 'patchy' nature, suggesting that the TVG fluctuates more rapidly, adjusting itself continually according to the backscatter strengths received (refer further on to figure 4.17). This effect is much more difficult to remove than by simply applying the correction function of figure 4.11, though without knowing the gain information, it is the best that one can do. The near-nadir brightness fluctuations do remain, but are dampened somewhat. Further descriptions of these characteristics are included in section 4.9.2.

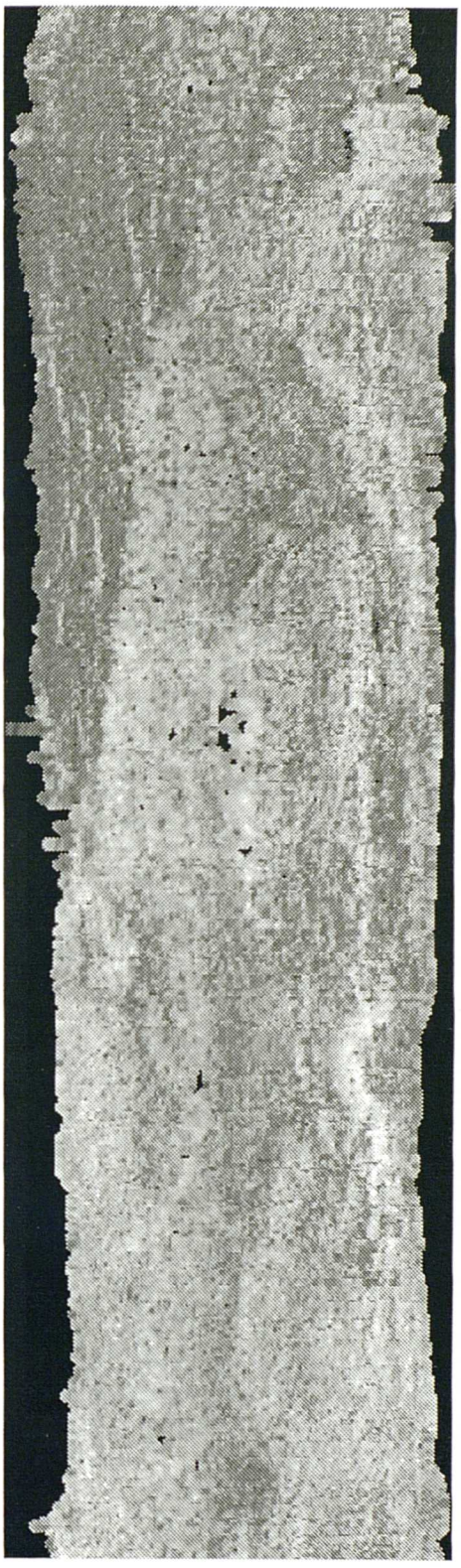


Figure 4.12 *Example of sidescan tile image after near-nadir empirical beam correction.*

The near-nadir correction function was therefore applied during the processing of all tile and geometric sidescan images. Figure 4.12 illustrates the result of applying the correction to the sidescan tile in figure 4.7. The near-nadir region is avoided for the later quantitative analysis.

4.6.3 Mosaicking

This procedure utilises the geometrically correct sidescan images and the *EASIPACE* image processing package (PCI, 1988). Associated geometric beam angle data are also mosaicked. Since these images have already been interpolated onto a uniform grid surface, each individual geometric image can be immediately placed according to its known UTM limits onto a basemap, encompassing the entire survey area. In order to combine all the geometric images from the survey, they must therefore be placed in turn on this final basemap. For speed, the whole process can be performed automatically where the most recent swath always overlaps the previous adjacent one, or vice versa. Talukdar and Tyce (1990) selected the average DN of overlapping regions for similar data, while Hughes Clarke *et al.* (1993) mosaicked beam angles simultaneously (like here) and determined the cut-lines automatically by choosing the backscatter value with the associated minimum beam angle. However, due to the poor quality of the dataset used here, it was essential to manually define a 'cut-line' between two overlapping swaths through an assessment of data quality and clarity of features. This procedure takes significantly longer. The two survey datasets were mosaicked separately since the first had already been completed before the second was surveyed, and then joined at the same latitudinal position as the bathymetry data.

For future reference, I detail the mosaicking procedure adopted within *EASIPACE*. Because the package is not designed for mosaicking sidescan sonar imagery, it is important to document the processing steps, which may not immediately appear to be the obvious way forward. Two basemaps must be created using the *CIM* routine in *EASIPACE* with a grid cell size of 50 m and dimensions of 1400 by 1400 pixels (for the December 1990 data case), since the grid is 70000×70000 UTM units (metres). The grid cell size selected is a compromise between the resolution differences between the depth and sidescan data; 81 depth values per ping and around 500 backscatter samples per ping. The UTM bounds are defined, or geo-referenced, for each basemap using *GEOSET*, with the same bounds as the bathymetry grid (section 4.4.1). All mosaicking is performed with 8-bit data, for efficiency, so all the binary geometrically correct files must firstly be linearly compressed. For simplicity, these two basemaps are named 1 and 2, where basemap

1 is required for intermediate processing purposes, while basemap 2 comprises the final mosaic. Figure 4.13a presents a flowchart of these procedures, and figure 4.13b indicates the relevant *EASIPACE* routines.

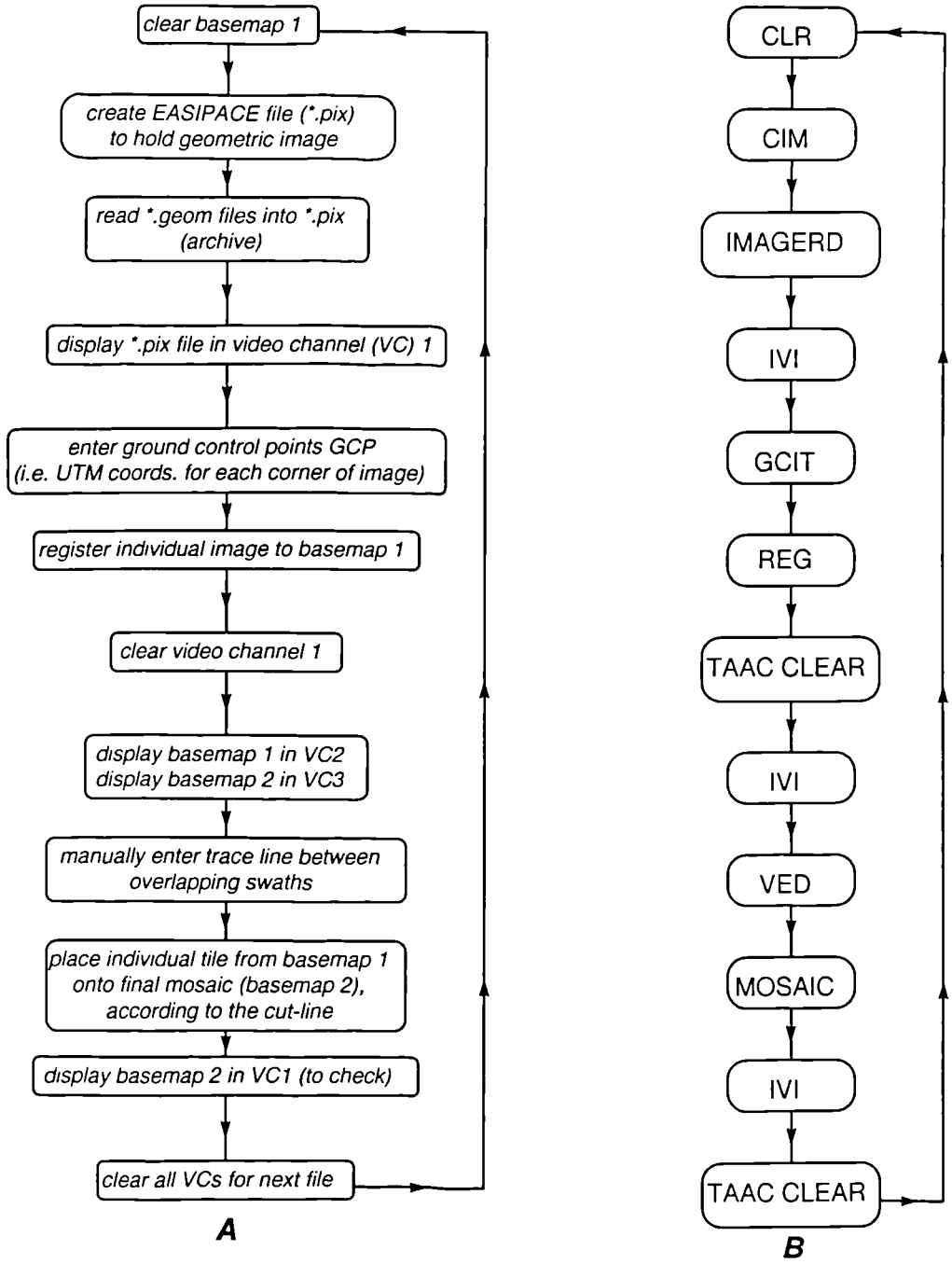


Figure 4.13 *Mosaicking procedure flowcharts: (a) descriptions; and (b) relevant EASIPACE routines.*

With reference to figure 4.13b, I shall make some clarifications regarding some of the parameter settings for the *EASIPACE* routines. Most are obvious, so only the possibly ambiguous ones will be mentioned. An *EASIPACE* database must have its

number of pixels as a factor of 8, so for simplicity, each individual database was set as 1024×1024 pixels (*CIM*), and the data were then read in according to its actual dimensions. That is, for the data input routine *IMAGERD*, '*dbiw*' is set to its default value and '*structur=nx, ny, <default>, <default>.....*', where *nx* and *ny* refer to the respective number of samples and rows of the geometric image. '*dbow*' is set to the same values as '*structur*'. When clearing a basemap using the routine *CLR*, all pixels in the image are set to a specified value, so '*valu*' is made equal to zero. The individual basemaps containing the geometrically correct images are first displayed in video channel 1 using *IVI*, step 4 of the procedure, and the parameters '*dbiw*' and '*vdow*' are set as: '*dbiw=0, 0, nx, ny*' and '*vdow=<default>*'. For the next routine, *GCIT*, '*dbiw*' remains the same, and '*dbgc=<blank>*' in order to create a new vector segment. The *GCIT* facility is used for entering the ground control points for placement of the individual images onto the mosaic. The prompt requires the entry of UTM coordinates, followed by the positioning of the cursor over the appropriate position on the image and then *<return>* is hit twice to confirm this ground control point. Four points are entered, corresponding with the corners of the image. For registering this individual image to basemap 1, '*dbgc*' is set to the number of the vector segment just created with *GCIT* and routine *REG* is run. The idea behind the second calling of the *IVI* image display routine is to distinguish the overlapping region between the completed mosaic so far and the next image to be incorporated. If the basemaps are displayed in the video channels as described, the green band on the 24-bit display shows the single geometric file alone in its correct geographic position, while the blue band displays the mosaic constructed thus far, excluding the new file in question; the overlapping region is therefore cyan. The vector editor (*VED*) is an interactive editor which enables the cursor to be dragged over the image, creating vector line segments and vertices every time key '*a*' is hit. The digitising mouse can be used to zoom in and out of the image whilst entering these vectors. The cut-line between overlapping swaths can thus be defined, mainly through an assessment of data quality and clarity of features. The vector polygon is best closed, and when exiting *VED*, it is imperative to remember to hit '*n*' to create a new vector segment. *MOSAIC* can combine overlapping images in any of three ways: firstly, a straightforward overlap; secondly, blending the two images through averaging techniques; and thirdly, if 'dbvs' is defined as the cut-line vector segment created with *VED*, then mosaicking will be performed accordingly. *MOSAIC* transfers the individual correctly positioned file stored in basemap 1 to the thus far completed mosaic held in basemap 2.

This set of routines have been incorporated into an *EASIPACE* procedure (*PASTE* - appendix 9). This questions whether the visual, interactive routine of cut-line definition is required before the registration of each file to the final mosaic (basemap 2). Without the cut-line facility, this process can be run quickly without any user input.

4.6.4 Destriping method

4.6.4a Introduction Figure 4.14 shows how the December 1990 acoustic backscatter data quality suffered greatly due to the poor weather conditions experienced during both cruises. Most notable is the degradation of the quality of certain tracklines due to the effect of ship and sea swell orientations. The degradation can be identified in two forms, both due to aeration around the transducer faces of the echosounder attenuating the received sonar signal: (A) the backscatter strengths of some individual pings are attenuated compared with neighbouring pings, resulting in a 'striped' appearance to the data; and (B) the average backscatter strength of some entire tracklines is decreased compared with adjacent tracks.

4.6.4b Previous work The destriping of sidescan sonar data has been previously attempted by Chavez (1986), Malinverno *et al.* (1990) and Le Bas *et al.*, (Le Bas, T.P., D.C. Mason, and N.W. Millard, TOBI image processing - the state of the art, submitted to *Geophysical Journal International,* 1993). They all dealt with actual line drop-outs inherent in GLORIA, SeaMARC I and TOBI data, respectively. These drop-outs are missing data, caused by the yaw of the towed vehicle. These image restoration techniques were all based on differential filtering principles.

Chavez (1986) solves the problem of line drop-outs in GLORIA data through the application of two different spatial filters; one high-pass (1 line by 71 samples) and one low-pass (21 lines by 71 samples). The high-pass filter leads to an image possessing the high frequency components of the image, except for the noise frequency, which the filter is designed to exclude, while the low-pass filter produces the low frequency components without the noise. The resultant image derived from high-pass filtering is subtracted from the original GLORIA image, then the result is simply added to the low-pass filtered image. The result is similar to the original image, but without the noise that the filters were designed to remove.

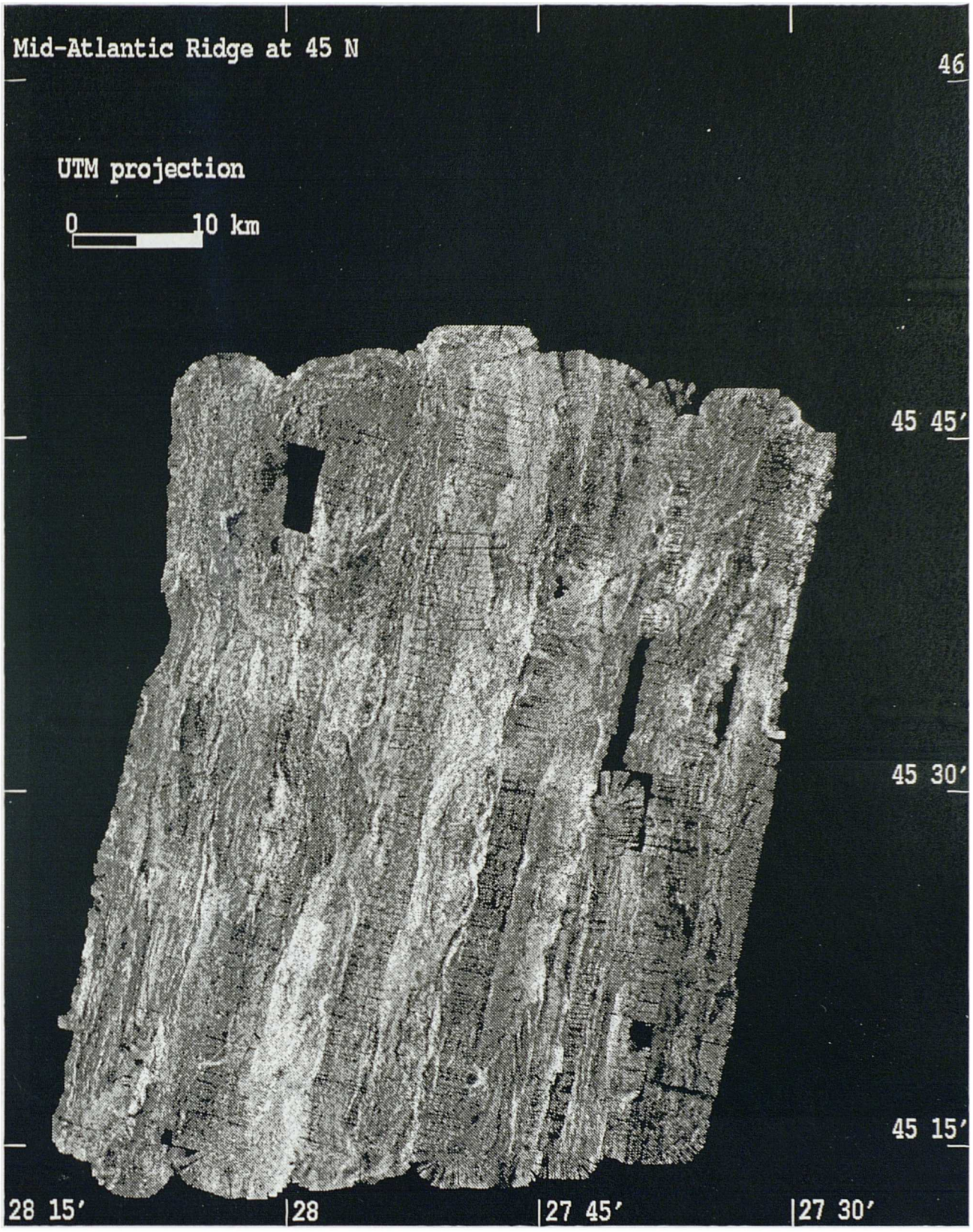


Figure 4.14 *Southern survey backscatter mosaic, prior to any noise reduction (white corresponds with a backscatter strength of -0.5 dB, black to -64 dB).*

Le Bas *et al.* (Le Bas, T.P., D.C. Mason, and N.W. Millard, TOBI image processing - the state of the art, submitted to *Geophysical Journal International,* 1993) present a more sophisticated line-drop-out correction designed for TOBI high resolution sidescan sonar data, again to reduce the effects of vehicle yaw, except here drop-outs do not necessarily appear symmetrical to port and starboard. Le Bas *et al.* passed a 7 line by 201 sample pixel kernel over a TOBI image, and calculated the ratio between average pixels of the central ping and the average over the rest of the kernel (three pings above and below). A threshold ratio can be established, by various means, and those pings identified as drop-outs are replaced by linear interpolation of their immediate neighbours.

Malinverno *et al.* (1990) applied the method of Chavez (1986) to drop-outs found in SeaMARC I data, but they also identified a similar problem to that experienced with the EM12 data, where certain pings appear 'darker' or 'brighter', compared with the majority of the data, generally caused by telemetry errors. These anomalous pings were detected by comparing the mean backscatter of an individual ping with that of its immediate neighbours, defining a 'bad' ping if its mean backscatter lies above or below a specified threshold percentage compared with its neighbours. The anomalous pings were replaced with a linear interpolation of adjacent pings.

Most of these documented sonar cases deal with the problem of complete drop-outs, i.e. missing data, but the EM12 problem involves the attenuation or 'darkening' of individual pings. All of this previous work also applies algorithms to sidescan data prior to mosaicking. However, the other problem with the EM12 (B) is the mismatching of grey levels of complete track swaths, so using mosaicked data facilitates the identification of these problem regions.

For the destriping problem (A), Poros and Peterson (1985) identified a similar scenario with striped noise on LANDSAT satellite imagery. The LANDSAT striping is often caused by slight errors in the calibration or a variable response of one of the 16 sensors, resulting in a periodic striping pattern. Since this noise is cyclical, it can be simply identified by comparing the mean grey levels for each of the 16 sensors once, defining a mean deviation, so if a mean scan exceeds this, it can be identified as being received from a noisy sensor. This procedure need only be performed once over 16 lines of the image, since the pattern then repeats itself. Two methods were investigated to alter the average grey level of these noisy sensor scans so they are

consistent with the good quality data: a radiometric equalisation and a histogram modification method.

The radiometric equalisation method was initially described by Algazi and Ford (1981), where they applied it to both periodic and non-periodic striping problems inherent in various satellite data. This method is based upon the sensor response functions relating input radiance to output voltage. For scan lines with different response functions, an equalisation curve can be generated according to a single chosen function by simply multiplying the noisy scanlines by the inverse of the anomalous function and the chosen function.

Horn and Woodham (1979) first reported the histogram modification method, and Kautsky *et al.* (1984) refined this with a smoothed histogram approach. Poros and Peterson (1985) identified the striped data throughout their LANDSAT image, once the responses of the 16 individual sensors had been established, since this pattern is periodic. They selected the reference histogram to only contain the data from the 'good' satellite sensors, and only the 'bad' sensor data were adjusted. They found this approach superior to matching all individual sensor histograms to a reference histogram composed of all sensors. Poros and Peterson (1985) successfully removed almost all periodic striping in their LANDSAT imagery using the histogram modification method, deducing that it was a superior method to the radiometric equalisation approach.

4.6.4c Destriping method The histogram matching technique is based upon the fact that a transformation exists relating any input histogram h, to some reference histogram h', where these histograms possess n and m number of grey levels, respectively. If H_h and $H_{h'}$ are the cumulative histograms of h and h', then T represents an exact transform matching H_h to $H_{h'}$, such that:

$$H_h(x) = H_{h'}(T(x)) \quad \text{where} \quad T = H_{h'}^{-1} H_h$$

and H^{-1} denotes the inverse transform of H. A look up table can be constructed by applying the inverse of the function $H_h(x)$ to $H_{h'}(x')$. The look up table is then applied to all data values which produced the histogram H.

EASIPACE (PCI, 1988) incorporates an algorithm for the histogram matching of complete images, generally used for balancing the histograms between the adjacent components of mosaics. The algorithm is principally the same as the histogram

modification method of Poros and Peterson (1985), so I have adapted it to remove both the image striping problems identified for the EM12 backscatter data. The first requirement before applying this method is to identify which of the data are poor quality pings. Once a mask comprising the poor quality data has been determined, two histograms may be generated; one from the good quality data and one from the degraded data.

4.6.4d Detection of poor quality data For the destriping problem (A) of individual pings with the EM12 backscatter data, the poor quality pings can be identified using some type of edge detection filter. A number of these were tried, all of which were linear convolution matrices which could be simply incorporated into *EASIPACE*. The best results were achieved with a directional Sobel operator:

$$\begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}$$

This effectively picks out the 'edges' of the 'bright to dark' regions, so an additional filter was designed to shift these edges down by one pixel to coincide with the bad pings:

$$\begin{pmatrix} 0 & 3 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

For the case of matching the histograms of complete trackline swaths (B), appropriate regions were selected by eye using the cut-out vectors, which were defined during the mosaicking procedure.

The histogram matching routine *MATCH* in *EASIPACE* requires a reference and a working image, such as desired by Poros and Peterson (1985); each of these working regions can be specified with a bitmap. In this situation, one image is used for both reference and working images, and two bitmaps are generated which I call the good and bad data bitmaps. The bad data bitmap is the thresholded result of applying the convolution filters to the original image, representing all the poor quality pings in need of radiometric enhancement. The good data bitmap is effectively the inverse of this, put through a logical 'AND' operation with a bitmap representing all data values other than the null value of the image matrix. The latter

is accomplished with a simple thresholding operation, since the null value is -64.5 (dB), and all data values are greater than this.

4.6.4e Applications

I Individual tile Figure 4.15a displays an individual EM12 backscatter tile, with all pings positioned horizontally, as described in section 4.6.1. This is clearly badly degraded with attenuated pings, or just parts of pings. Figure 4.15b is the result of applying the two bad ping detection convolution matrices, and forms the bad data bitmap. Correspondingly, figure 4.15c is the good data bitmap. The histogram matching was attempted by computing histograms for a specified number of lines at a time. I thought at first that a small number of lines per cycle of maybe 3, 5 or 7 pixels would be optimal, effectively matching poor quality pings with the histograms of their adjacent good quality neighbours. However, running 50 or 100 lines per cycle over the whole tile proved just as effective, and more rapid. Figure 4.15d presents the destriped image; clearly, the quality is greatly improved.

II Individual geometric image Figure 4.16a shows a geometric image, that is each pixel is placed in its correct geographic position, while figures 4.16b and 4.16c illustrate the derived good and bad data bitmaps. The edge detection routines were tried out on rotated datasets, with pings oriented horizontally; though since our tracklines were only oriented between 013° and 019°, no improvement in the anomalous ping detection capability was noticeable. Figure 4.16d displays the improved image, again the technique proved successful.

III Whole mosaic This technique was applied separately to the northern and southern areas. The ping detection performed equally well on the original and rotated datasets, so rotation was not necessary. The destriping histogram matching procedure (A) was attempted in several ways, including one track at a time and all the mosaic at once. The latter proved just as successful and took considerably less time.

The complete trackline matching (B) was applied in a similar manner; Cervenka and de Moustier (1993) applied a similar principle to mosaicked SeaMARC II sidescan imagery. Matching the 'bad' data histograms to the histograms of only the two immediately adjacent swaths yielded the best results. One must exercise care with enhancing mid-ocean ridge data in this manner, since backscatter strengths do physically vary with distance off-axis, concurrent with increased sedimentation

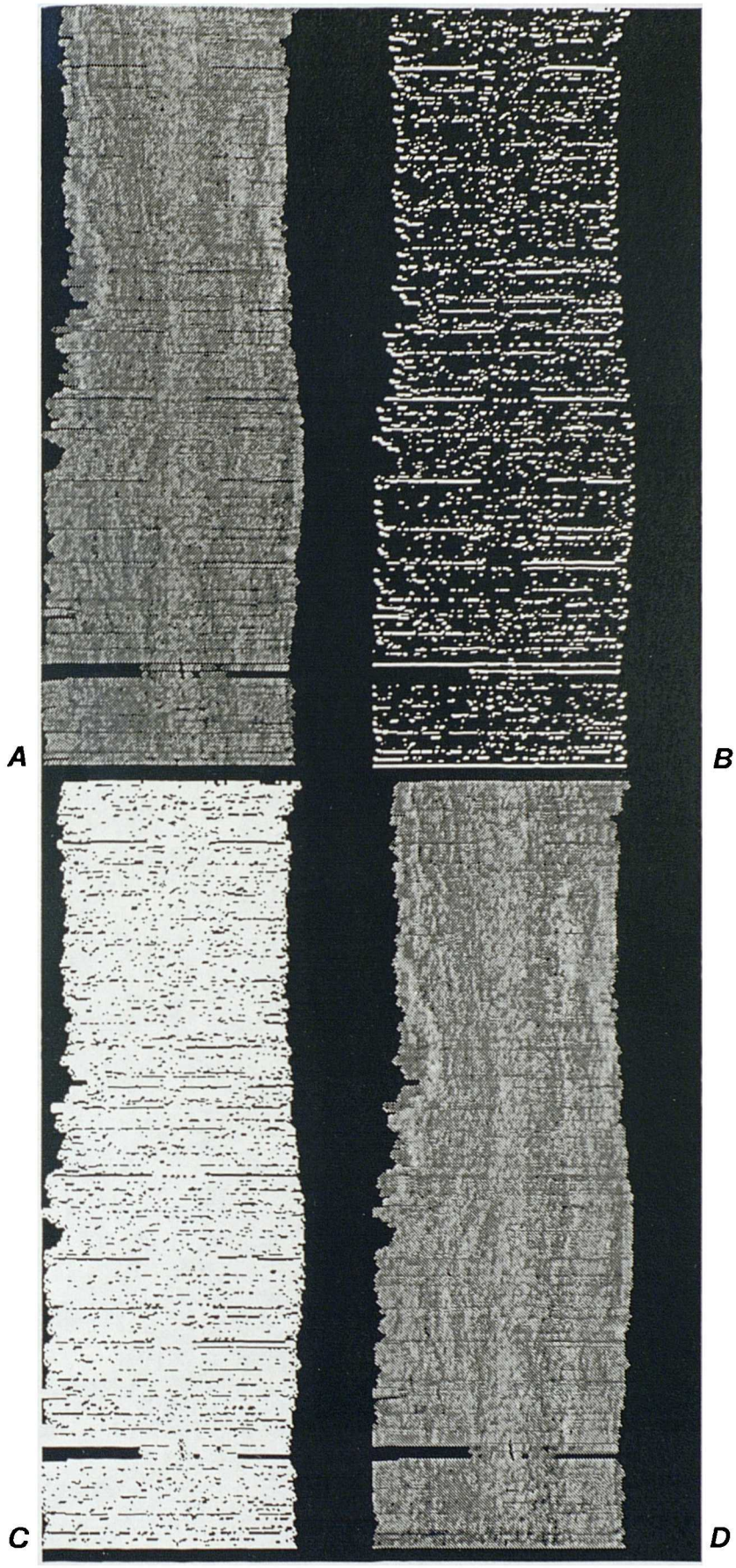


Figure 4.15 *Example of backscatter tile image destriping: (a) original image; (b) bad data bitmap; (c) good data bitmap; and (d) resulting destriped image.*

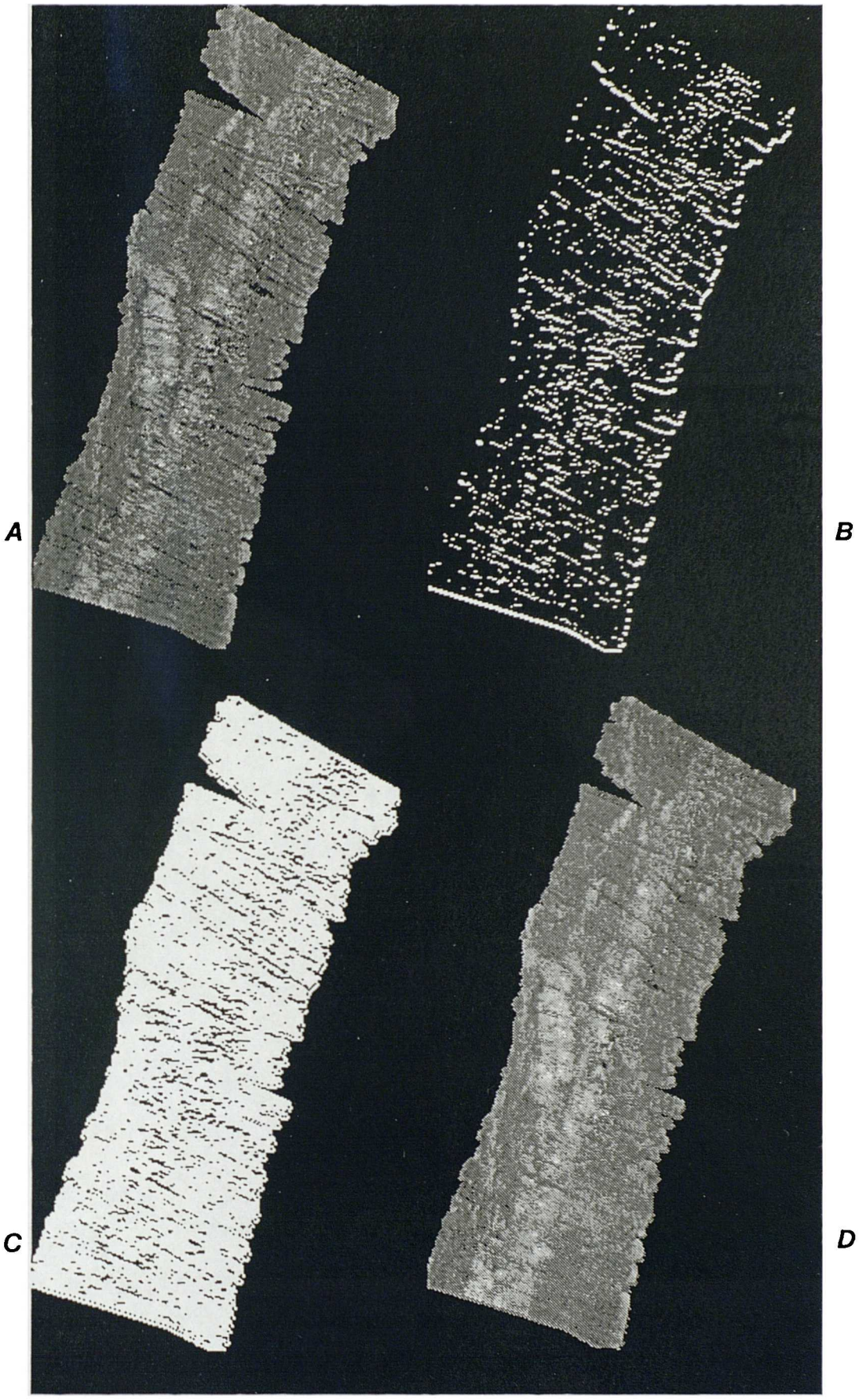


Figure 4.16 *Example of backscatter geometric image destriping: (a) original image; (b) bad data bitmap; (c) good data bitmap; and (d) resulting destriped image.*

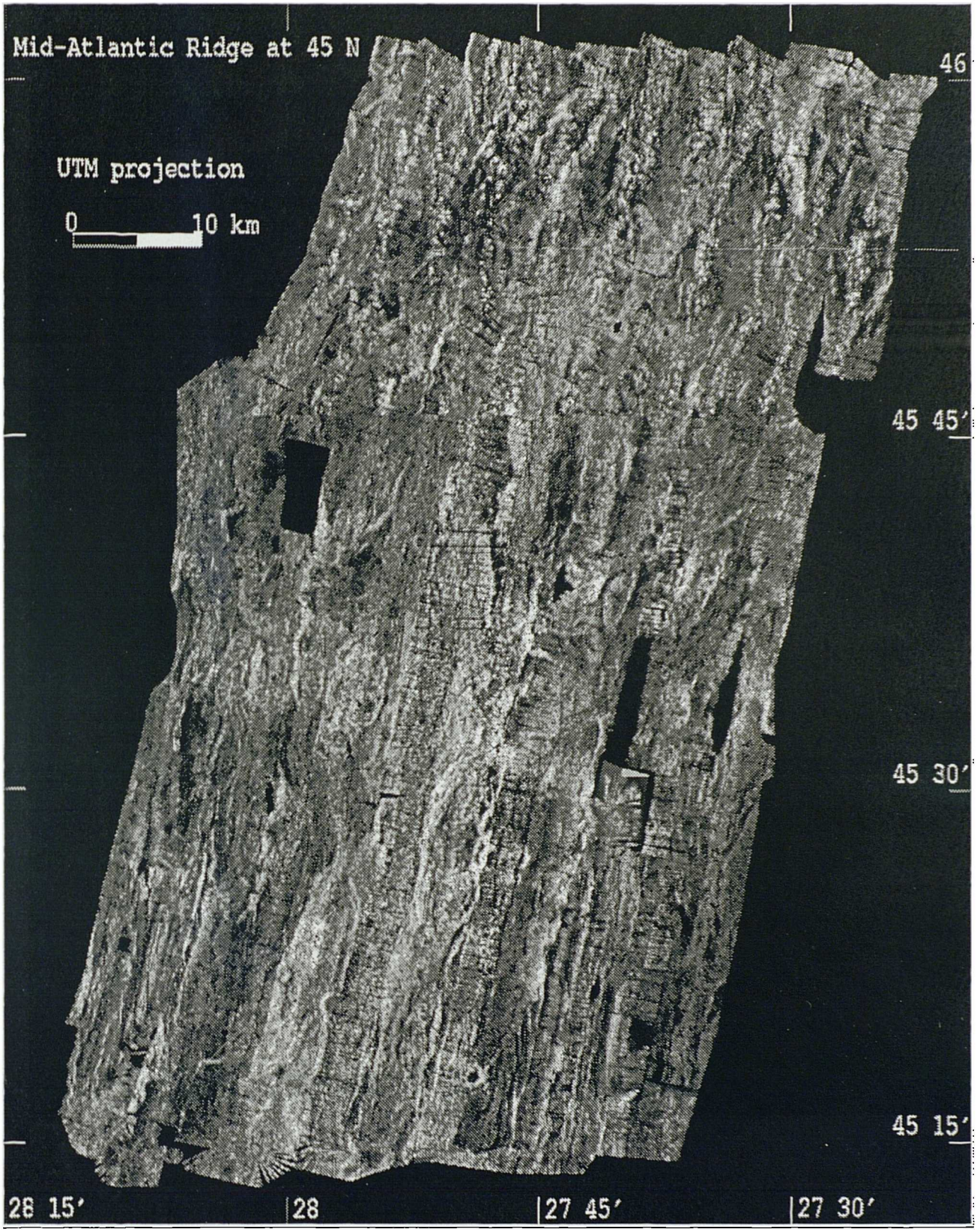
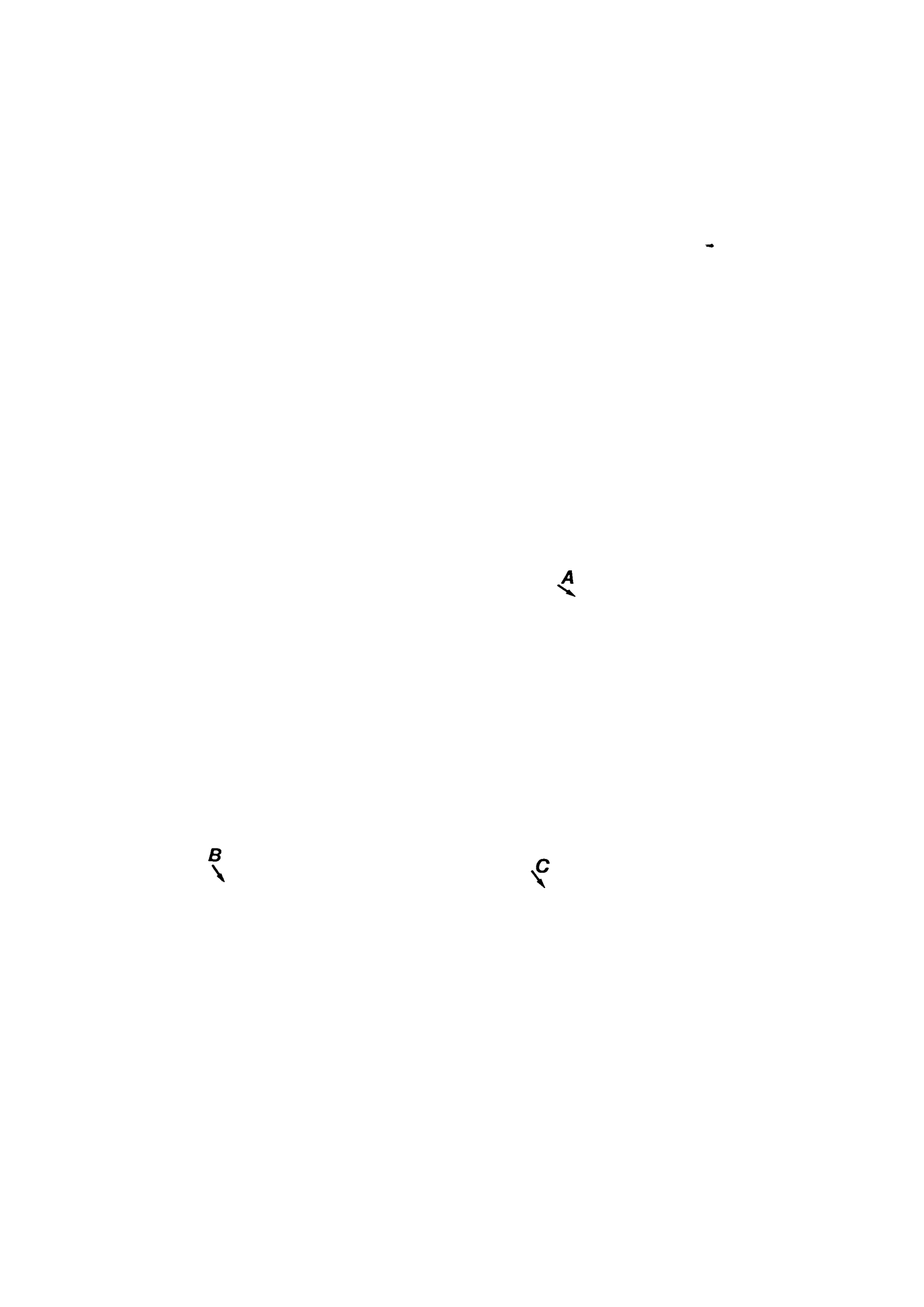


Figure 4.17 Backscatter mosaic for combined survey data, after noise reduction (white corresponds with a backscatter strength of -0.5 dB, black to -64 dB).


A
B
C

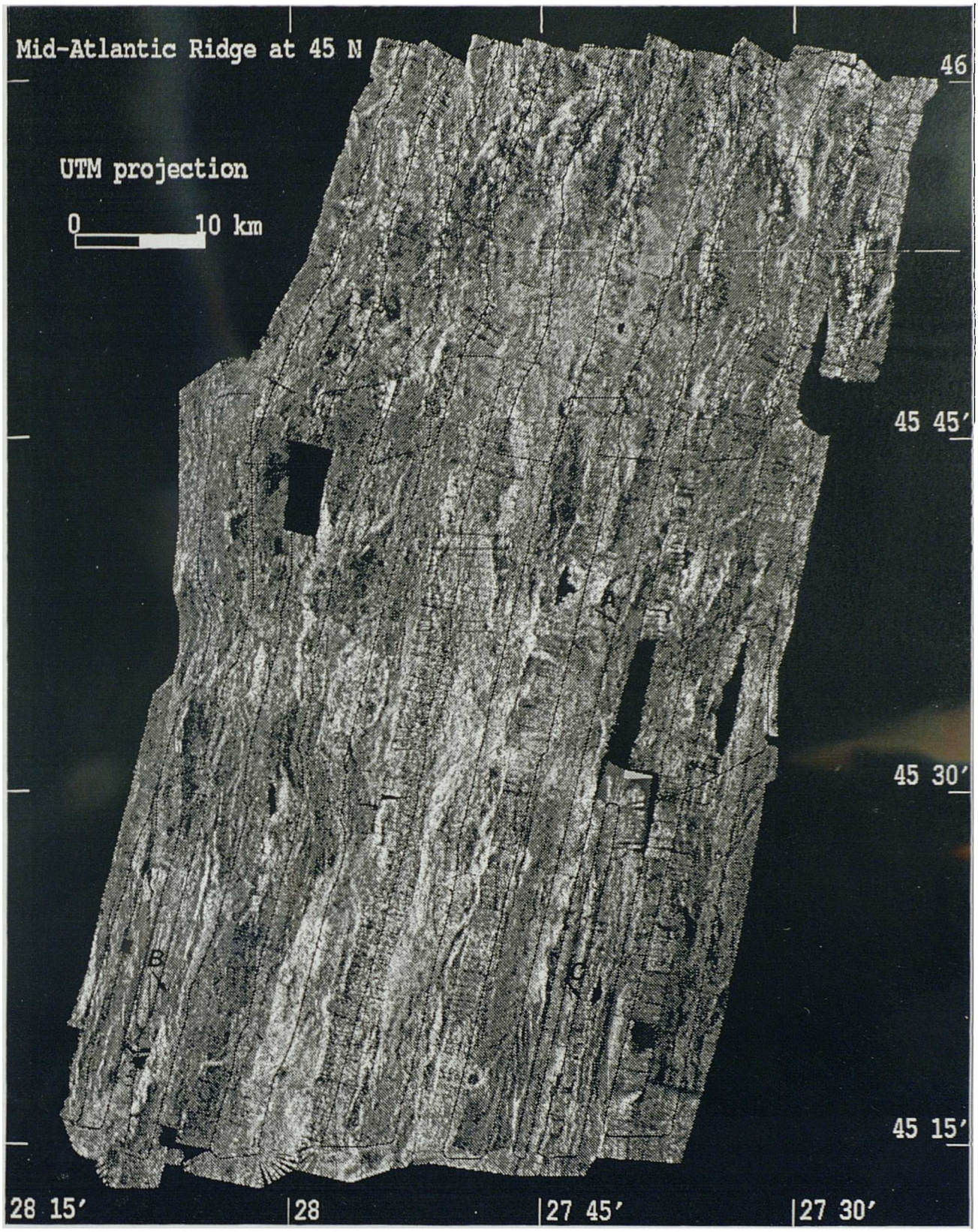


Figure 4.18 *Backscatter mosaic for combined survey data, after noise reduction (white corresponds with a backscatter strength of -0.5 dB, black to -64 dB). Ship tracks are overlain as solid black lines. Additional annotations correspond with figure 4.28.*

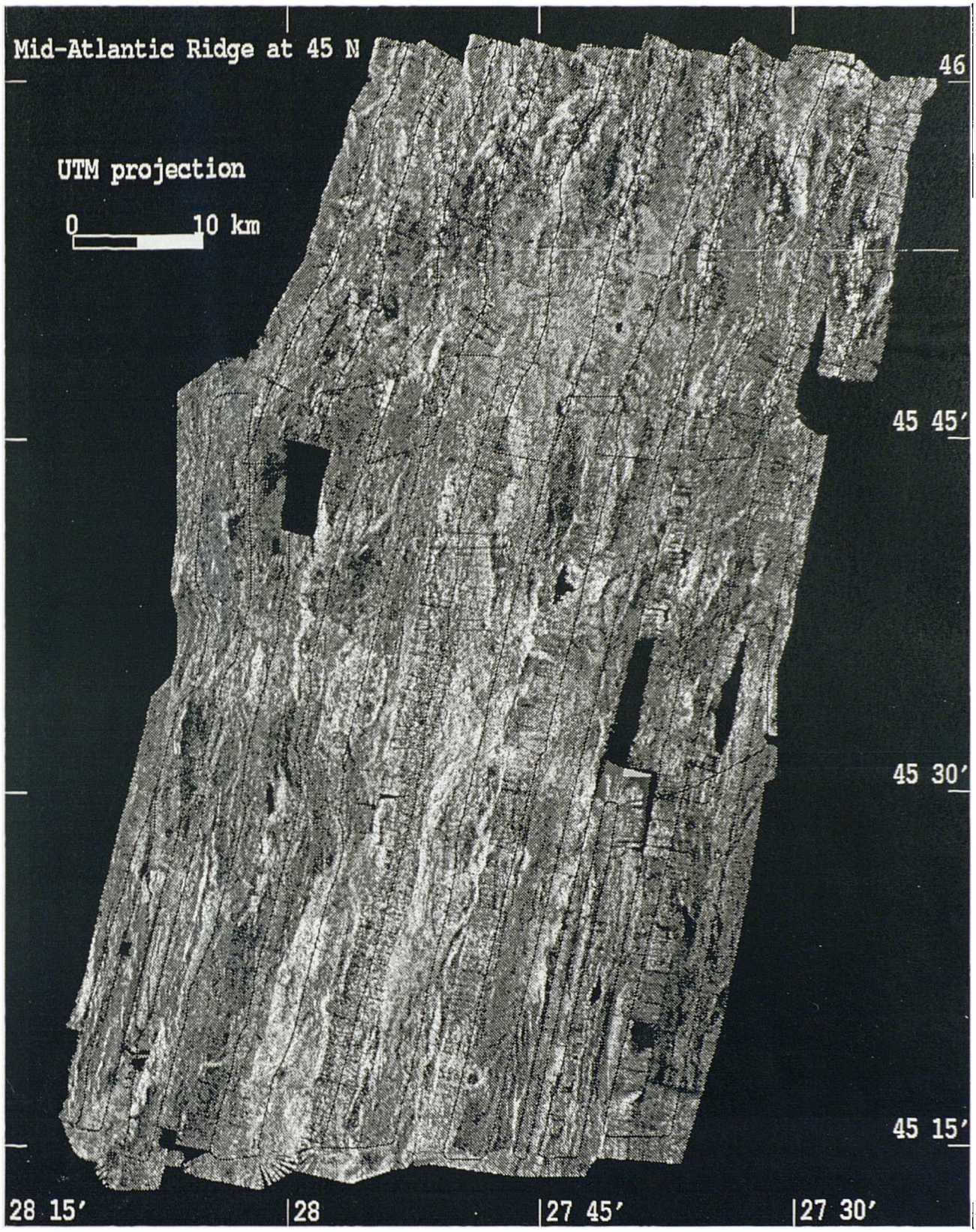


Figure 4.18 *Backscatter mosaic for combined survey data, after noise reduction (white corresponds with a backscatter strength of -0.5 dB, black to -64 dB). Ship tracks are overlain as solid black lines. Additional annotations correspond with figure 4.28.*

(Searle and Laughton, 1981; Hagen *et al.*, 1990). The tracklines are oriented parallel to the ridge axis, so the reference histograms were generated from the good quality tracklines either side of the poor quality line, in order to balance out the results. 50 lines per cycle were used when matching, occasionally over-compensating some of the data; however, this was proved to provide the best compromise.

Figure 4.17 presents the improved mosaic; a considerable reduction of the striping and mis-matched trackline grey levels is apparent (the additional image enhancement routines described in section 4.6.5 have also been applied). Figure 4.18 is identical, except that the ship tracks are superimposed. The quality of some of the swaths is so poor, that no technique is capable of radical restoration. A more sophisticated approach for bad ping detection, prior to mosaicking at the initial stage of processing, may also improve the results.

4.6.5 Image Enhancements

These were performed after, rather than before, mosaicking. This was of benefit for a balanced radiometric contrast enhancement, and the first smoothing filter would also then be able to remove small gaps between swaths. The anti-speckle filter should not be affected whenever it is applied.

4.6.5a Grey level contrast enhancement

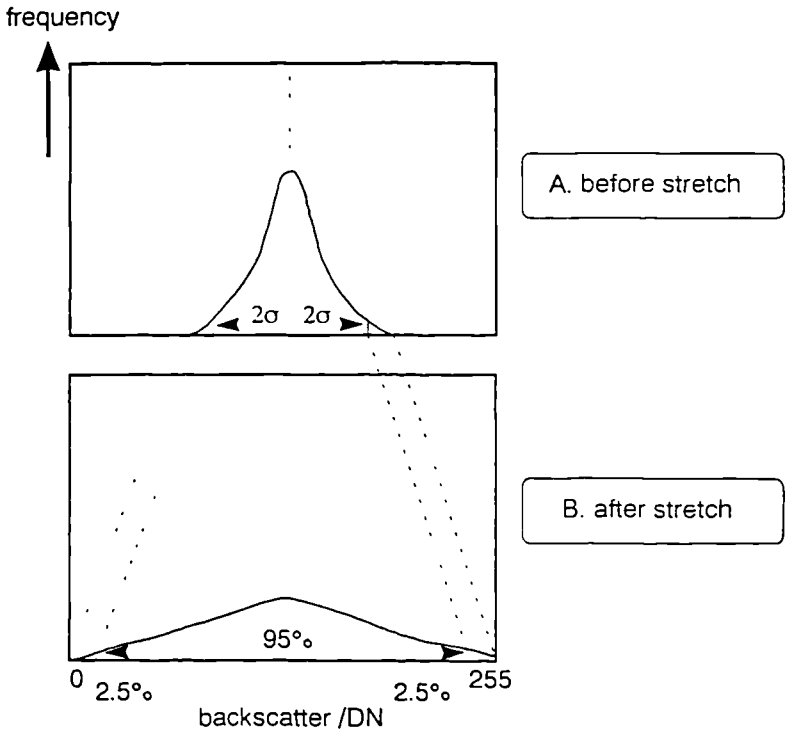
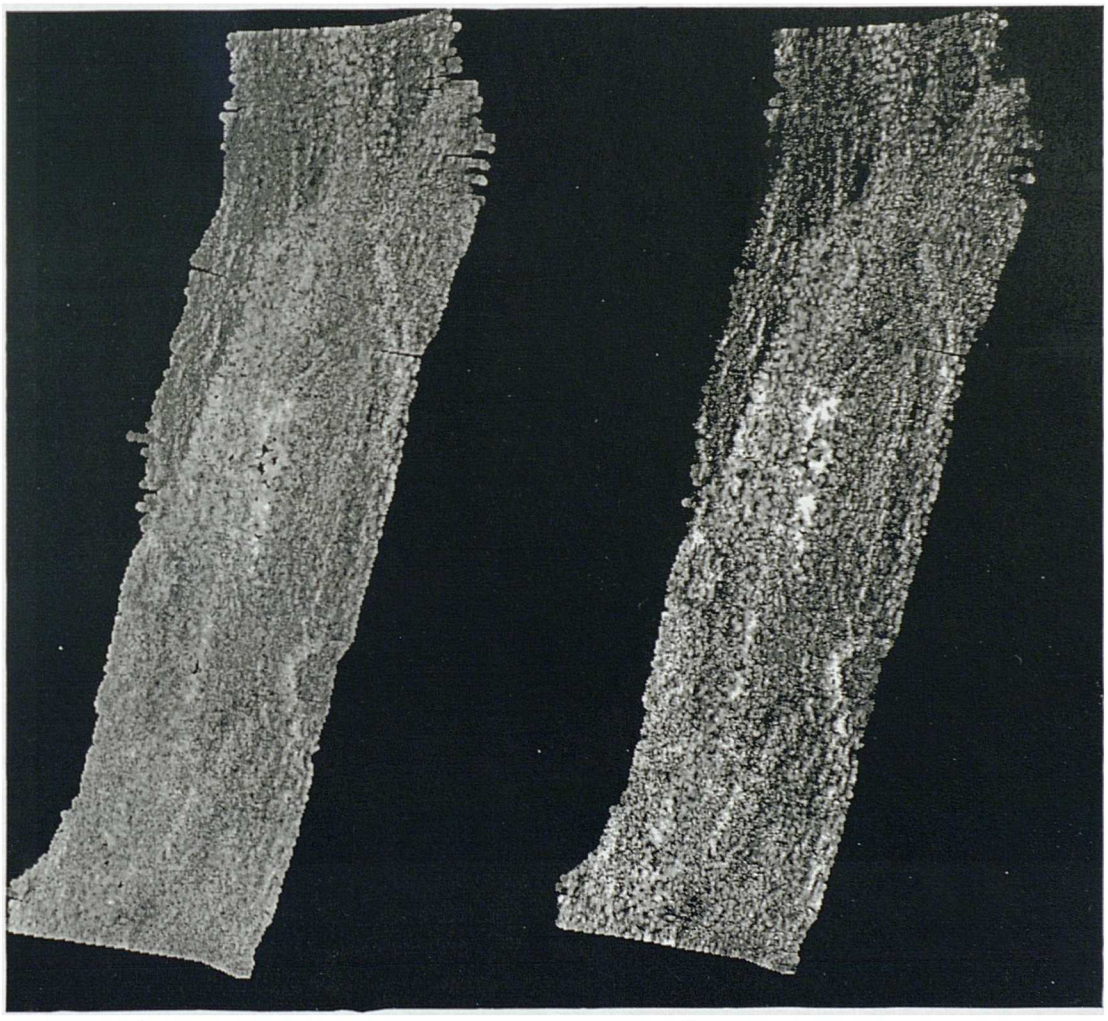


Figure 4.19 Principles of Drieman contrast stretch.

The grey level contrast of the sidescan mosaic, currently displayed as linearly compressed decibel backscatter values, is enhanced using a non-linear contrast stretch, often described as a Drieman Stretch (PCI, 1988), (*RSTR* routine in *EASIPACE*). Figure 4.19a illustrates an approximately Gaussian distribution for the image data. The mean and standard deviation of the image are calculated and two grey levels are determined: two standard deviations above and below the mean, respectively. All pixels with digital number (DN) equal to zero are ignored. The data between the minimum pixel DN, i.e. the cut off, and the lower grey level are mapped to the bottom 2.5% of the grey level scale; the data between the two grey levels are placed through 95% of the grey scale; while the remaining data lying between the upper grey level and saturation are assigned to the top 2.5% of the scale (see figure 4.19b). Figure 4.20a and b respectively display a sidescan image before and after this radiometric correction has been applied.



A **B**

Figure 4.20 *Example of sidescan image: (a) before; and (b) after, Drieman contrast stretch.*

4.6.5b Noise removal Two filters are applied to the image. One is designed to fill in some of the data gaps (*smooth image.f* - appendix 10). A 5x5 pixel window is passed over the image; any pixel of DN 0 is replaced by an average of its neighbouring pixels, only calculating from those which are greater than zero. A slight disadvantage is that this will remove small acoustic shadows, though with the small incident angles, these are few anyway. Figure 4.21a and b illustrate the data before and after applying this filter.

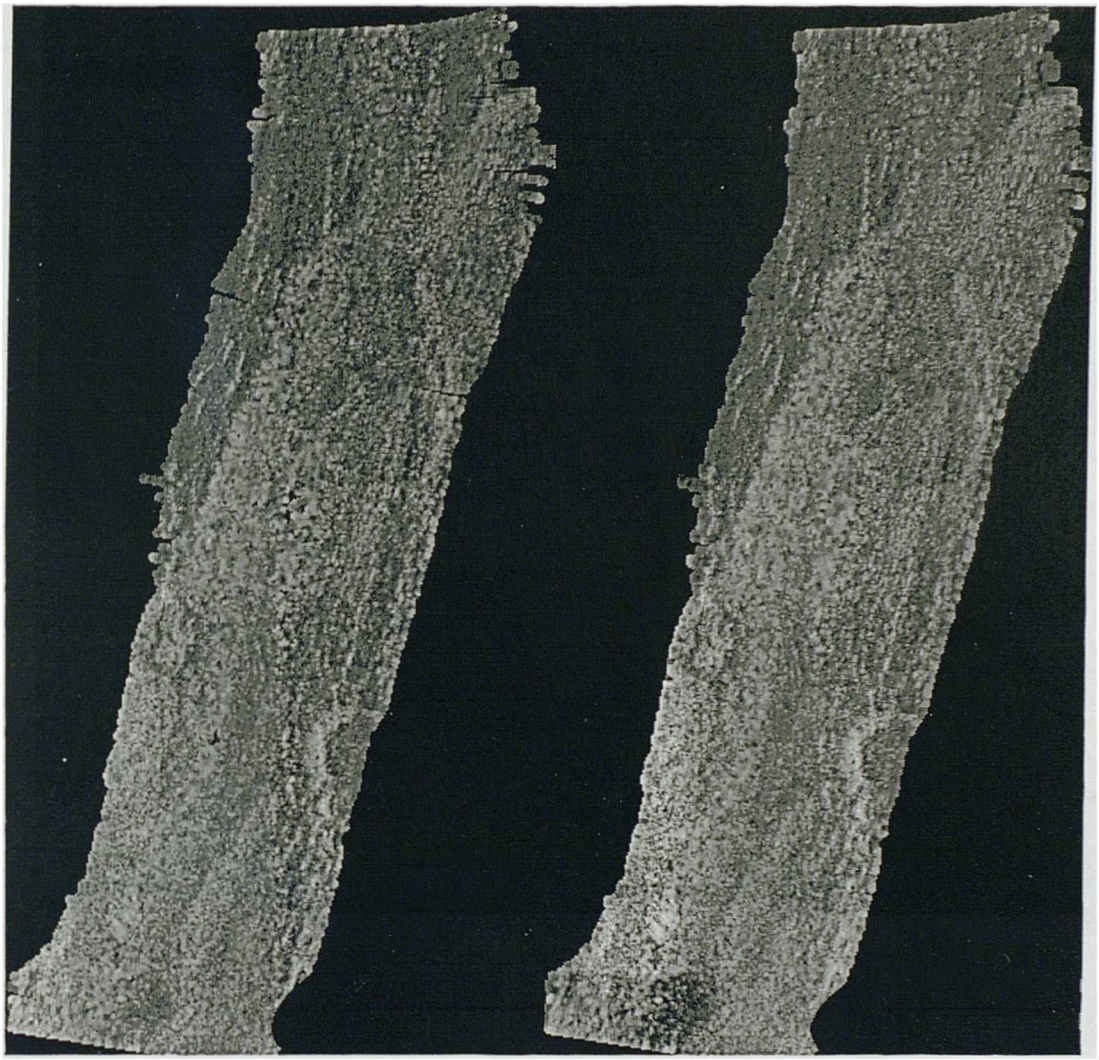


Figure 4.21 *Example of sidescan image: (a) before; and (b) after, smoothing filter.*

The second filter is an adaptive filter (*FAD* in *EASIPACE*; see PCI, 1988), devised to remove high frequency noise (speckle) while preserving high frequency features (edges). Speckle noise is an inherent feature of monochromatic radiation, e.g. laser (Goodman, 1976), radar (Frost *et al.*, 1982) and sonar, resulting from the

constructive and destructive interference from a number of scatterers on a rough surface. Chavez (1986) describes speckle removal filters based on the identification of 'noisy' pixels by thresholding a high-pass spatial filtered version of the data, setting these to DN 0, then applying a filter similar to the first noise suppressing filter described in this section.

A filter window of size 3x3 pixels is used to define a circularly symmetric filter with a set of weighting values, M, calculated for each pixel (Frost *et al.*, 1982), where:

$$M = e^{-AT}$$

and:

$$A = DAMP\left(\frac{V^2}{I^2}\right)$$

T = absolute value of pixel distance from the centre pixel to its neighbour in the filter window
$DAMP$ = damping factor (use default of one here)
V^2 = grey level variance in filter window
I^2 = square of mean grey level in filter window

Thus the resulting grey level, R, for the smoothed pixel is:

$$R = \frac{\left(\sum_{i=1}^{i=n} P_i M_i\right)}{\left(\sum_{j=1}^{j=n} M_i\right)}$$

where:
$P_1 \ldots\ldots P_n$ = the grey levels for each pixel in the filter window
$M_1 \ldots\ldots M_n$ = the weights (as defined above) for each pixel in the filter window

Figure 4.22a and b respectively present data before and after applying this adaptive filter, with a window size of 3×3 pixels.

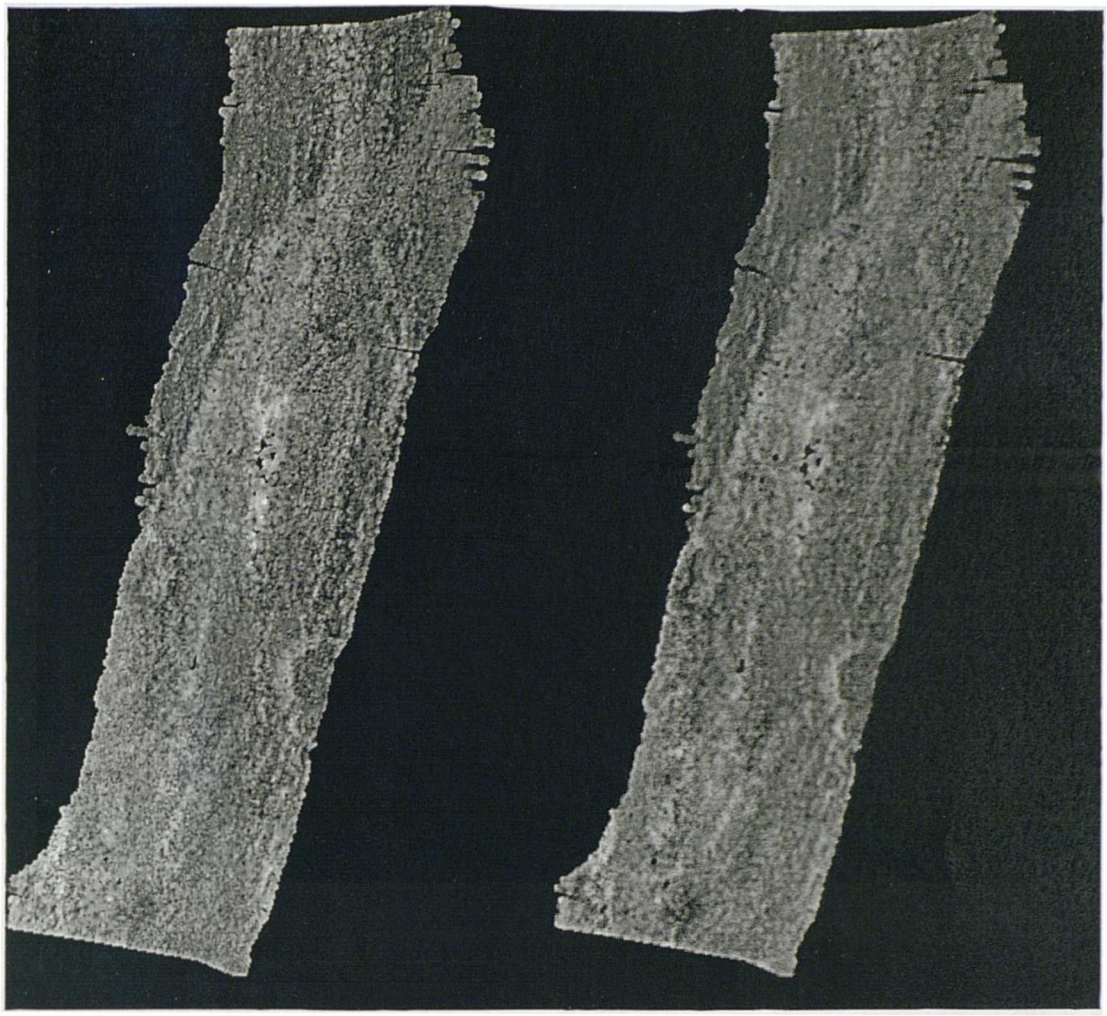


Figure 4.22 Example of sidescan image: (a) before; and (b) after adaptive anti-speckle filter.

Figure 4.17 and 4.18 illustrate the resulting sidescan mosaic, once all the above improvements (sections 4.6.4-4.6.5) have been applied (figure 4.18 shows the ship track information aswell). The data will be described in section 4.8.2 and analysed further in chapter 5.

4.7 MARINE GEOGRAPHIC INFORMATION SYSTEM

A geographical information system is primarily a database, commonly used for handling a number of different types of co-registered data. Typical, land-based systems can include images such as satellite or aerial photography data and vectors such as drainage patterns and geological boundaries. These data are all geographically registered together and can be easily accessed and viewed in any combination. For marine data, the same principles can be applied. At this stage, only

processed data has been included, but a potential scheme could be to organise the database, so raw instrument data (such as parameters associated with particular beams, including quality factors etc.), often necessary for quantitative analyses, can also be called up from selected sites, akin to a system currently being developed by Shippey *et al.* (1993).

4.7.1 *EASIPACE* data storage principles

The *EASIPACE* image processing package stores all information in the form of a database; this includes both image channels and vector segments. The image channels comprise data such as the bathymetry grid, sidescan mosaic etc. with either 8, 16 or 32 bits per pixel. Additional images generated by producing shaded relief maps, combining datasets etc. may also be stored as image channels. The vector channels comprise many different types of information: these can include straightforward bitmaps, geometric information, such as the geographic referencing information or the ground control points used for mosaicking, or actual vector data, such as ship track information. The *EASIPACE* setup in Durham includes a 24-bit TAAC application accelerator monitor which enables the display of three 8 bit images at any one time, plus up to 10 graphics planes.

4.7.2 Incorporation of vector information

Vector information, such as the ship track, tickmarks, survey times etc., can be incorporated into an *EASIPACE* database file as bitmap segments, displayed in any of ten available graphic planes (*IVB*) on the display or 'burned' onto an actual image channel (*MAP*). Point or line vectors can simply be translated from the archived navigation files (section 4.5) into the *EASIPACE* database using *VREAD*, refer to the *EASIPACE* manual for formats (PCI, 1988). Points can be displayed as asterisks and lines as a continuous line. Figure 4.18 illustrates the superposition of shiptrack information onto the sidescan mosaic. Numbers referring to the time of the relevant tickmark can be generated using the 'graphics label' routine in *DCP* on a graphics channel on the VDU, then saving this as a database segment using *VIB*. Labels can be constructed in the same manner.

4.8 DISPLAY TECHNIQUES

To aid comparison between the bathymetry and sidescan datasets, visualisation techniques of the combined datasets have been investigated. Searle *et al.* (1990) examined some of these techniques for the combined display of GLORIA sidescan

and Sea Beam bathymetry data using the *MIPS* image processing package (Chavez, 1986). The co-registered nature of the EM12 sidescan and bathymetry data should provide an excellent subject for a combined display.

4.8.1 Theory

The display techniques attempted here combine three image channels, where the colour definition of an image can be represented either in terms of the intensity, hue and saturation (IHS), or in terms of its red, green and blue components (RGB). Intensity refers to the lightness (or darkness) of a colour, zero intensity (DN 0) is equivalent to black, while full intensity (DN 255) would be white. Hue refers to the actual colour, represented as a continuous colour wheel through the 8 bit range of the data; blue is equivalent to DN 0, going through cyan and green to yellow (DN 128), then red, magenta and round to blue again at DN 255. Saturation refers to the amount of colour present, zero saturation (DN 0) shows no colour, while full saturation would show full colour for red, green and blue (at the expense of other hues).

Byte images representing one particular dataset can be selected according to whether they are best represented as the intensity or hue of an image, while saturation is chosen to remain at a constant level of DN 190, selected from experimental trial and error. For a combined display of these IHS images, they must be converted to their RGB components. These three output channels can now be displayed on a 24-bit monitor or converted for hardcopy. The IHS to RGB conversion is implemented using the *RGB* routine in *EASIPACE* and the conversion algorithms are as follows:

$$R = \frac{\sqrt{3}}{3} I * \left(\frac{442}{255}\right) + \frac{\sqrt{6}}{6} X - \frac{\sqrt{2}}{2} Y$$

$$G = \frac{\sqrt{3}}{3} I * \left(\frac{442}{255}\right) - \frac{\sqrt{6}}{6} X + \frac{\sqrt{2}}{2} Y$$

$$B = \frac{\sqrt{3}}{3} I * \left(\frac{442}{255}\right) - \frac{\sqrt{6}}{3} X$$

where

$$X = S * \left(\frac{208.2066}{255}\right) * \cos\left(H * \left(\frac{360}{255}\right)\right) * \left(\frac{\pi}{180}\right)$$

$$Y = S * \left(\frac{208.2066}{255}\right) * \sin\left(H * \left(\frac{360}{255}\right)\right) * \left(\frac{\pi}{180}\right)$$

4.8.2 Method

The EM12 sidescan mosaic is already held in an 8 bit *EASIPACE* database, however at this stage, the bathymetry is in the *GMT* grd format. Figure 4.2 displays the bathymetry contour map. *grd2raw* (courtesy of Dave Stevenson, Department of Geological Sciences, University of Durham) converts this grid into a byte image, where a minimum depth of 750 m is represented by DN 0, while the maximum depth of 3600 m is equivalent to DN 255. The surroundings where no data exists, also comprise DN 0. It makes more sense for shallow depths to appear as large values and deep as low (because the numbers are really negative). *PBMplus* software (Poskanzer, 1991) is used to flip the image to the required orientation (*pnmflip*), since *GMT* and *EASIPACE* orientate their grids differently, and to invert the grey levels (*pnminvert*), so the shallowest seafloor is equivalent to DN 255. The bathymetry data is now read into *EASIPACE*, but the surroundings with no data now comprise DN 254 or 255; a threshold (*THR*) is therefore applied to set these to zero. A shaded relief image is generated from the bathymetry data (*REL* routine in *EASIPACE*), where the intensity of each pixel is determined according to the cosine of the angle between the normal vector to the surface and the direction of illumination (grey levels are normalised to DN 0-255); an azimuth of 60° clockwise from north and an elevation of 60° above the horizontal are selected here (figure 4.23). The *REL* routine requires floating point input data, so the input image must initially be converted back to its original metre values, using *SCALE*. A new *EASIPACE* database is created with a large number of image channels, three of which comprise the sidescan, bathymetry and shaded relief data.

The range of hue can be controlled by *SCALE* routine, and a constant saturation image can be created using *CLR* and setting 'valu' parameter to DN 200 (figures 4.24, 4.25 and 4.26) or DN 128 (figure 4.27). Some of the permutations of our three input images are described below.

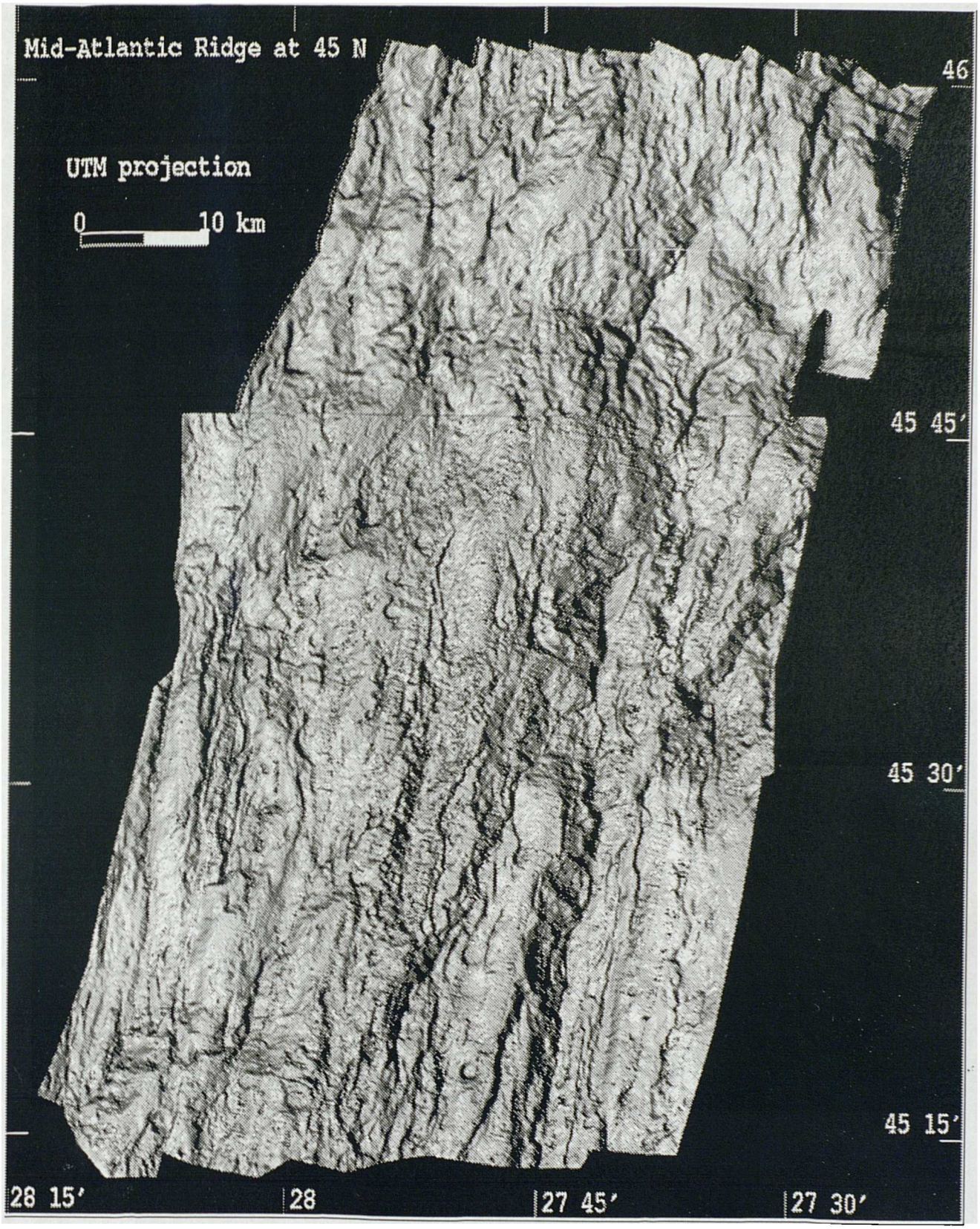
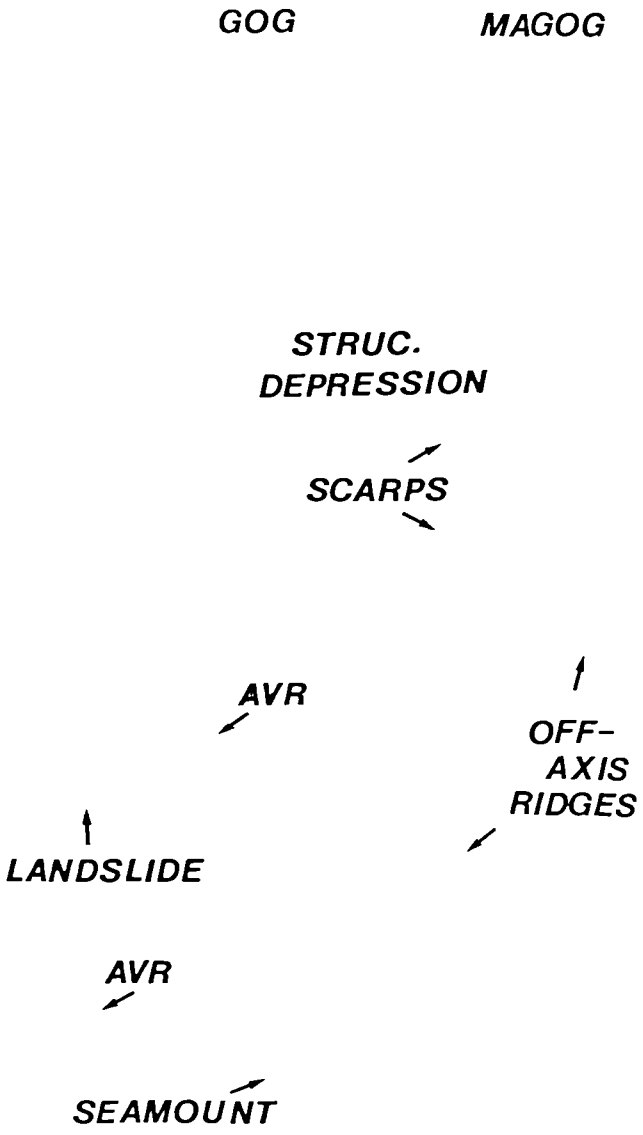


Figure 4.23 *Shaded relief image generated from the bathymetry data from both surveys, illumination is at 60° clockwise from north and elevation is at 60° above the horizontal.*


GOG
MAGOG
STRUC.
DEPRESSION
SCARPS
AVR
OFF-
AXIS
RIDGES
LANDSLIDE
AVR
SEAMOUNT

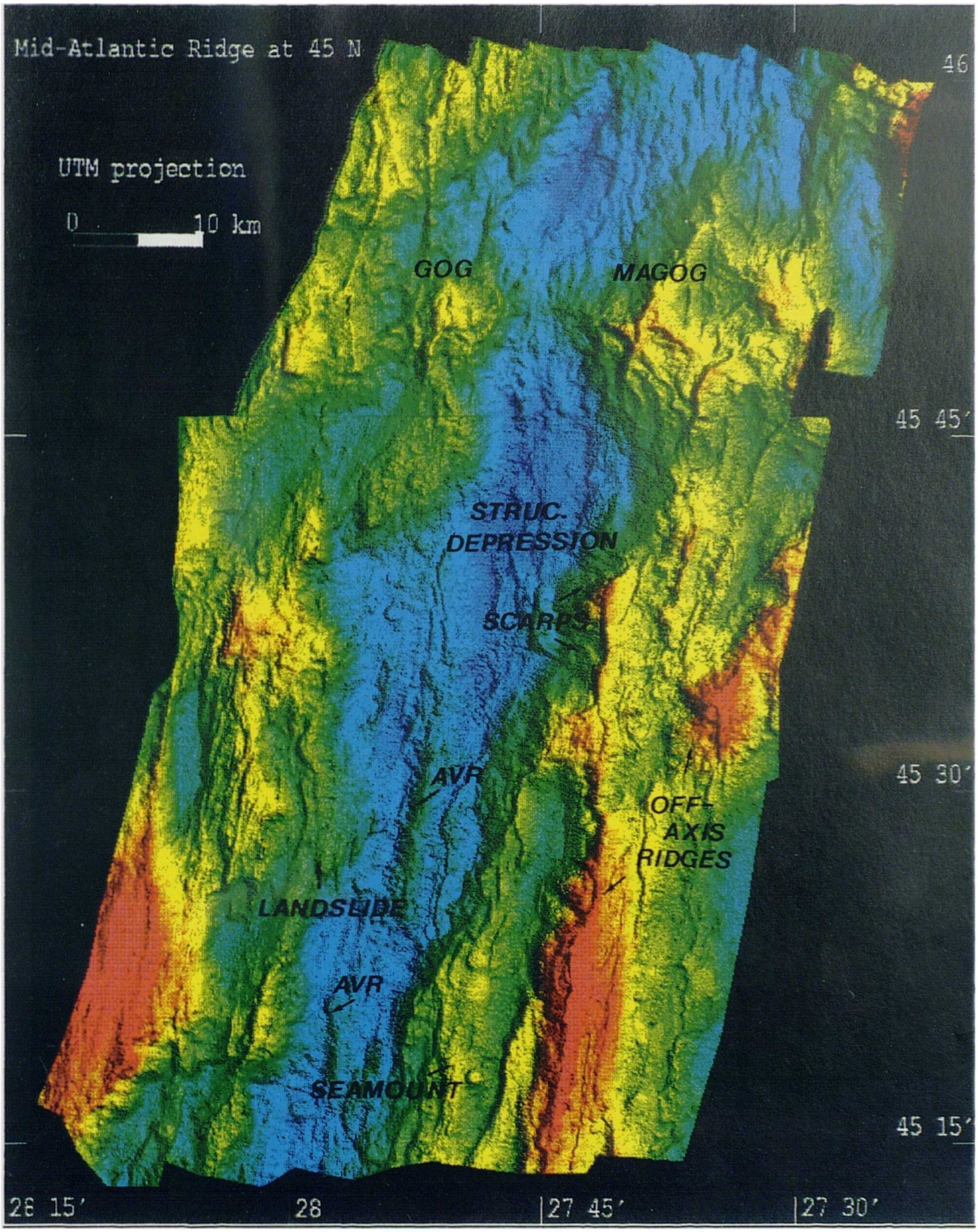


Figure 4.24 *The intensity of this figure is the shaded relief image of figure 4.23, while the hue represents the depth values, Magenta shows the minimum depth of 750 m, while blue corresponds with the maximum depth of 3600 m.*

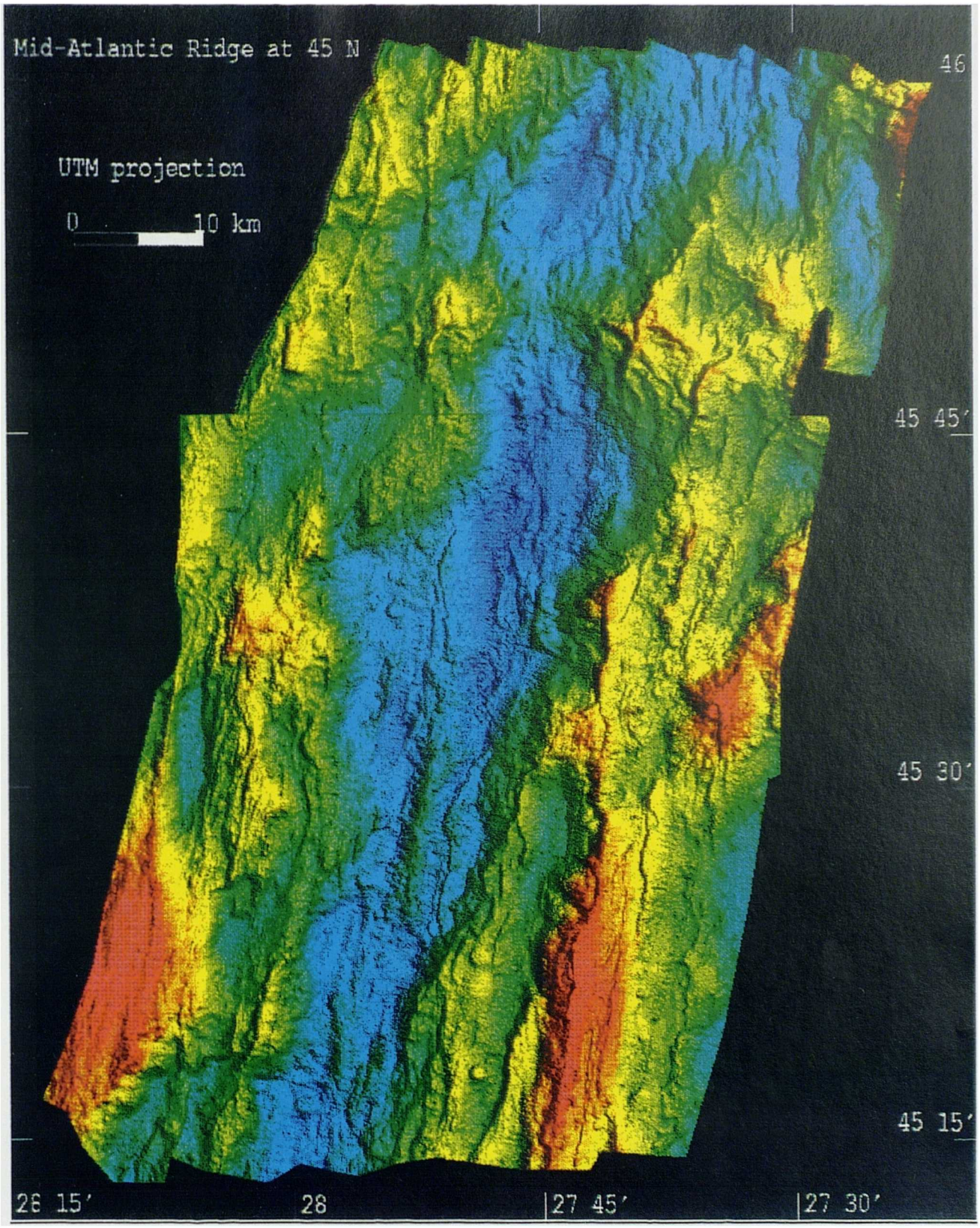


Figure 4.24 *The intensity of this figure is the shaded relief image of figure 4.23, while the hue represents the depth values, Magenta shows the minimum depth of 750 m, while blue corresponds with the maximum depth of 3600 m.*

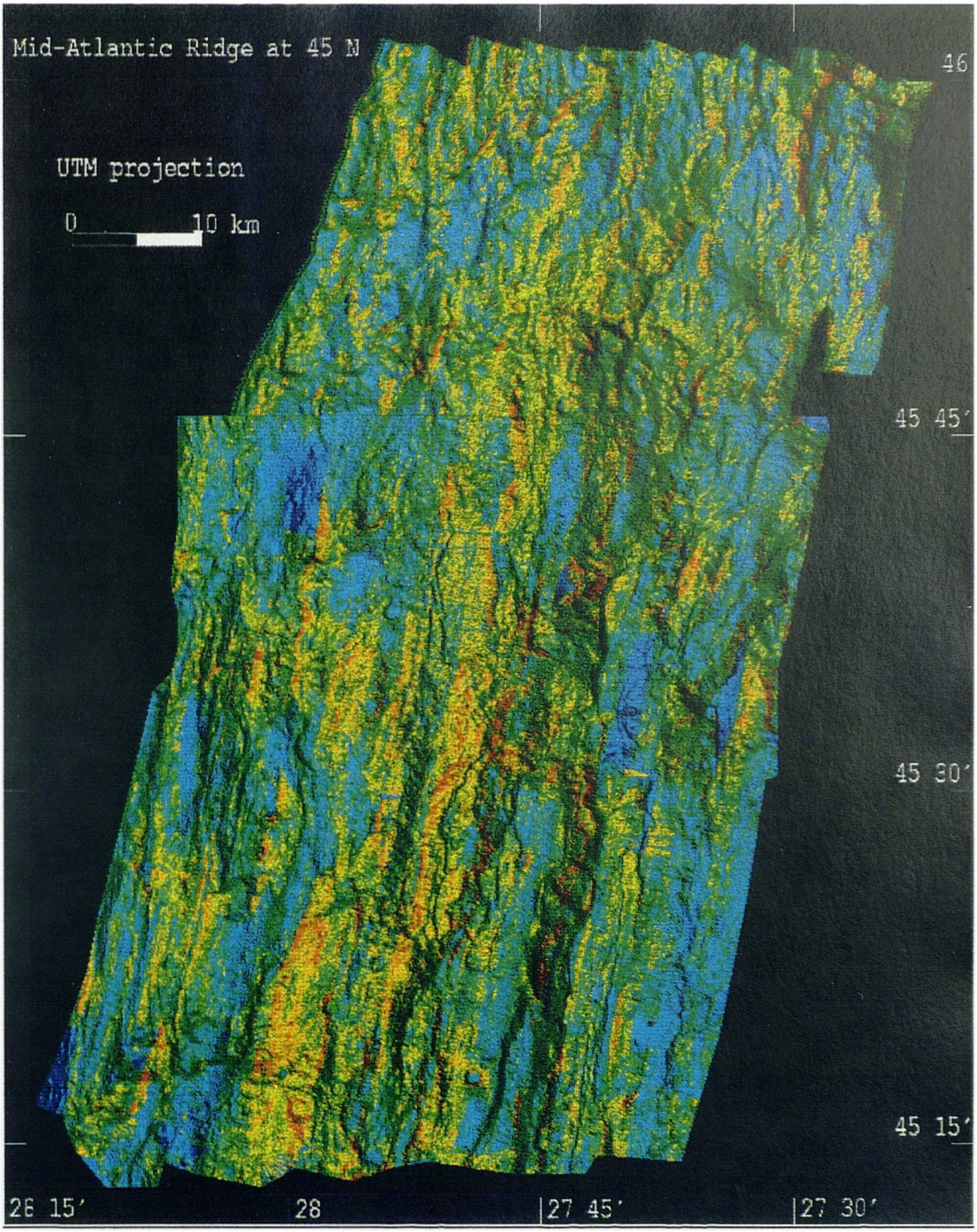


Figure 4.25 *The intensity of this image is the same shaded relief display of the bathymetry data as shown in figure 4.22. The hue represents the acoustic backscatter strengths (logged), where the most highly backscattering surfaces are shown as red (-0.5 dB) and the lowest as blue (-64 dB).*

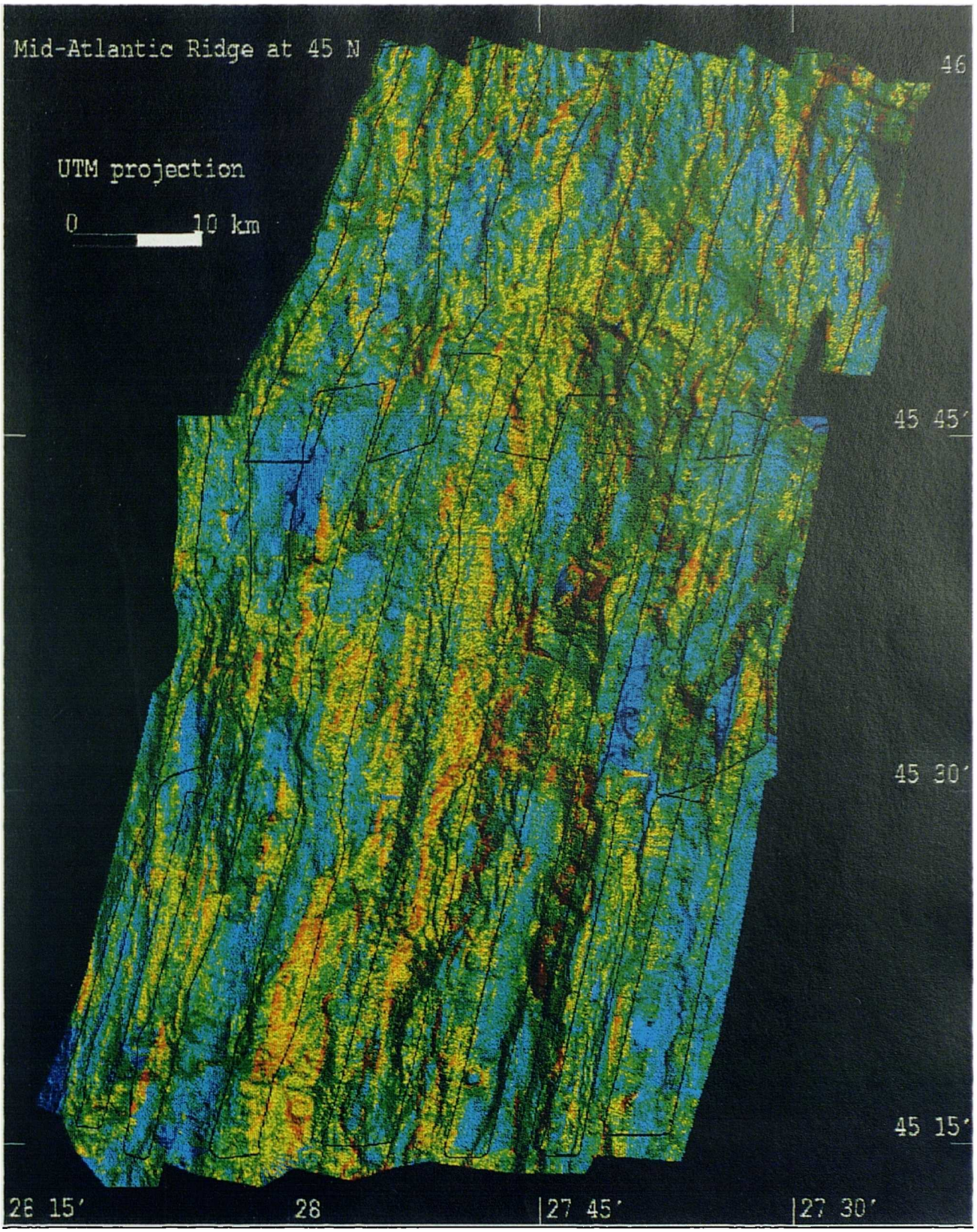


Figure 4.26 *This the same image as figure 4.25. Ship track information is overlain.*

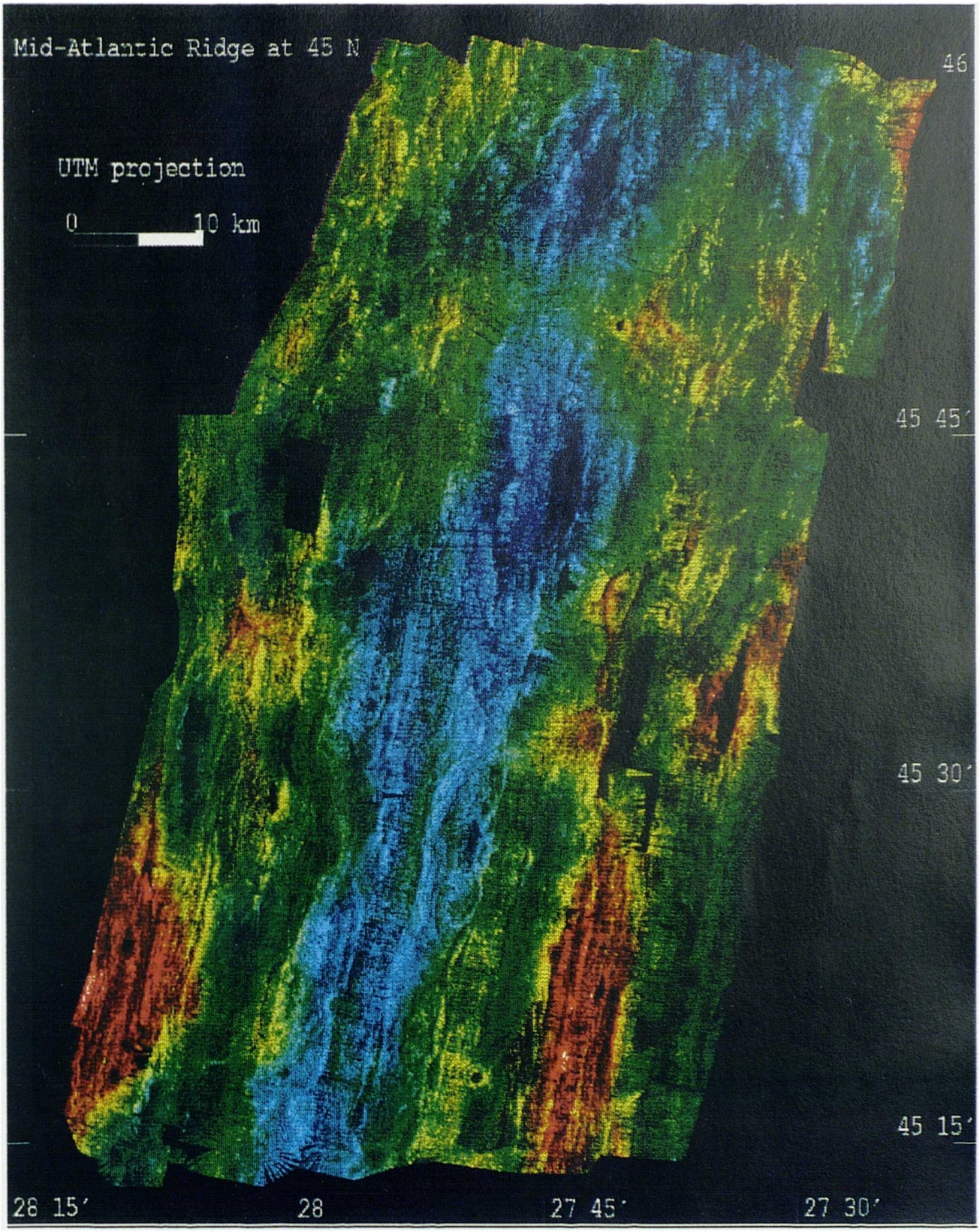


Figure 4.27 *The backscatter strengths form the intensity of this image, while the hues are the depths (same colour scale as figure 4.24).*

4.8.3 Results

4.8.3a Coloured bathymetry, intensity shaded relief This provides an excellent display for interpretation of the bathymetry data. Features such as volcanic ridges are highlighted with the intensity variations of the shaded relief image, whilst the hue allows an illusion of depth to be introduced (magenta represents a minimum depth of 750 m and blue corresponds to a maximum depth of 3600 m). Figure 4.24 illustrates this display, enhancing the geological interpretation in section 4.9.1.

4.8.3b Coloured sidescan, intensity shaded relief The interpretation of sidescan data over the entire colour scale is difficult due to the widely fluctuating nature of the backscatter characteristics at mid-ocean ridges. Therefore, the range of hues was constrained, so the grey scale of the sidescan hue image was linearly compressed to DN 0-175, displaying the low backscatter pixels (-64 dB) as blue and the most highly backscattering surfaces (-0.5 dB) as red. The bathymetry grey scale image was originally tested as the intensity image, but no useful effect was gained; the shaded relief surface provided a more informative backdrop (see figure 4.25). Figure 4.26 illustrates the same data, but with the ship track superimposed. The highly backscattering properties of the median valley are clearly highlighted, as is the loss in quality of some of the swaths. However, this technique particularly brings out the highly reflective properties associated with some of the fault scarps and lower backscattering properties of the sediments (refer to section 4.9 for further details).

4.8.3c Coloured bathymetry, intensity sidescan This display also aids the comparison between the two datasets. Of particular note in figure 4.27 are the intensity variations within the median valley and some fine scale faulting associated with the crestal mountains, markedly on the extreme south western ridge of the image.

4.9 RESULTS

4.9.1 Geology

Valsami *et al.* (Valsami, E., J.R. Cann, R.C. Searle, M. Ackers, and J.A. Keeton, Morphological characteristics of the Mid-Atlantic Ridge at 45°N: implications for magmatic and tectonic processes, manuscript in preparation, 1993) and Ackers *et al.* (1992) present a detailed geological description of the area, a brief overview of

which is presented here. Figures 4.2, 4.23 and 4.24 illustrate some aspects of the bathymetry of the survey area.

The slow spreading (1cm/year half spreading rate) MAR axis at 45°30'N is expressed as a median valley, striking 020°, between 8.5 and 22 km wide, assuming it is bounded by the 2750 m contour, and approximately 3100 m deep. Axial volcanic ridges (AVRs) can be identified on the valley floor, striking somewhat obliquely to the median valley at 010° (e.g. central 'green' area, labelled in figure 4.24). The ridges are right-laterally offset and the offset regions are characterised by structural depressions (e.g. 'blue' area labelled north of the ridges in figure 4.24). The remains of ancient volcanic ridges and depressions can be clearly traced off-axis. Towards the north of the area (near 45°15'N), the median valley is constricted by two large volcanoes ('Gog' and 'Magog', Aumento *et al.,* 1971) respectively west and east of the ridge axis. The median valley shallows by more than a kilometre here. Individual circular volcanic seamounts are observed on the valley floor. The median valley walls rise by between 300-1000 m and sometimes comprise multiple parallel fault scarps. These median valley parallel faults extend throughout the survey area, principally inward facing, although a small proportion are outward facing. Numerous landslides have been identified from the morphology of some of these scarps.

With regard to the lithology of the region, summarising Aumento *et al.* (1971), the axial volcanic ridges would be composed of freshly formed basalt, with structural depressions forming pockets of sediments and the sediment blanket generally thickening with increased distance from the ridge axis. Fault scarp exposures can yield more massive doleritic and gabbroic textures, with talus ramps at their feet.

4.9.2 Sidescan Data Description

Figures 4.17 and 4.18 illustrate the complete EM12 sidescan mosaic, and figures 4.25, 4.26 and 4.27 show some combined displays. The description of the backscatter data will be addressed one cruise dataset at a time as I feel some of the characteristics of the backscatter data altered in the time between the two cruises. With regard to the southern, December 1990 survey, the track lines run approximately NNE to SSW. Unfortunately, since the eastern portion of the survey was undertaken in a Force 8 gale, the data quality clearly suffers as mentioned previously in section 4.6.4. The loss in quality of the data is expressed as complete 'holes' in the survey where we were unable to pick up any returns. Even through the median valley where the weather conditions had improved, it is possible to identify in which direction the ship was steaming, with rougher conditions resulting in generally

'darker' swaths where the received signal has been attenuated. This degraded data has been improved somewhat with the methods described in section 4. In general, the quality of at least half the sidescan data was severely affected by the poor weather.

Through comparison with the bathymetry data, a broadly bright distribution of pixels is associated with the median valley, representing the high backscattering properties of rough unsedimented basaltic terrain. Some small seamounts are picked out well. Elongated bands of high backscattering pixels coincide with many of the fault scarps, particularly the multiple faulting in the south eastern median valley wall. The backscatter generally decreases with increasing distance from the ridge axis and structural depressions clearly exhibit the lowest backscatter values.

The data collected from the second survey (March 1993) chiefly look different because a revised EM12 operating system appears to have altered the near-nadir time-varying gain (TVG) correction (see section 4.6.2). The interpretation of the backscatter data is confusing in this region, where highly backscattering linear fault scarps are prolific, oriented parallel to the ship's track. The superposition of trackline data on figures 4.18 and 4.26 aids the distinction between artifacts and true scatterers.

Again, it is possible to identify the highly backscattering linear features associated with the fault scarps, interspersed with a surface possessing weaker backscatter strengths. The region of the image coinciding with the volcanoes, Gog and Magog, exhibits an intermediate scattering strength. These results suggest that the seafloor possessing the highest scattering strengths are the fault scarps; however a component of backscatter strength due to seafloor slope is likely here. The flatter region of intermediate scattering strength indicates the high backscatter strength of basaltic lava which would be expected in this locality. Finally, the low backscattering regions further off-axis can be attributed to the increased sedimentation here.

4.9.3 Comparison with GLORIA data

GLORIA 6.5 kHz long range sidescan sonar data were collected from this area in 1975 (Searle *et al.*, 1978; Laughton and Searle, 1979) and are displayed in figure 4.28. GLORIA is a traditional surface-towed regional mapping sonar tool, insonifying a swath of seafloor 50-60 km wide (Somers *et al.*, 1978), producing pixels of 45×160 m in size (Searle *et al.*, 1990). The pixel size is comparable with the EM12 data, but the incident beam angles are restricted to a range of 0-45° for the EM12, compared with 0-80° for GLORIA (respectively Pøhner and Lunde, 1990;

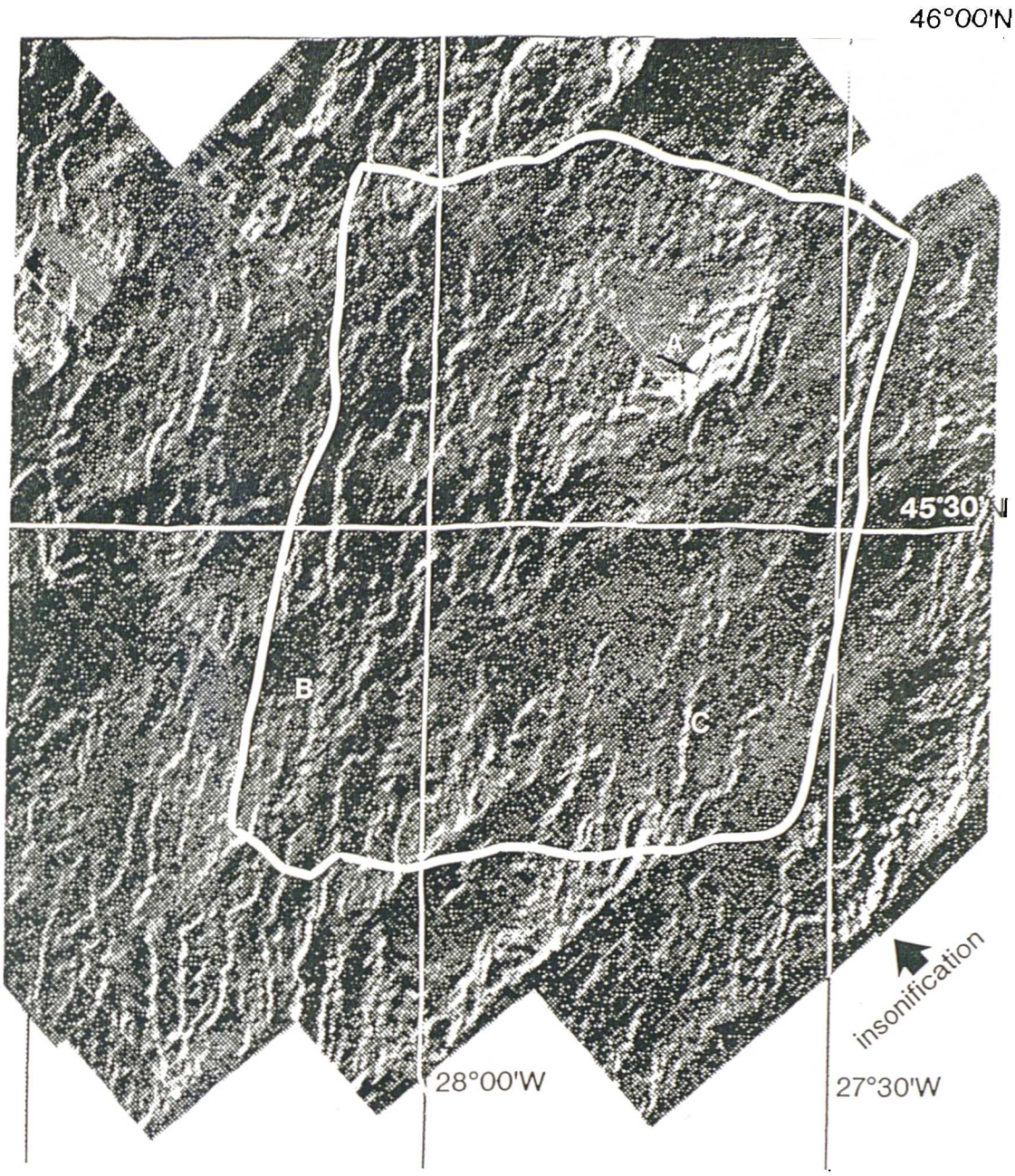


Figure 4.28 GLORIA data of the survey area, outline of southern survey is shown. The annotations relate to features also identified in figure 4.17.

Searle *et al.*, 1990). Figure 4.29, from Hughes Clarke *et al.* (1993), confirms the angular biases for examples of both types of data. The 45°N GLORIA data are recorded in analogue form and have not been converted to backscatter strengths; they must be considered to be purely relative. The 45°N GLORIA survey lines were run oblique to the spreading axis in order to search for possible fracture zones. The GLORIA mosaic is presented in figure 4.28 on a Mercator projection, while all the

EM12 data are shown with a UTM projection. The approximate location of the December 1990 EM12 survey is superimposed.

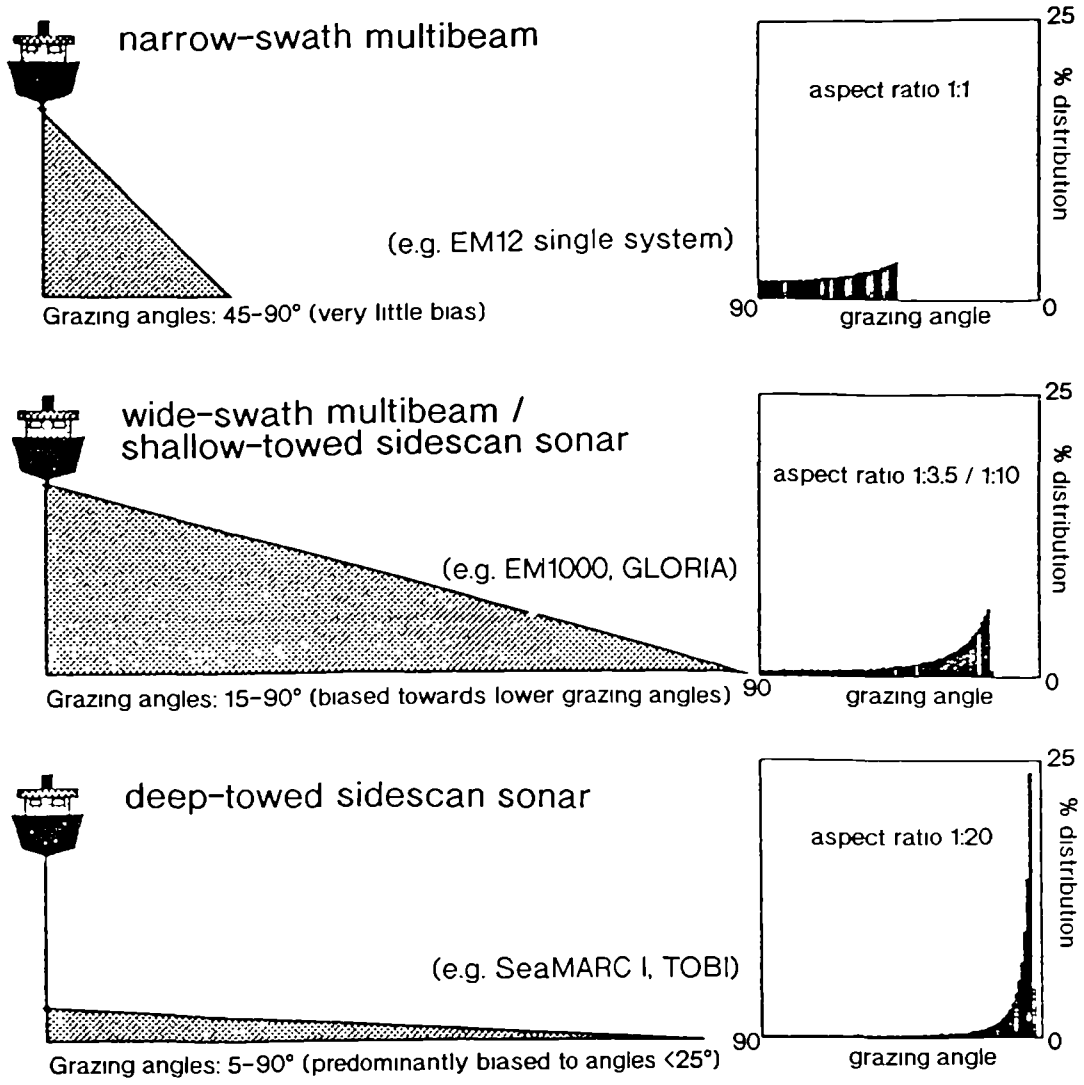


Figure 4.29 Angular geometries and biases of various sonar systems (adapted from Hughes Clarke et al., 1993).

The GLORIA data predominantly show fault scarps parallel to the spreading direction. One highly backscattering faulted feature can be discerned in both datasets in the north eastern portion of the EM12 survey, bordering the median valley (labelled feature A in figure 4.17 and 4.28). Noting the direction of insonification of the GLORIA data, feature B is an inward facing fault (to the ridge axis) and feature C is an outward facing fault. These features are correspondingly identified in figures 4.17 and 4.28.

Clearly, the appearance of the two sidescan datasets varies greatly as well. From the GLORIA data, it is difficult to pick out features other than faults, particularly variation in the inherent reflectivity of the terrain. This may be due to the fact that the GLORIA data were processed with an automatic gain control (AGC) (Searle *et al.*, 1990). There are also certainly some navigational inaccuracies in this dataset, which was navigated using dead reckoning and Transit satellite fixes. One must note that the EM12 sidescan data has had the effects of varying beam angle removed (section 5.3, chapter 5), assuming Lambert's Law; we are therefore not strictly looking at the same physical data. The incident beam angle of the GLORIA instrument can reach 80°,whilst the maximum angle for the EM12 is 45° (figure 4.29), thereby severely affecting the angle of incidence (ϕ) and thus backscatter (refer to chapter 5). Also, the operating frequency of the two systems is different, 6.5 kHz for GLORIA and 13 kHz for EM12; however, section 5.2 demonstrates that although frequency variations can have an important effect on the backscattering characteristics of the seafloor, the variation of backscatter with frequency over rugged terrain, such as a mid-ocean ridge, is negligible. The resolution differences are usually the most significant characteristic of various surveying tools (Kleinrock *et al.*, 1992; and section 3.2.1c).

Summarising this comparison, it is clear that imaging the same surface with different instruments, with their own specific characteristics, at various orientations, can wildly affect the imaged results. The 45°N area looks completely different when surveyed by GLORIA compared with the EM12 multibeam backscatter measurements. This agrees with the findings of Davis *et al.* (1987) and Kleinrock *et al.* (1992).

4.10 CONCLUSIONS

Processing techniques for the EM12 sidescan data have been developed, through the writing of new software and incorporation into existing image processing packages. The result is a geometrically correct mosaic of the survey area, with all the data stored in what can be described as a marine geographic information system. The sidescan data have been displayed in combination with the bathymetry, in order to aid the geological interpretation of the area. Chapter 5 now follows on with an acoustical investigation of the backscatter relationship to bathymetry; a first stage in lithological classification.

CHAPTER 5
EM12 BACKSCATTER AND BATHYMETRY RELATIONSHIPS

5.1 INTRODUCTION

Chapter 4 described the acquisition of a co-registered dataset of bathymetry and backscatter data from the 45°30'N area on the axis of the Mid-Atlantic Ridge (MAR). The amplitude of a backscattered signal is dependent upon two factors: the intrinsic backscattering strength of the seafloor material and the angle with which the acoustic beam strikes the seafloor. Since the EM12 gathers high resolution bathymetry readings alongside the backscatter data, accurate bottom slope measurements may be determined. In turn, these allow the calculation of true angles of incidence of the sonar beams, knowing their angles of arrival at the transducer faces.

This investigation of the EM12 backscatter and bathymetry characteristics was conducted on the December 1990 dataset, which was logged with each amplitude sample positioned relative to the ship; the beam amplitude waveforms were not retained. de Moustier and Matsumoto (1993) state four methods for characterising co-registered backscatter and bathymetry data (chapter 2, section 2.1), though only two of these are possible with these data. Ideally, one is able to determine whether the measured backscatter variations are due to incidence angle or seafloor type by deriving an angular dependent function of acoustic backscatter for the survey area, and normalising all backscatter measurements to it. Alternatively, if the angular responses themselves are sufficiently distinct between bottom types, then they may provide a suitable classification characteristic. A desired output would be a geographical map of pure backscattering strengths for the survey, providing a classification of the lithological characteristics of the seafloor material.

Many of the conventional interpretations of sidescan datasets are purely qualitative, but here, with this co-registered dataset of bathymetry and backscatter information, I attempt a completely quantitative approach. Ideally, this will lead to a classification of bottom type, but at least will provide important information for sidescan interpreters to consider, regarding how various instrumental and geological features affect their sidescan observations.

5.2 PREVIOUS WORK

5.2.1 Background

Figure 5.1 illustrates some of the processes by which sound is scattered back from the seabed. Johnson and Helferty (1990) outlined some of these backscattering characteristics, whereby a sonar pulse is received as a variable combination of two forms: backscattered energy, which is diffracted due to the microtopography of the bottom material; and specularly reflected energy.

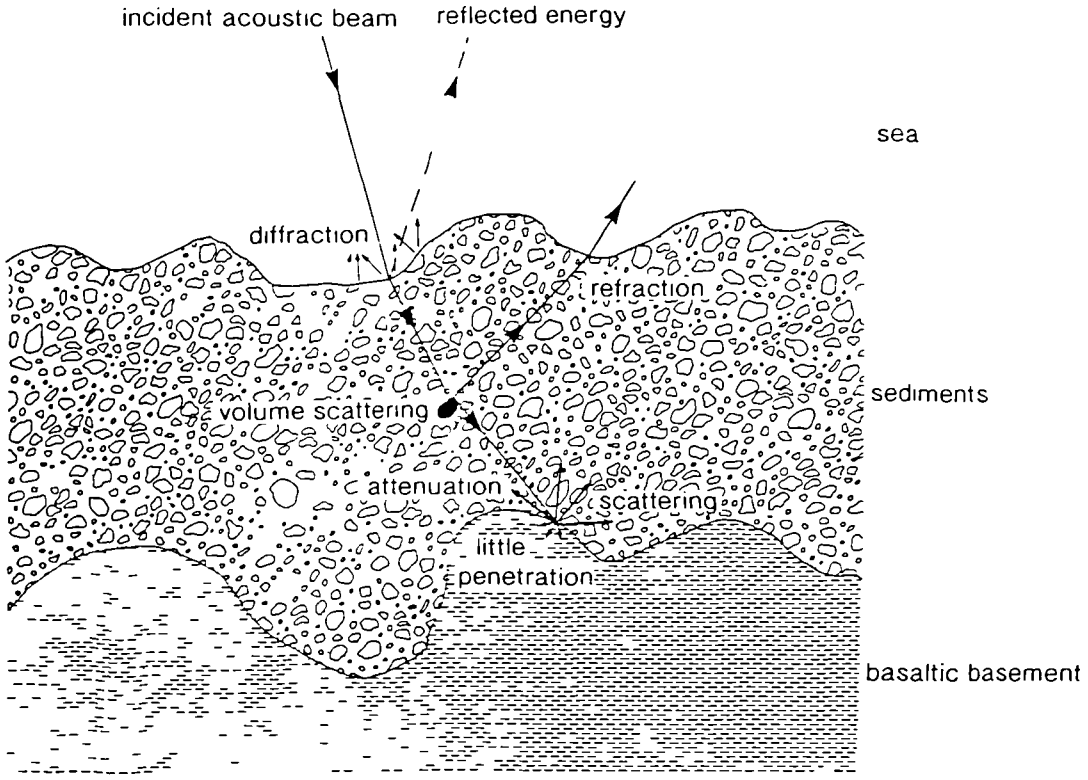


Figure 5.1 Acoustic scattering behaviour at a typical seawater/seafloor interface.

As the incident beam strikes the seafloor, it is backscattered firstly due to the impedance contrast between the seawater and the material comprising the upper surface of the seafloor. Acoustic impedance is defined as ρv (bulk density × compressional wave velocity). The bulk density is primarily controlled by the porosity of the material; whereas the compressional wave velocity is dependent upon the bulk modulus and bulk density. The upper layer in this example may be considered to be composed of sediments (and in varying thicknesses over the MAR survey area). Hamilton and Bachman (1982) and Hamilton (1970) summarised some

basic physical properties of marine surface sediments, corrected to a temperature of 23°C and a pressure of 1 atm. Some relevant parameters are presented below:

	overall bulk density /g cm^{-3}	sound velocity /ms^{-1}	impedance /g cm^{-2} s^{-1} ×10^5	Rayleigh reflection coefficient
abyssal plain (turbidite)	1.454 - 1.352[a]	1528 - 1503[a]	2.6795 - 1.8911[b]	0.2627 - 0.0941[b s/w]
abyssal hill (pelagic)	1.347 - 1.414[a]	1522 - 1493[a]	2.1615 - 2.1118[b]	0.1596 - 0.1477[b s/w]
seawater	1.0475[b]	1530[b]	1.6027	
basalt	2.7 - 3.2	6000 - 7000	16.2 - 22.4	0.8199 - 0.8665 [b/ps]

[a] from Hamilton & Bachman (1982)
[b] from Hamilton (1970)
[s/w] sediment/water coefficient
[b/ps] basalt/pelagic sediments coefficient

Table 5.I - Some marine sediment properties

The Rayleigh reflection coefficient is defined in Hamilton (1970) as:

$$R = \frac{\rho_2 v_2 - \rho_1 v_1}{\rho_2 v_2 + \rho_1 v_1}$$ at normal incidence

where $\rho_1 v_1$ is the impedance of the first medium (e.g. water), and $\rho_2 v_2$ is the impedance of the second medium (e.g. sediments). The coefficient expresses the ratio of the amplitude of the reflected wave to the incident wave. For example, at the water/sediment interface, 15% of the energy is reflected and 85% transmitted into the sediment; at the sediment/basement rock interface, 85% of energy is reflected and only about 15% is transmitted. It is important to note that this is a very simplified model, and the results become less reliable for rough surfaces. Mitchell (1989) inferred that little refraction would be expected at the water sediment interface, since Snell's Law states that

$$\sin\theta_b = \left\{1 - \left[\frac{v_{sed}}{v_{water}}\cos\theta_g\right]^2\right\}^{1/2}$$

where θ_g is the grazing angle of the sonar beam at the transducers, θ_b is the grazing angle at the seafloor after refraction through the water column, v_{water} is the sound velocity in water and v_{sed} is the sound velocity in the sediments.

Surfaces are described as acoustically smooth or rough. Generally, smooth surfaces can be defined as possessing topographic wavelengths and amplitudes of less than half the acoustic wavelength (12 cm is the acoustic wavelength of the EM12), and rough surfaces as possessing larger topographic wavelengths (Rayleigh, 1894). More

specifically, the 'Rayleigh parameter' distinguishing smooth and rough surfaces states that for a smooth surface, the amplitude of surface perturbations should be

$$< \frac{\lambda}{8 \cos \theta_g}$$

that is ~4.4 cm for the EM12 (assuming a grazing angle of ~70°). Ulaby *et al.* (1982) are more rigorous and define that the standard deviation of the surface roughness height should be

$$< \frac{\lambda}{32 \cos \theta_g}$$

which is 1.1 cm for θ_g = 70°. Therefore, flat lying sediments with a grain size of a few microns (Hamilton and Bachman, 1982), would be described as acoustically smooth, and pillow lavas, about 1 m in size, would be acoustically rough.

5.2.2 Theoretical

Jackson *et al.* (1986) attempted to model the backscattered signal from sediments and assumed that backscatter was a combination of both interface and sediment volume scattering. For grazing angles of less than 70°, Jackson *et al.* (1986) used the *composite roughness model* (equation #21 in Jackson *et al.*, 1986) to model the interface scattering component, based on ocean surface modelling concepts presented by McDaniel and Gorman (1982, 1983). Here, the interface is treated as the sum of scattering from two roughness scales, where the sound is scattered by diffraction from surfaces with a radius of curvature less than the acoustic wavelength and reflected randomly from surfaces with roughness scale greater than or equal to the acoustic wavelength. For steep grazing angles (> 70°), the entire composite roughness model is replaced by the *Kirchoff approximation* for steep angles to model the interface scattering term (equation #38 in Jackson *et al.*, 1986), which assumes that the scattering surface is relatively smooth, and so the radius of curvature of the seafloor surface must not be 'much smaller' than the acoustical wavelength. The sediment volume scattering term is based upon the composite roughness model. In all, the modelled backscatter is dependent upon five parameters: sound speed ratio, density ratio, 2 roughness parameters and a volume scattering parameter. These equations are quite complicated, so refer to Jackson *et al.* (1986) for their definitions and explanations; figure 5.2 presents some plots of the combined theoretical models. Jackson *et al.* (1986) confirmed that higher backscatter occurs from a rough surface than from a smooth one and that the volume scattering effect is relatively important compared with surface roughness scattering for soft, fine grained rather than sandy sediments.

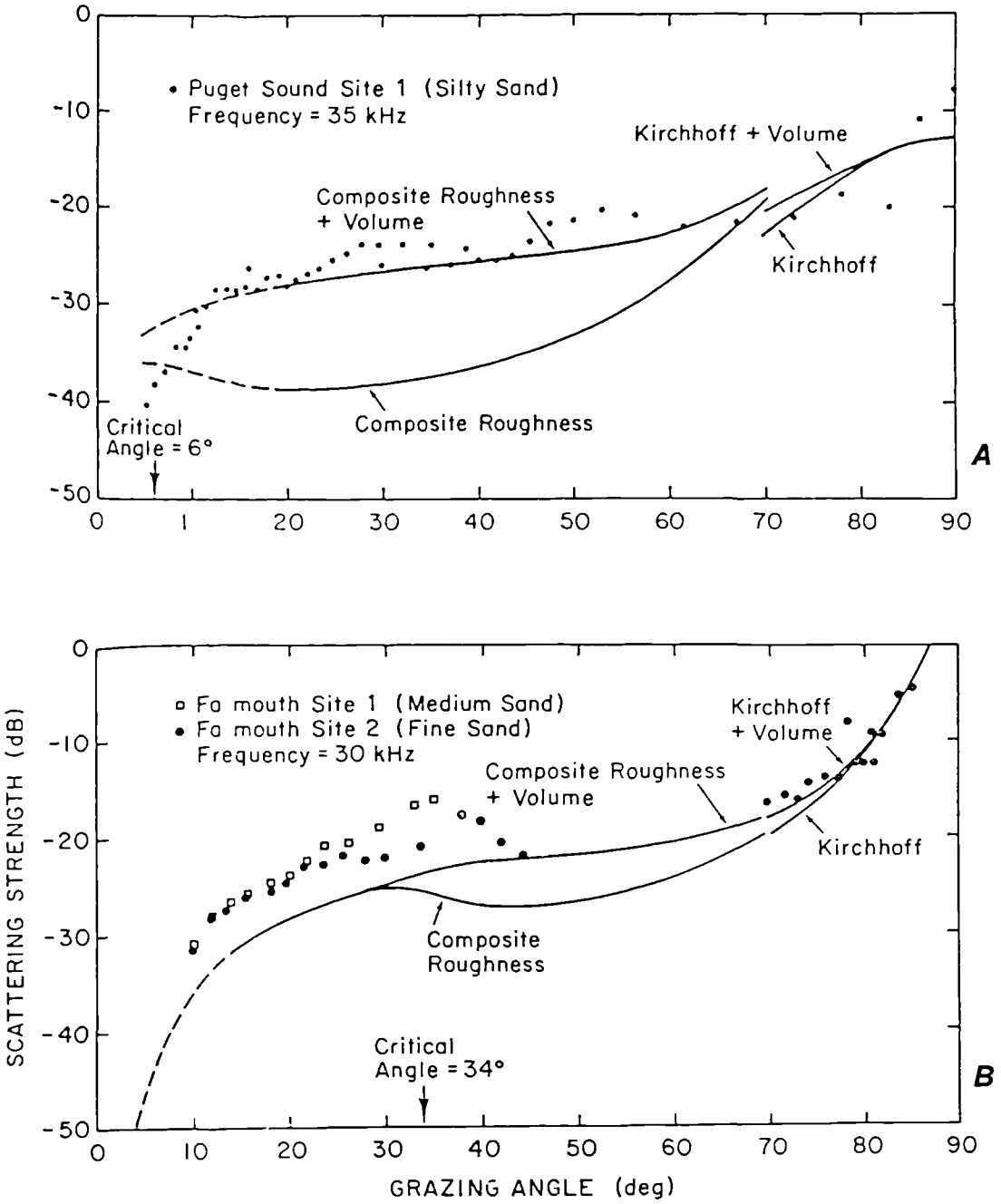


Figure 5.2 *Theoretical modelling of backscatter strengths against grazing angles (from Jackson et al., 1986). The individual points refer to actual data collected (from sedimented sites of varying grain size - parts (a) and (b) refer to the two experimental locations), while the continuous lines represent the theoretical models, indicating which components of the models have been used to generate each portion of the curve. A complete description is provided in Jackson et al. (1986).*

Caruthers and Novarini (1993) modelled swath bathymetry and sidescan sonar data using both Helmholtz/Kirchhoff theory described in Caruthers *et al.* (1992), based on the principle that the complex scattered field for a point source and a 2-D surface is calculated according to a cosine squared insonification function. Caruthers and Novarini (1993) also used the *bistatic scattering strength model*, which is based on the composite roughness theory, where diffuse scattering from the microroughness of the seafloor surface based on Lambert's Law is combined with a component due to scattering from larger scale features. Caruthers and Novarini (1993) compared these models with data from a site on the western flank of the Mid-Atlantic Ridge (~26°N) and from simulated geomorphology, and concluded that the bistatic scattering strength model provided a good representation of the more complicated Helmholtz/Kirchhoff model.

In addition to the volume scattering phenomenon, it must be noted that sound energy will quickly attenuate as it passes through sediments. Kibblewhite (1989) summarised the attenuation literature and suggests that sound energy attenuates at a rate of 0.4-1.0 dB m^{-1} for the 13 kHz frequency of the EM12 through fine abyssal sediments, compared with a standard rate in water (α in section 5.3) of 1.1 dB km^{-1} (Simrad, 1992). Mitchell (1993) estimates an attenuation rate of 0.2-0.4 dB km^{-1} for hemipelagic sediments, computed from GLORIA (6.5 kHz) backscatter strengths and a simple acoustical model. Relating these coefficients to the EM12 system, Stoll (1985) notes that the assumption of a constant attenuation coefficient with varying frequency is unacceptable.

In figure 5.1, I have illustrated basement material below the sediment layer. At a mid-ocean ridge, this would be typically composed of basalt. Due to a high impedance contrast for a solid rock, little sound would penetrate through and the majority would be scattered back into the sediments. Experimental work described below clearly confirms the recorded backscatter difference, and the consequent contrasts of impedance and small scale roughness contrast between bare rocks and fine-grained sediments.

5.2.3 Experimental

It must be noted that the seafloor backscatter strengths recorded by the EM12 (and EM1000, GLORIA etc.) are uncalibrated, so specific numerical comparisons between these and other datasets are not valid. However, the general broad relationships identified by other workers regarding backscatter strength and angular response differences between bottom types should be considered.

Workers such as McKinney and Anderson (1964) and Wong and Chesterman (1968) presented plots of acoustic backscatter against grazing angle for various bottom types, measured with simple transducer configurations at frequencies 100 and 48 kHz, respectively. They identified a relationship between the proportion of the sonar energy backscattered and the particulate nature of the seafloor, where the highest average backscatter values were associated with exposed bedrock and gravel, and the lowest readings corresponded with fine muds (refer to table 5.II).

seafloor material	average backscatter /dB	source
bare rock	-18	M&A '64
gravel	-18	W&C '68
sand	-24	M&A '64
	-28	W&C '68
silt	-32	M&A '64
clay/mud	-38	M&A '64
	-43	W&C '68

Table 5.II - Intrinsic backscatter strength for various seafloor materials

Additionally, all these lithologies exhibited an approximately Lambertian backscatter dependence on grazing angle. Lambert's Law, defined in Urick (1983) as a rule for scattering of both light and sound from very rough surfaces, states that the magnitude of acoustic backscatter is proportional to the square of the sine of the grazing angle. However, the specific shape of the response curves did vary according to bottom type. For large grazing angles, high value specular reflection and sediment volume scattering became significant. Patterson (1967) observed an interruption to this Lambertian dependence, where, as backscatter strength was expectedly decreasing with decreasing grazing angles, it increased again at a grazing angle of 30°. He inferred this could be possibly due to the presence of bare rock outcrops amongst sediments, confirmed with ground truth photography. This point is relevant to mid-ocean ridge geology, characterised by numerous fault scarps, volcanic cones and ridges, interspersed with flatter areas where sediments are more likely to settle.

For comparison purposes with the EM12 data, it is necessary to consider the variation of backscatter with frequency of the sonar equipment. Urick (1983) describes some results suggesting that frequency dependence varies with bottom type. Rough, heavily dissected bottoms, such as those at the MAR, showed no backscatter dependence with frequency over a range of 1-30 kHz; whilst smooth sedimented bottoms did exhibit a strong frequency/backscatter relationship .

Nolle *et al.* (1963) physically modelled some of these acoustic phenomena in a water filled tank, and discovered that bottom roughness rather than bottom type, affects the measured backscatter strength. They noted that only a minor roughening of the smooth sand at the bottom of the tank considerably increased the backscatter return.

Gardner *et al.* (1991) compared GLORIA 6.5 kHz long range sidescan sonar backscattering characteristics (uncalibrated also) with detailed sediment coring and photographic ground truthing. They found that there was a correlation between sediment character and backscatter, though this sometimes yielded unexpected results. For instance, a predominantly sand filled core produced lower backscatter with GLORIA than one mainly composed of mud. Also, they deduced from photographic evidence, that bed roughness had a negligible effect on backscatter return.

More recent observations of the angular dependence of backscatter data have been made by de Moustier (1986). He derived backscatter strength data from the Sea Beam 12 kHz multibeam echosounder over three ground-truthed terrain types: manganese nodules, lava flows and hemipelagic sediments. de Moustier (1986) deduced that backscatter strength was more sensitive to bottom type for the manganese nodule area, to bottom roughness over the lava flows and a combination of the two over the sediments. Plots of backscatter strength versus grazing angle yielded a considerable difference in backscatter level between the three sites, but little variation in the angular response was exhibited.

de Moustier and Alexandrou (1991) looked at Sea Beam 12 kHz multibeam echosounder backscatter data collected over deeply sedimented areas and iteratively found a good agreement of backscatter strength with the theoretical functions of Jackson *et al.* (1986). Due to the geometry of the Sea Beam system, similar in characteristics to the EM12, only beam angles of 0-20° from vertical are achieved, so the Kirchoff approximation model for steep angles was employed for the interface scattering term. As mentioned earlier, the radius of curvature of the insonified area should be large compared with the acoustic wavelength; this was a reasonable assumption here where backscatter from smooth sediment fields was being measured. Volume scattering was modelled according to Jackson *et al.* (1986), but the shadowing term was omitted due to the small beam angles of the Sea Beam system.

de Moustier and Alexandrou (1991) deduced that the backscatter signal was little affected by any contribution from sediment volume scattering.

Talukdar and Tyce (1992) used Fox and Hayes' (1985) power law description of seafloor bathymetry in conjunction with Jackson *et al.*'s (1986) backscattering model, to model backscatter variations with incident angles out to ±20° for three contrasting geological areas. Talukdar and Tyce (1992) stated that the Sea Beam bathymetry is inadequate to truly infer the spectral characteristics of the topography, suggesting that higher resolution bathymetry is really desirable. Through curve fitting, they did however confirm the influence of small-scale roughness in an abyssal plain area on backscatter caused by Rayleigh scattering, and the dominance of large-scale roughness effects over the mid-ocean ridge.

Dziak *et al.* (1993) applied curve fitting procedures to fit Jackson *et al.*'s (1986) model to Sea Beam backscatter data, in an attempt to estimate topographic power spectra from a site on the Juan de Fuca Ridge. Although about 100 m is the minimum power spectral limit for Sea Beam data, this application enabled the spectra to be estimated at the centimetre scale. The technique successfully discriminated ground-truthed characteristics over the study area.

Hughes Clarke (1993) described an experiment conducted in the Bay of Fundy, Canada; a site experiencing particularly large tidal ranges. Multibeam sonar surveys, using Simrad EM100 and EM1000 systems, the shallow water equivalents to the EM12 equipment used in the MAR survey and with frequencies of 95 kHz, were completed whilst the tide was in and direct detailed ground truthing was accomplished during the inter-tidal periods. Hughes Clarke (1993) presented backscatter angular response curves for the 95 kHz system; the highest backscatter measurements of approximately -20 dB were associated with gravelled areas, then lower readings over glacial till (~ -23 dB), then bedrock (~ -30 dB) and finally the lowest strengths came from the fine grained mud flats (~ -35 dB). The backscatter amplitude decreased with increasing incidence angle for all bottom types, but at differing rates. Hughes Clarke (1993) concluded that at this stage, although a tentative angular correction could be made to dampen this incidence angle effect on backscatter variation, it was not possible to totally remove the angular variation effects for a complete seafloor classification, and possibly the shape of the angular response curves may provide an additional characteristic of bottom types as well as the mean backscatter level.

Finally, with regard to mid-ocean ridge research, Mitchell and Somers (1989) calculated backscatter strengths from GLORIA 6.5 kHz long range sidescan sonar data and Sea Beam bathymetry information. These data, from the South West Indian ridge, exhibited characteristically complex mid-ocean ridge terrain, but nevertheless yielded a Lambertian backscatter dependence on grazing angle. Mitchell (1991) deduced that the largest backscatter variations were in fact due to variations in seafloor material, where a backscatter variation of 20-30 dB was measured between sediments and basement rock. Bottom slope only provided a secondary effect, with backscatter variations of 10-15 dB.

Summarising, backscatter variations decrease with grazing angle and have been modelled to fit both Lambertian and more sophisticated Helmholtz/Kirchoff, composite roughness, and bistatic scattering strength model responses. Bottom type affects backscatter strength, with gravels and bedrock yielding around a 25 dB increase in backscatter strength compared with fine muds, and a different angular response shape. Additionally, the roughness of the bottom material considerably affects the backscatter return.

5.3 EM12 BACKSCATTER MEASUREMENTS

Before proceeding with an analysis of the EM12 backscatter data collected at the MAR, it is important to note precisely what data are being dealt with. That is, in what form is the backscatter data logged by the EM12 operating system?

Urick (1983) defines the fundamental sonar equation for an echosounder as:

$$EL = SL - 2TL + BS \qquad (5.1)$$

where EL is the received echo level, SL the transmitter source level, $2TL$ the two way transmission loss in the water column and BS the backscattering strength of the target. The transmission loss (in dB) is related to the range, R, and absorption coefficient in water, α, by:

$$2TL = 2\alpha R + 40\log R \qquad (5.2)$$

The backscattering strength depends upon the seabed conditions, defined by the intrinsic backscattering strength of the seafloor, BS_B, and the insonified area, I, as:

$$BS = BS_B + 10\log I \qquad (5.3)$$

where

$$I = \theta_T \theta_R R^2 \qquad \text{around normal incidence } (\phi \approx 0) \qquad (5.4)$$

$$I = \frac{\frac{1}{2} c \tau \theta_T R}{\sin \phi} \qquad \text{elsewhere} \qquad (5.5)$$

and ϕ is the angle of incidence, clarified in figure 5.3. θ_R is the receiver beamwidth, θ_T, the transmitter beamwidth, τ the pulse length and c is the velocity of sound in sea water. Therefore, putting equations (5.1), (5.2) and (5.3) together gives:

$$EL = SL - 2\alpha R - 40\log R + 10\log I + BS_B \qquad (5.6)$$

The EM12 outputs these backscatter strength values (BS_B) compensated for the source level (SL), including transmission and reception beam pattern effects, and the absorption coefficient (α) (E. Hammerstad, personal communication, 1993). This value has had no time- or angle-varying gains applied to it (Pøhner and Lunde, 1990), just a series of linear gain steps to ensure the echo level remains within the dynamic range of the recorder. Hammerstad (personal communication, 1993) states that all gain variations have been compensated for in the logged data. There is no absolute calibration available for the echo level voltage measurements, so all the data must be regarded as purely relative values.

In order to reduce the dynamic range of the recorded data, the backscatter measurements have been modified according to Lambert's Law based on grazing angles computed assuming a flat bottom (Simrad, 1992). These modified backscatter values logged by the EM12 will be referred to as 'logged' backscatter in the future. As explained in section 5.2.3, Lambert's Law has often been used to describe the backscattering function with varying incidence angle, ϕ, again refer to Urick (1983); Simrad (1992) assume this for all incidence angles greater than 25°:

$$BS_B = BS_N \qquad \text{around normal incidence } (\phi \approx 0)$$
$$BS_B = BS_O \cos^2 \phi \qquad \text{elsewhere, where} \qquad (5.7)$$

BS_N is the intrinsic backscattering strength of the target at normal incidence, and BS_O is the strength at oblique incidence. Putting all of the above into a single equation gives (Simrad, 1992):

$$EL = SL - 2\alpha R - 20\log R + 10\log(\theta_T\theta_R) + BS_N \qquad (\phi=0)\ (5.8)$$

$$EL = SL - 2\alpha R - 30\log R + 10\log\left(\frac{\frac{1}{2}c\tau\theta_T\cos\phi}{\tan\phi}\right) + BS_O \qquad (\phi>10°)\ (5.9)$$

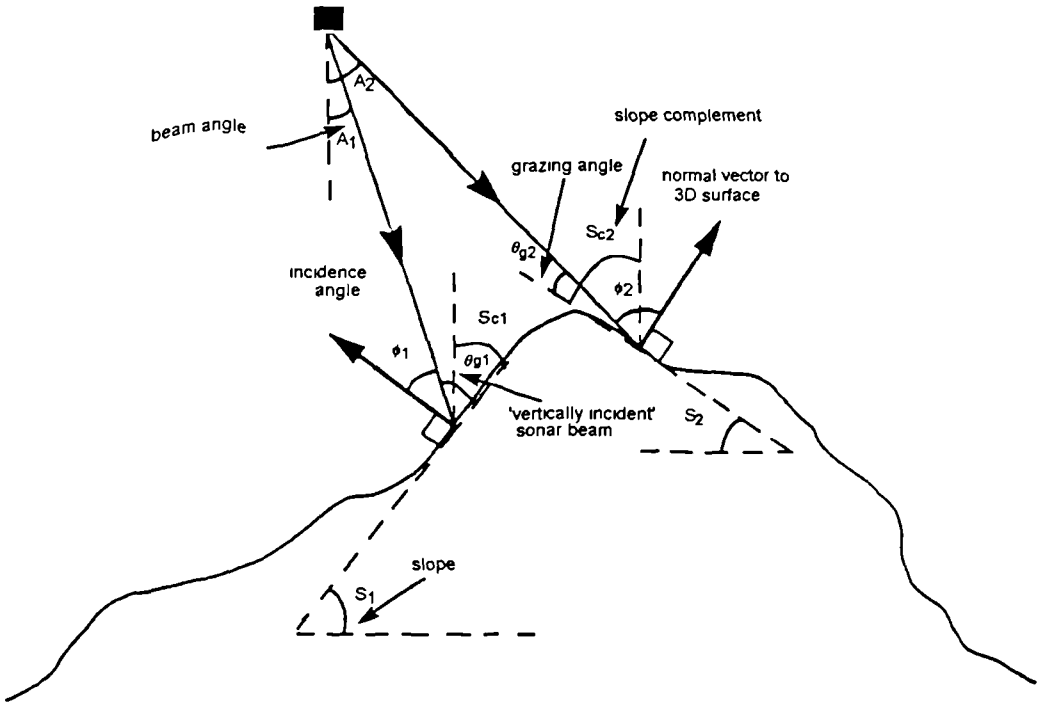


Figure 5.3 *Geometry of sonar beam: two examples (subscripted '1' and '2'), defining the various angles referred to in the text. A is the beam angle; S the seafloor slope; ϕ the angle of incidence; θ_g its complement, the grazing angle; and S_c is the slope complement.*

In order to compute the 'primary' backscatter strengths (BS_M) measured at the transducers from the 'logged' backscatter values (BS_C), E. Hammerstad (personal communication, 1993) suggests performing the following correction:

$$BS_M = BS_C - 10\log\left[\left(\frac{\cos\phi}{\tan\phi}\right)\left(\frac{\tan A}{\cos A}\right)\right] \qquad (5.10)$$

where A is the incidence angle assuming a flat bottom, i.e. the beam angle (refer to figure 5.3). Equation 5.11 removes the flat-bottom Lambert's Law correction and takes into account along-track slopes in the area correction (E. Hammerstad, personal communication, 1993):

$$BS_M = BS_C - 10\log\left[\left(\frac{1}{\sin\phi}\right)\left(\frac{\tan A}{\cos A}\right)\right] \qquad (5.11)$$

5.4 ANALYTICAL METHOD

5.4.1 Experimental Sites

The basis for the quantitative analysis of the acoustic backscatter and bathymetry data was the selection of small experimental sites from the survey area. These areas were selected purely from the bathymetry data and their corresponding geological interpretation; however, to ensure reasonable backscatter data quality, these data were consulted too.

We collected no direct ground truthing data for the area, but detailed dredging, coring and bottom photography were accomplished in the 1960s (summarised in Aumento *et al.*, 1971). Also, we can infer lithogical knowledge about this area of the MAR, through comparison of the bathymetry from other regions of the ridge and ground truth data collected there. Ground truthing includes rock dredging, bottom photography and also, to a degree, high resolution sidescan sonar (e.g. Cann *et al.*, 1992).

Eight areas were chosen (refer to figure 5.4 for locations), some with the aim of isolating single homogeneous terrain types for backscatter dependence derivations, e.g. purely bare rocks or deep sediments. Ideally, for a sufficient range of incident beam angles (0-45°) for this 'single-pass' data (Hughes Clarke, 1993, had the benefit of overlapping swaths), these experimental areas must encompass at least half a swath's width of data; however a balance must be struck between the size of the site and the likelihood of encountering more than one bottom type within the site. Potentially composite seafloor type areas, e.g. a mixture of bare rocks and sediments within a single site, were also selected for analysis, more characteristic of a mid-ocean ridge, in order to try and characterise regional backscatter trends due to geological processes.

An axial volcanic ridge (AVR) from within the median valley (site G) would be composed of freshly erupted, hummocky basalt, with little or no sediment cover (Cann *et al.*, 1992), forming a single terrain type. A further three volcanic ridges (J, T and H) were selected with increasing distance from the ridge axis (7, 17 and 19

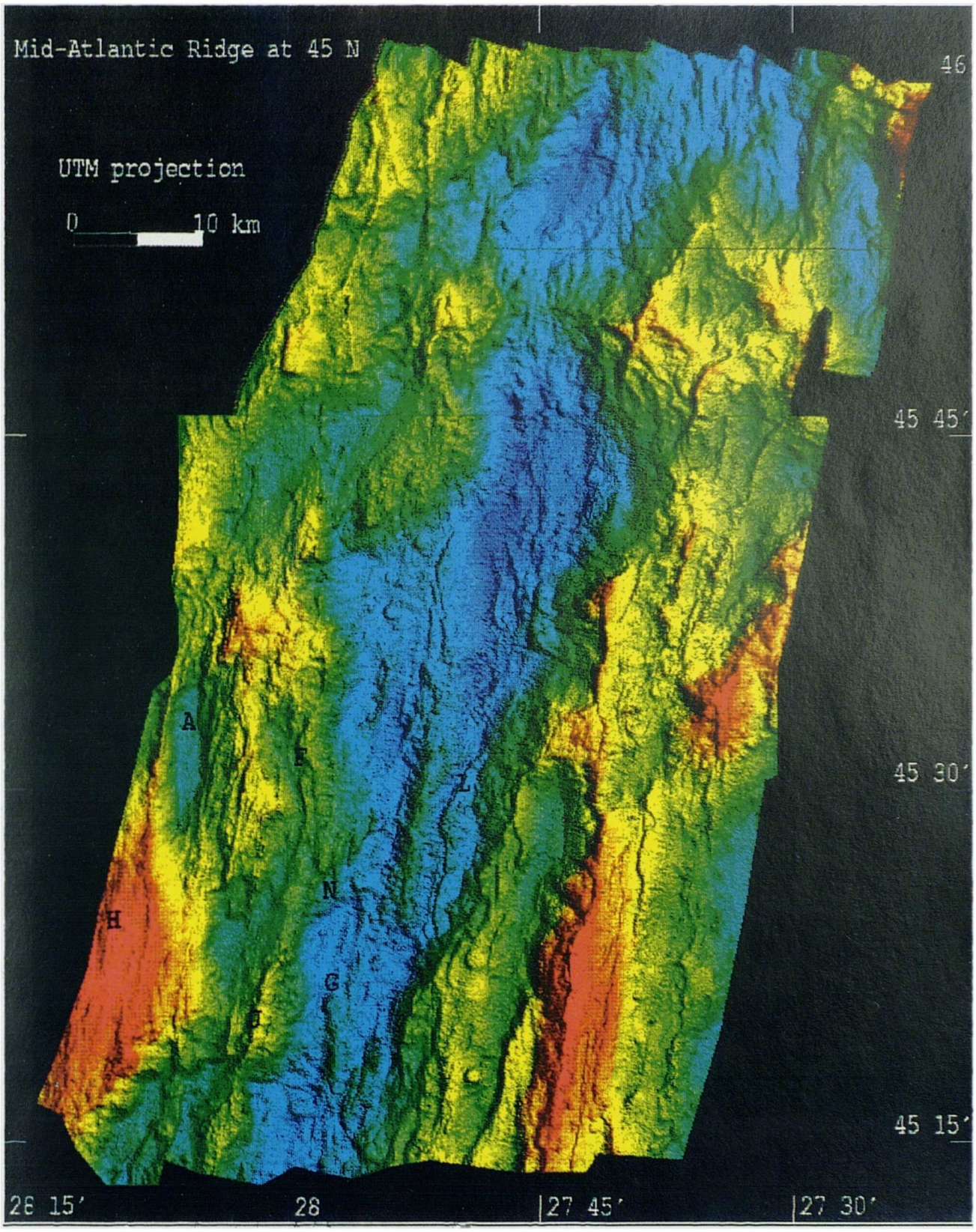


Figure 5.4 *Location of experimental sites on shaded relief bathymetry image (figure 4.24).*

km, respectively), the geology of which will be described alongside the results; obviously these will be considered to comprise composite terrain types, due to the variety of geological processes at work (e.g. sedimentation and faulting). A portion of an off-axis basin (A), which would be expected to form a pond for sediments, was selected as another single homogeneous terrain type. Keen and Manchester (1970) confirmed this ponding effect with detailed sediment profiling conducted in the vicinity, i.e. over a basin just to the west of the major peak in the south west of the survey area, where they concluded that sediments tended to slump down into off-axis valleys from the surrounding ridges and estimated an average sediment thickness of 25 metres. One of the landslides (N), identified from its three dimensional morphology (Valsami, E., J.R. Cann, R.C. Searle, M. Ackers, and J.A. Keeton, Morphological characteristics of the Mid-Atlantic Ridge at 45°N: implications for magmatic and tectonic processes, manuscript in preparation, 1993), was also chosen as a homogeneous experimental site, which may show the effects of roughness on the backscattered signal from talus, the broken up remains of a fault scarp. Finally, since mid-ocean ridges are dominated by numerous fault scarps, two of the experimental sites (F and L) are sections of opposite bounding walls of the median valley. These last sites are of composite terrain type as small multiple fault scarps appear to be interspersed with flatter regions.

5.4.2 Techniques

Numerous plots of backscatter against grazing angle have been produced for the experimental sites described above. No ray bending corrections were applied prior to the calculation of grazing angles, but de Moustier (1986), amongst others, concluded that at such small beam angles, the refraction effects are inconsequential. The backscatter mosaics presented in figures 4.17, 4.24 and 4.25 comprise 'logged' backscatter values (the BS_C term of equation 5.11); it is therefore appropriate to consider analytical work on these data as well as the true acoustical relationships of backscatter strengths and grazing angles for comparison purposes. Additionally, it is interesting to investigate whether there is any backscatter dependency with slope distributions, e.g. is the intrinsic backscattering strength of the fault scarps higher than that of the flatter, possibly sedimented, regions inbetween? Grazing angle variations between 90 and 45° may result from either the simple beam angle span over flat terrain, or the interaction of the beam angles with rough terrain. Looking at backscatter and slope distributions may provide a further characterisation in this respect, as well as possibly confirming whether or not the true grazing angle responses are artifacts. The following analyses therefore consider 'logged' backscatter strengths (BS_C) versus the complement of seafloor slopes (the contrived equivalent

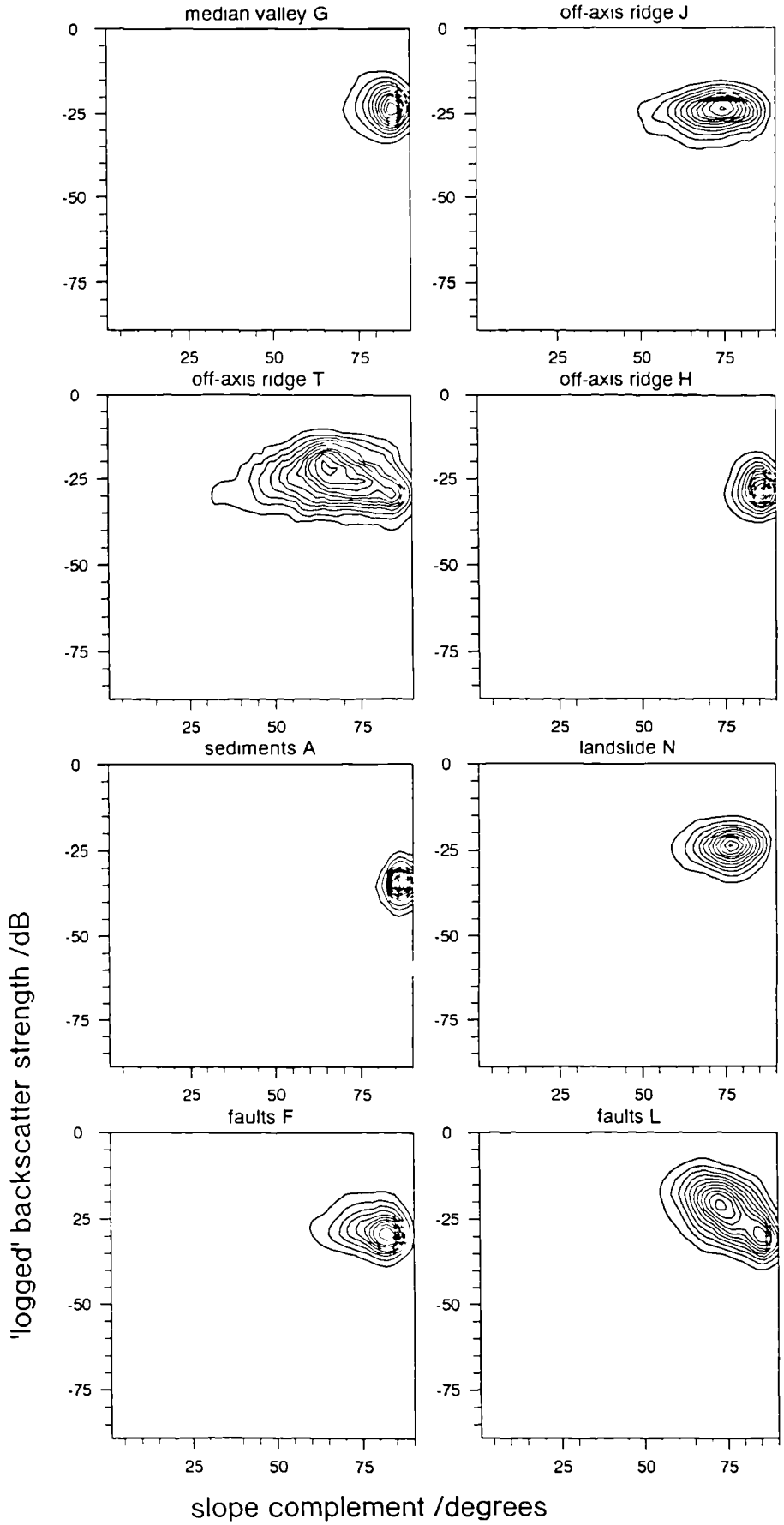


Figure 5.5 *Contour plots of 'logged' backscatter strength (BS_c) (dB) against slope complement (degrees), for all eight experimental sites. The contours represent 1 dB/1 degree bins and the contour values increment by 2 pixels, with the lowest at a frequency of 2.*

to grazing angle for the 'logged' backscatter data - see next section); and 'primary' backscatter strengths (BS_M) (gained following the removal of the first order Lambert's Law correction) versus grazing angles (computed from beam angles and a three dimensional terrain model).

5.4.2a Logged backscatter versus seafloor slope complement Firstly, some plots were constructed to investigate the relationship between the sidescan mosaic and bathymetry images. The first order Lambert's Law correction assuming a flat bottom has been applied to these backscatter data before logging. For a direct comparison with the grazing angles in the next section, the complements of the seafloor slope were selected as suitable representations of the bathymetry data. As the backscatter variation through the beam angle sweep has been removed assuming all scattering is Lambertian, the data are as they would have been recorded with a uniform, vertically incident beam angle. Therefore, the effective grazing angle in this case is, in fact, the complement of the seafloor slope. The slope complement is obviously 90° for a horizontal seafloor, and 0° for the vertical case. These were computed as shown in figure 5.3, where two examples of the slope complement definition are illustrated. In each example, S_c represents the slope complements, ϕ denotes angles of incidence, θ_g grazing angles, S the slope of the seafloor (shown in two dimensions rather than the three actually used) and A the beam angle.

The 'logged' backscatter strength (BS_C) versus slope complement data may provide information about whether there is any relationship between slope and intrinsic backscattering strength of the seafloor material. It is possible in this region that there may be a backscatter dependence on seafloor slope, not purely grazing angle, since the material comprising the high angle fault scarps may have an intrinsically higher backscatter strength compared with the flat, possibly sedimented, areas in between.

Figure 5.5 illustrates plots of this 'logged' backscatter (BS_C) against slope complement. The contours represent 1 decibel / 1 degree bins, with a 2 pixel contour interval, though the shape of the distribution rather than the actual contour values is most important, since the experimental sites extend over different sized areas.

5.4.2b Primary backscatter versus grazing angle The backscatter measurements logged by the EM12 have had a Lambert's Law correction applied, which must be removed in order to retrieve the 'primary' backscatter value measured at the transducers (BS_M). In effect, the *10 log (tan A / cos A)* term of equation (5.11),

section 5.3 should be removed from the 'logged' backscatter value (BS_C). Figure 5.6 plots this correction term and figure 5.7 illustrates an example of how the apparent backscatter variation is changed. Basically, the near-nadir backscatter is seen to increase considerably, whilst the outer beam backscatter values remain almost unaltered. It is not known what corrections the EM12 operating software has applied to this near-nadir peak in backscatter strength, so the data within 10° to port and starboard of vertical must be ignored.

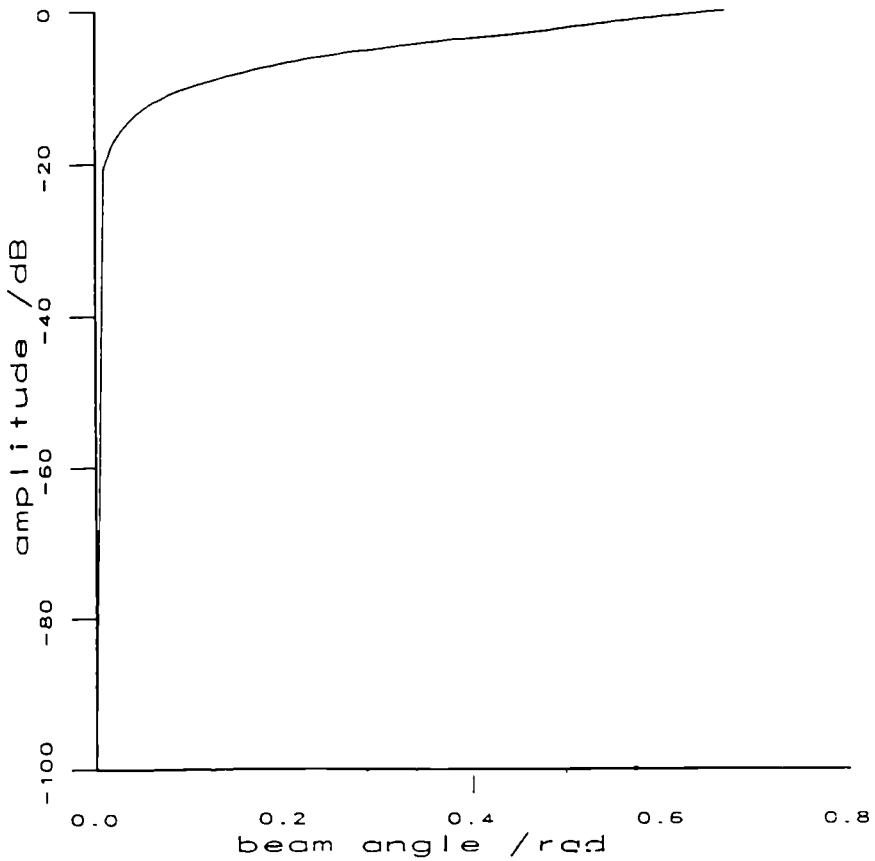


Figure 5.6 *Simrad's (1992) first order Lambert's Law correction function (assuming a flat bottom).*

A grazing angle, θ_g, can now be calculated using the beam angle information, A, (i.e. the angle with which the sonar beam is received at the transducers) and the bathymetry terrain model, through the determination of the normal vector to the three dimensional surface. Figure 5.3 displays the geometry, remembering three dimensions are actually considered rather than the two illustrated here.

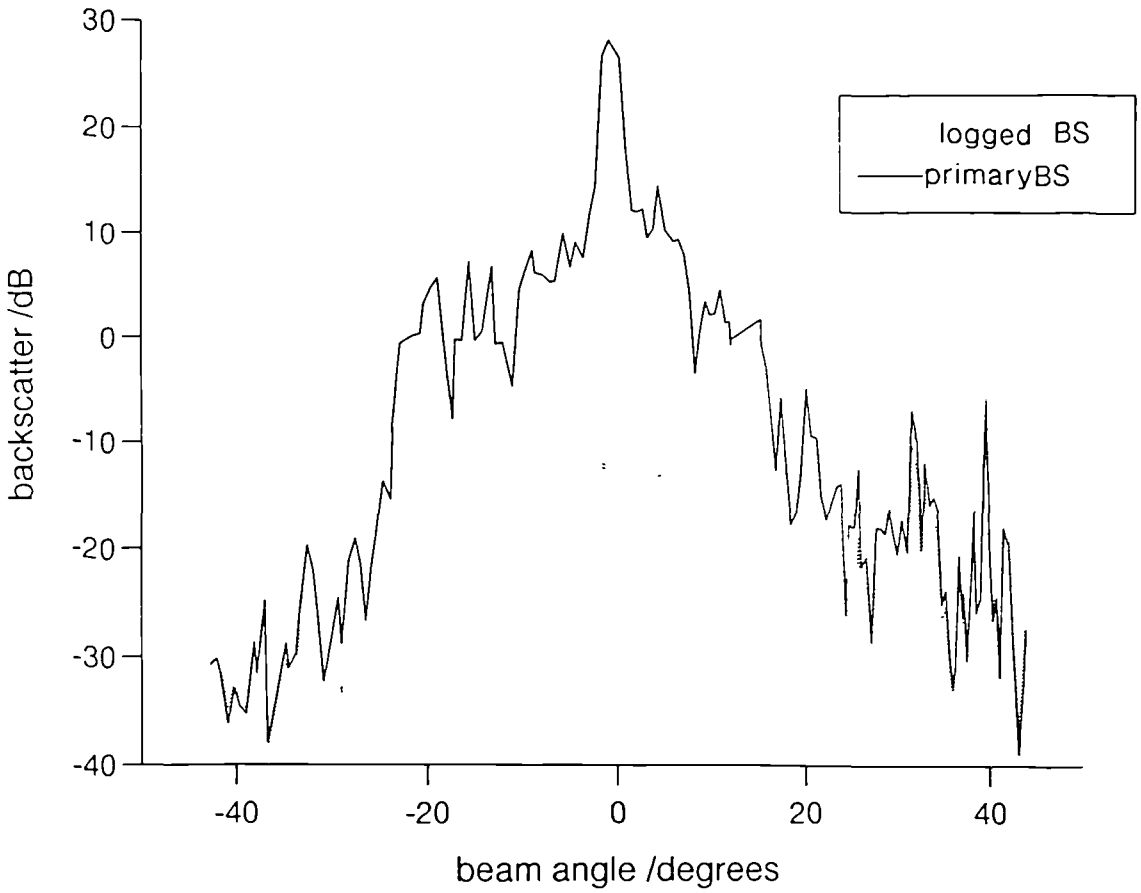


Figure 5.7 *The two curves illustrate the difference between the 'logged' (BS_C) and 'primary' (BS_M) backscatter strengths, plotted against beam angle A. The 'primary' backscatter strengths are derived from reversing the first-order Lambert's Law correction (figure 5.6) that is performed by the EM12 operating system prior to logging. Note that the precise nature of backscatter strengths near-nadir (± 10°) is not known.*

Obviously, it is important to note whether the bathymetry is sloping towards or away from the sonar equipment; an edge orientation image and gradient image were therefore constructed from the appropriate bathymetry data. The grazing angle configuration could now be compensated for accordingly. If the seafloor is sloping away from the transducers, and the seafloor slope is greater than the beam angle, the grazing angle is set to a value of zero.

Contour plots of 'primary' backscatter strengths (BS_M) plotted against grazing angle were constructed (figure 5.8), while plots of the modal backscatter value for

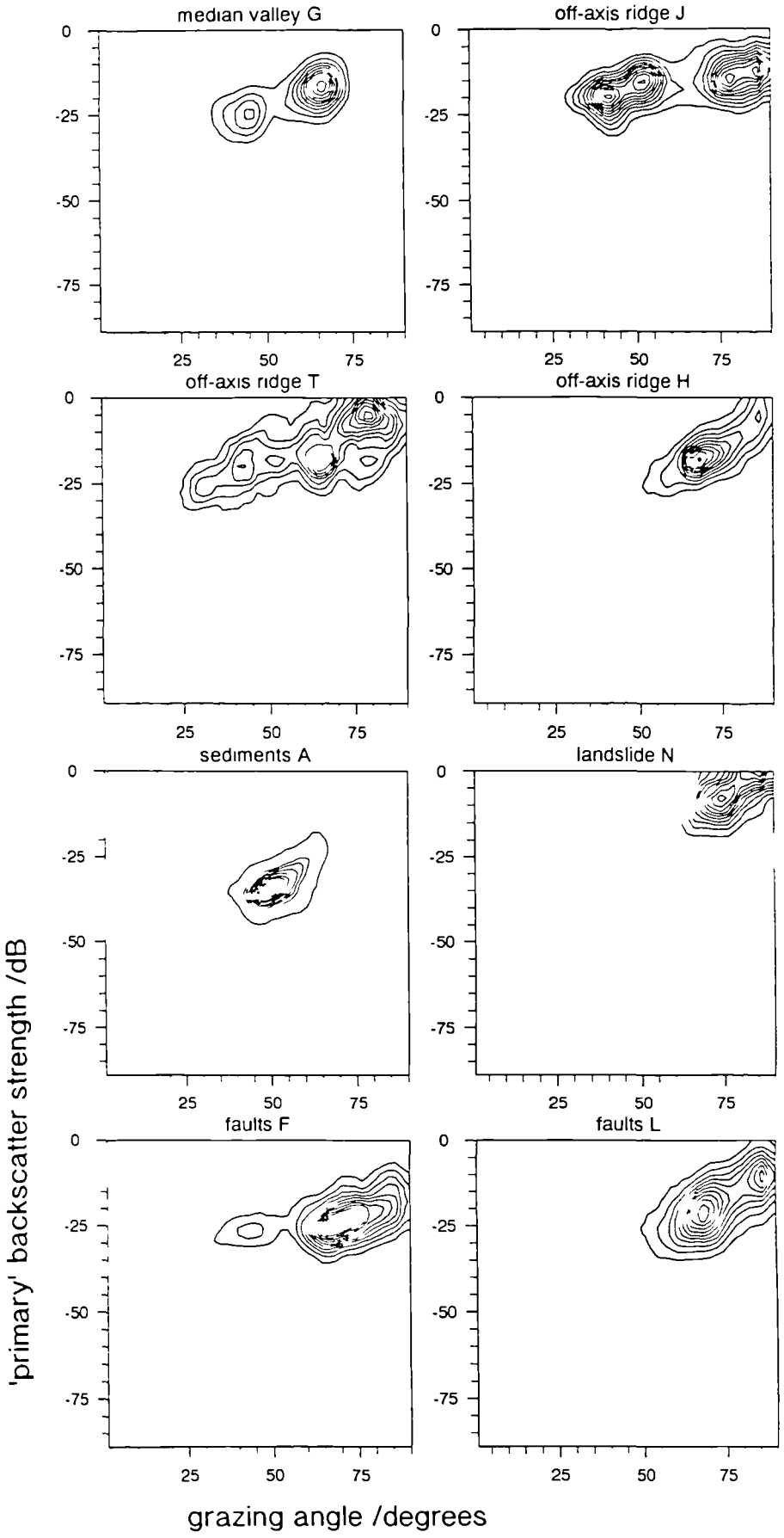


Figure 5.8 *Contour plots of 'primary' backscatter strength (BS_M) against grazing angle, for all eight experimental sites. The contours represent 1 dB/1 degree bins and the contour values increment by 2 pixels.*

each grazing angle were also produced for backscatter angular dependency determinations (figure 5.9).

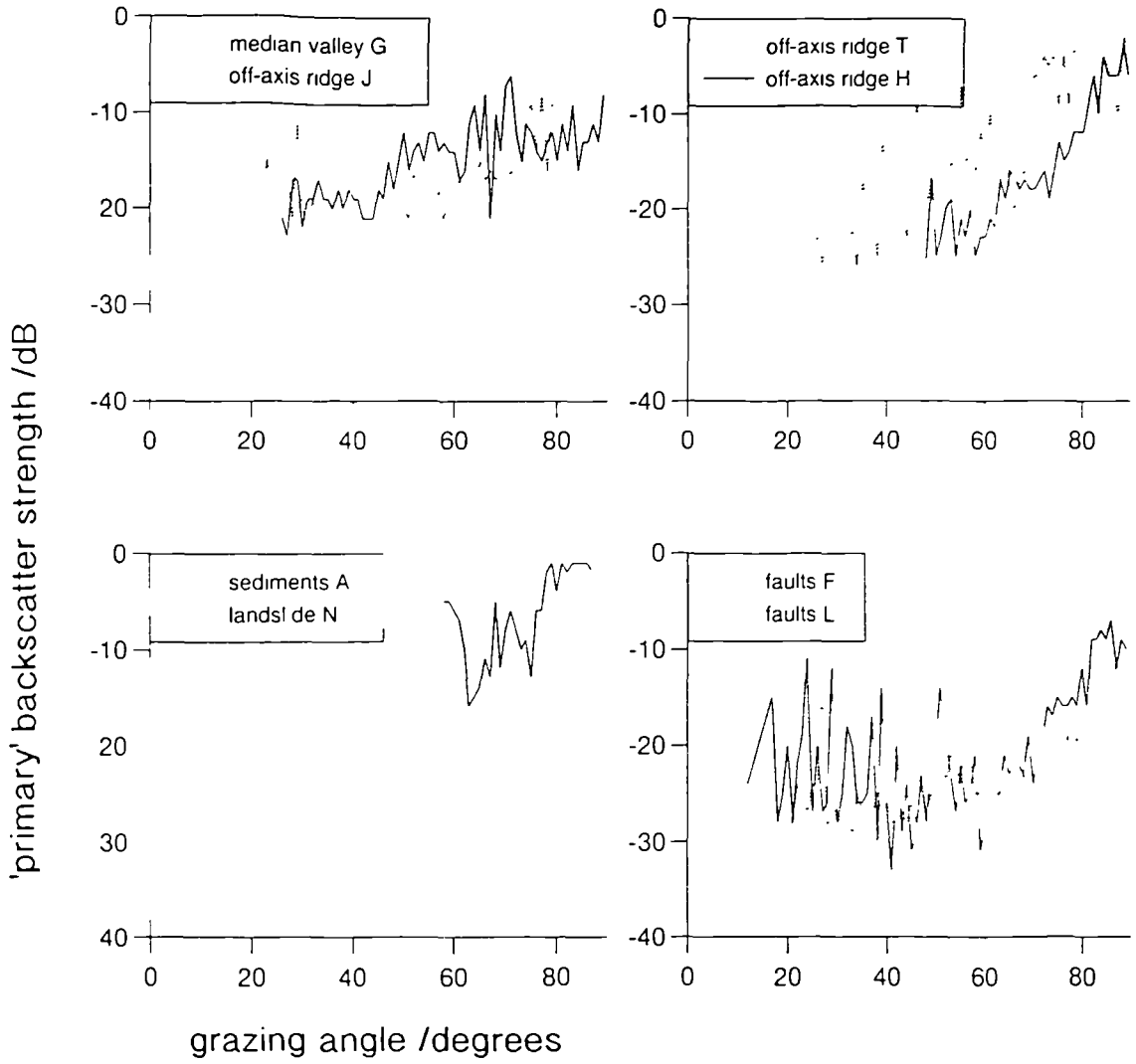


Figure 5.9 *Modal 'primary' backscatter strengths plotted for each 1 degree bin of grazing angle, for the eight experimental sites.*

5.5 ANALYTICAL RESULTS

5.5.1 Logged backscatter versus slope complement

Table 5.III summarises some basic statistics from the plots of figure 5.5:

Locality	BS mode /dB	Min BS /dB	Max BS /dB	S_c mode /°	Min S_c /°	Max S_c /°
G - AVR(1)	-22	-33	-12	85	70	90
J - off-axis ridge (2)	-24	-35	-15	72	50	88
T - off-axis ridge (3)	-23	-40	-10	65	30	90
H - off-axis ridge (4)	-30	-39	-18	85	75	90
A - sedimented basin	-37	-45	-25	86	80	90
N - landslide	-25	-34	-18	77	56	85
F - fault scarps	-30	-39	-18	83	60	90
L - fault scarps	-20	-40	-8	70	56	90

Table 5.III - 'Logged' backscatter versus slope complement statistics

Two of the lithologically most diverse terrain types, each hopefully conforming to a single (but different) homogeneous bottom material, are the AVR and sedimented basin, denoted G and A respectively. Both exhibit a small range of 'logged' backscatter (BS_C) and slope complement, indicating that the seafloor is approximately horizontal (between 0 and 20° slope) within the resolution of the echosounder and the 'logged' backscatter is also approximately uniform over the slope range, suggesting that each of the two sites is of a homogeneous bottom type. The 'logged' backscatter (BS_C) of the two terrain types differs by some 15 dB. The landslide N possesses approximately similar backscatter strengths to the median valley, but a larger range of slope complements, due to its inherent slope. The median valley bounding fault L comprises multiple fault scarps, where backscatter sharply increases towards steeper seafloor slopes. This forms one of the composite terrain types described earlier. One aim of this study is to attempt to distinguish whether this dependency is purely due to grazing angle variations or whether the intrinsic backscatter strength is dependent upon slope angle. The backscatter dependence on slope complement for the faults at location F appears to be more closely related to the characteristics of the landslide at N rather than the faults at L, maybe these scarps are draped with talus.

It is important to remember that the backscatter versus seafloor slope statistics will be strongly dependent on the grid size used for the bathymetry. Coarser grids will smooth out the topography, and hence the distribution of slopes. The bathymetry and backscatter data were acquired in their raw format and gridded in the same manner through simple binning and interpolation, with a grid size in both cases of 36 m (chapter 4).

Of particular interest is the relationship between the 'logged' backscatter strength (BS_C) and slope complement at locations G, J, T and H, corresponding with volcanic

ridges at increasing distances from the ridge axis (respectively 0, 7, 17 and 19 km). A typical evolutionary scenario through time for such ridges would involve a freshly erupted basaltic AVR becoming increasingly broken up by faulting with time due to thermal cooling associated with lithospheric extension (e.g. Ballard and van Andel, 1977; Luyendyk and Macdonald, 1985; and Parson *et al.*, 1993). However, this faulting process is also concurrent with increased sedimentation with distance from the ridge axis. Eventually, the faulting ceases and the rocks become totally buried by the sediments. The associated plots for these locations in figure 5.5 appear to be consistent with this geological theme, showing progressive increase in range of slope complement (except for H) and a modest decrease in backscatter coefficient, with distance off-axis.

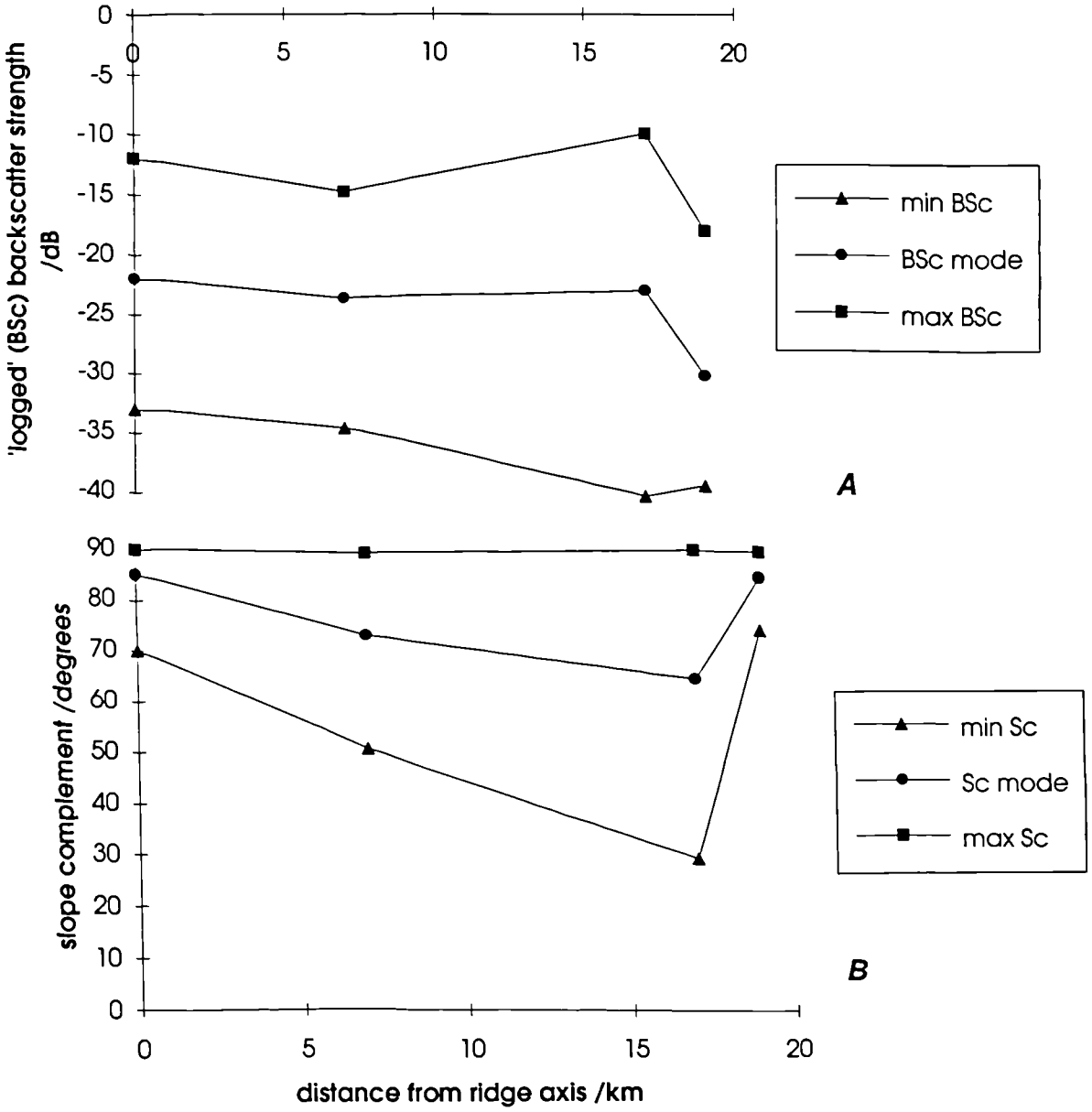


Figure 5.10 *(a) 'logged' backscatter strength (BS_c), and (b) slope complement, variations for volcanic ridges at increasing distances from the ridge axis.*

Figure 5.10 plots 'logged' backscatter (BS_C) and slope complement against distance from the ridge axis for the sites G, J, T and H. As mentioned above, AVR G comprises small ranges of high 'logged' backscatter strength (BS_C) and slope complements, indicative of median valley floor terrain; J has an overall lower 'logged' backscatter strength with the introduction of steeper slopes related to the initiation of faulting; T possesses a very broad distribution of both 'logged' backscatter strength and slope complements; while ridge H exhibits the lowest 'logged' backscatter strength of the selection but small ranges of both variables. There is no variation in 'logged' backscatter with slope angle for all cases except ridge T, where recorded backscatter increases with decreasing slope up to a slope complement of 65°, then decreases again with steeper slopes and lower slope complements. The question regarding whether these skews are associated with a backscatter dependency on slope, also postulated as a possible cause for the skew observed for the faults at L, will be addressed in section 5.6.3. Due to the return of a low slope angle range at ridge H, it is proposed that the fault scarps may now have been totally buried by sediments. The 'logged' backscatter strength (BS_C) at H is, however, still higher than at the basin A, suggesting that the sonar beam has not yet been fully attenuated before reaching the volcanic basement. The sediment blanket at the ridge can be said to be less thick than the sediments ponded in the basin, with the maximum thickness of the sediments at H constrained by the penetration depth of the sonar. Both H and A are at comparable distances from the ridge axis. An estimation of the thicknesses of sediments at these sites will be attempted in section 5.5.3.

5.5.2 Primary backscatter versus grazing angle

The most striking observation of the relationship between the 'primary' backscatter strength (BS_M) and grazing angle is the clear correlation of increasing backscatter strength with grazing angle for all eight experimental localities (figure 5.8). The shape of the angular backscatter functions are similar to the theoretical functions in Jackson *et al.* (1986) (figure 5.2). It is important to note that the modal 'primary' backscatter strength (BS_M) plots, which express the angular response function best, exhibit a far higher second order variability at lower grazing angles, particularly those less than 40°, than at the higher angles. This is due to there only being a small sample of data available in this region, undermining the reliability of the data. Additionally, the spatial bathymetry resolution must be noted, since variations in backscatter due to topographic relief not resolved by the bathymetric system will be incorporated into the measured angular response. The statistics of the results from each site are presented in table 5.IV. The rate refers to $\frac{\partial BS}{\partial \theta_g}$, the slope of the plots,

and the range of backscatter is measured in the direction orthogonal to the main slope.

Locality	Min BS /dB	Max BS /dB	BS range /dB	Min θ_g /°	Max θ_g /°	rate /dB deg^{-1}
G-AVR (1)	-32	-7	18	35	74	0.36
J-off-axis ridge (2)	-31	-3	19	28	90	0.12
T-off-axis ridge (3)	-33	0	27	24	90	0.34
H-off-axis ridge (4)	-29	0	16	50	90	0.49
A-sedimented basin	-45	-18	20	36	66	0.67
N-landslide	-19	0	16	58	90	0.47
F-fault scarps	-35	-7	20	33	90	0.40
L-fault scarps	-35	0	23	50	90	0.67

Table 5.IV - 'Primary' backscatter versus grazing angle statistics

The 'primary' backscatter strength (BS_M) and grazing angle relationships between the two distinct bottom types of median valley AVR (G) and sedimented basin (A) clearly exhibit contrasting characteristics. The binned contour plots still show a small range of 'primary' backscatter strength (BS_M) for both sites, but a wider spread in grazing angles is seen, compared with the results presented of 'logged' backscatter (BS_C) versus slope complement in section 5.5.1. The effects of varying beam angle have now been taken into account. Through the determination of the 'primary' backscatter strengths (BS_M) and grazing angles, the backscatter range for a particular grazing angle has decreased from those 'logged' strengths (BS_C) plotted against slope complement in figure 5.5 (section 5.5.1). The former range of 'logged' backscatter strength (BS_C) is now accounted for by the angular backscatter response curve. This angular response is greater for the sedimented basin (0.67 dB deg^{-1}) compared with the median valley floor (0.36 dB deg^{-1}). Hughes Clarke (1993) also observed a steeper angular response for muds compared with bare rock and glacial till. The difference in average backscatter strengths between sites G and A (exposed volcanics and sediment pond) is 12 dB, though the backscatter varies for each of these sites by more than twice as much over the grazing angle range. This contradicts Mitchell's (1991) conclusions where the backscatter variation between bottom types was more significant than bottom slope effects.

The landslide N possesses the highest 'primary' backscatter strength (BS_M) of all the experimental sites, with most returns being above -10 dB. I propose that this may be due to two possible factors. Firstly, the intrinsic backscattering strength of the landslide surface may be greater than that of the surrounding seafloor. The landslide would be expected to comprise rocks with a similar acoustic impedance to the

median valley volcanics, since it would be materially composed of weathered basalt and maybe some lower crustal rocks (Cann *et al.*, 1992). Therefore, I suggest the 'primary' backscatter difference (BS_M) between G and N would be primarily controlled by the roughness of the seafloor. Site G would be composed of individual pillow lavas of approximately 1 m scale; site N would comprise talus blocks broken up from pillows, and would therefore be a smaller size. The acoustic wavelength is 12 cm for the EM12, so the broken up, irregular blocks comprising the landslide would increase surface diffraction of the incident sonar beams and consequently increase the surface's backscatter strength. Another partial cause of the high primary (BS_M) backscatter returns associated with the landslide could be due to the high grazing angles striking this particular site, as near-normal incidence specular reflections can greatly increase the amount of acoustic energy received at the transducers. However, surface roughness must be considered as a dominant cause for the 'primary' backscatter difference between localities G and N.

The fault scarps imaged at locations F and L exhibit opposite backscatter versus grazing angle trends to those identified in section 5.5.1, now the 'primary' backscatter (BS_M) and true grazing angles have been computed. This revised angular response of higher 'primary' backscatter strengths at higher grazing angles indicates that the high 'primary' backscatter bands characteristic of the fault scarps could be due to high energy normal incidence specular reflections. The opposite sharp correlation in section 5.5.1 might have been explained thus; however a more detailed investigation on individual profiles will be given in section 5.6.3. The grazing angle range for these fault scarps is greater than for the flat surface of the sediments at A, as high angle fault scarps are present together with gentler sloping terraces. The angular response curves, $\frac{\partial BS}{\partial \theta_g}$, for these composite bottom type localities lie within the range of those for A and G. The range of 'primary' backscatter strength (BS_M) for a single grazing angle is greater for the faults than for A and G, confirming that one is dealing with more than one bottom type.

Regarding the off-axis ridge evolution through time, a similar story as revealed in section 5.5.1 can be derived from the relevant plots of figure 5.8. The range of 'primary' backscatter (BS_M) and grazing angle increase with distance from the ridge axis, for sites J and T, presumably due to increased faulting and sedimentation. The 'primary' backscatter strength of J and T is in fact higher than for the median valley G, with a considerably less steep angular response gradient. These sites possess the steepest slope angles of the test areas (section 5.5.1) and are clearly composite

terrain types due to the wide range of backscatter for any individual grazing angle. A variation of bottom type with slope may provide a possible explanation for the trend observed in figure 5.5 of section 5.5.1 (site T). The previous explanation for the 'primary' backscatter strength (BS_M) versus grazing angle characteristics of the buried ridge H is not quite so clear now. Extrapolating the trend of angular backscatter response, H does possess 'primary' backscatter strengths lying between those of G and A. Although the slope range is approximately the same for all three sites, the grazing angles are greater for H and the backscatter may have been artificially strengthened due to specular reflections here, since it is of a generally higher strength that G or A. The gradient of the backscatter angular response is halfway between that of A and G, again suggesting an intermediate stage of sedimentation.

5.5.3 Estimation of sediment thickness from backscatter strengths

It has been proposed that since site H possesses primary backscatter strengths (BS_M) and an angular response between those of sites G and A, then it represents an intermediate stage of sedimentation. Mitchell (1993) presents a model relating backscatter strengths to sediment thickness.

5.5.3a Theory Mitchell (1993) derived a simple model for the determination of sediment thicknesses from backscatter strength information. The model is based on a simplified scenario of figure 5.1, where a plane wave is incident upon a horizontal layer of attenuating sediments, overlying a highly reflective basement material. The intensity of the acoustic signal is reduced by the pressure transmission coefficients ($T_{WS}^2 \times T_{SW}^2$) at the sediment/water interface. Mitchell (1993) states that the effective backscatter strength of the sedimented surface is:

$$BS = BS_1 + 20\log_{10}(T_{SW}T_{WS}) - \frac{2\alpha h}{\sin\theta_g}.$$

BS_1 is the backscatter strength of the basement material (volcanics here), θ_g is the grazing angle of the refracted wave, α is the sediment attenuation coefficient (dB m^{-1}) and h is the thickness of the sediment layer.

Mitchell (1993) explains that the transmission coefficients can be ignored because their variations with grazing angle are small compared with the attenuation response. The grazing angles used have not been corrected for refraction effects, since as stated in section 5.2.1, the velocity ratio of sea water to pelagic sediments is small (1.01), so for the grazing angles associated with the EM12 (45°-90°), a maximum of

only 0.6° of refraction would occur. This effect was considered negligible for this simple model. Refer to Mitchell (1993) for further descriptions of the assumptions of the model.

In order to apply this model to sites G, H and A, the simplest set of assumptions is considered:

- Site G possesses no sediment cover
- Sites H and A are draped with sediments of different thicknesses, and the sonar beam is not fully attenuated in either case.

So the backscatter strengths for the three sites become:

$$BS_G = BS_V(\theta_g)$$
$$BS_H = BS_V(\theta_g) - \frac{2\alpha h_H}{\sin(\theta_g)}$$
$$BS_A = BS_V(\theta_g) - \frac{2\alpha h_A}{\sin(\theta_g)}$$

and the backscatter strength differences between sites G & H and G & A become:

$$BS_{(G-H)} = \frac{2\alpha h_H}{\sin(\theta_g)}$$
$$BS_{(G-A)} = \frac{2\alpha h_A}{\sin(\theta_g)}$$

Therefore, if the backscatter strength differences between G & H and G & A are plotted against $1/\sin(\theta_g)$, and the attenuation coefficient is known, then the sediment thicknesses of sites H and A can be determined.

The attenuation coefficient of pelagic sediments has been estimated by Kibblewhite (1989) and described in section 5.2.1 to lie between 0.4-1.0 dB m^{-1} at a frequency of 13 kHz. Mitchell (1993) calculated a value of 0.2-0.4 dB m^{-1} for hemipelagic sediments at a frequency of 6.5 kHz. Kibblewhite (1989) infers a

doubling of attenuation coefficient with frequency at this frequency range. The attenuation coefficient selected for this experiment is therefore 0.6 dB m^{-1}.

5.5.3b Results

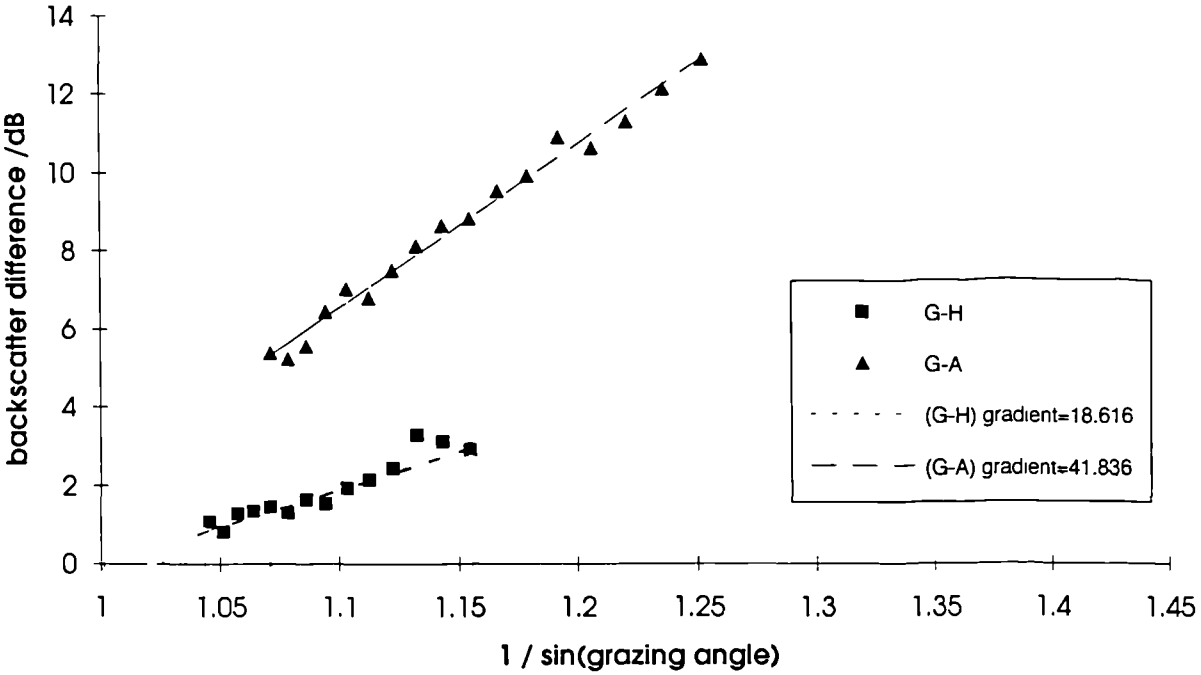


Figure 5.11 *Sediment thickness estimations: the backscatter differences between two sedimented sites (A & H) and an unsedimented reference site (G) are plotted against the reciprocal of the sine of the grazing angle. Best fit regression lines are plotted, with appropriate gradients.*

Figure 5.11 presents the backscatter difference between sites G & H and G & A plotted against $1/\sin(\theta_g)$. The modal 'primary' backscatter (BS_M) data displayed in figure 5.9 are used; the highly variable, sparsely sampled, low grazing angle data were omitted and the responses were smoothed prior to determining their differences. A best-fit regression line was fitted through the straight line portions of the G & H and G & A curves, with respective confidence levels of 95 and 99%. The gradient of G & H was -18.60 ± 2.41 dB, and G & A was -39.48 ± 1.45 dB. From the equations stated above, the sediment thicknesses can be determined:

$h_H = 15.50 \pm 2.01$ metres

$h_A = 34.90 \pm 1.21$ metres

5.5.3c Discussion There is obviously the possibility that site G is draped with a sediment layer as well; in this case, it is only possible to determine the difference in thickness between sediments at sites H and A. This would be 19.40 m. Ballard and

van Andel (1977) dived in similar regions of the MAR (37°N) and observed only a very light dusting of sediments over the axial zones, so the assumption that G possesses negligible sediment cover is probably valid. Also, the sonar signal may be fully attenuated at site A, in which case the determination of the sediment thickness at A by this method only presents a minimum thickness. Mitchell (1993) discusses the problem of estimating penetration depths of sonars, and suggests penetration depths of between 5-30 m at a 30° grazing angle for this frequency. Since the grazing angles for the straight line segments of G & A range from 53 to 69° and G & H from 62 to 70°, then the penetration depths will be greater and the sediment thickness model assumptions and results should be reasonable. There is a problem if I consider the situation presented in Mitchell (1993) where backscatter strengths are available from a sedimented lava flow and a deep sediment field. To determine either the attenuation coefficient or sediment thickness of the former site, Mitchell (1993) adopts the following equation:

$$BS_G = BS_V(\theta_g) - \frac{2\alpha h_G}{\sin(\theta_g)}$$

$$BS_A = BS_S(\theta_g)$$

where BS_S is the surface backscatter strength of 'normal' thick sediments. The backscatter difference between sites G and A become:

$$BS_{(G-A)} = BS_V(\theta_g) - BS_S(\theta_g) - \frac{2\alpha h_G}{\sin(\theta_g)}$$

Mitchell (1993) assumes the angular dependencies of BS_V and BS_S are equal and his resulting gradient of the backscatter difference plotted against $1/\sin(\theta_g)$ was negative. The gradient derived here for $BS_{(G-A)}$ was positive, so there is clearly a problem. The assumption that the angular dependencies of the two bottom types are the same or that the sediments imaged by the EM12 at A have fully attenuated the sonar signal must be invalid in either one, or both, of the cases. The results presented in this chapter agree with some of the literature (e.g. Hughes Clarke, 1993), whereby a steeper angular response is observed for fine-grained sediments than bare rock. Mitchell (1993) must clearly have recorded the opposite case, or the sediments imaged by the EM12 at site A have not fully attenuated the sonar signal and the angular response was purely due to the attenuation effect of a sediment layer overlying a more highly reflective basement.

The tentative sediment thicknesses estimated for sites A and H, of 33 and 16 m, respectively, relate to equal distances from the ridge axis (19 km or 1.9 Ma, assuming a half spreading rate of 1 cm/year). Balsam (1988) suggests a sediment accumulation rate for the MAR of 10 mm/ka, yielding an estimated sediment thickness at sites A and H of 19 m. This is consistent with the results achieved here. The lower value for the off-axis ridge H and the higher value for the basin A may agree with the observation of Keen and Manchester (1970), where they suggested, from sediment profiling in the vicinity, that sediment slumps down from the crestal ridges into the basins.

5.6 SEAFLOOR CLASSIFICATION

Backscatter angular response curves have been derived for various bottom types. If these trends are sufficiently distinct, then they alone can be used to distinguish bottom types. If they are similar, then the angular effect can be removed and the resultant backscatter strengths present purely bottom type variations. The responses presented here appear to fall into the intermediate case, and the tentative removal of the angular backscatter variation is attempted below.

5.6.1 Removal of angular backscatter variation

Ideally, now the backscatter angular response curves have been derived for various single homogeneous seafloor types, it should be possible to normalise all 'primary' backscatter (BS_M) returns accordingly to produce intrinsic backscattering strength values related to particular bottom types. However, there is the problem that the gradient of these response curves varies according to bottom type, making a total removal of angular variation impossible. It was still deemed reasonable to normalise the 'primary' backscatter measurements according to one of the single bottom type responses, which would optimally remove the angular variation, but may only effectively dampen this variation for lithologies with steeper response curves than the normalisation function, or slightly reverse the angular trend for bottom types with less steep angular gradients. The median valley (area G) results were selected as a suitable response curve for normalisation.

A six term polynomial was fitted through the data from the modal 'primary' backscatter plot G of figure 5.9. Only the modal backscatter values for grazing angles greater than 40° were included from the dataset, due to the unreliability of the sparsely sampled low angle data. However, the 'primary' (BS_M) backscatter values for

these low grazing angles were extrapolated to a constant value of -25 dB to complete the function. Figure 5.12 presents the angular response function used for normalisation. The data were normalised to a grazing angle of 70° (approximately the mean value for the 45-90° range), corresponding with a 'primary' backscatter strength of -15 dB, using this algorithm:

$$BS_{norm}(\theta_g) = BS_M(\theta_g) - BS_f(70) / BS_f(\theta_g)$$

BS_{norm} normalised backscatter strength
BS_M primary backscatter strength
BS_f angular response function

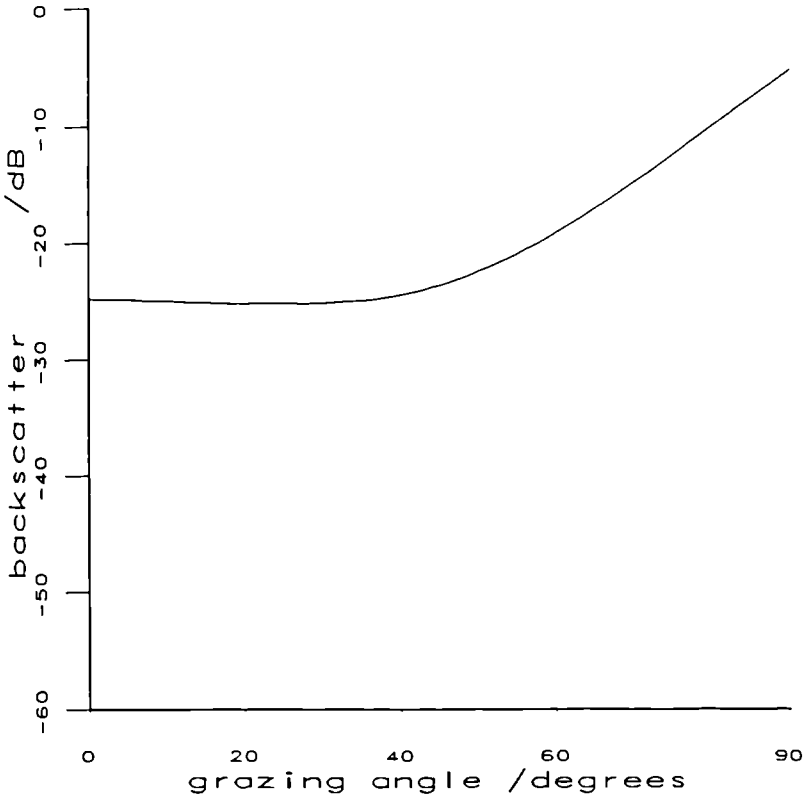


Figure 5.12 Backscatter normalisation function, derived from plot G of figure 5.9, after smoothing and extrapolation.

5.6.2 Normalised backscatter strength results

Figure 5.13 displays modal backscatter strength plots following normalisation to a median valley angular response function at a grazing angle of 70°. The median valley curve of normalised backscatter strength against grazing angle now exhibits a flat trend, with second order fluctuations due to small scale bottom type

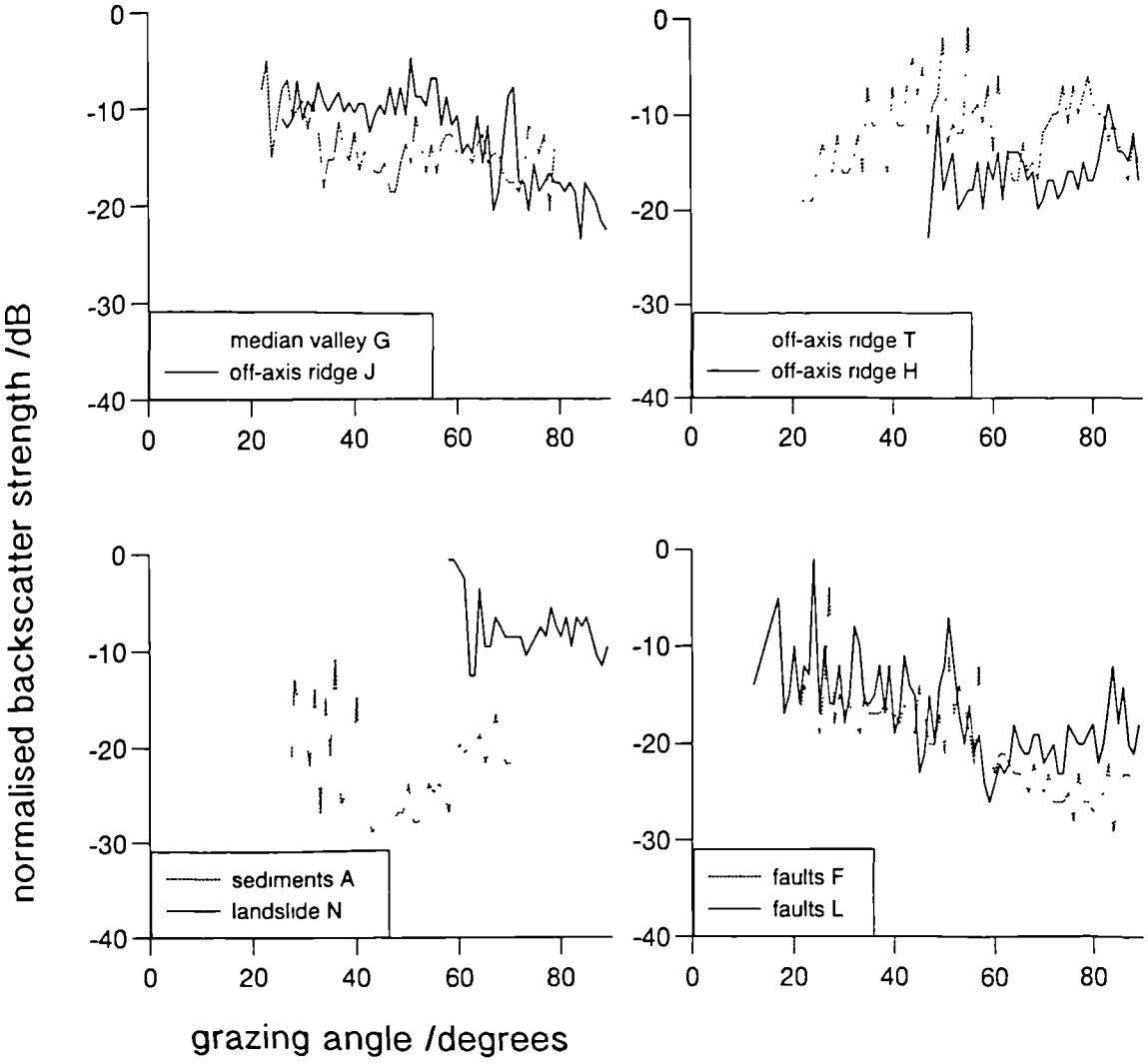


Figure 5.13 *Normalised modal backscatter strengths plotted against grazing angle, for the eight experimental sites. Backscatter strengths were normalised to a grazing angle of 70°.*

inhomogeneities or speckle (see section 4.6.5b). The sites of landslide N, off-axis ridges T and H, and faults F also show approximately flat responses for grazing angles of less than 40°, where a sufficiently large sample size of data is available. Off-axis ridge J and faults L have been over-compensated for, with a revised trend of increased backscatter for lower grazing angles, while the sediments A still exhibit an under-corrected opposite response. Areas J and L may involve some contribution from high angle fault scarps possessing a higher backscatter strength than the material in between; area A purely relates to the steeper angular trend widely observed for fine grained muds. These persistant responses would be expected to produce systematic grazing angle dependent trends in classification, resulting in some 'striping' in images over flatter areas, as witnessed by Hughes Clarke (1993). This effect is not however evident here.

Once a flat response has been achieved, one can deduce that all backscatter variations are now solely due to seafloor type. Figure 5.14 presents some histograms of the normalised backscatter data for some of the experimental sites; noise and acoustic shadows of -64 dB were excluded by thresholding. The distributions are clearly different for the landslide, median valley volcanics and sediments. The median valley volcanics and sediments have modal backscatter values of approximately -14 and -28 dB, respectively. A division separating the two distributions can be selected at -20 dB. The landslide histogram lies above these two, with a modal backscatter value of -9 dB; the histogram is skewed with a tail of lower backscatter, possibly due to acoustic shadowing or small pockets of sediments. The faults lie between the median valley and sediments histograms, with a modal backscatter value of -19 dB. The faults histogram is wider than the single bottom type responses, presumably because this is a composite bottom type area, extending the tails of the histogram out to high backscatter strengths on the fault scarp surfaces and low backscatter strengths inbetween.

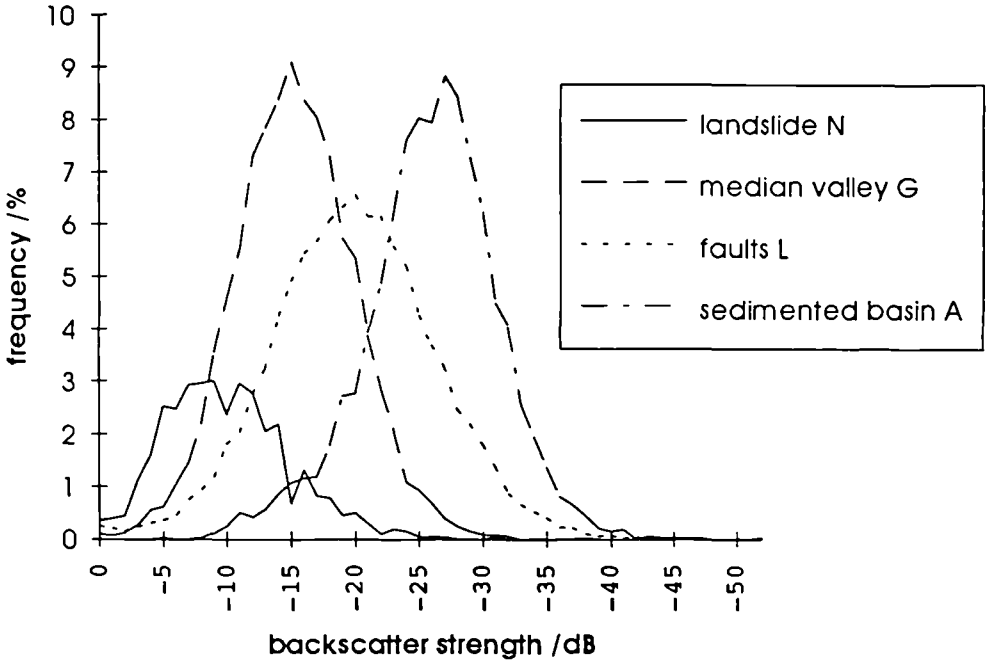


Figure 5.14 *Backscatter strength histograms for four of the experimental sites.*

Figure 5.15 presents three examples of 'logged' backscatter (BS_C) tiles (a, b, and c), containing the G, N, A and L experimental sites (see figure 5.4 for the geographic locations of these). Figure 5.16 presents a preliminary classified version of these tiles, after normalisation, and thresholded according to the -20 dB backscatter division

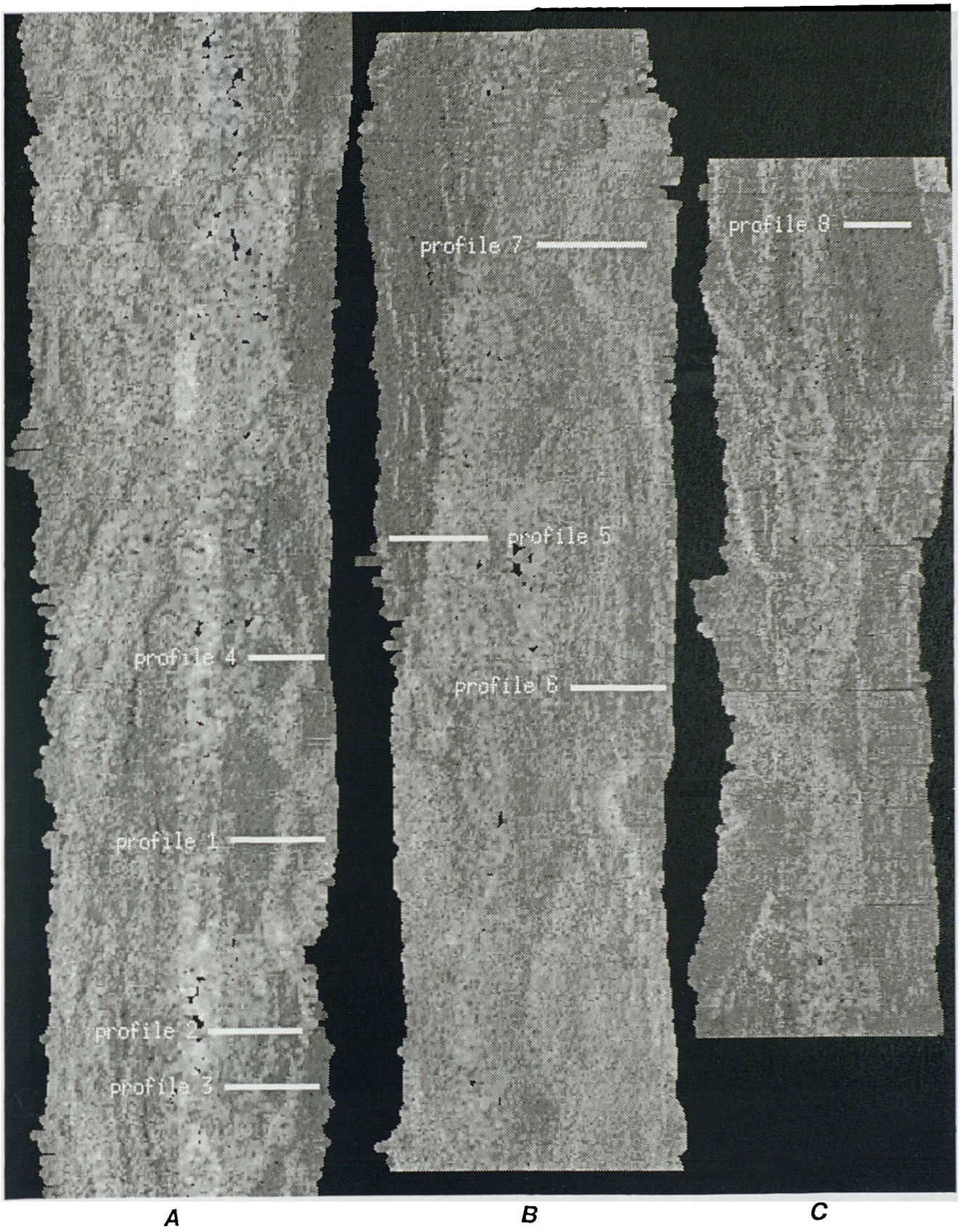


Figure 5.15 *'Logged' backscatter strengths of three individual sidescan tiles ((a), (b) and (c)). White are the highest backscatter strengths (-0.5 dB), black the lowest (-64 dB). The labels refer to the single pixel profiles described in section 5.6.3.*

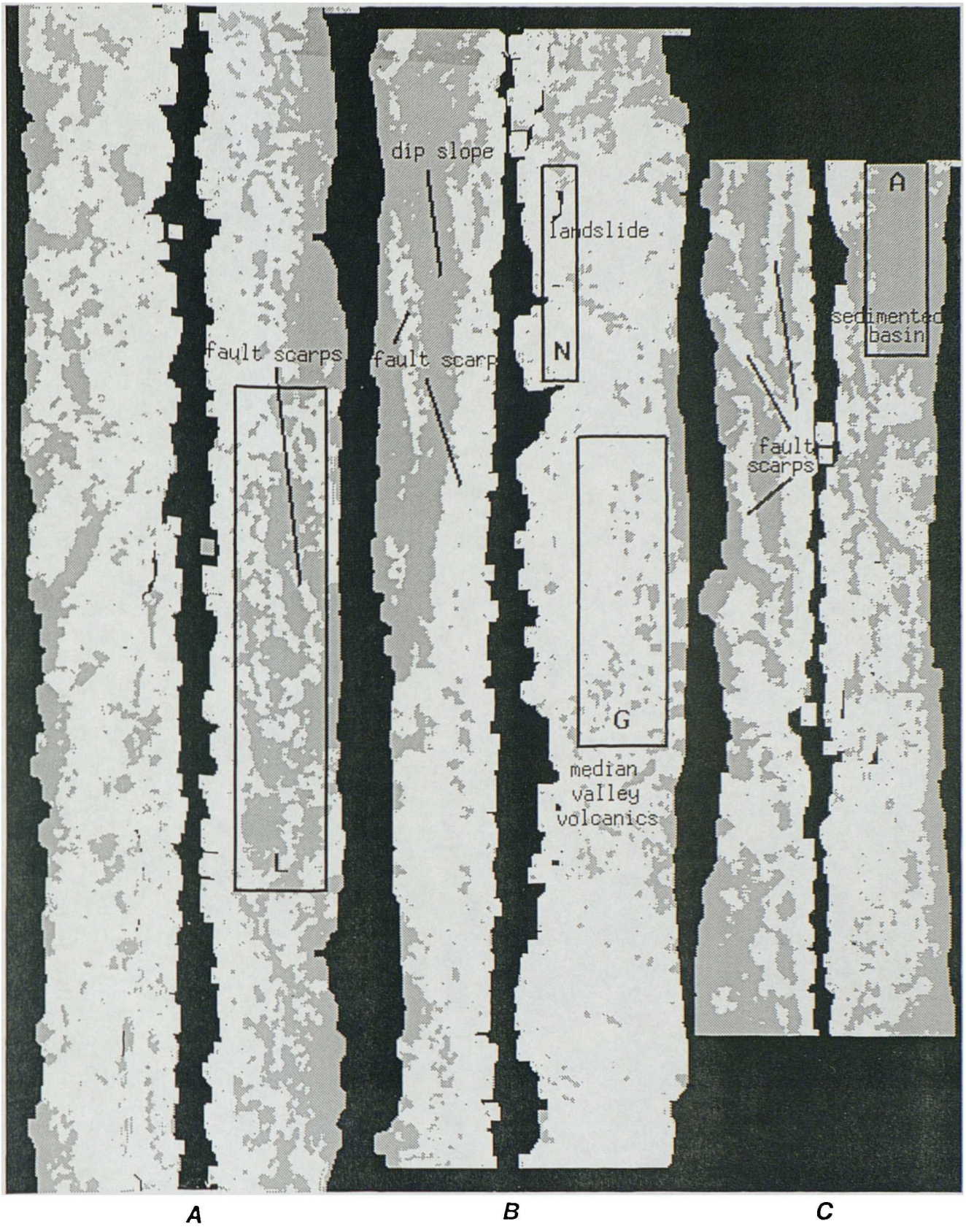


Figure 5.16 *Classified normalised backscatter strengths for the three tiles ((a), (b) and (c)) of figure 5.15, encompassing experimental sites L, N, G and A (labelled). Classification was performed with a simple thresholding operation. White represents all normalised backscatter strengths greater than -20 dB, and grey shows all normalised strengths less than -20 dB. Some geological features are annotated.*

between sediments and bare rock. A median filter of 5×5 pixels has been passed over the image data and some of the gaps were filled using the algorithm described in section 4.6.5b.

The sedimented basin A in tile 'c' and the dip slope of the median valley fault scarp in tile 'b' are the most extensive regions classified as sediments. Within the sediments of the latter, a long linear 'bare rock' scatterer can be identified, corresponding with a small scarp. The majority of tile 'b' is classified as comprising highly backscattering exposed volcanic rock, while tile 'a' possesses a combination of the two bottom types. The faulted site L in tile 'a' does not obviously classify as linear bands of bright material, however the angular response for this heterogeneous bottom type site was over-compensated for here, thus lowering the normalised backscattering strength of these scarps and therefore would not necessarily show up when classified. The peak of the histograms of the faults at L lies directly at the bisection of the bare rock volcanics and sediments distributions. Tile 'c' exhibits a relatively large surface area of sediments, particularly in the 'top left' portion of the image, where numerous long linear features can be identified, corresponding with fault scarps.

This is very much a rudimentary attempt at seafloor classification; the results are crude and not necessarily reliable. The shape of the angular responses may provide as good a classification as normalisation, though either alone is probably insufficient. Further work is clearly required with better quality data, ground truthing to confirm results, and maybe comparisons with other sidescan systems.

5.6.3 Single pixel profiles through experimental sites

All of the analyses performed so far have concentrated on regional trends of backscatter characteristics, based on areas selected according to their geological terrains, either as single lithologies or more complex settings. In this section, some profiles have been selected through these data with the aim of clarifying some of the findings and to pinpoint some of the backscatter characteristics associated with fault scarps, in an attempt to finally address whether the numerous bright linear bands of backscatter are actually consistent with varying angles of incidence or bottom type. Figure 5.15 illustrates the location of these profiles.

5.6.3a Fault scarp - profile 1 Profile 1 in site L is located over a number of multiple fault scarps comprising the eastern median valley bounding wall. Figure 5.17a presents a 250 m section of a fault scarp and a portion of its flat base. The

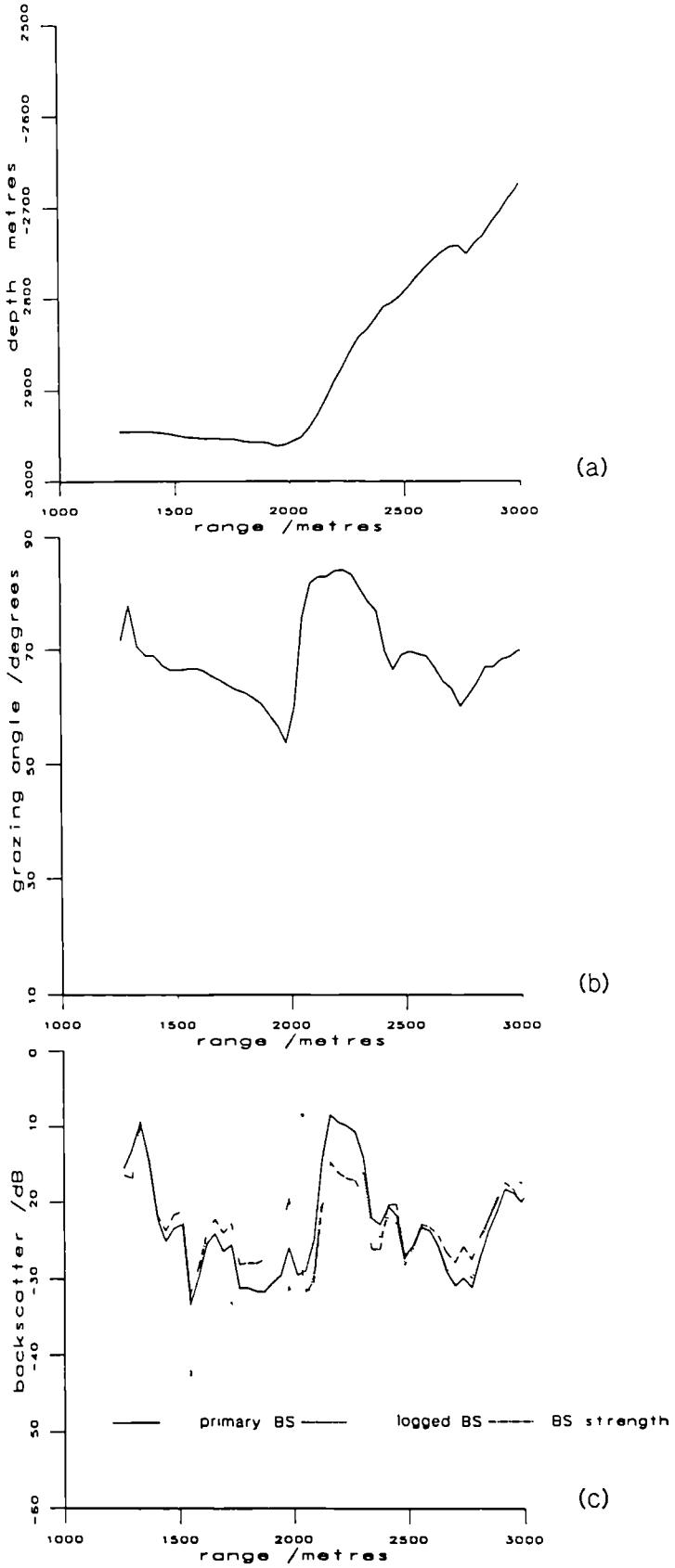


Figure 5.17 Single pixel profile 1 (fault scarp): (a) depth; (b) grazing angle; and (c) primary, logged and (normalised) backscatter strength.

computed grazing angle profile from the slopes and beam angle data, displayed in figure 5.17b, appears quite different. The grazing angle decreases slowly from 70-55°, as expected, for the flat base of the fault, then increases steeply to a peak of ~85° over the fault scarp. But as the beam angle becomes larger than the angle of the scarp, the grazing angle sharply decreases again, with a second order trough (60°) corresponding with a small scale break in slope on the scarp face. It is important to note with this geometry that the horizontal extent of this high grazing angle region is approximately 400 m, compared with the kilometre width of the fault scarp.

Figure 5.17c presents all three stages of backscatter determinations, which all closely follow the grazing angle variations, with higher backscatter correlating with high grazing angles. The difference between the 'logged' (BS_C) and normalised backscatter strength responses confirms the invalidity of Simrad's (1992) flat bottom Lambert's Law assumption. Even the normalised backscatter strength exhibits a peak at -15 dB, 10 dB greater that its surroundings, coincident with the grazing angle maximum. The angular response for area L, after normalisation, showed the opposite trend. The normalised backscatter strength does possess the smallest fluctuations compared with the 'logged' backscatter (BS_C), which is lower closer to nadir due to the first order Lambert's Law correction. The 'primary' backscatter strengths (BS_M) would be equivalent to backscatter strengths computed from GLORIA data (uncalibrated too), for example. It is not obvious from the normalised backscatter strength curve that backscatter varies with seafloor slope, though this is suggested from plot L in figure 5.13. The peak of all the backscatter curves is narrower again than the grazing angle peak, making it only a third of the width of the fault scarp. It is therefore very important to note, from a sidescan interpreter's point of view, that the steady change in beam angle can significantly affect the backscatter responses expected from simple fault scarp geometries. Clearly here, if an interpretation was based purely on the sidescan data, the width of this fault scarp would be greatly underestimated. Searle (personal communication, 1993) described submersible observations of similar scarps, where slightly less steep portions of the scarps were sometimes draped with sheets of talus or sediments. Talus and fault scarp materials both possess intrinsically high backscatter strengths, though the energy backscattered from sediments is far lower. This may provide a possible explanation for the smaller backscatter compared with grazing angle peaks. The plots of 'logged' backscatter (BS_C) against slope complement for the fault scarp sites F and L in figure 5.5 do not suggest a critical slope (e.g. angle of repose of sediment or talus) at which the backscatter strength changes for the scarps.

A conclusion from this profile would be that despite normalisation attempts, backscatter does still vary positively with grazing angle, and no definitive variation of backscatter strength with slope is apparent. It is also shown to be vital to consider true grazing angles rather than slopes when interpreting backscatter data.

5.6.3b Fault scarp - profile 2 This fault scarp, selected again from area L, appears very similar to that shown in profile 1, and now has a vertical displacement of 400 m, with little of the foot of the scarp included (refer to figures 5.18a, b and c). The grazing angles again look considerably different; high values of between 80-90° for the initial uniform gradient scarp, then two sharp troughs going down to 50° corresponding with minor interruptions to the slope; the grazing angles between these troughs decrease from 80 to 65°. Excluding the near-nadir effects, the backscatter strengths generally decrease from -5 to -30 dB, following the very general trend of the grazing angle response, but not as closely as the backscatter of profile 1. For instance, the two 50° troughs in grazing angle do not correspond with backscatter lows. Again, backscatter does generally increase with increasing grazing angle, contrary to the general trend described for this area in section 5.6.2. The width of the highly backscattering region is approximately 1 km, less than the width of the fault scarp (1.5 km). In conclusion again, no confirmation can be made about the material composition of the scarps, and grazing angle remains the major controlling factor on backscatter strength.

5.6.3c Fault scarp - profile 3 Again, this profile was selected from site L and the results are plotted in figure 5.19a, b and c. Here however, there is just a small region of steep slope, 300 m horizontal extent and 200 m relief, with a much flatter depth profile closer to nadir, maybe comprising smaller scarps or talus. The grazing angle profiles contain a number of sharply fluctuating peaks and troughs; the peaks have values of about 85° and the troughs 50-60°, with one particular trough corresponding with a true acoustic shadow with an angle of less than 10°. The backscatter ('logged') recorded here is ~-30 dB, implying contributions to the scattering on this area of seafloor from the surrounding slopes. The associated backscatter curves are also steeply fluctuating, between about -10 and -45 dB, in accordance with the grazing angle trend. The highest 'primary' (BS_M) and 'logged' (BS_C) backscatter peaks coincide with the steep scarp face, but the highest normalised backscatter peak occurs closer to nadir on the flatter terrain, resulting in a possible distorted interpretation from the backscatter data alone. It can be deduced that backscatter does correlate positively with grazing angle and there is no obvious

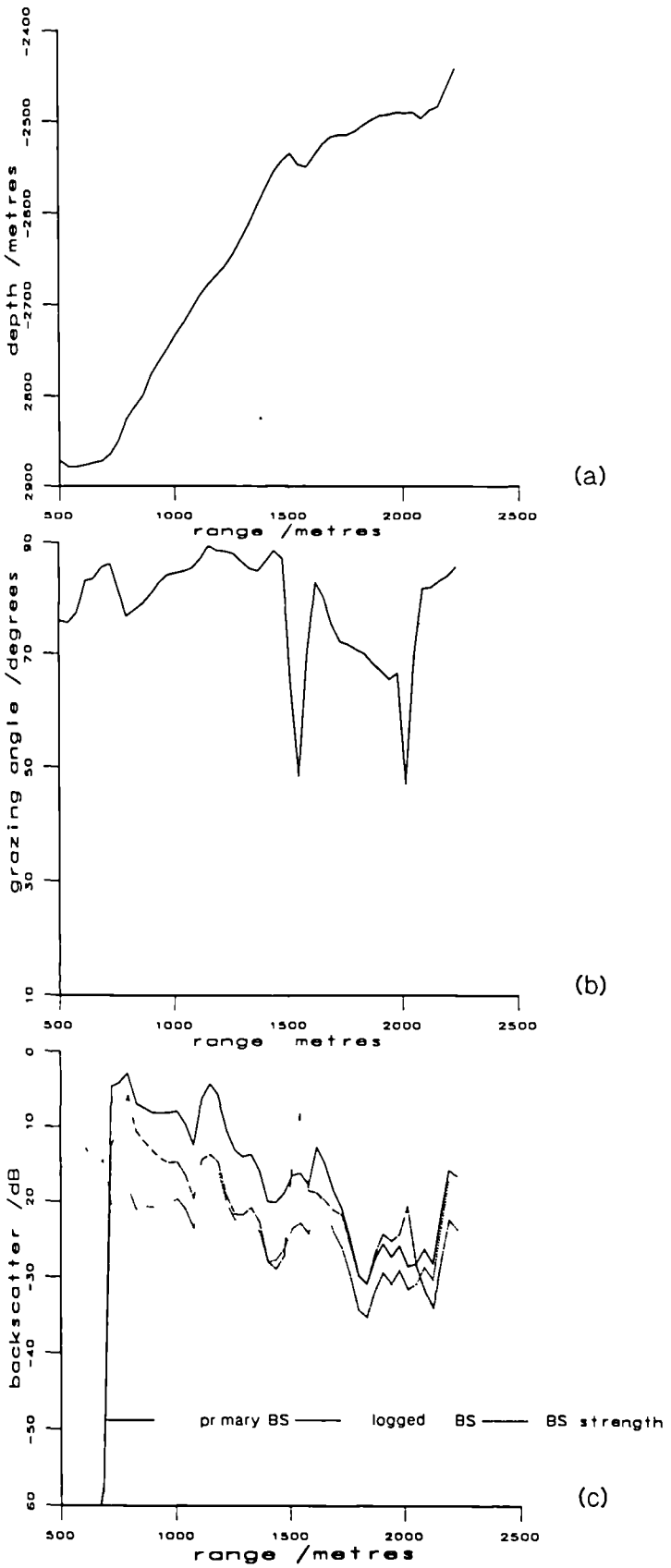


Figure 5.18 Single pixel profile 2 (fault scarp): (a) depth; (b) grazing angle; and (c) primary, logged and (normalised) backscatter strength.

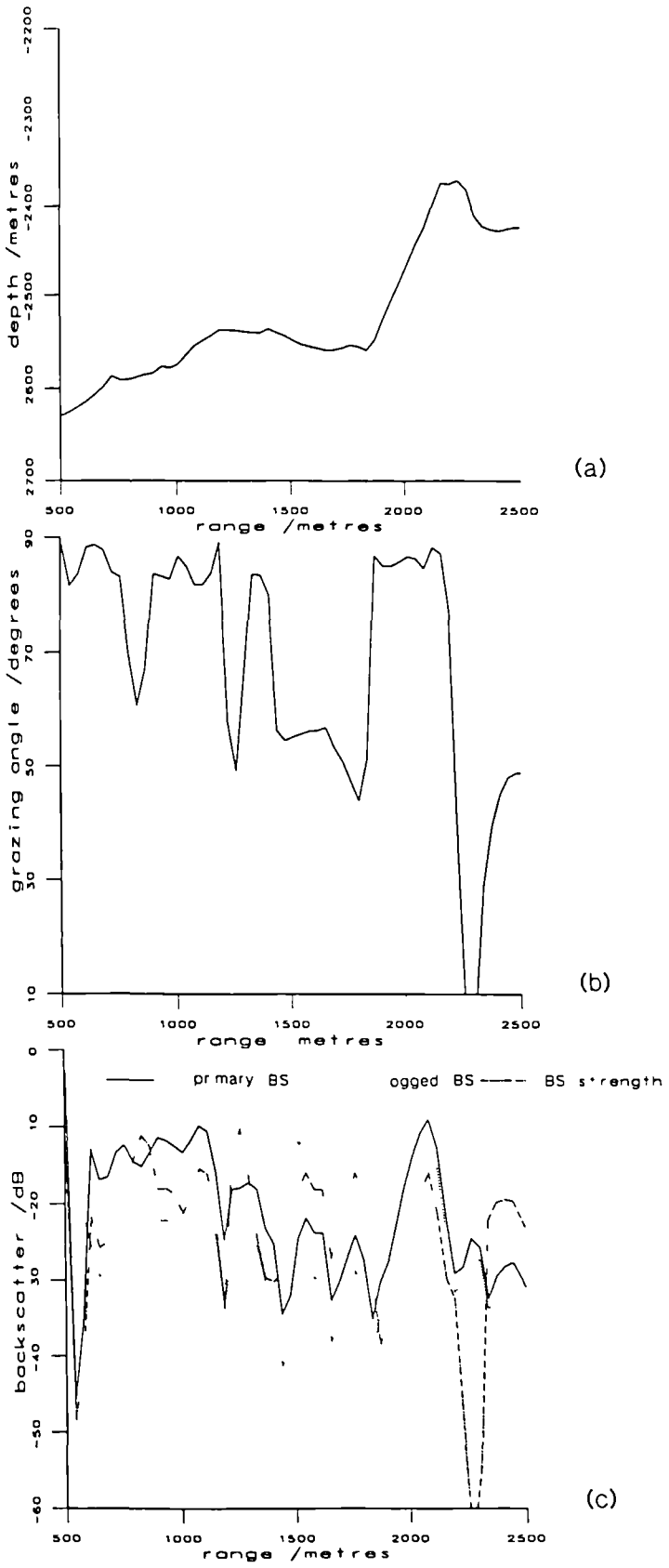


Figure 5.19 *Single pixel profile 3 (fault scarp): (a) depth; (b) grazing angle; and (c) primary, logged and (normalised) backscatter strength.*

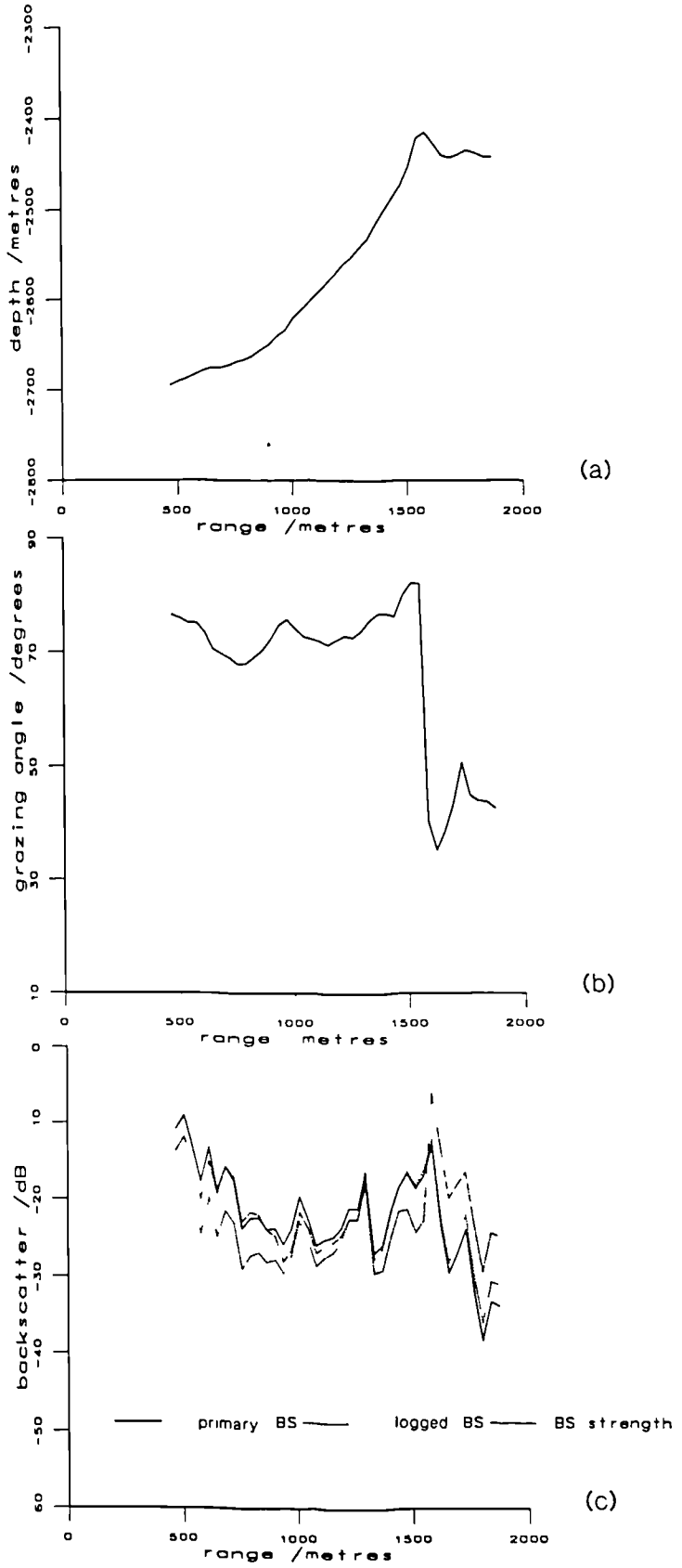


Figure 5.20 *Single pixel profile 4 (fault scarp): (**a**) depth; (**b**) grazing angle; and (**c**) primary, logged and (normalised) backscatter strength.*

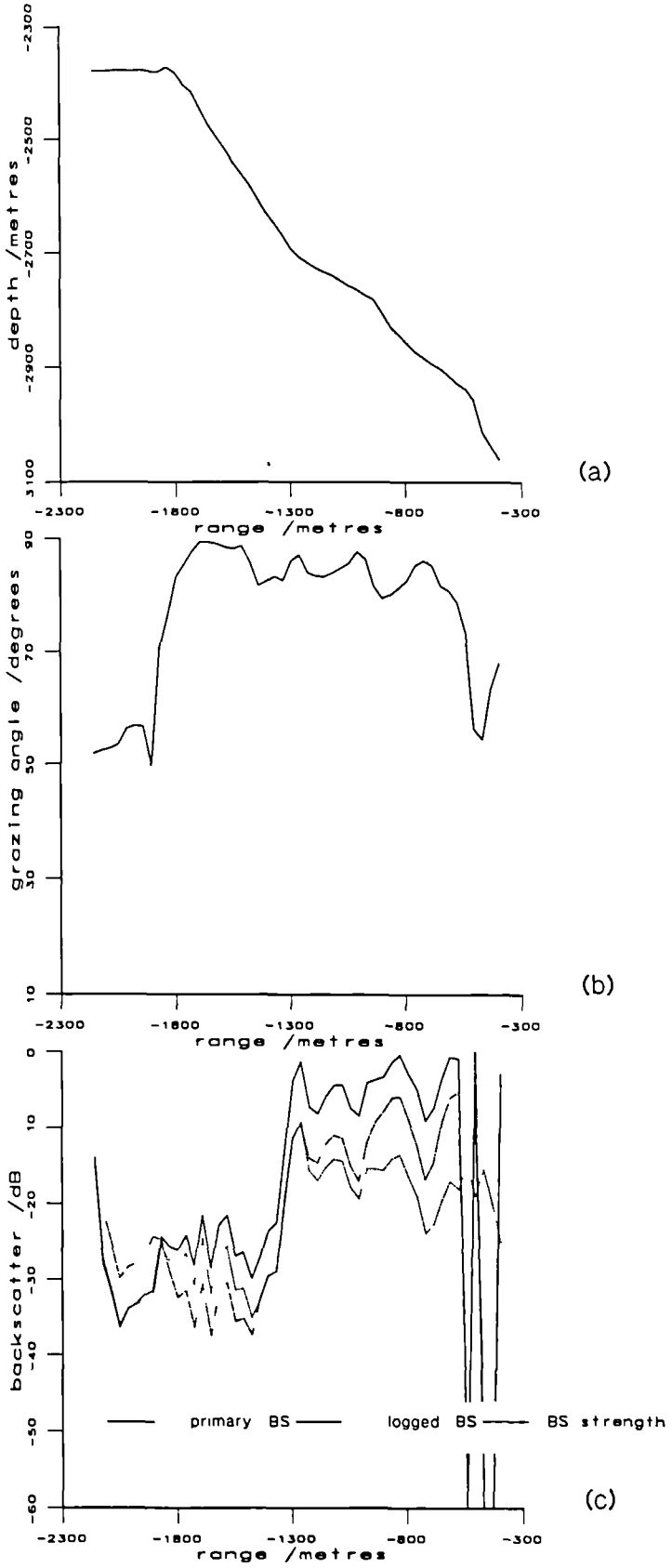


Figure 5.21 *Single pixel profile 5 (fault scarp): (a) depth; (b) grazing angle; and (c) primary, logged and (normalised) backscatter strength.*

backscatter strength peak corresponding with the scarp, that is distinctive from other peaks. The precise backscatter/grazing angle relationship is not predictable.

5.6.3d Fault scarp - profile 4 This is the final fault scarp profile taken from site L, exhibiting some 300 m relief and plotted in figures 5.20a, b and c. The grazing angle drops off considerably from 80-30° at the brim of the scarp, where it slopes away from the insonifying beams. This response is perfectly reflected by all backscatter responses, varying from -30 up to -10, then down to -40 dB at the top of the scarp. The backscatter response observed for this profile correlates simply with grazing angle, which in turn could be qualitatively expected from the bathymetry.

5.6.3e Fault scarp - profile 5 This profile was taken over the western median valley bounding fault, captured in figure 5.16b of the classifications in section 5.6.2, and plotted in figure 5.21a, b and c. The range values are negative because this scarp was insonified to the port side of the ship. The scarp has a relief of 700 m, over a horizontal extent of 1.5 km. The grazing angles exhibit a broad peak of approximately 85° over about 1.3 km of the scarp's horizontal range, with lows of about 50° at its crest and base. The backscatter curves are somewhat enigmatic, with high 'primary' strengths (BS_M) of around -5 dB and normalised strengths of about -12 dB, over just 750 m of the width of the scarp nearest nadir, then they plummet down to -30 dB at greater range. An immediate explanation for this phenomenon is not readily apparent, though the higher backscatter region towards the foot of the scarp may possibly be associated with talus. The outcome is a backscatter peak far narrower in horizontal width than its associated fault scarp.

5.6.3f Further profiles Figure 5.22a, b and c show a profile (#6) taken over experimental site G, the region used for the determination of the backscatter normalisation function. The hummocky terrain of only 60 m amplitude, possesses grazing angles of between 55-70°, with one trough corresponding to ground sloping away from the direction of insonification. The normalised backscatter is, as expected, independent of these grazing angles, with an approximately flat response of -16 dB; the intrinsic backscattering strength of the bottom material.

A profile (#7) was also taken over the landslide N (figure 5.23a, b and c) where the backscatter response of normalised strengths of -10 dB are independent of the grazing angle variations. This confirms the high intrinsic backscatter strength of the landslide material.

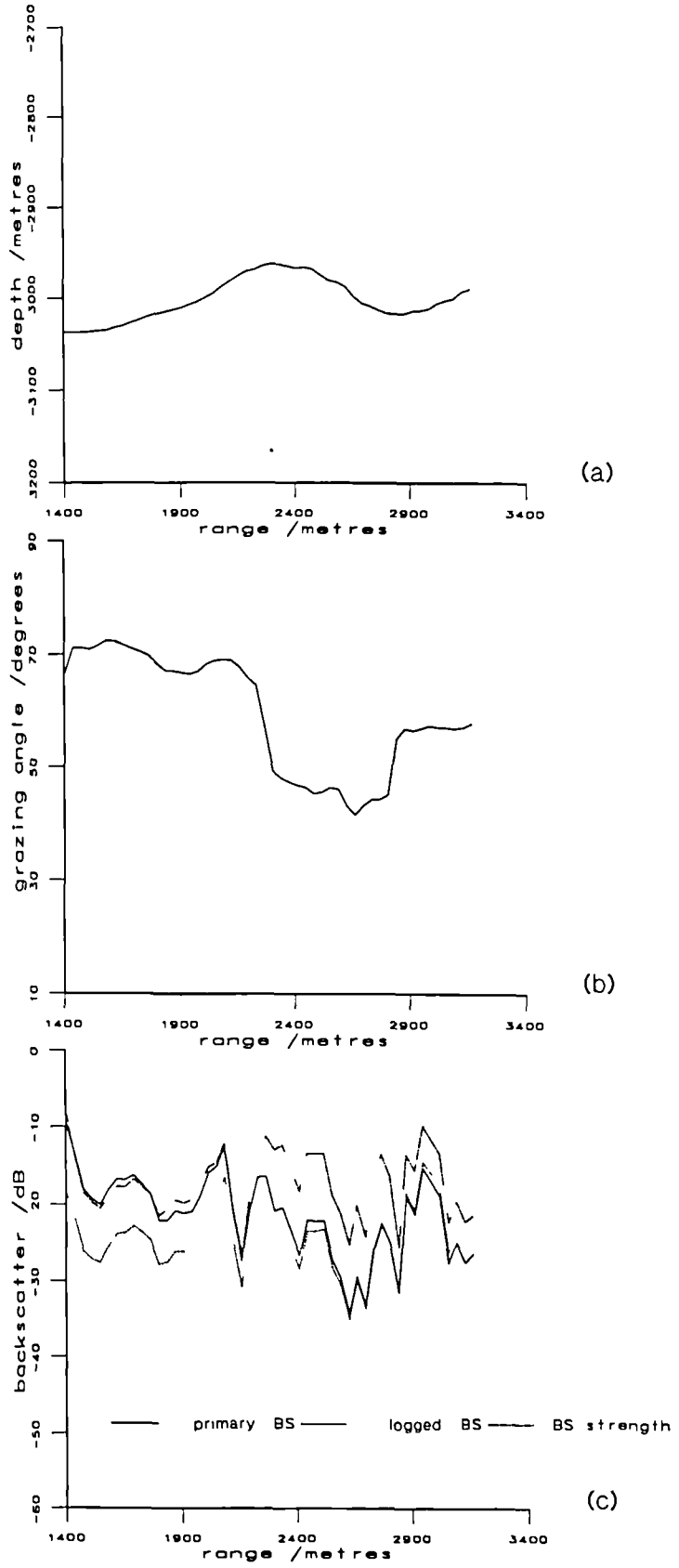


Figure 5.22 *Single pixel profile 6 (AVR): **(a)** depth; **(b)** grazing angle; and **(c)** primary, logged and (normalised) backscatter strength.*

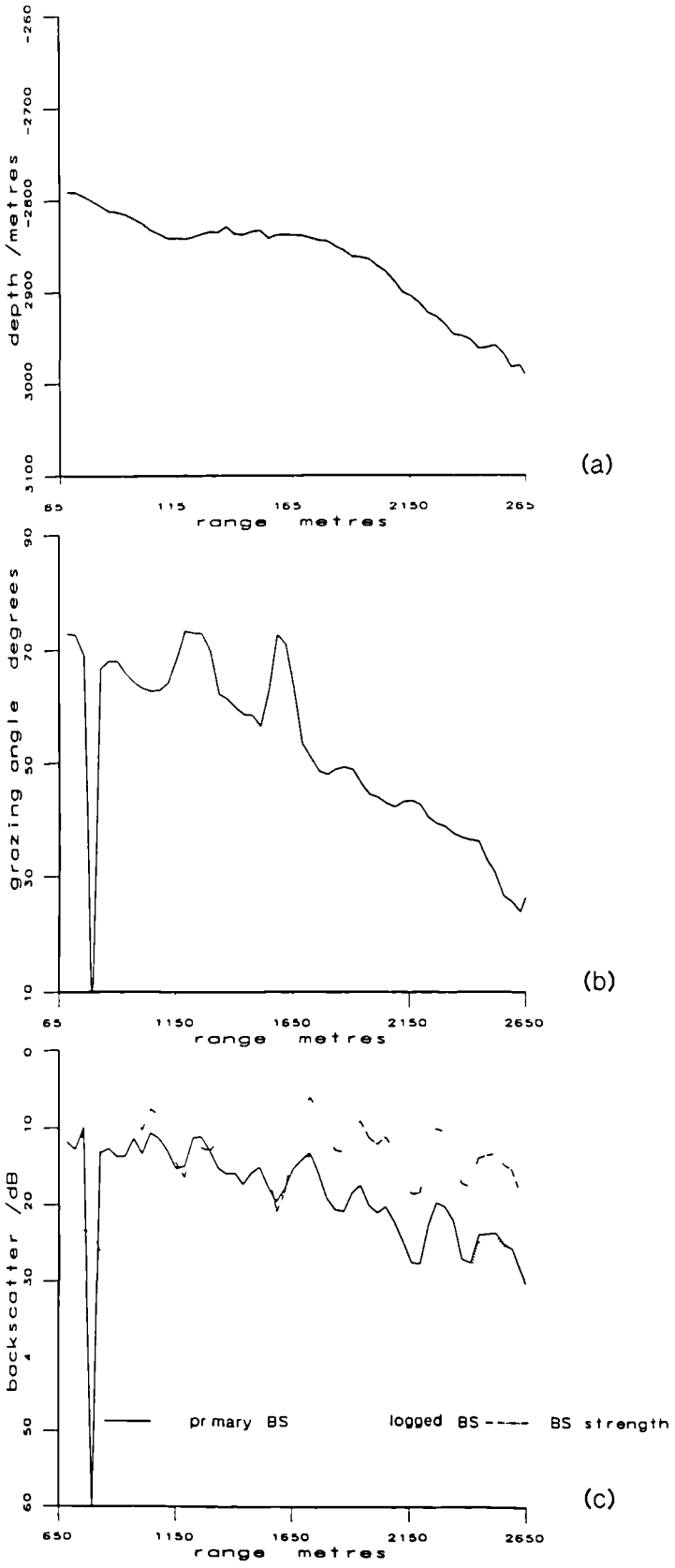


Figure 5.23 *Single pixel profile 7 (landslide): (a) depth; (b) grazing angle; and (c) primary, logged and (normalised) backscatter strength.*

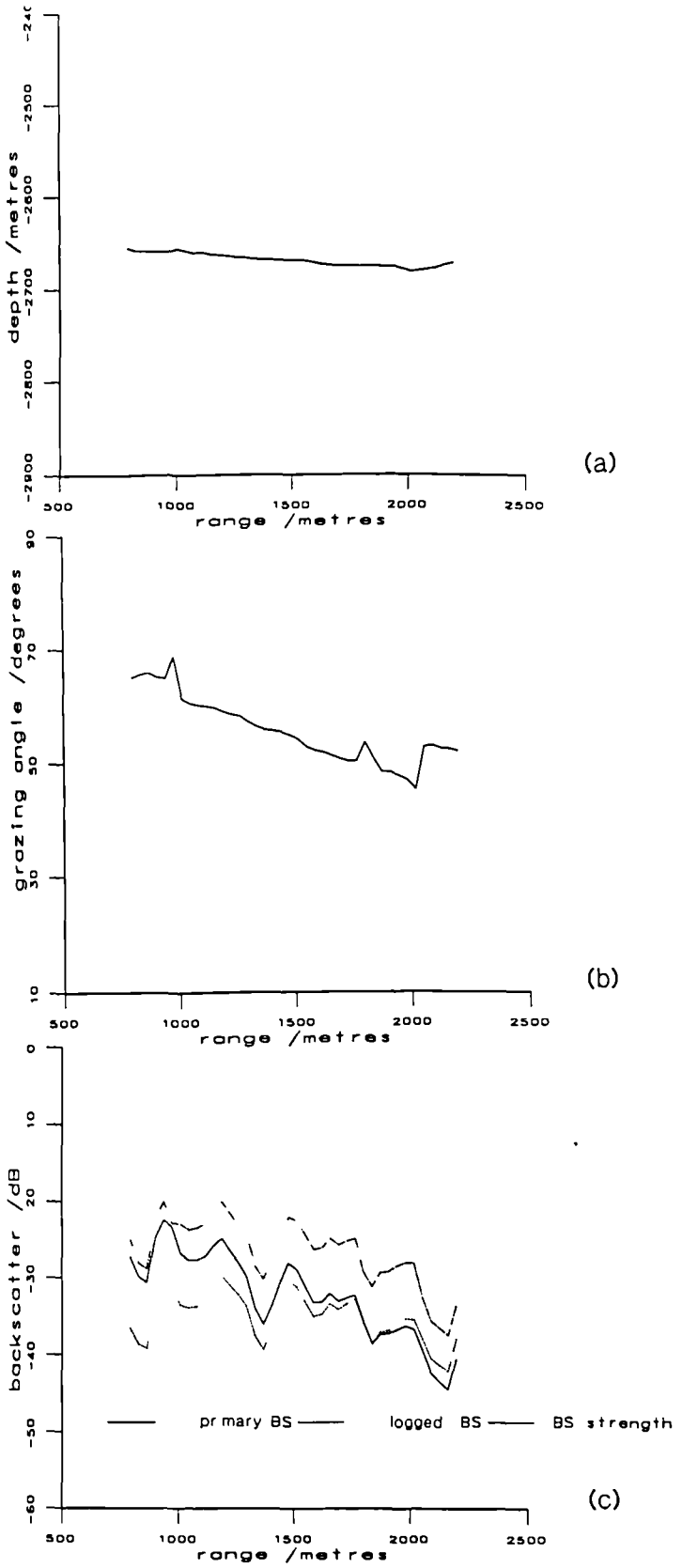


Figure 5.24 *Single pixel profile 8 (sedimented basin): (a) depth; (b) grazing angle; and (c) primary, logged and (normalised) backscatter strength.*

Finally for completeness, a profile (#8) was taken over the sedimented site A. Figure 5.24a, b and c display the flat topography of the floor of this basin, reflected by the grazing angle and low backscatter values of between -23 and -37 dB for the normalised strengths and -25 and -44 dB for the 'primary' (BS_M) readings. The normalisation has slightly dampened the angular effect on backscatter.

5.7 RELATIONSHIP TO OTHER DATA

With regard to previous work, it is very difficult to make direct comparisons since most recording systems (e.g. EM12, EM1000 and GLORIA) are uncalibrated; however, the magnitude, or at least contrast, of backscatter between bottom types is generally comparable with those measured by others. There is little mid-ocean ridge backscatter strength data available, but my results compare favourably with those reported by McKinney and Anderson (1964) from coastal locations using a frequency of 100 kHz. For basalt, the EM12 returned backscatter strengths of -10 to -27 dB, whilst McKinney and Anderson (1964) gathered strengths averaging -18 dB for an unspecified bedrock type. Hughes Clarke (1993) measured backscatter strengths of between -26 and -39 dB for an unspecified bedrock type. Mitchell and Somers (1989) computed backscatter strengths for mid-ocean ridge basalt of -37 dB. These latter two measurements are considerably lower than those measured with the EM12; they were collected with 95 and 6.5 kHz systems respectively.

The backscatter strengths measured by the EM12 for the sedimented basin, expected to comprise pelagic muds, were between -19 and -38 dB. This compares with an average value of -38 dB (at 100 kHz) from McKinney and Anderson (1964) for mud and -3 to -35 dB by de Moustier and Alexandrou (1991) (at 12 kHz) for foraminiferal ooze. Understandably, the sediment types are not all the same, and as Hamilton and Bachman (1982) state, properties do vary between different sediment types; however the values are comparable. McKinney and Anderson (1964) reported a backscatter contrast between bare rock and sediments of 20 dB, while we suggest an uncalibrated value of approximately 10 dB.

Mitchell (1991) observed a greater backscatter strength contrast between bottom type (20-30 dB contrast) than over bottom slope for a single lithology (10-15 dB). The measurements from the EM12 over this axial region of the MAR yielded the opposite case (respectively 12 dB for bottom types compared with approximately 25 dB for slope variations).

Hughes Clarke (1993) normalised 95 kHz EM100 backscatter measurements to the mean angular response of his survey area which is composed of bedrock outcrop and glacial till. Due to the variation of angular response with bottom type, he still found a resultant angular response from muds of -10 dB over the grazing angle range, this is comparable to our remaining trend. The shape of the angular dependent functions are similar to the theoretical responses derived by Jackson *et al.* (1986).

5.8 DISCUSSION

The overall aim of this type of work is to provide an automatic classification of seafloor type; this has been demonstrated to be a complex task. The primary assumption for these particular analyses is that the geological interpretation of the area is correct. Clearly, direct ground truthing of the sample sites is necessary; with bottom photography, dredging, coring, or high resolution sediment profiling; the last two would enable examinations of the sub-surface penetration of the acoustic energy.

Due to the angular dependence on bottom type, it is not possible to totally remove the angular effect without having additional information about the seafloor material. Possibly texture measurements, such as those discussed by Reed and Hussong (1989), or the spectral methods of, for example, Pace and Gao (1988), may be employed, though the quality of this present EM12 dataset questions the feasibility of these methods. Even if bottom types can be distinguished from backscatter strength information, other quantitative methods must be employed at a mid-ocean ridge, such as automatic detection techniques for faults (Shaw, 1992), for a bottom type classification, due to the complex nature of the geology.

The modelling of backscatter angular responses is clearly a sensible way forward; Jackson *et al.*'s (1986) model is the most generally applied to date. The controlling factors of this model are the sound speed ratio, density ratio and roughness parameters (volume scattering can probably be ignored for these relatively unsedimented regions). The first two parameters can be estimated from the literature and ground truthing; the third cannot be assessed as topographic spectral parameters can only be computed down to wavelengths of 100 m or so from multibeam bathymetric data (Dziak *et al.*, 1993). Dziak *et al.*'s (1993) curve fitting method, based on Jackson *et al.*'s (1986) theory, for estimating small scale topographic roughness power spectra from acoustic backscatter data, may be worthy of investigation. However, if large scale anisotropies are present, the model is not valid

as the seafloor no longer possesses the assumed Gaussian roughness spectra on which the theory is based; this is certainly possible for the 45°N MAR data.

5.9 CONCLUSIONS

The backscatter strengths presented in this chapter provide an important addition to the small library of true acoustic backscatter data collected by low frequency sonars over mid-ocean ridge geology. From the selected geological settings, a distinct backscatter strength difference was identified between single bottom types and from the geological interpretation of the bathymetry data. It has been suggested that the acoustic impedance and maybe some roughness characteristics influenced the backscatter strength contrast between the sediments and bare rock volcanics. It has been proposed that the roughness of the seafloor may have caused the backscatter difference between the landslide and exposed volcanics. It is important to note that the grid size used for the bathymetry (100 m) may affect the backscatter-slope statistics, as a coarse grid may tend to flatten out some of the topography, possibly leading to the flatter distributions of figures 5.5, 5.8 and 5.9.

A general angular dependent backscatter function was derived based upon the response yielded from the median valley terrain. Backscatter strengths were normalised according to this, and a tentative attempt at quantitative classification was made. This brought limited success but it was clear that some angular dependency still existed, which was dependent upon bottom type.

Regarding the backscattering characteristics of the fault scarps, the individual profiles did not confirm any distinctive variation of intrinsic backscatter strength with seafloor slope; bottom type inhomogeneities were better highlighted by the wider range of backscatter identified for individual grazing angles on the binned contour plots of 'primary' backscatter strength *(BSM)*. The individual profiles were more important in warning sidescan interpreters just how much a varying beam angle can affect grazing angle determinations, for a system such as the EM12 imaging a mid-ocean ridge, where beam angles are of a comparable size to seafloor slopes.

Finally, this quantitative approach for analyzing bathymetry and acoustic backscatter datasets has helped to confirm the evolutionary state of axial volcanic ridges as they are transported off-axis. This phenomenon is not immediately obvious when simply viewing the sidescan mosaic.

CHAPTER 6
TEXTURE ANALYSIS OF TOBI DATA

6.1 INTRODUCTION

Recent numerous surveys of the Mid-Atlantic Ridge (MAR) (Parson *et al.*, 1993; Searle *et al.*, 1992; Cann *et al.*, 1992; and Murton *et al.*, 1993) using the Institute of Oceanographic Sciences' high resolution sidescan sonar, TOBI (Flewellen *et al.*, 1993) having prompted this research into finding a technique suitable for classifying the imaged bottom types.

Texture analysis has been widely used in many forms for many diverse disciplines. Texture is a property innate in all images and is not a function of image intensity, though it is strongly related to it. A human observer can recognize and describe different textures as being, for example, coarse, smooth or rippled; however, these phenomena are far more difficult to quantify. There are two approaches to texture analysis: structural and statistical. The structural method characterises texture in terms of its basic elements or primitives, which are the constructional units of the image. There are various statistical approaches: common ones include auto-covariance, power-spectral, Laws' 'texture energy transform' and grey level co-occurrence methods. Whichever texture analysis method is chosen, the result is a set of feature vectors associated with each pixel in an image. These vectors discriminate between the different texture types. Effectively, the feature vectors replace the image plane by describing the original image in Euclidean n-dimensional feature space. The feature images generated provide the input into standard classification procedures, such as those commonly used for classifying multispectral satellite imagery.

6.2 DATA

As described in chapter 3, the TOBI sidescan instrument operates at a frequency of 30 kHz, producing a resolution of some 4×7 m at high grazing angles and 42×2 m at far range (Flewellen *et al.*, 1993). 8000 samples are stored for each ping, though they are decimated across-track by a factor of 8 to achieve a geometric balance with the along-track sampling rate. Each swath is 6 km across and only poor data are collected within an approximately kilometre wide strip near-nadir.

The TOBI sidescan sonar data collected by Cann *et al.* (1992) and others have principally imaged portions of the MAR within the median valley. As chapter 2 describes, the axis of the MAR is the site for generating new oceanic crust, so the most important bottom types imaged in this region are the various forms of volcanic morphologies: Smith and Cann (1993) identify some of these. Probably the most prolific are the irregular, coalesced piles of hummocky lava generally comprising the axial volcanic ridges. These would be composed of individual pillow lavas, ~1 m across, so would be described as acoustically rough by the Rayleigh coefficient (chapter 5, section 5.2.1) since the wavelength of the TOBI sidescan system is ~5 cm, and would therefore appear as bright, highly reflective surfaces, though none are individually resolved. The hummocks themselves range from the resolution limit of TOBI to about 200 m in diameter. Anything larger than this arbitrarily defined value would be described as a seamount - these may be 'hummocky' or 'flat-topped' or many other forms and are principally identified by their circular shape and associated shadow; they will be omitted from further description until chapter 7. Volcanic eruptions can occur as sheetflows; these would again produce high backscattering characteristics since they would again be acoustically rough for TOBI's parameters, but appear fairly homogeneous themselves in the imagery. Often these sheetflows are strongly fissured , producing a third volcanic morphology to consider. The pillow lavas may also be fissured.

As the newly accreted seafloor is transported away from the ridge axis, additional geological processes become dominant; namely tectonism and sedimentation. As the lithosphere cool and extends, the volcanic terrain is increasingly broken up by faults; some of these appear as numerous minor features cross-cutting the volcanic units and sometimes the displacement is accommodated within one huge scarp. These scarps would be composed of weathered basalts, dolerites and possibly gabbroic or even ultramafic materials. Their acoustic properties would be similar to those of the volcanic material, though since they are insonified at higher incidence angles, their measured backscatter increases and they appear as 'bright' or shadowed (according to insonification direction) linear bands. These scarps are subject to mass-wasting processes and a number of landslides have been identified by Cann *et al.* (1992). These would be composed of broken up, weathered pillows and maybe lower crustal materials, and would thus possess highly backscattering properties (chapter 5). As well as discrete landslides, many active faults will have a fairly continuous talus ramp along their bases, which is also likely to be strongly backscattering.

Sufficient sediment cover attenuates the sonar signal, beginning to obscure the buried base rock and eventually, once the penetration depth of the sonar has been exceeded, absorbs all the sonar energy so none is backscattered, except at the sediment/water interface itself. The individual particles comprising the pelagic sediments would be smaller than the acoustic wavelength of TOBI, so would also be described as acoustically smooth, however ripples etc. may cause roughness.

The bottom types imaged at the MAR can be categorised as follows:

Material variations:	sediments fresh volcanics older igneous rocks
Acoustic roughness variations:	landslide talus fresh volcanics
Morphologies:	hummocks fissures sheetflow scarps

A number of different scales clearly occur. Material and acoustic roughness variations are best analysed through true backscatter strength and corresponding bathymetry data. We do not have backscatter strength information for the TOBI data, so effects of variable gain settings etc. do occur. There is also no co-registered bathymetry at a comparable resolution so variations due to incidence angles cannot be determined. Equally important for MAR terrains are the distributions of volcanic morphologies, which require a quantitative description. Regions comprising no information must also be identified and excluded. The aim here is to encompass the characterisation of all bottom types imaged by TOBI with one technique.

6.3 CLASSIFICATION

Whichever way texture analysis is performed, a number of feature images are generated from the original dataset. These must be analysed and collated into different bottom types using a classification method.

There are two main methods for classifying image data: unsupervised and supervised classification. Unsupervised classification is often described as clustering,

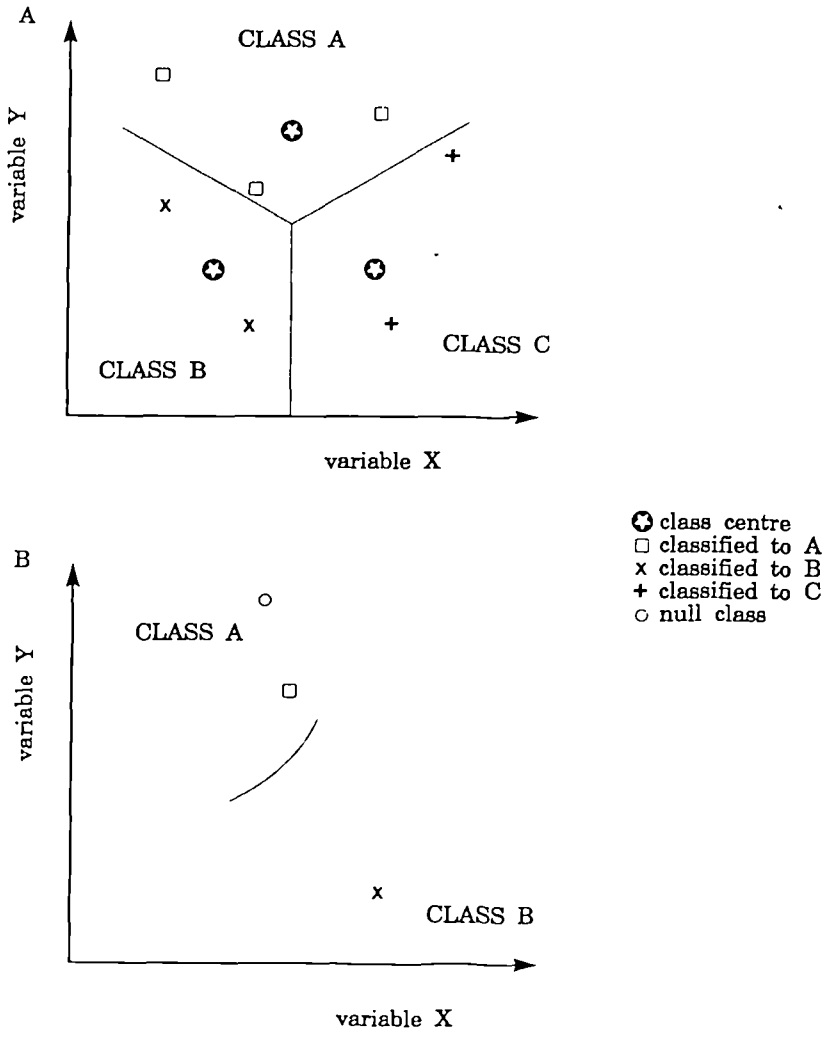


Figure 6.1i *(a) Minimum distance classification method; (b) maximum likelihood classification method. The ellipses are defined around the class values in standard deviation units.*

where natural groupings of feature vectors are determined and assigned to appropriate classes. Hartigan (1973) provides a good overview of these methods. Supervised classification requires the selection of training areas based on supplementary (ground truth) information, encompassing representative samples of every class (bottom type) within the image. Once these training areas have been obtained, class signature statistics, such as mean vector and standard deviation, can be computed for each class training area. All the pixels in the image are then assigned to the class to which they are most similar, using one of the many available classification routines. I have used supervised classification for this study, since it operates in a similar manner to how a geologist would go about classifying or mapping the dataset. Sidescan sonar bottom types would be identified by eye, generally based on their tonal and textural appearance, relevant ground truth information would lead the interpreter to deduce them to be a particular bottom type. Then, when this texture is observed again, it would be assumed that it falls into the same bottom type class.

The supervised classification routines tested here are maximum likelihood and minimum distance classification. The maximum likelihood classifier (Duda and Hart, 1973) uses the Gaussian threshold stored in each class signature to determine whether a given pixel falls within a class or not. This principle assumes that the classes in the input data possess a Gaussian distribution. The threshold is the radius (in 3 standard deviation units) of a hyperellipse surrounding the mean of the class in feature space. If the pixel falls within the hyperellipse, it is assigned to this class. If the pixel does not fall into any class, it is assigned to a null class. In detail, for each class ($i=1, ..., n$), one must determine whether x lies within the hyperellipsoid for this class, that is

$$(x-\mu_i)^T C_i^{-1}(x-\mu_i) \leq T_i^2$$

where x is the pixel vector of the grey levels, μ is the mean vector for class i, ^{T} indicates the transpose of a matrix, C_i^{-1} is the inverse of the covariance matrix for class i and T_i is the threshold value for class i (always set at the default value of three standard deviations). If x is not within any hyperellipsoid, then it is assigned to the null class, otherwise $G_i(x)$, the result for class i on pixel j, is computed for each class and the pixel is assigned to the class where $G_i(x)$ is a maximum.

$$G_i(x) = -\tfrac{1}{2}(x-\mu_i)^T C_i^{-1}(x-\mu_i) - d/2\log(2\pi) - \tfrac{1}{2}\log(|C_i|) + \log(P_i),$$

where d is the number of channels in the classification. P_i is the a priori probability for class i; this can be set to be biased towards a particular class, but I have not used this facility here. This equation is known as the Mahalanobis classifier (Duda and Hart, 1973).

The minimum distance classifier (e.g. Hodgson, 1988) effectively assigns each pixel to the class which has the minimum Euclidian distance between the pixel value and the class mean, i.e. the result for class i on pixel x is

$$G_i(x) = \sum_{j=1}^{nchannels} (x_j - \mu_j)^2 .$$

x is the pixel vector of grey levels and μ is the mean vector for class i. The minimum distance classifier is a faster classification method than the maximum likelihood approach, but generally produces poorer classification results. However, if the maximum likelihood classifier's multivariate Gaussian distribution does not hold, then the minimum distance approach may prove superior.

A signature separability measure is a useful tool to determine the distinction between training classes prior to performing classification. The Bhattacharrya distance separability measure is used here and, without going into further details, it outputs a separability measure where a value of 0-1.0 indicates very poor separability, 1.0-1.9 represents poor separability, while a value of 1.9-2.0 implies a good separability. Poor separability measures imply that the two class signatures are statistically very close to each other, and adjustment of input image channels and/or modification of training areas may improve the separability and therefore classification.

6.4 ANALYTICAL OUTLINE

One of the most commonly used and successful statistical texture analysis methods is the classification of statistics computed from grey level co-occurrence matrices (GLCMs). The grey level co-occurrence method is one of the most widely used texture analysis techniques, and was first introduced by Haralick *et al.* (1973). This method has been applied to 'medium' resolution (5×40 m pixel) sidescan sonar data by Reed and Hussong (1989). A novel technique based on fractal principles has been recently developed by Linnett *et al.* (1991, 1993) and successfully classified

very high resolution (few cm) bottom types. This chapter tests both of these techniques on TOBI sidescan imagery collected at the MAR (Cann *et al.*, 1992).

6.5 THE GREY LEVEL CO-OCCURRENCE MATRICES METHOD

6.5.1 Introduction

Haralick *et al.* (1973) computed a number of feature vectors based on the GLCM associated with each pixel in an image, at four different angles. They successfully classified various textures associated with sandstone photomicrographs, aerial photography and satellite imagery (with a classification accuracy of between 82 and 89%). The method has since been employed extensively, and some of the more relevant publications to my proposed application are those of Weszka *et al.* (1976), Shanmugan *et al.* (1981), Sali and Wolfson (1992), Pace and Dyer (1979) and Reed and Hussong (1989).

Weszka *et al.* (1976) performed a comparative study of various texture measures for the purpose of terrain classification of Landsat satellite data, and found the GLCM method provided a classification accuracy of over 90%, yielding better results than alternative Fourier and first order statistical methods. Shanmugan *et al.* (1981) applied the GLCM technique to radar imagery, which is closely akin to sidescan sonar data. They were able to yield different statistical features for various pre-determined bottom types. Sali and Wolfson (1992) applied a combination of first order statistical information, GLCM features and the fractal dimension of SPOT high resolution satellite imagery and successfully segmented the terrain types using unsupervised classification methods.

Pace and Dyer (1979) applied GLCM texture analysis to various sediment types, imaged by a 48 kHz sidescan sonar. They used 8 of Haralick *et al.*'s (1973) features and introduced 3 new ones, yielding a 'considerable success' in discriminating the bottom types, utilising a minimum distance classifier.

Reed and Hussong (1989) performed texture analysis on sidescan sonar imagery using GLCMs. Their data were SeaMARC II high resolution sidescan sonar data (pixel size of 5 m across-track and 40 m along-track); the GLCM method clearly distinguished between ground-truthed bottom types, such as lava flows, sedimentary bottoms and basaltic outcrops. The texture analysis work of Reed and Hussong

(1989) is described in further detail in Reed (1987), and since it is related to this application, it will be closely referred to throughout the proceeding description.

6.5.2 Theory

6.5.2a The Grey Level Co-occurrence Matrix The GLCM is a matrix constructed from the relationship of a given pixel to its specified neighbours. From this matrix, various statistical features may be computed, which can then be input into a number of standard classification procedures.

If the image I under analysis possesses N_x cells in the horizontal direction, N_y cells in the vertical direction, and N_g grey levels, then $L_x = \{1, 2, ..., N_x\}$ can represent the horizontal spatial domain, $L_y = \{1, 2, ..., N_y\}$ the vertical spatial domain, and $G = \{1, 2, ..., N_g\}$ the set of N_g grey levels. Each GLCM is a square matrix of dimension N_g comprising relative frequencies P_{ij} with which two neighbouring image cells, separated by distance d at angle ϕ, occur, one with grey level i and the other with grey level j. There is therefore a different matrix associated with each ϕ and d. For example:

$$P(i, j, d, 0°) = \#\{((k,l),(m,n)) \in (L_y \times L_x) \times (L_y \times L_x) \mid k - m = 0, |l - n| = d, I(k,l) = i, I(m,n) = j\}$$

$$P(i, j, d, 90°) = \#\{((k,l),(m,n)) \in (L_y \times L_x) \times (L_y \times L_x) \mid |k - m| = d, l - n = 0, I(k,l) = i, I(m,n) = j\}$$

where # denotes the number of elements in the set.

For example, considering a window of imagery

4	4	3	2	1
4	3	3	3	2
4	3	2	2	1
4	4	2	0	0
4	2	0	0	0

the GLCM, evaluated at 0° (i.e. in the vertical (down) direction) for a distance 1, appears as:

neighbour pixel value

reference pixel value	0	1	2	3	4
0	2	0	0	0	0
1	1	0	1	0	0
2	2	1	1	1	0
3	0	0	2	2	1
4	0	0	1	1	4

Before calculating feature statistics from the GLCM, the elements of the matrix are normalised to DN 0-255.

6.5.2b GLCM statistical features Haralick *et al.* (1973) suggested a number of statistical features which can be extracted from GLCMs. Eleven of these are described below:

A. Angular Second Moment

$$ASM = \sum_{i}\sum_{j}\{p(i,j)\}^2$$

This statistic is smallest when $p(i,j)$ are all as equal as possible, and is highest when the elements of the GLCM are clustered along the main diagonal. In effect, it is a measure of image coarseness.

B. Contrast

$$CON = \sum_{n=0}^{N_g-1} n^2 \left\{ \sum_{i=1}^{N_g}\sum_{\substack{j=1 \\ |i-j|=n}}^{N_g} p(i,j) \right\}$$

The image contrast measure derived from the GLCM is a weighted measure of deviation of entries from the main diagonal.

C. Correlation

$$COR = \frac{\sum_i \sum_j (ij) p(i,j) - \mu_x \mu_y}{\sigma_x \sigma_y}$$

μ_x, μ_y, σ_x and σ_y are the means and standard deviations of p_x and p_y. This statistic yields information about the degree of linearity of the image.

D. Variance

$$VAR = \sum_i \sum_j (i-\mu)^2 p(i,j)$$

The standard first order statistical variance measure applied to the GLCM.

E. Inverse Difference Moment

$$IDM = \sum_i \sum_j \frac{1}{1+(i-j)^2} p(i,j)$$

This statistic is a measure of local homogeneity. According to Sali and Wolfson (1992), the IDM feature provides 'bigger scores to images containing big blobs and having slow changes in grey levels'.

F. Sum Average

$$SAV = \sum_{i=2}^{2Ng} i p_{x+y}(i) \quad \text{where} \quad p_{x+y}(k) = \sum_{\substack{i=1 \\ i+j=k}}^{Ng} \sum_{j=1}^{Ng} p(i,j), \; k=2, 3, ..., 2N_g$$

Effectively, the features related to the sum of the GLCM refer to the statistical measures orthogonal to the main diagonal of the matrix, summed along these diagonals, e.g.

$\ddots$	S_3	S_4	S_5	S_6
S_3	S_4	S_5	S_6	S_7
S_4	S_5	S_6	S_7	S_8
S_5	S_6	S_7	S_8	S_9
S_6	S_7	S_8	S_9	$\ddots$

the numbers refer to k in the above equation, all S values with the same k number are summed to yield p_{x+y}.

G. Sum Variance

$$SVAR = \sum_{i\ 2}^{2Ng} (i - SENT)^2 p_{x+y}(i)$$

H. Sum Entropy

$$SENT = -\sum_{i} \sum_{j} p_{x+y}(i) \log\{p_{x+y}(i)\}$$

I. Entropy

$$ENT = -\sum_{i} \sum_{j} p(i,j) \log(p(i,j))$$

This is largest when all pixel pairs occur with the same frequency, such as in a perfectly random image. The entropy measure decreases as the structure of the image increases.

J. Difference Variance

$$DVAR = \text{variance of } p_{x-y} \quad \text{where} \quad p_{x-y}(k) = \sum_{i=1}^{Ng} \sum_{\substack{j=1 \\ |i-j|=k}}^{Ng} p(i,j), \; k=0, 1, ..., Ng-1$$

The difference refers to the sum of elements of the GLCM parallel to the main diagonal, such as

$\ddots$	S_1	S_2	S_3	S_4
S_1	S_0	S_1	S_2	S_3
S_2	S_1	S_0	S_1	S_2
S_3	S_2	S_1	S_0	S_1
S_4	S_3	S_2	S_1	$\ddots$

the numbers refer to k in the above equation, all S values with the same k number are summed to yield p_{x-y}.

K. Difference Entropy

$$DENT = -\sum_{i=0}^{N_g-1} p_{x-y}(i)\log\{p_{x-y}(i)\}$$

Any combination of these 11 features at various look directions and distances can now be selected for classification. The features shall be referred to by their 3 or 4 letter identifiers assigned here. All features are normalised to DN 0-255 over the whole image.

6.5.2c Previous theoretical investigations into GLCM technique A kernel size must be defined for investigating the neighbouring connectivities of each pixel; this kernel is the size of the window of pixels from which the GLCM is computed (e.g. section 6.5.2a example). Pratt (1978) states that in order to obtain sufficient statistical confidence in the GLCM method, each window should contain at least NG^2 pixels, where NG is the number of grey levels to which the image has been quantised. This rule has no particular theoretical basis, but simply ensures that a sufficiently large sample size is taken to provide a meaningful estimate for the GLCM joint probability distribution. If a large window size is selected, there is the risk that GLCMs may be generated over portions of different image textures. Also, a border around the edge of the image of width (KS-1)/2, where KS is the kernel size, will be lost. If the image is quantised to a very small number of grey levels, then a loss in accuracy will occur, particularly if certain textures are concentrated towards one end of the grey scale. A compromise is clearly essential; Pratt (1978) suggests a 16 grey level image and a kernel size of between 30 and 50 pixels per side. Reed (1987) also found 16 grey levels to be best, with a 49×6 pixel kernel size.

Reed (1987) recommended performing the quantisation of a typical 256 grey level image by simply dividing each input intensity by the maximum of the desired range of quantisation and rounding to the nearest integer. He felt this was superior to some of the more complicated quantisation methods used by Haralick *et al.* (1973). I have followed Reed's (1987) advice and used one of the *PBMplus* routines (*pnmdepth*) (Poskanzer, 1991) to perform the grey scale compression.

Reed (1987) performed a detailed theoretical investigation of the GLCM textural analysis on synthetic sidescan sonar data, which comprised various combinations of sinusoidal patterns. He investigated the effect on GLCM statistical features of varying the wavelength of features, the stationarity of an image (i.e. its mean intensity), its noise sensitivity, rotation invariance and wavelength mixing. Since the TOBI data is at a higher resolution than Reed's (1987) SeaMARC II (~6 m compared with 40 m, respectively), the effect of some of these phenomena may change and be worthy of further investigation. Reed (1987) investigated the effect of five of Haralick's (1973) statistics only (ASM, CON, COR, IDM and ENT), plus a sixth of his own derivation (ISO: the difference between orthogonal GLC matrices - a measure of isotropy).

Reed (1987) identified a sensitivity to wavelength with ASM, CON, ENT and ISO; but found no effect with COR and IDM. If the wavelength of the synthesised texture was greater than the kernel size, then the classification accuracy declined dramatically, as the texture would only be partially sampled. This suggested that it would be prudent to measure the 'wavelengths' of the texture associated with each of my bottom types and therefore to ensure that the kernel dimensions were larger. Some of the analyses described later do investigate the effects of varying image resolution and kernel size on classification accuracy, to verify Reed's (1987) deductions.

No effect of image stationarity on statistical features and classification accuracy was confirmed by Reed (1987), so any problems due to gain variations of the TOBI data should not affect the results. This phenomenon is not investigated any further.

Regarding noise sensitivity, Reed (1987) observed that by adding 12.5% random noise to his synthetic sidescan images, classification accuracy was reduced by 50%; the addition of 50% noise made the textures indistinguishable. This suggests that the input images should be as clean as possible. Since only small samples of TOBI data have been selected for this preliminary investigation, chosen for their high quality, the effect of noise should not really be a problem. If the technique is to be applied to a wider field of data, Le Bas (1993) describes some methods for cleaning up the principal artifacts associated with TOBI data, which could be applied before performing texture analysis.

Reed (1987) rotated his synthetic textures and found that CON, IDM, ENT and ISO were invariant under rotation, while ASM and COR were not. Since the feature vectors are evaluated for the four angles of 0, 45, 90 and 135°, no further examinations in this field will be attempted.

Finally, the phenomenon of wavelength mixing was investigated by Reed (1987), where he synthesized the effects of two simultaneous scales of roughness; one larger than the kernel size (>250 m) and one much less (~20-40 m). ASM, CON, ENT and ISO showed little or no contribution from the longer wavelengths, although COR and IDM were affected by mixing the shorter wavelengths with a signal of wavelength ten times the kernel size (2.5 km), which he suggested may actually prove beneficial for classification purposes. The effect of wavelength mixing was not explored any further by Reed (1987), and will not be assessed any further in this study.

6.5.3 Principal Component Analysis

Davis (1973) outlines the basis for principal component analysis (PCA), alternatively known as the Karheunen-Loev or Hotelling transform. PCA is a linear transform which rotates the axes of image space onto the lines of maximum variance. The principal components are simply the eigenvectors of the variance-covariance matrix generated from a series of input image channels. The transformed image channels, or eigenchannels, are orthogonal with respect to each other, since the variance-covariance matrix is by nature symmetrical. The input channels are not necessarily orthogonal as they may be highly correlated. The variance implicit in the original channels is forced into the new uncorrelated channels and the eigenchannel with the highest eigenvalue contains more variance than the subsequent eigenchannels possessing decreasing eigenvalues. A new set of n-dimensional feature vector eigenchannels can be generated by using the eigenvectors as loadings on the original input feature vector image channels, for example

$$Y_1 = a_1X_1 + a_2X_2 + \ldots + a_nX_n$$
$$Y_2 = b_1X_1 + b_2X_2 + \ldots + b_nX_n$$
$$\vdots$$
$$Y_n = m_1X_1 + m_2X_2 + \ldots + m_nX_n$$

where Y_1-Y_n are the eigenchannels, X_1-X_n are the input channels and a_1-a_n are the eigenvector loadings for eigenchannel 1, through to m_1-m_n which are the eigenvectors for eigenchannel n.

In the following analyses, image classification has both been performed on the transformed eigenchannels and selected raw input feature vector images, based upon the weighting of their principal components. The higher the eigenvector loading and the larger the associated eigenvalue, the more significant this particular input feature vector image is in terms of data variance.

6.5.4 Software

The GLCM generation programs are included in the appendices. *get_glcm.c* (appendix 11) computes a GLCM for a single specified pixel in an input *PGM* format (Poskanzer, 1991) image. *get_glcm* operates on a command line interface as:

get_glcm infile xwinsize ywinsize xpixel ypixel opfile0 opfile45 opfile90 opfile135

Displaying these GLCMs is useful for the initial determination of the number of grey levels, kernel size, image resolution, etc. *get_stats.c* (appendix 12) also operates on a command line structure, where flags may be used to select any one of the eleven of Haralick's (1973) statistical features (denoted a-k), at any one of four angles (0, 45, 90 and 135°), at a distance of 1 unit. The kernel size in x and y dimensions must also be specified, so the usage is:

get_stats infile xwinsize ywinsize ang [-a] [-b] ..... [-k] > opfile

These programs utilise input and output subroutines for *PGM* images, these were: *readpgmhead.c* (appendix 13); *writepgmhead.c* (appendix 14); *readline.c* (appendix 15); and *writeline.c* (appendix 16).

All classification routines used were part of the *EASIPACE* image processing system (PCI, 1988). One simply creates an *EASIPACE* database, reads in the PGM files, then generates class statistics (*CSG*), investigates training area separability (*SIGSEP*), and classifies via maximum likelihood (*MLC*) or minimum distance (*MINDIS*) classifiers. The entire classification procedure was incorporated into a shell (*CLASS*) and an additional shell (*ACC*) (appendices 17 and 18, respectively)

was generated to produce a measure of classification accuracy through spatial comparison with a ground truth image.

6.5.5 Preliminary Analysis

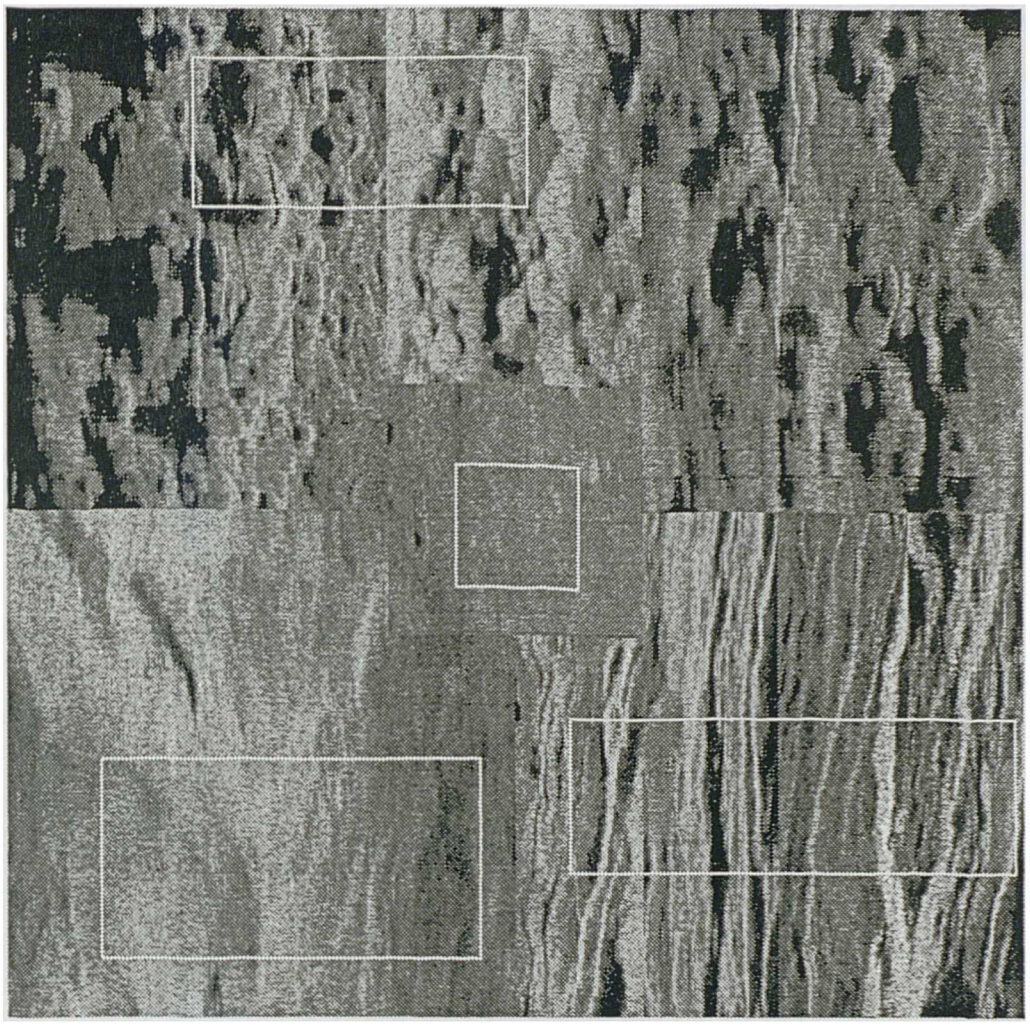


Figure 6.1 *Composite TOBI test image (top - hummocks; bottom left - landslide; bottom right - fissures; centre - sheetflow). The boxes represent the training areas used for the GLCM texture analysis testing.*

A composite image was constructed from portions of TOBI data collected on the Mid-Atlantic Ridge (MAR) at 24-30°N, and described in detail by Cann *et al.* (1992), see figure 6.1. Four of the principal bottom types identified by Cann *et al.* (1992), and confirmed by dredging and bottom photography, were the prolific hummocky basaltic terrain which formed the majority of the neo-volcanic zones of the MAR here; landslides, which were also confirmed to be present here by Tucholke (1992) through bathymetric analysis and were commonly found at the ends of the spreading segments; sheetflows, which be composed of the same material as the

hummocks, but clearly at a different roughness scale; and fissures which were particularly prominent in one segment imaged by Cann *et al.* (1992) where no neo-volcanic zone was identified bathymetrically, but can also be observed in smaller expanses throughout the rest of the data. The composite image comprises the four bottom types, though the hummocks and fissures are mosaics of various sections of data, since it was difficult to find extensive areas of these single bottom types.

6.5.5a Selection of initial GLCM parameters Section 6.5.2c advises using a 16 grey level input image, and consequently also recommending a kernel that contains at least 16^2 or 256 pixels. The TOBI data have been decimated by a factor of eight across-track to produce square pixels of dimension 6×6 m, which I shall describe as full resolution data in the following text. The pixels and therefore the kernel size are geometrically square, and as all kernels inherently comprise an odd number of pixels on each side surrounding each single pixel, the minimum size for a kernel would be 17×17 pixels, or 102×102 m for full and 204×204 m for half resolution data. Additionally, since Reed (1987) concluded that the kernel size should be larger than the wavelength of the texture, the effective wavelength of the hummocks was estimated to be 180 m and that of the fissures (orthogonal to their alignment) was 50 m. This suggests that either full resolution data should be analysed with a minimum kernel size of 30 pixels, or half resolution data with a minimum kernel size of 15 pixels. No obvious wavelength was measurable for the landslide and sheetflow bottom types.

6.5.5b Generation of GLCMs for specific pixels Figures 6.2-6.5 display contour plots for GLCMs associated for a single pixel, one for each bottom type of the composite image, and evaluated four times for the four look directions. All GLCMs are computed for a lag of one pixel. The GLCMs are displayed with their origin at the bottom left, rather than the conventional top left; the main diagonal of the matrix would therefore be from bottom left to the top right of the plot.

Figure 6.2 illustrates GLCMs generated for a full resolution image, i.e. kernel size of 102×102 m, and pixel size of 6 m. There is a clear distinction of the GLCM characteristics for the landslide and sheetflow compared with the hummocks and fissures. The former possess closely packed, high value matrix elements, whilst the latter appear more spread out. The differentiation between the landslide and sheetflow and between the hummocks and fissures are not clear. Reed (1987) did state that the kernel should be larger than the texture wavelength, which it is not here.

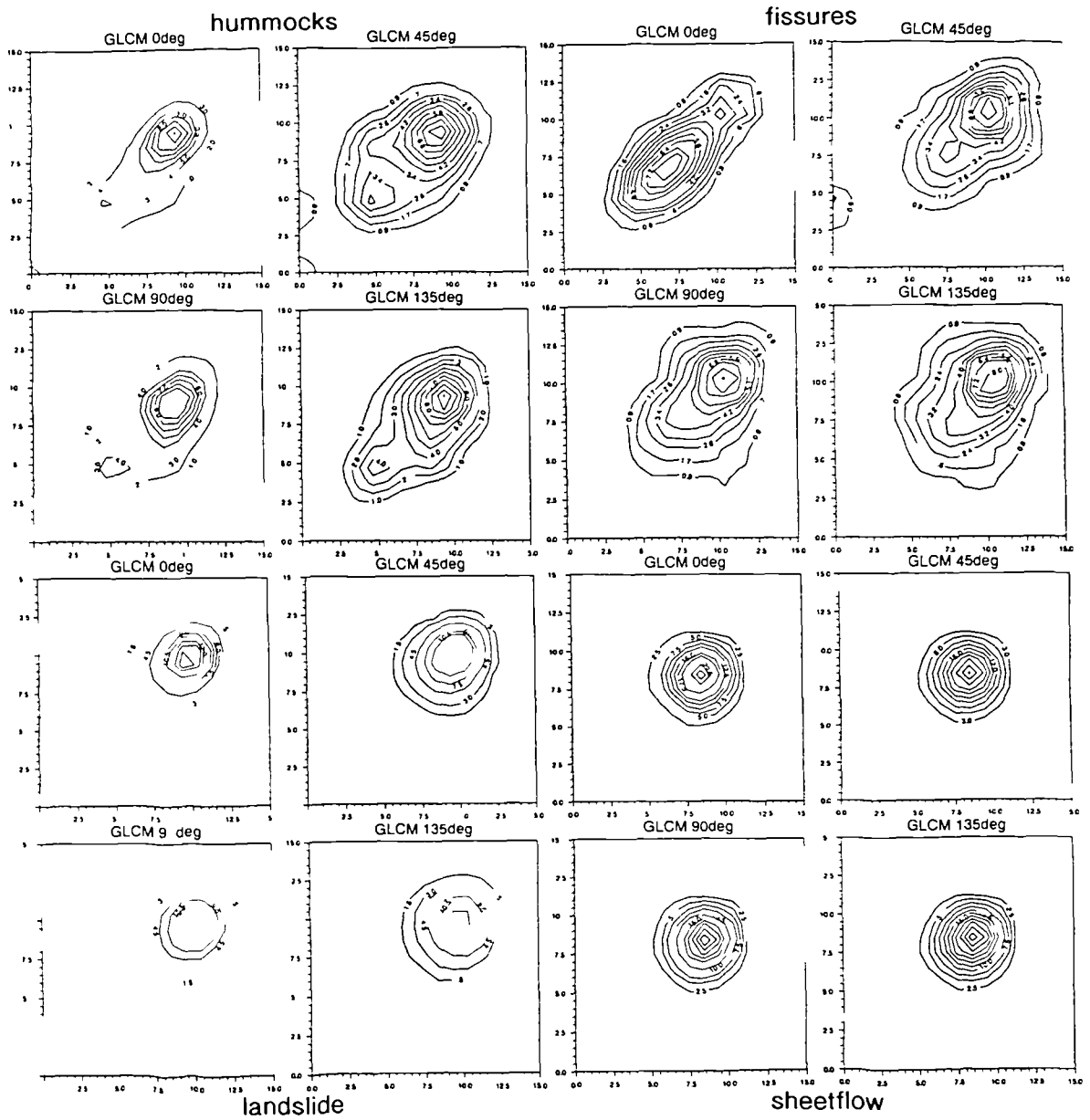


Figure 6.2 *GLCM contour plots computed from a 16 grey level, full resolution image (pixel 6×6 m, kernel 102×102 m), for specific pixels, at angles of 0, 45, 90 and 135°, and for a lag of 1 pixel. Four GLCMs are determined for each bottom type (hummocks, fissures, landslide and sheetflow). The matrix origin is at bottom left, so the main diagonal is from bottom left to top right.*

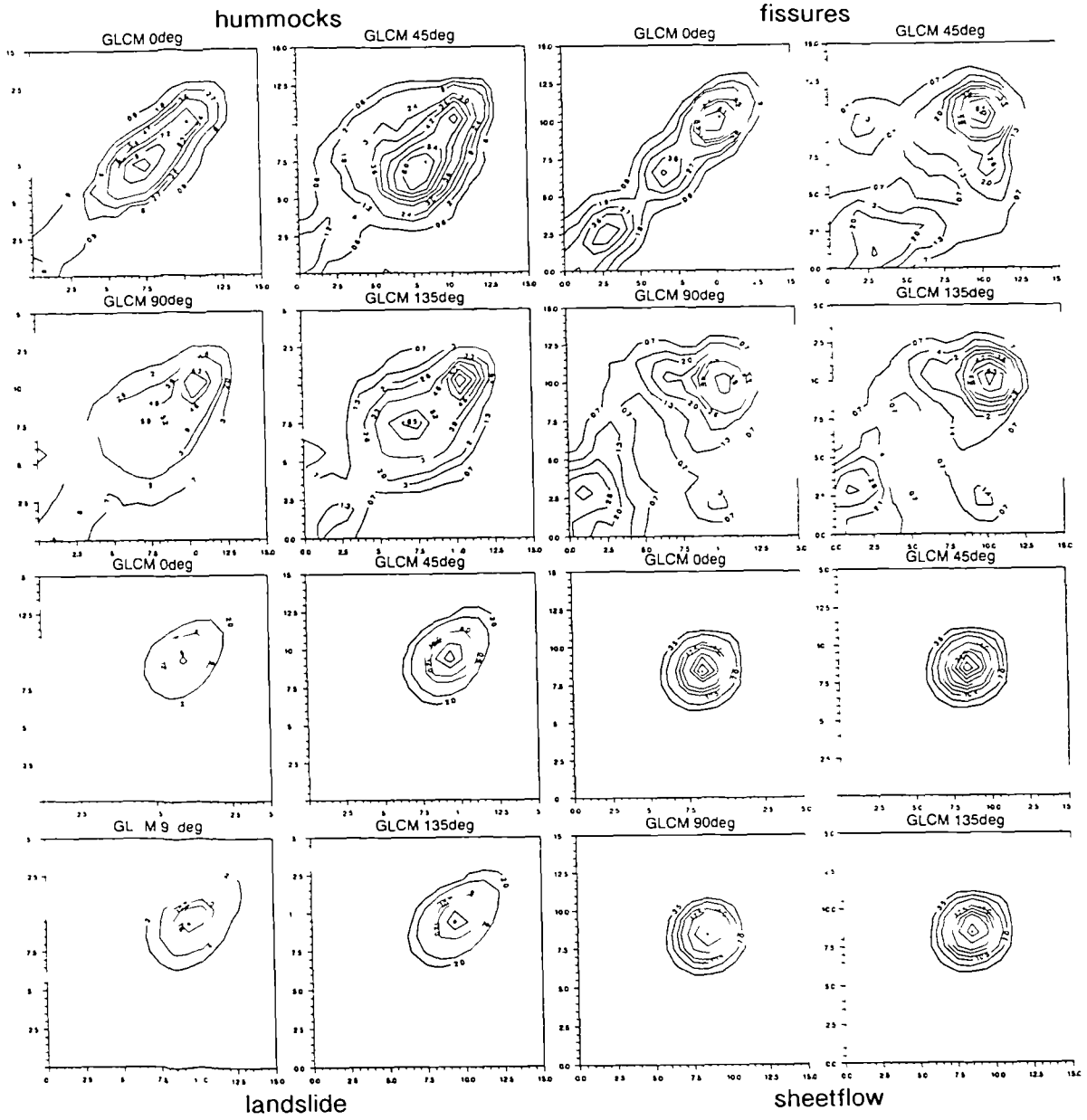


Figure 6.3 GLCM contour plots computed from a 16 grey level, half resolution image (pixel 12×12 m, kernel 204×204 m). Further details as figure 6.2.

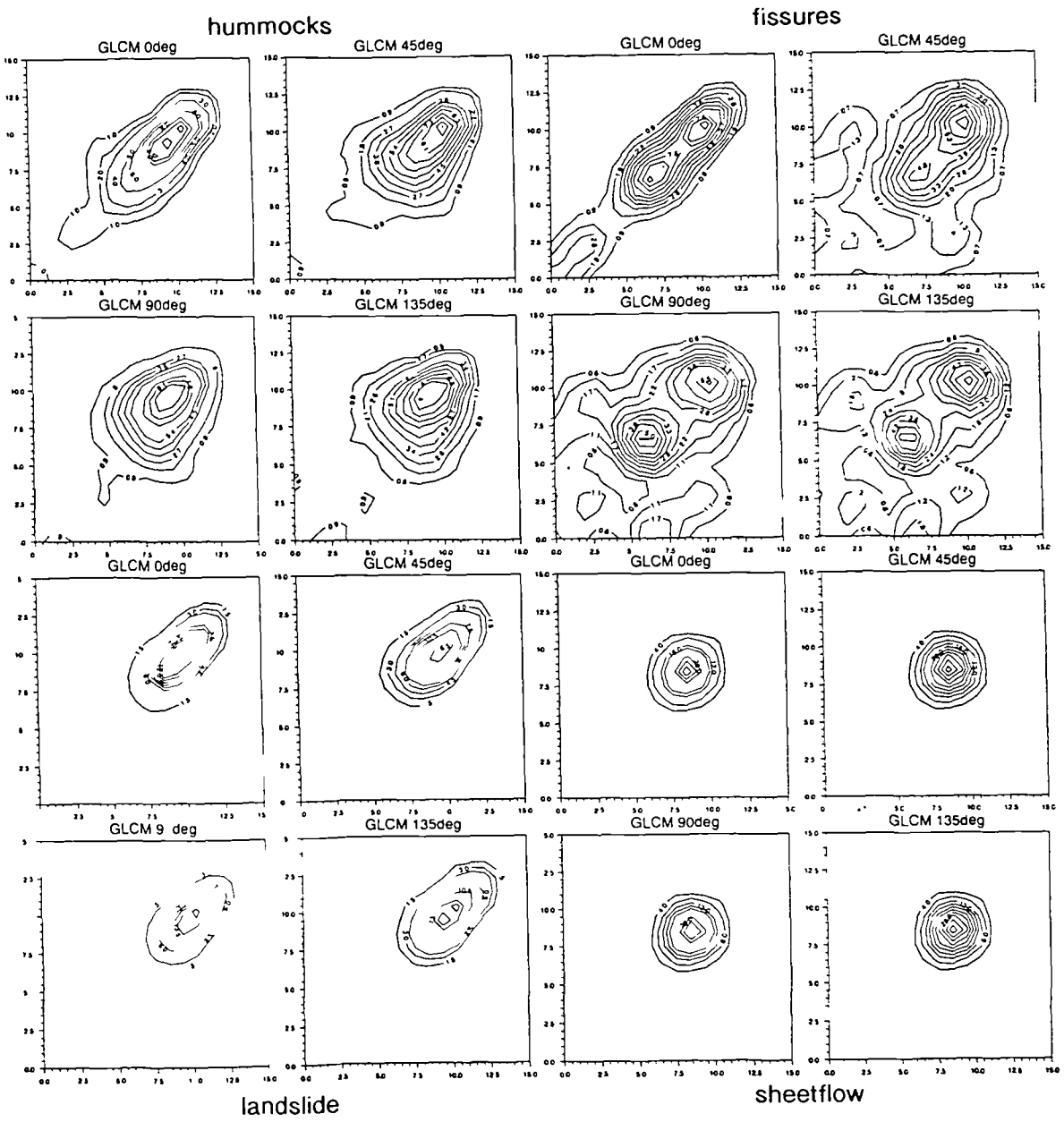


Figure 6.4 *GLCM contour plots computed from a 16 grey level, quarter resolution image (pixel 24×24 m, kernel 408×408 m). Further details as figure 6.2.*

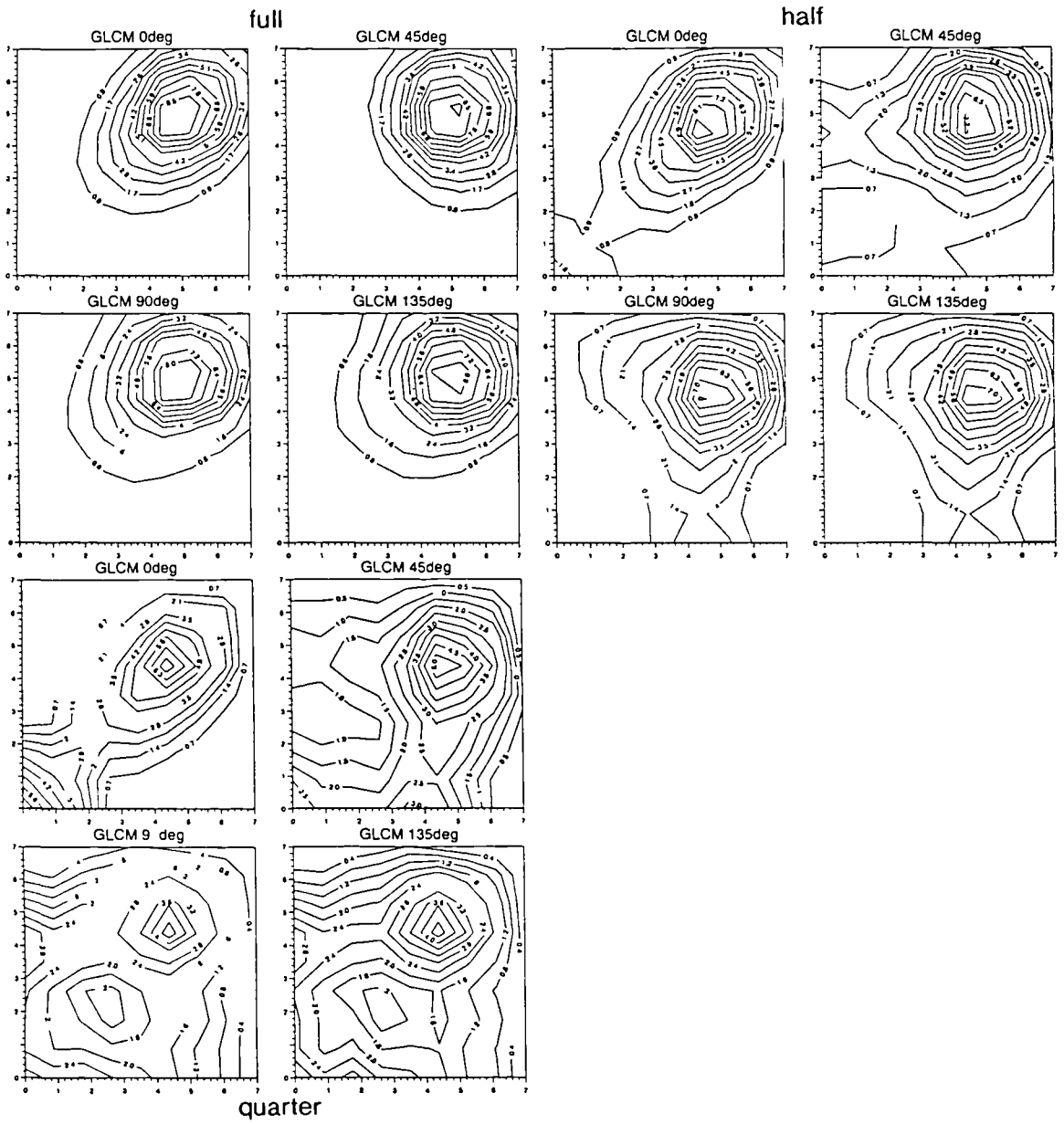


Figure 6.5 *GLCM contour plots computed from an 8 grey level image, at full, half and quarter resolution (pixels respectively 6×6, 12×12, and 24×24 m; kernels respectively 102×102, 204×204, and 408×408 m). These were all computed for a specific pixel within the hummocks training area site. Further details as figure 6.2.*

Figure 6.3 illustrates the GLCM distribution for a half resolution image, possessing a pixel size of 12 m and kernel size of 204 m. The most notable feature here is the angular effect on the matrices, particularly for the fissures. The characteristics at 0° (parallel to the fissures' trend) show tightly constrained matrix elements along the main diagonal; the GLCMs for the other angles are distributed much wider throughout the matrix. This is seen to a much more minor effect with the hummocks, indicating a slight 'vertical' trend, which may be due to the fact that this angle is perpendicular to the insonification direction, so no shadowing occurs. The landslide and sheetflow are now more clearly distinguised, with some trend along the main diagonal for the former, but none evident for the latter.

Figure 6.4 shows the GLCMs for a quarter of the original resolution, that is a pixel size of 24 m and a kernel dimension of 408×408 m. The characteristics are very similar to those observed in figure 6.3 at a half resolution, but do not appear to provide any significant supplementary information. Clearly one is beginning to lose a substantial proportion of the data at this resolution.

Finally, figure 6.5 shows the effect of generating GLCMs for an input image quantised to just 8 grey levels. Clearly, the bottom types are not so well distinguished since the data are confined to fewer matrix elements than before, restricting the possible distribution variations. One can conclude at this stage that quantising the original image to just 8 grey levels is not satisfactory since better results were achieved with 16 grey levels.

6.5.5c Histogram distributions of GLCM statistical features *get_stats* was used to calculate Haralick's (1973) 11 statistical features from the GLCMs generated for every pixel in the composite bottom type image. The statistics were computed for a half resolution, i.e. 256×256 pixels, image with a 17×17 pixel kernel and 16 grey levels. One run of the program, i.e. computing one statistical feature at one angle, took approximately ten minutes on a Sun SPARC 10 processor. This obviously requires a large amount of computational power, which is why most of the testing has been performed on half resolution data. Additionally, it is possible that not all these statistical feature vectors (44 of them) are required. However, they are necessary for these initial investigative steps. Since the programs operate under a command line structure, a batch of runs can be simply incorporated into a shell and then run overnight. The identical training areas for the four bottom types (figure 6.1), were the same regions selected in the fractal texture analysis section (figure 6.18b). Histograms were computed for each of these training regions. Additionally, simple

statistics such as the mean and standard deviation (SD) of these distributions were calculated. Since 44 images were generated, just six of the histogram characteristics will be described here; the reader may note that these were selected due to their interesting attributes and that the statistical features not described are therefore not particularly distinctive. The next section (6.5.5d) will summarise these.

Figure 6.6a illustrates the histograms computed for the four bottom types for the correlation statistic COR, measured at 0°. As stated in section 6.5.2b, COR measures the linearity of an image. The hummocks, fissures and landslide exhibit similar means (of DN 99, 105 and 90, respectively) and standard deviations (DN 30, 32 and 25, respectively); however the sheetflow possesses a much lower mean of DN 73 and only a SD of DN 8. It can therefore be concluded that the sheetflow is represented by low linearity (COR) values at 0°, whilst the other bottom types are all characterised by higher values, and are not individually distinctive.

The histograms for the inverse difference moment (IDM) statistic for 0° are displayed in figure 6.6b; this provides information about the homogeneity of an image. There are large means (DN 116 and 120, respectively) and SDs (DN 24 and 27, respectively) for the hummocks and fissures. The respective means are far lower for the landslide and sheetflow, DN 75 and 52, as are their SDs, DN 14 and 8. The landslide and sheetflow are therefore distinguishable, and are distinct from the hummocks and fissures, though these latter two cannot be resolved.

Figure 6.6c presents the histograms for the entropy (ENT) statistical feature, which is expected to yield lower values as the structure of an image increases. As predicted, the sheetflow possesses the highest mean (DN 221), with a small SD (DN 8); the landslide a little lower with a mean of DN 203 and SD of DN 11. The hummocks and fissures appear very similar again, with respective means of DN 155 and 167, and SDs of 29 and 25. These lower values make sense as one would consider there to be more structure in the highly fluctuating hummocks and fissures, compared with the landslide and sheetflow.

The histograms of the angular second moment (ASM) feature vector, representing the coarseness of the image are shown in figure 6.6d. The hummocks, fissures, landslide and sheetflow have respective means of DN 83, 75, 52 and 37, with SDs of DN 28, 25, 13 and 10. ASM does distinguish all bottom types quite well, though the visual histogram distributions of the hummocks and fissures still bear some striking similarities.

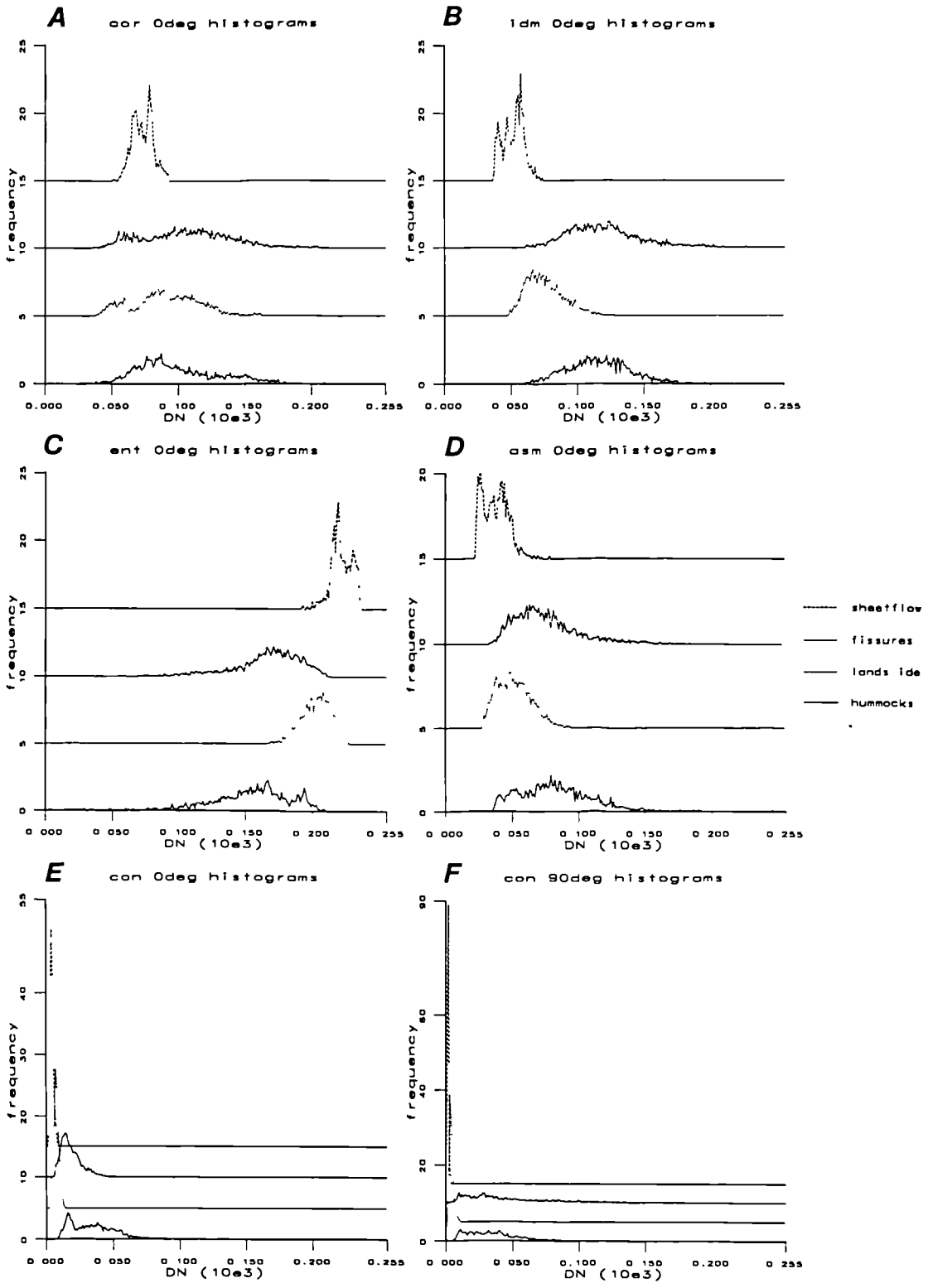


Figure 6.6 *Histograms of selected GLCM statistical features: (a) COR 0°; (b) IDM 0°; (c) ENT 0°; (d) ASM 0°; (e) CON 0°; and (f) CON 90°. From top to bottom, each graph presents the responses for the sheetflow, fissures, landslide and hummocks training areas.*

Finally, the contrast (CON) feature is considered, both at angles of 0 and 90° (figures 6.6e and f), since Reed (1987) notes that this particular statistic varies strongly with look direction. So comparing the angular effects, the hummocks training area has means of DN 34 for both 0 and 90°, and respective SDs of 16 and 19; therefore statistics computed for both look directions are very similar. Whereas the fissures possess a mean of DN 18 at 0° (parallel to the fissures) plus a SD of DN 7, then a mean of DN 50 and SD of DN 37 at 90° (orthogonal to the fissures). Consequently, a clear contrast is observed for the fissures at 0 and 90°. The CON values for the landslide and sheetflow are quite similar: landslide means of DN 7 and 4 for 0 and 90° with SDs of 2 for both; and means of DN 4 and 2 for the sheetflow and SDs of DN 1.

The conclusion from viewing the histograms of these feature statistics is that neither one statistic at one angle, one statistic at all angles, or all statistics at one angle will yield high quality classifications. Additionally, all 44 of the statistics are not required, as many of the statistics do appear very similar. This latter point is illustrated and described better in the next section (6.5.5d).

6.5.5d Angular variation of GLCM statistical features Figure 6.7 presents all the mean feature vectors for all the computed statistical features and for the four training areas, against the four look directions of 0, 45, 90 and 135°. This section summarises what the statistical features actually represent, and this provides a qualitative assessment of their potential for characterising the TOBI texture types under consideration. The major points gained from figure 6.7 are:

I Angular variations Strong angular variations were noted in section 6.5.5c between the CON measurements at 0 and 90°, with a mean difference of DN 32 between the two. The mean feature vectors at angles of 45 and 135° are similarly distinct from those at 0°. All the entropy features (ENT, SENT and DENT) have relatively high mean vectors at 45 and 90°, compared with those at 0 and 135°; however, this was found to be true for all bottom types, not just fissures, and so is of not much use in this respect. IDM increases for fissures at 0°, but not for the other bottom types. Since IDM measures local homogeneity of the image, it is sensible that it should be relatively larger parallel to the fissures' alignment. ASM varies according to look direction, but not relative to a specific bottom type. Finally, the mean feature vector of COR decreases for all bottom types from 45 to 0°, except fissures where it increases. COR measures linearity, so one would expect a feature vector computed parallel to these fissures to yield a high magnitude. Reed (1987) stated that CON,

ENT and IDM were strongly affected by angular variations, but ASM and COR were not; these results do not fully verify his conclusions.

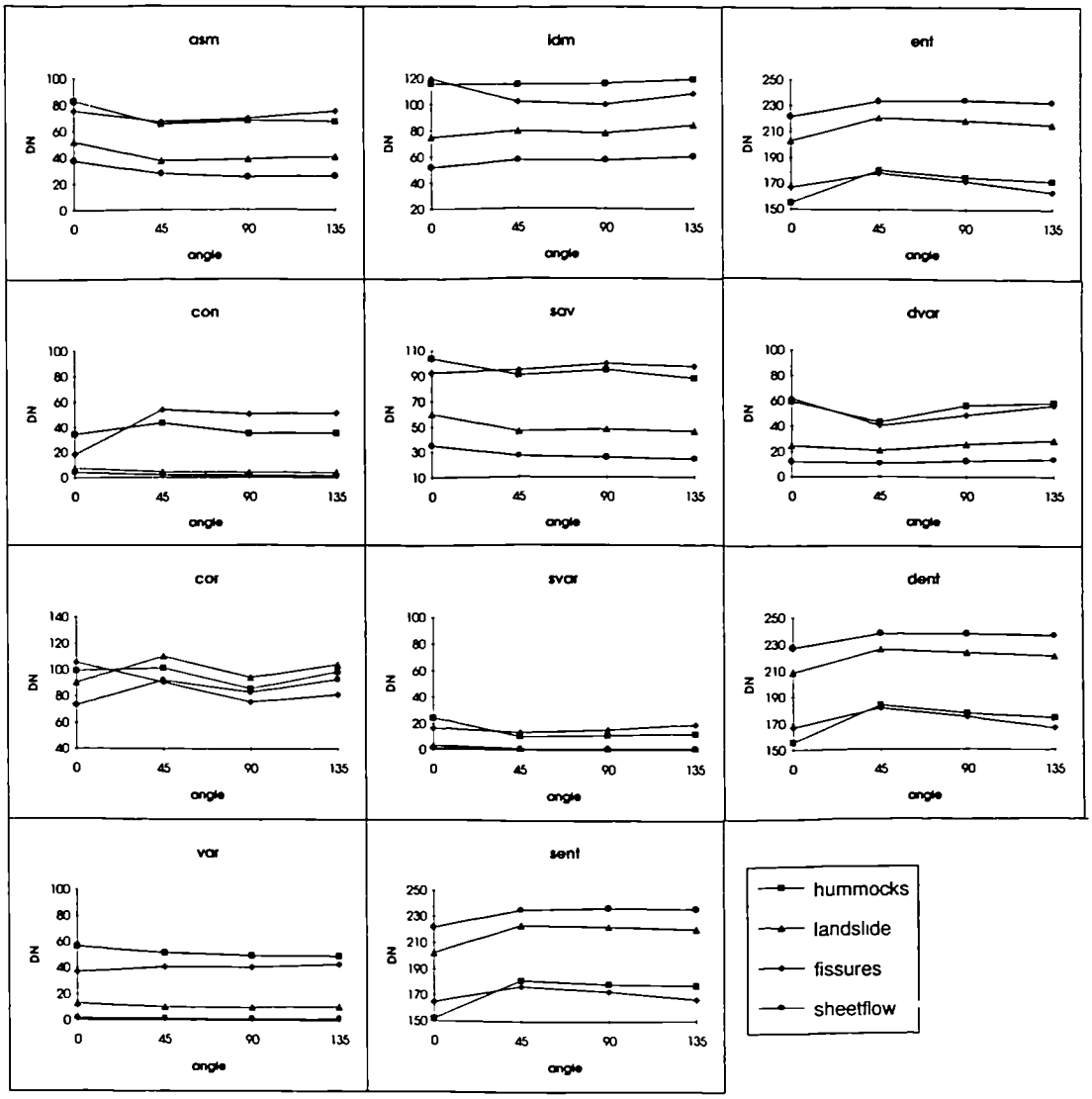


Figure 6.7 *Angular responses of all 11 GLCM statistical features. DN is plotted against angle, and four curves corresponding with average values for each bottom type training area are plotted on each graph.*

II Bottom type variations What about the magnitude of the difference between the mean feature vectors and bottom types, discounting any angular variations? Again, section 6.5.5a has already discussed these to a degree, so some additional information for completeness is recorded here. Feature mean vectors, such as IDM and VAR appear to distinguish between bottom types fairly well, though since differences are of minimum magnitude DN 5 and standard deviations can reach DN 25, they cannot alone be used for classification purposes. Some other features are

only able to separate groups of bottom types, for example, ASM, SAV, DVAR and the entropy measurements can only discern between landslide or sheetflow on the one hand, and then the hummocks or fissures on the other, though the hummocks and fissures cannot themselves be divided. Alternatively, CON and VAR can isolate hummocks and fissures, or landslide and sheetflow as pairs, but not separately.

III Similarity between features Comparing the overall responses of features, it is clear that some exhibit very similar characteristics; notably the three entropy features: ENT, SENT and DENT. These features are all related to the structure of the image (mean vector increases as structure decreases). Point IV details the usefulness of the sum and difference statistics in addition to those computed simply from the rows and columns of the GLCM, but it is clear here that no extra entropy information is gained from calculating all three features.

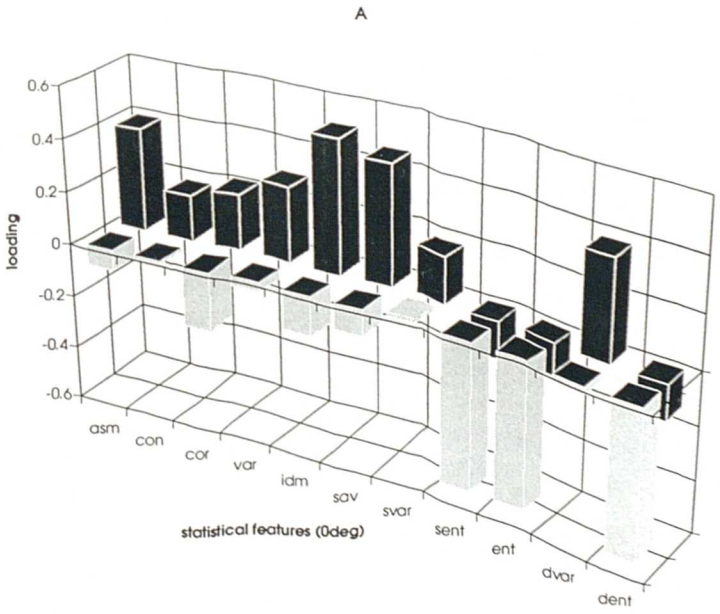
IV Sum and difference statistics A final point considers the justification of computing sum and difference statistics. Re-iterating section 6.5.2b, the sum features calculate statistics orthogonal to the main diagonal of the GLCM, while the difference features are computed from sums parallel to and at equal distances from the main diagonal. The sums therefore represent the statistics of the sums of variable connectivity magnitudes, while the differences are the statistics of the sums of equal connectivity magnitudes. Point III found no real profit for computing the three entropy measurements for these particular texture types. However, VAR, SVAR and DVAR exhibit some potentially important and informative variations. VAR distinguishes between all bottom types, and has different angular responses for hummocks and fissures. It decreases at 0° for the fissures, due to being parallel to their alignment, and increases for the hummocks, where they must possess a higher variability. In contrast, SVAR possesses a puzzling angular variation for the hummocks, but not for the fissures, while the two responses for DVAR are quite similar.

6.5.5e Principal Component Analysis of Statistical Features The theory of PCA is discussed in section 6.5.3. Sections 6.5.5c and d above describe subjectively how particular features at particular angles may be useful in distinguishing between the bottom types under investigation; this will now be examined in a quantitative manner by discussing the results of PCA applied to selected feature images. Some sort of selection of a limited number of feature images prior to classification is required since the *EASIPACE* (PCI, 1988) class signature generator (*CSG*) only allows a maximum of 16 input channels. In fact, for statistical validity, a training site

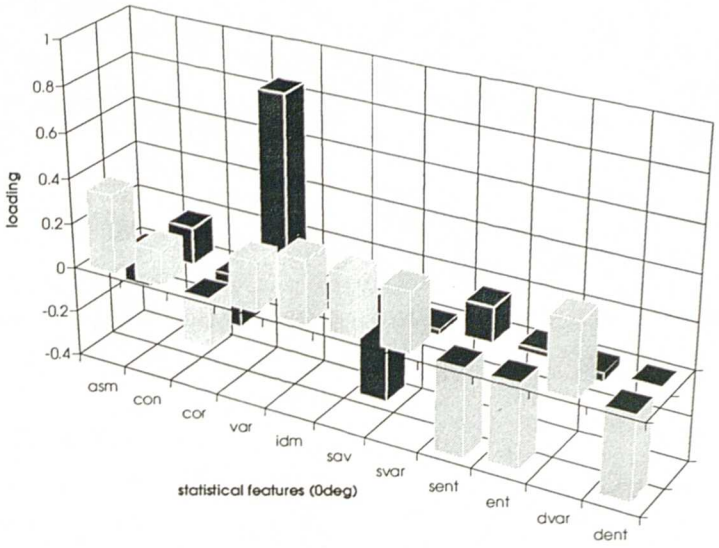
must contain a minimum of $5\ (n^2 + n)$ pixels, where n is the number of input channels to be used for classification (PCI, 1988). The smallest training area for this composite image at a resolution of 12 m per pixel contains 1056 pixels, limiting the maximum number of channels for classification purposes to 14. This section will just describe some of the magnitudes of the principal components, and section 6.5.6b(I) will deal with the merits of performing PCA prior to classification.

All data were normalised to DN 255 and the *PCA* utility in *EASIPACE* (PCI, 1988) was implemented; a limit of 16 input channels is set, so PCA could not be performed on all 44 feature vector images at once. Instead, two groups of input channels (feature vectors) were selected: (1) all eleven features, at 0° only; and (2) a subjectively determined set. The latter particularly defies some of the basics of PCA since the selections are somewhat subjective, however the number of raw channels had to be limited somehow. The selections were based upon the distinctive histogram and angular feature vector responses in figures 6.6 and 6.7.

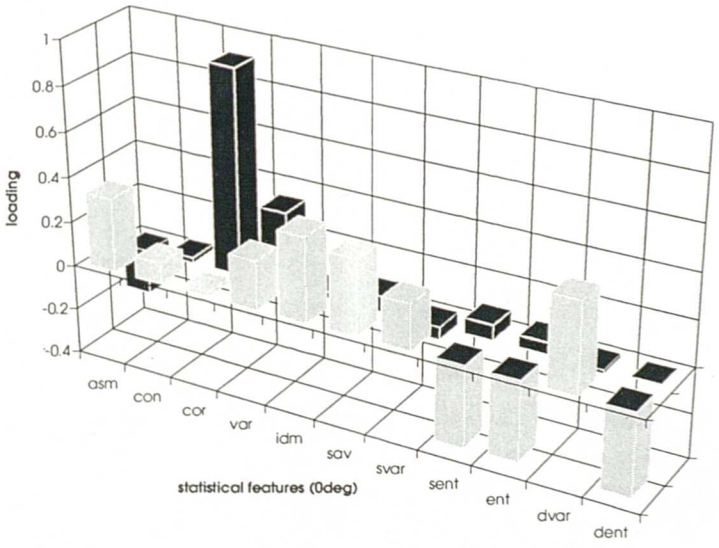
I PCA on 11 statistical features at 0°



B


1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
loading
asm
con
cor
var
idm
sav
svar
sent
ent
dvar
dent
statistical features (0deg)

C


1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
loading
asm
con
cor
var
idm
sav
svar
sent
ent
dvar
dent
statistical features (0deg)

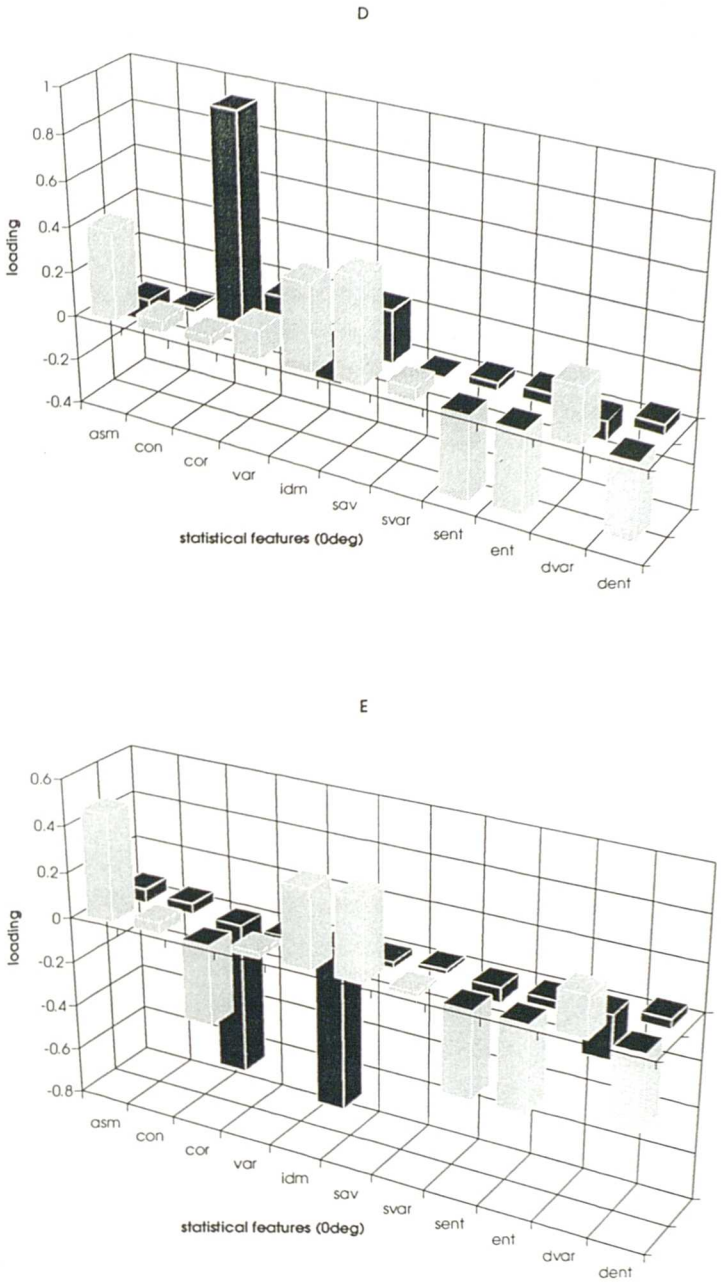


Figure 6.8 *Eigenvector loadings for all statistical features at 0° only. The graph in the foreground presents eigenchannel 1, whilst the graph in the rear shows eigenchannel 2. The eigenvector weightings are plotted for: (**a**) the whole image; (**b**) the hummocks training area; (**c**) the fissures training area; (**d**) the landslide training area; and (**e**) the sheetflow training area.*

Figure 6.8 presents plots of the eigenvector loadings, or principal components, for the whole image and its constituent training regions. The 'front' plot represents

the eigenvectors associated with eigenchannel one (EC1), while the 'rear' corresponds with those for eigenchannel 2 (EC2). Table 6.I summarises the results:

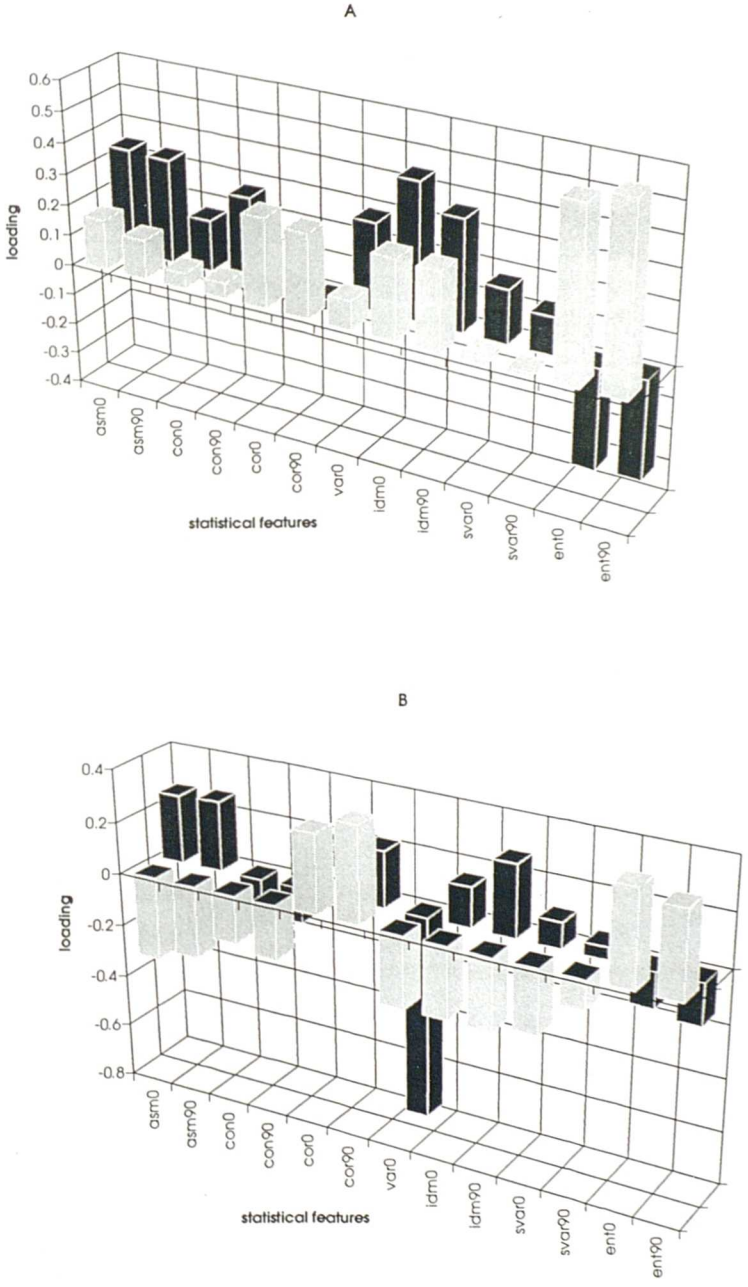
Region	Cumulative variance		Significant eigenvector weightings	
	EC1	EC2	EC1	EC2
whole	63	91	ENT SENT DENT	IDM SAV ASM
hummocks	68	88	ENT SENT DENT	VAR SAV
fissures	74	92	ENT SENT DENT	COR VAR
landslide	54	94	SAV IDM ASM	COR SAV
sheetflow	94	not significant	ASM COR IDM	not significant

Table 6.I - PCA results for all features at 0°

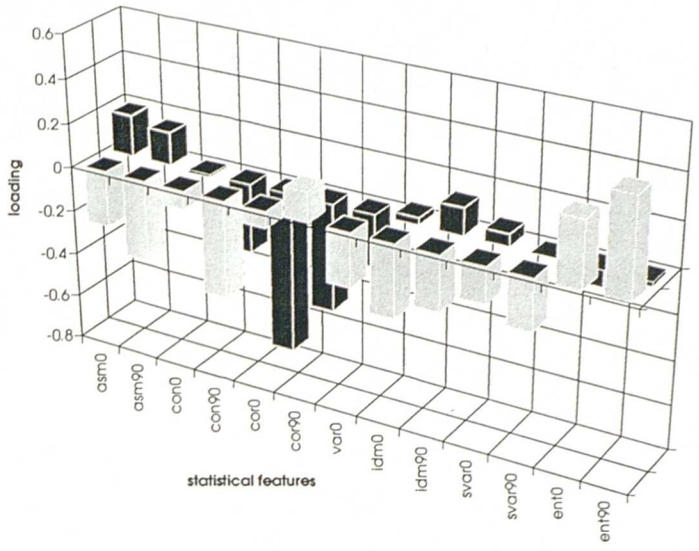
With regard to the overall image and the hummocks and fissures training areas, the variance of EC1 is mainly restricted to the three entropy measurements (ENT, SENT and DENT). In all, these are distinct from the other features, but the difference between the three of them does not appear to be significant. This is concerning since the overall variance of the entropy features is relatively high and led to the question of whether feature channels should be standardized or not; this is addressed in part III of this section.

Regarding the fissures training area, since the statistical features are all at 0° here, they are unlikely to optimise the alignment of the fissures. The landslide input feature vector channels must in general be less correlated than the hummocks and fissures, because the variance encompassed in EC1 is just 54%, though adding up to 94% with EC2. The entropy measurements are superseded here by IDM, SAV and ASM for EC1; then COR and a minor contribution of VAR for EC2. Finally, for the sheetflow training area, different features again appear to contribute to the variance of the input channels, though these channels are more highly correlated than before. EC1 comprises 94% of the variance, so the components of EC2 can be disregarded. ASM, COR and IDM exhibit the largest principal components.

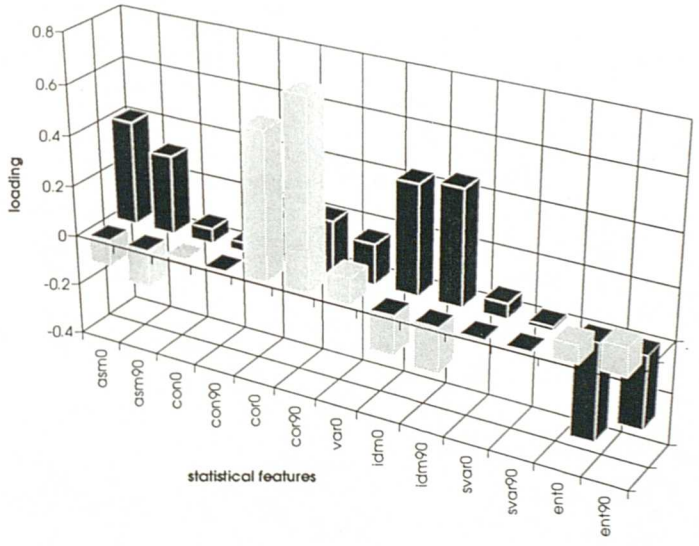
II PCA on selected statistics at various angles These statistics were chosen somewhat subjectively from the information presented in sections 6.5.5c and d, rather than the possibly dubious PCA results of part I, and they also constitute the various combinations of feature vectors which were used for the classification attempts in table 6.IV. These PCA results are presented below in figure 6.9 and in table 6.II.



C


0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
loading
asm0
asm90
con0
con90
cor0
cor90
var0
idm0
idm90
svar0
svar90
ent0
ent90
statistical features

D


0.8
0.6
0.4
0.2
0
-0.2
-0.4
loading
asm0
asm90
con0
con90
cor0
cor90
var0
idm0
idm90
svar0
svar90
ent0
ent90
statistical features

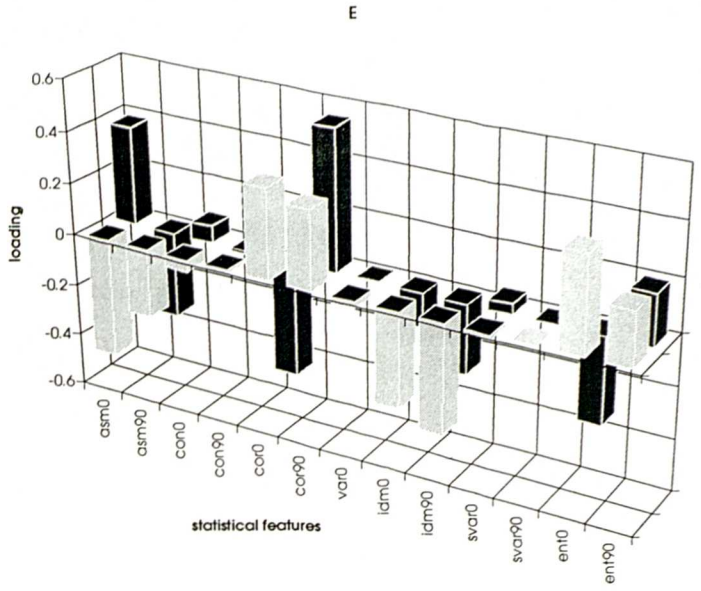


*Figure 6.9 Eigenvector loadings for selected statistical features from eigenchannels 1 (front) and 2 (rear). The eigenvector weightings are plotted for: (**a**) the whole image; (**b**) the hummocks training area; (**c**) the fissures training area; (**d**) the landslide training area; and (**e**) the sheetflow training area.*

Region	Cumulative variance		Significant eigenvector weightings	
	EC1	EC2	EC1	EC2
whole	52	83	ENT90 ENT0 COR0	IDM0 IDM90 ENT0
hummocks	52	78	COR90 ENT0 ENT90	VAR0 IDM90 ASM90
fissures	62	78	ENT90 CON90 ASM90	COR0 COR90
landslide	54	90	COR0 COR90	IDM0 IDM90 ASM0
sheetflow	87	not significant	IDM90 ASM0 ENT0	not significant

Table 6.II - PCA results for selected features

The features with the highest votes in table 6.II are IDM90, ENT0, followed by COR0, COR90 and ENT90.

III To standardize or not? Davis (1973) presents an excellent explanation of the fundamentals of PCA. The results of PCA above are clearly biased towards those input feature vector channels with the highest variance, i.e. the entropy features. There is a common problem in multivariate analysis of whether to standardize a set of input channels or not. The process of standardization subtracts from every input channel pixel, the mean for the whole channel, and then divides it by the standard deviation. The transformed pixels would therefore have a mean of zero and standard deviation (and variance) of 1. Since principal components are purely the eigenvectors of the variance-covariance matrix of the dataset, the effect of standardization is to make the variance-covariance matrix equivalent to a simple correlation matrix. Davis (1973) states that standardization may have a great influence on the structure of the variance-covariance matrix and subsequently the PCA results, and that standardization is recommended if the set of input channels comprises a complete variety of measurement units, but not if the units are consistent across the variables. Since these data are all normalised, one may assume that they do not require standardizing. As stated above though, there is concern that the results of the PCA are too heavily biased towards the highly variable input channels, even though some of Davis' (1973) examples exhibit similar characteristics.

PCA has been applied to equalised input feature vector channels. Equalisation is similar to standardization as it balances out the channel variances, but not the means which may be important. For example, the unequalised variance of ENT at 0° is 4330, while the equalised value is 5155; the unequalised variance of SVAR at 0° is 243 and it is 4665 once equalised.

Figure 6.10 represents the eigenvector loadings for the 11 features at 0° and the selected statistics after the input channels have been equalised, for the whole image only. Clearly the results are vastly different, all features have approximate equal loadings for EC1, though the entropy measures are still prominent in EC2. Similarly for the selected features, similar magnitudes of principal components are seen in EC1, except for COR0, COR90, ENT0 and ENT90, which are the four main components of EC2. Even if the PCA is more statistically valid after equalisation, the results are not much use as they possess approximately equal biasing for all input channels. The selection of feature vector channels for classification was therefore chosen according to the guides of the unequalised PCA, and additional trial and error.

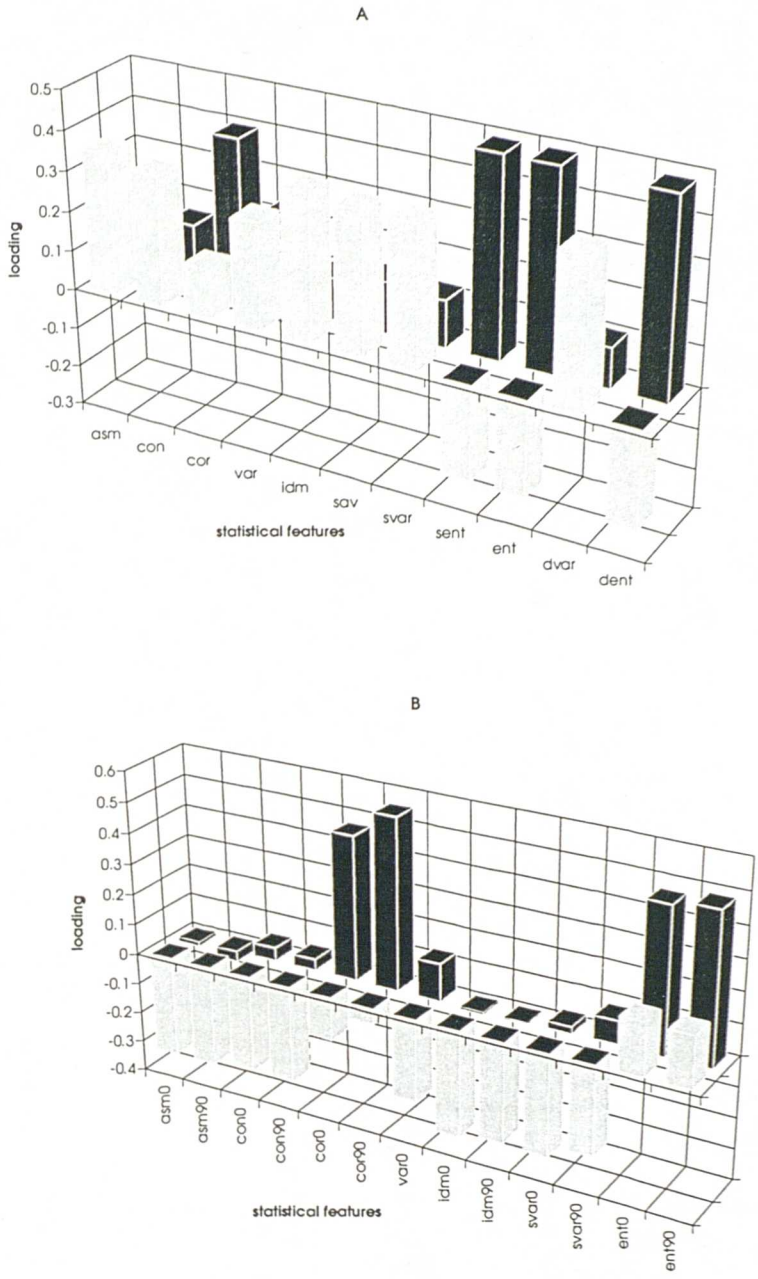


Figure 6.10 *Eigenvector loadings for equalised whole images from eigenchannels 1 (front) and 2 (rear). The eigenvector weightings are plotted for: (**a**) all statistical features at 0°; and (**b**) selected statistical features.*

6.5.6 Classification

Section 6.5.5b recommended the optimum GLCM parameters to be half resolution data (256×256 test image) with 12 m per pixel, 16 grey levels and a 17×17 pixel kernel size. The initial classification attempts used PCA to combine objectively all the features at each of the angles, followed by a combination of all of the data.

Then several experimental runs were performed through selections of various combinations of the feature images. These were based on the PCA results of section 6.5.5e and further refinements. The merits of whether to classify eigenchannels or just the raw feature vector channels with the largest principal components, whether to classify raw normalised or equalised feature images, or whether to perform maximum likelihood or minimum distance classification are explored. Finally, to verify the initial GLCM parameter assumptions, an investigation is presented into the effects of varying image resolution and kernel size.

6.5.6a Objective classification results The first attempts at classification were to take the 11 feature vector images for each angle, perform PCA on each group of 11 features, and classify the resulting two eigenchannels with the highest variances. Classification was also performed on all 8 eigenchannels to combine all the angular information. The results are presented in table 6.III:

Angle /°	separability index	classification accuracy /%
0	1.31	46.2
45	1.33	48.8
90	1.34	53.0
135	1.30	53.6
all	1.79	44.5

Table 6.III - Objective classification results

These results are unsatisfactory, with all separabilities in the category of 'poor separability', and the classification accuracies bearing this out with only about half of the bottom types in the test image being correctly identified.

6.5.6b Selective classification results Before describing the results, a few important points had to be addressed first:

I With or without PCA? Before performing these classifications, the question whether to classify raw feature vector channels or eigenchannels was first addressed. Selection F in table 6.IV was classified using both methods; the former yielded a separability of 1.84 and a classification accuracy of 65.6, while the latter produced respective values of 1.40 and 51.4. This was established by further tests, to prove that it is best to use the raw feature vectors, which is how table 6.IV was constructed.

II Use normalised or equalised feature vector images? Again, prior to performing classification on the selections of feature vector images, the problem of whether to classify normalised or equalised images was examined. Classification was

performed on both equalised and unequalised feature vector images of example K from table 6.IV; the former was classified with an accuracy of 64.7% and the latter 66.0%. Further examples confirmed that the raw normalised feature vector images were the best input channels for classification.

III Maximum Likelihood or Minimum Distance Classification? Several identical groups of features were classified using both the maximum likelihood and minimum distance classifiers. The maximum likelihood method consistently yielded the best results. For example, run F produced a classification accuracy of 65.6% for maximum likelihood, but only 52.9% with the minimum distance method.

IV Effect of post-classification filtering Figure 6.11c displays the best classification result of 66.0% obtained for iteration K. Small fluctuations of classes can be observed due to slight inhomogeneities, probably due to both the data and technique. A simple median filter (5×5 kernel) can remove these to improve the classification accuracy to 69.8% (figure 6.11d). Similar grades of improvement were gained with other iterations, so the technique is recommended, though not yet applied to these experimental results.

ID	asm 0	asm 90	con 0	con 90	cor 0	cor 90	var 0	idm 0	idm 90	svar 0	svar 90	ent 0	ent 90	sep	class acc
A								✓	✓			✓	✓	1.57	53.7
B						✓	✓		✓			✓		1.55	50.6
C				✓	✓	✓							✓	1.65	56.7
D					✓	✓		✓	✓					1.54	54.7
E	✓								✓			✓		1.44	49.5
F			✓	✓			✓					✓		1.84	65.6
G			✓	✓			✓			✓	✓	✓		1.93	58.2
H			✓	✓	✓	✓	✓					✓		1.90	54.3
I			✓	✓	✓	✓		✓	✓					1.89	52.7
J			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		1.97	35.8
K			✓	✓				✓	✓					1.78	66.0
L	✓	✓	✓	✓	✓	✓		✓	✓			✓	✓	1.95	34.1
M			✓	✓			✓	✓	✓			✓		1.90	58.3
N							✓	✓	✓			✓		1.65	57.6
O					✓	✓	✓					✓		1.64	53.3
P					✓	✓		✓	✓			✓		1.54	54.7
Q					✓	✓						✓		1.46	51.8
R	✓				✓	✓			✓			✓	✓	1.77	52.3
S					✓	✓						✓	✓	1.61	52.3
T					✓	✓		✓	✓			✓	✓	1.75	49.6
U		✓		✓	✓	✓							✓	1.73	54.9
V					✓	✓						✓	✓	1.72	54.1

Table 6.IV - Selective feature classification results

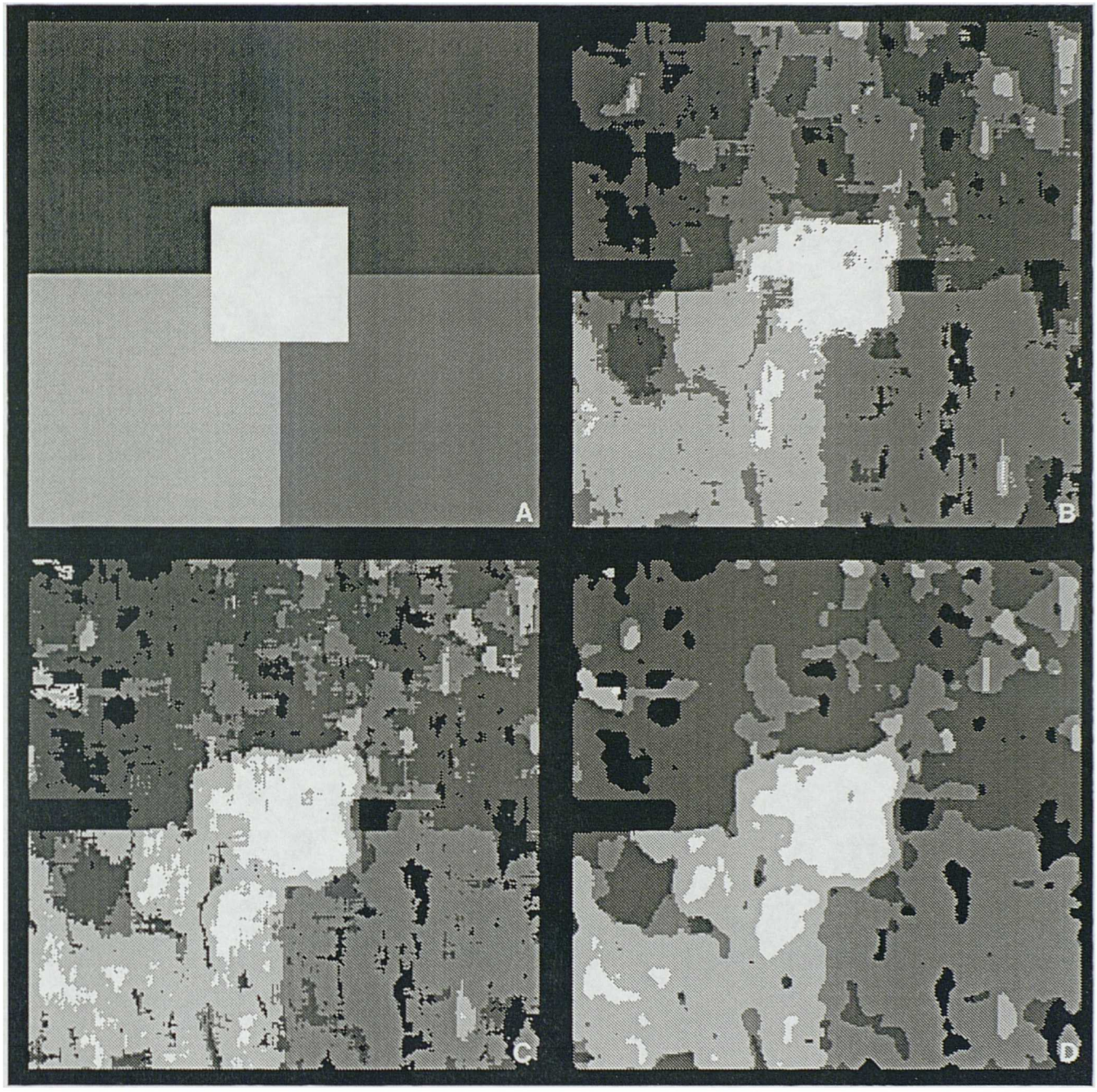


Figure 6.11 *Classification results: (a) ground truth image; (b) classification F (table 6.IV); (c) classification K (table 6.IV); and (d) is the result of applying a 5×5 pixel median filter to (c).*

Table 6.IV presents the classification results, both in terms of class separability and classification accuracy, of selected feature vector images. The ticks denote which features were used in each classification attempt.

Twenty two iterations of the features are presented, with varying degrees of success. Generally, higher class accuracies were obtained when classifying a smaller number of channels (see K and L), though larger separability indices were attained with increasing numbers of input channels. This suggests that the separability guide to classification accuracy may not be reliable.

The first five classification attempts, A-E, respectively selected the features with the highest principal components for the whole region, hummocks, fissures, landslide and sheetflow training sites (table 6.II). Further combinations of these features and many others were attempted. For example, the features that Reed (1987) selected, just at 0 and 90° rather than the four angles that he used, only produced an accuracy of 34.1% test L). The best classification accuracies obtained were 65.6 and 66.0% for experiments F and K; the corresponding separabilities were 1.84 and 1.78 (not the highest). These are respectively displayed in figure 6.11b and c, along with the corresponding ground truth image (6.11a).

6.5.6c Verification of assumed GLCM parameters All classification performed so far has assumed a half resolution (12 m per pixel) input image with 16 grey levels and a kernel size of 17×17 pixels. The suitability of these parameters is tested below.

I Effect of varying resolution This is equivalent to Reed's (1987) testing of the wavelength sensitivity of the GLCM texture analysis technique. The two most successful runs of section 6.5.6b (K & F) were tested at varying image resolutions: full, half and a quarter, plus intermediate values. Figure 6.12a displays how the mean feature vectors for the whole image vary with pixel size. All except the entropy measures become more spread out at lower resolutions, possible suggesting better classification potential. Figures 6.12b and c respectively present the separability and classification accuracies of feature selections K and F. The separability indices increase with pixel size, indicating that the classification statistics generated for each training area become more distinctive as the TOBI data is decimated. However, the response of the classification accuracies with resolution was different for the two feature selections. The accuracy of K increases with pixel size, while that of F

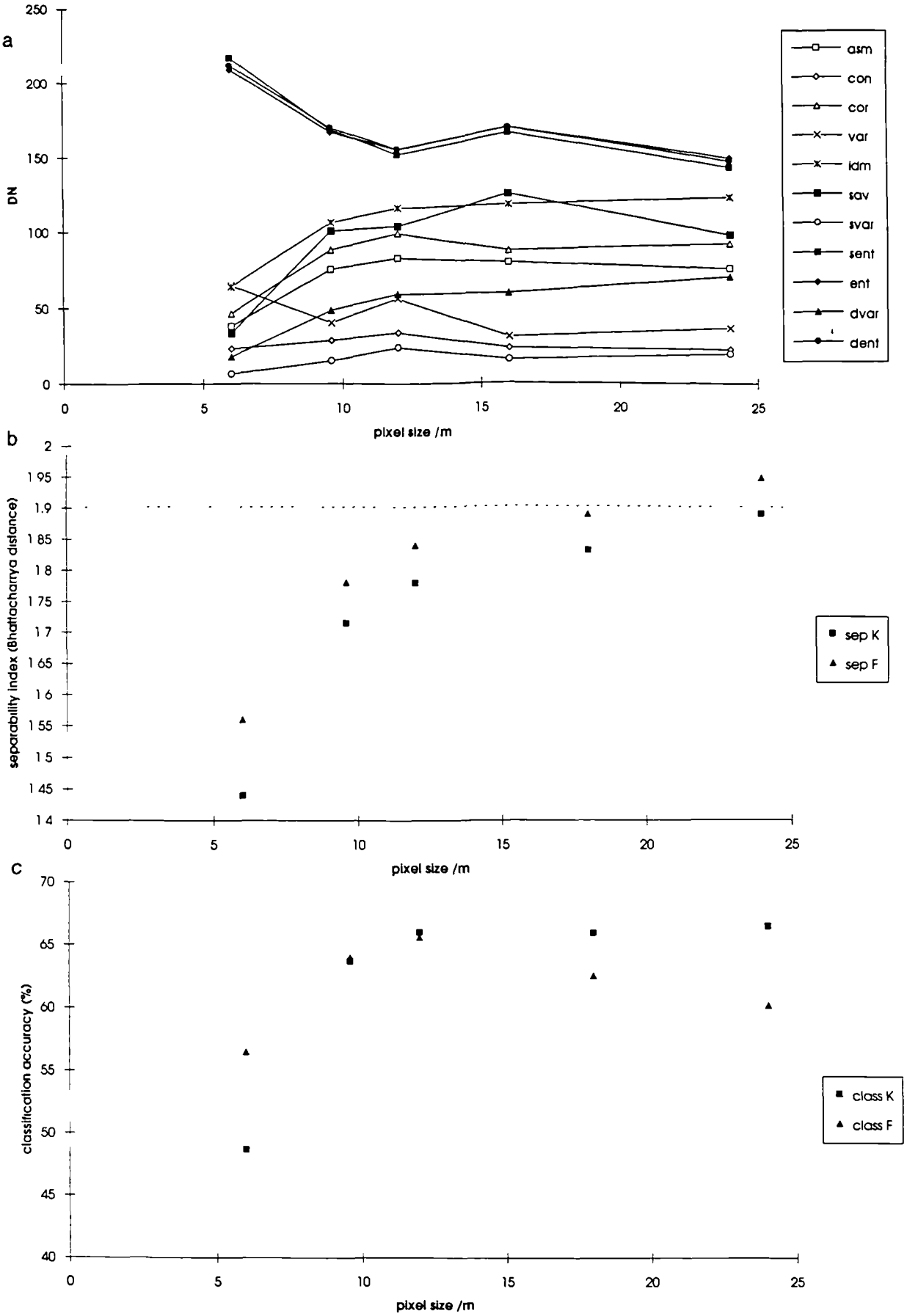


Figure 6.12 *Statistical variation with image resolution: (a) DN plotted against image pixel size for 11 curves corresponding with each statistical feature vector (key); (b) separability index plotted against pixel size for F and K statistical selections; and (c) classification accuracy plotted against pixel size for F and K statistical selections.*

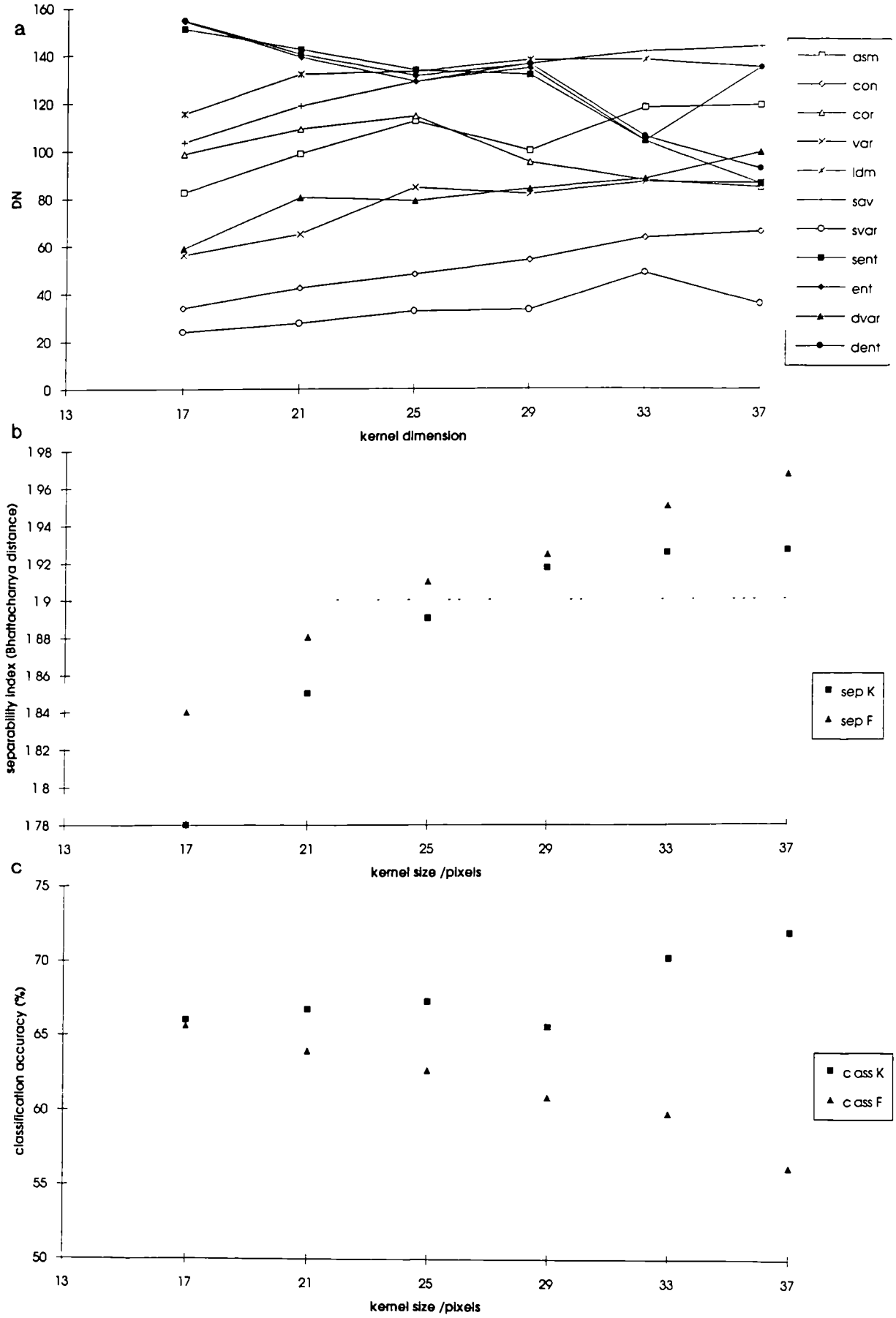


Figure 6.13 *Statistical variation with kernel size: (a) DN plotted against kernel dimension (all square) for 11 curves corresponding with each statistical feature vector (key); (b) separability index plotted against kernel dimension for F and K statistical selections; and (c) classification accuracy plotted against kernel dimension for F and K statistical selections.*

increases to a peak at a pixel size of 12 m, then decreases again. The assumption of using a half resolution, 12 m per pixel test TOBI image was therefore valid.

II Effect of varying kernel size Reed (1987) investigated this, concluding that it was sensitive to the size of feature being examined. Section 6.5.5a states that the maximum 'wavelength' of features is 180 m, or 15 pixels at 12 m per pixel resolution, and a 17×17 kernel is required to fulfill the number of grey levels criterion. Testing is presented here on square kernels of 17 to 37 pixels per side. Figure 6.13a displays the 11 mean feature vectors plotted against kernel size; the responses are rather messy, but the means generally increase with kernel size, except for the entropy features, but exhibit no greater distinction at any particular kernel size. Figure 6.13b shows that the separability indices increase with kernel size, and figure 6.13c presents the class accuracies for feature selections K and F. Increased kernel size (*KS*) improves the classification accuracies for K (to over 70%), but degrades them for F (to less than 60%). Additionally, since information of width $(KS - 1) / 2$ is lost around the texture and image boundaries, the 17×17 pixel kernel can be confirmed as the best size to use.

6.5.7 Conclusions

The GLCM texture analysis technique has been tested extensively on a composite image of typical MAR TOBI 'bottom types', i.e. both different volcanic morphologies probably possessing similar acoustic characteristics and also materials with different acoustic properties. All of Haralick *et al.*'s (1973) statistical features were computed at four angles, and principal component analysis and iterative testing attempted to deduce the optimum combination of these features for classification. The highest classification accuracy achieved for this test image was 66.0%.

6.6. THE FRACTAL METHOD

6.6.1 Introduction

Many naturally occurring phenomena are extremely complex, and therefore forming descriptions using traditional geometric methods becomes inadequate. In view of this, Mandelbrot (1983) introduced the concept of fractal geometry and with this was able to synthesise many images of natural scenes, such as mountains, clouds and planets. Linnett *et al.* (1993) used fractal geometry to produce synthetic images of the seabed. Fractal methods have also been used to describe topographic

characteristics. Burrough (1981) introduced some fractal dimensions (see next section for definitions) computed for various landscapes. More specifically, Malinverno (1989) and Goff and Jordan (1988) have utilised various fractal-based methods to quantify seafloor bathymetric variations.

Fractals have been applied for texture analysis purposes by Keller *et al.* (1987), who used fractal geometry to segment natural silhouetted scenes. Pentland (1984) measured the power spectrum over small image areas and used the fractal dimension as an image descriptor. Peleg *et al.* (1984) measured the grey level surface area (this is equivalent to the length of a line in two dimensions) of an image at several resolutions, finding that the area increased at finer resolutions. This increase in area with resolution enables the fractal properties of the image to be determined, which can then be used for texture analysis. Linnett *et al.* (1991; 1993) developed a texture classification scheme based on fractal principles, which they were able to apply in real time to very high resolution (resolution of a few centimetres) sidescan sonar data. The principle behind this analysis is simple and the classification accuracy was excellent. The theoretical outline for Linnett *et al.*'s (1991; 1993) algorithms follows.

6.6.2 Theory

Mandelbrot (1983) found that the length of many curves could be approximated by

$$L(\lambda) = k\lambda^{1-D}$$

where $L(\lambda)$ is the length of the curve measured using a unit length of λ, k is a constant and D is the fractal dimension of the curve. One way of measuring this dimension is to locate points at a distance λ either side of the curve, forming a strip 2λ wide. The area of this strip divided by 2λ provides an approximation to its length. As the magnitude of λ decreases, the length of the curve increases. Extending this to a two dimensional image, an upper level at distance λ above the surface and a lower level at distance λ below can be determined. The grey level of the image at pixel location i,j is $g(i,j)$, and the upper (u) and lower (l) images (blankets) are initially defined to be equal to this. For any point i,j, the blankets can then be computed as

upper blanket: $$u_\varepsilon = \max\left(\left[u_{\varepsilon-1}(i,j)+\lambda\right],\left[u_{\varepsilon-1}(m,n)\right]\right) \qquad (6.1)$$

lower blanket: $$l_\varepsilon = \min\left(\left[l_{\varepsilon-1}(i,j)-\lambda\right],\left[l_{\varepsilon-1}(m,n)\right]\right) \qquad (6.2)$$

where *(m,n)* is the position of a pixel surrounding *(i,j)* in a 3×3 kernel, 4-connected neighbourhood , λ is the level by which the surface is raised, and ε is the iteration number. The volume enclosed between the two layers is

$$V_\varepsilon = \sum_{ij} (u_\varepsilon(i,j) - l_\varepsilon(i,j))$$

Mandelbrot (1983) related the area to the fractal dimension as well, in the same manner as the length

$$A(\lambda) = k\lambda^{2-D}$$

The exponent is simply raised by one.

Linnett *et al.* (1993) deduced that it would not be useful to use successive iterates of the unit length measurement for statistical classification purposes. Instead, the phenomenon of directionality is adopted, where equations (6.1) and (6.2) can be used to define the upper and lower blankets (*u* and *l*) in different directions. Linnett *et al.* (1991) used seven different directions based on a 3×3 pixel grid:

The original image is therefore transformed into seven feature images and a pixel at any position has associated with it a feature vector $X=\{x_1,x_2,....,x_7\}$. This information is used to assign one texture to the pixel under consideration, given its 7 dimensional feature vector.

Classification is based on a supervised technique, where information about individual textures, from user-specified training areas, is known before a

classification result can be achieved. A maximum likelihood discriminant analysis is performed.

6.6.3 Software

Software, written in the C programming language, by L. Linnett of the Department of Computing and Electrical Engineering, Heriot-Watt University, Edinburgh was acquired, and modifications were made to enable a common PGM format interchange (Poskanzer, 1991). This allowed the simple display and combinations of original data and results. The procedure for the application of these programs is presented in figure 6.14.

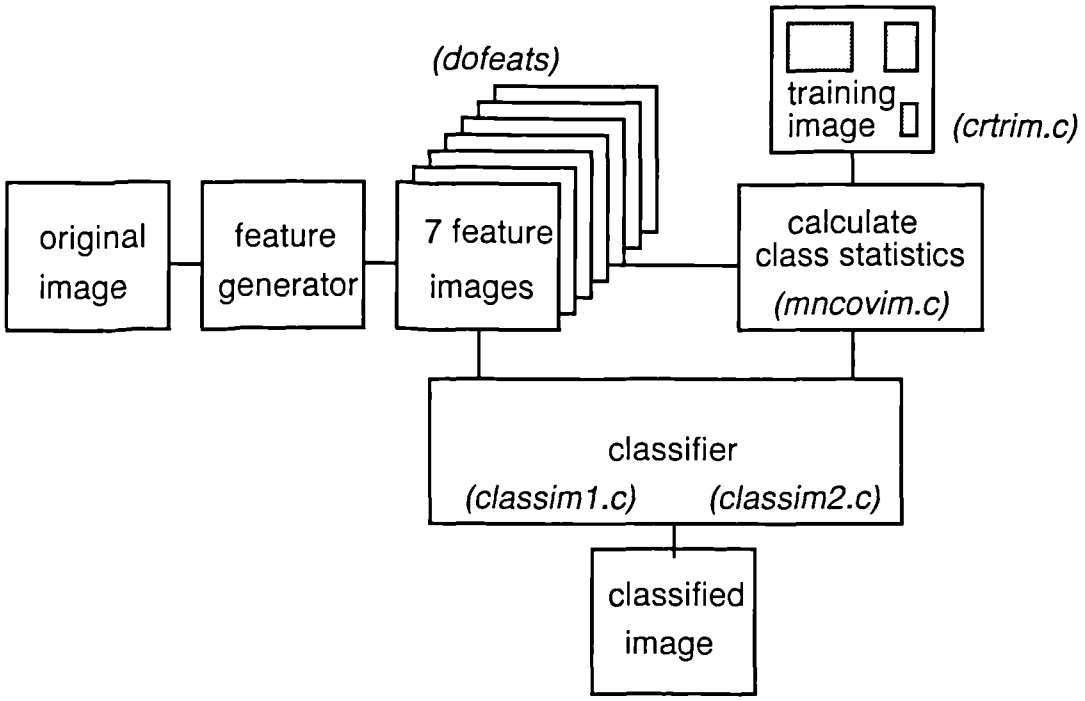


Figure 6.14 Flowchart of fractal texture analysis procedure, based on Linnett et al.'s (1993) software (program or shell names are in italics).

A 3×3 pixel neighbourhood is always assumed for each pixel, and the 7 directional feature images are constructed assuming a level (λ) of 1, though λ may be altered. The upper and lower blankets are determined as in equations (6.1) and (6.2), the difference between them is calculated, then an averaging filter is applied (of default dimension 29 pixels, though this may be altered) to produce the feature image.

6.6.4 Preliminary analysis

Initial testing of the fractal texture analysis technique was applied to the same composite bottom type image of TOBI high resolution sidescan sonar data from the Mid-Atlantic Ridge as used for the GLCM texture analysis (section 6.5, figure 6.1) (figure 6.18a displays the image again). The primary examination of the fractal texture analysis technique was applied to the full resolution test image, following Linnett *et al.*'s (1991) default parameter settings.

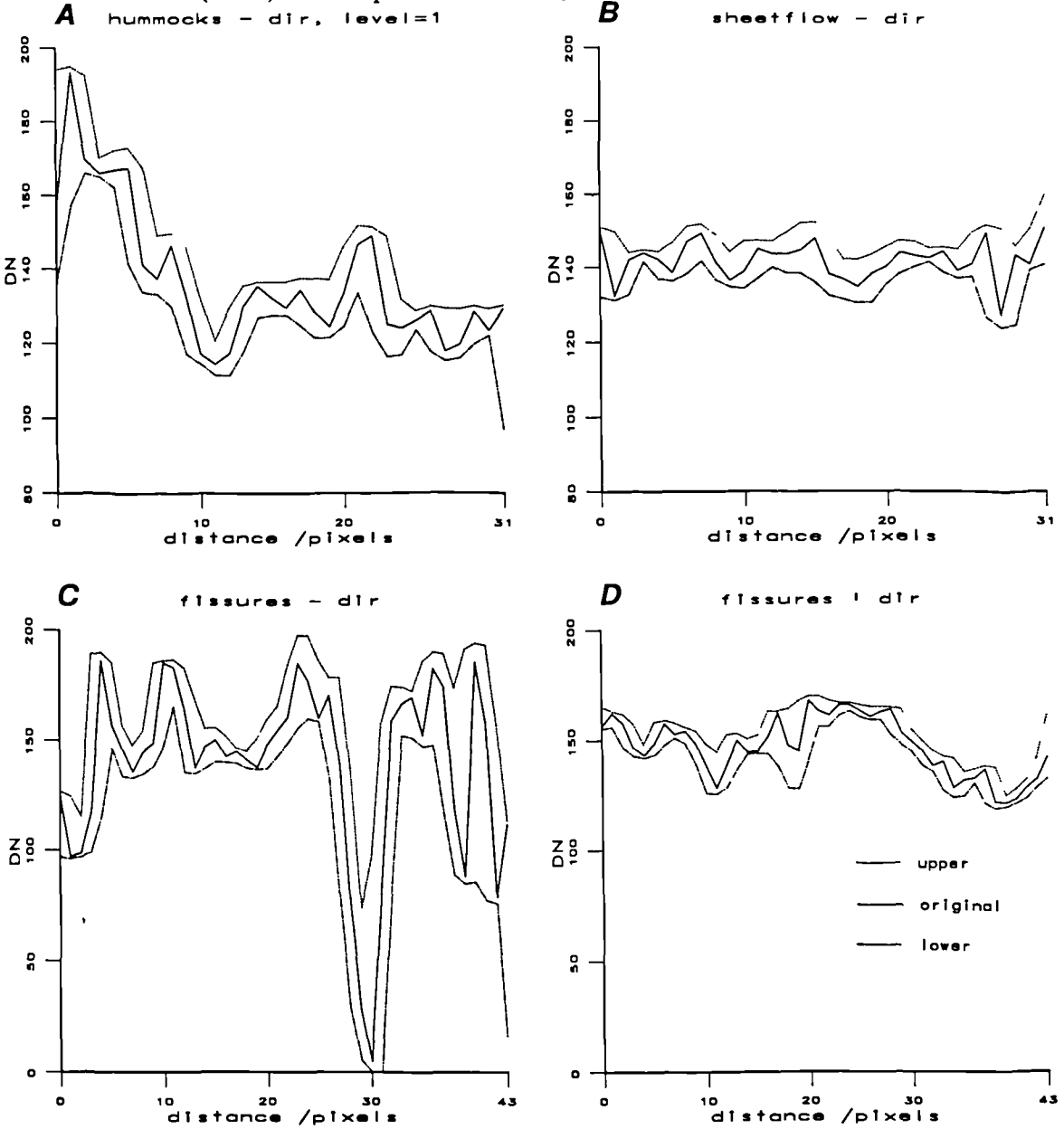


Figure 6.15 Examples of blanket plots for profiles over: (a) hummocks, 90° (horizontal direction - feature #1); (b) sheetflow, 90°; (c) fissures, 90°; and (d) fissures, 0° (vertical direction - feature #3). The solid line represents the actual grey level variations along the profile, whilst above are the upper, and below the lower blankets. All blankets are computed for level (λ) = 1. The profiles are taken in the same directions as the features.

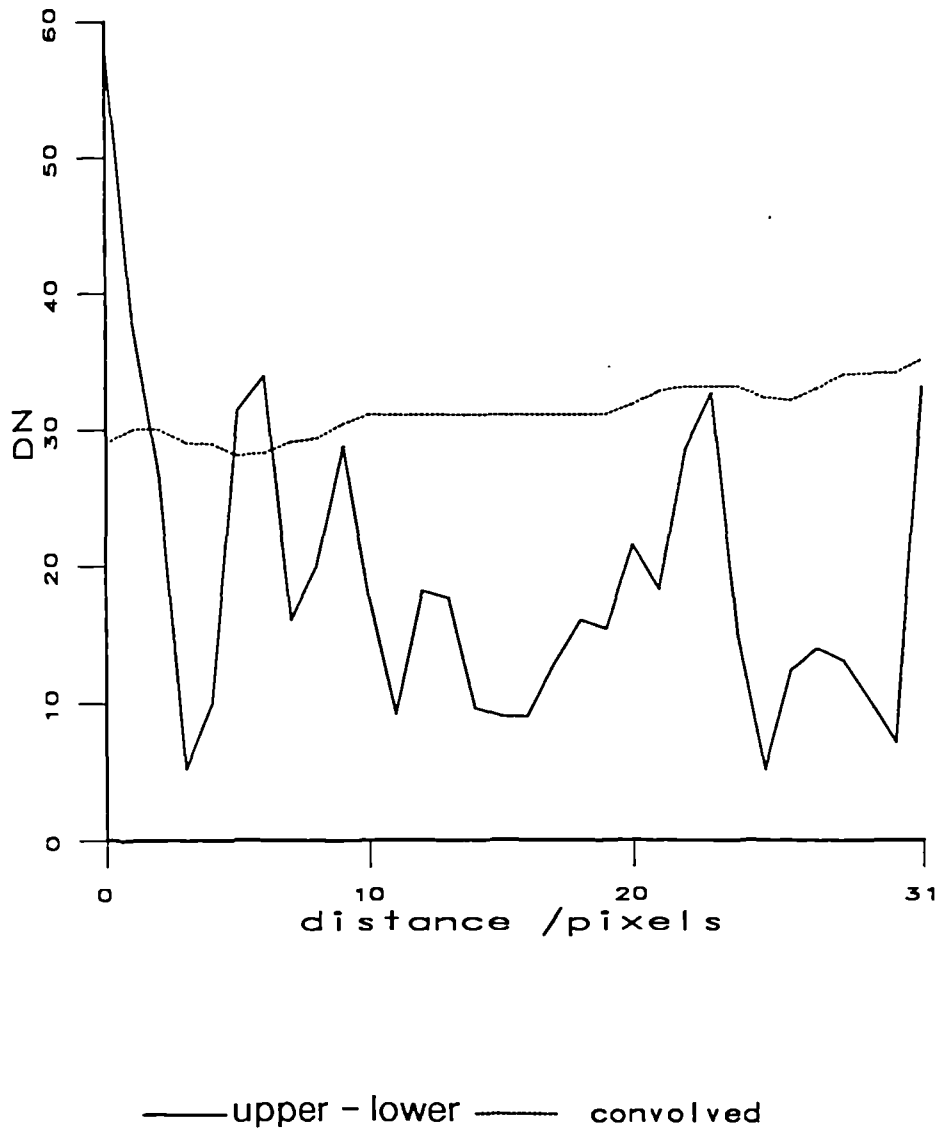


Figure 6.16 *The difference between the upper and lower blankets for figure 6.15a (hummocks, 90°), and the averaged result (convolved with a kernel size of 29 pixels).*

Figure 6.15a displays profiles taken across some of the test image bottom types, and their corresponding upper and lower blankets. Figure 6.15a presents the effect of determining the two blankets over a single profile within the hummocky terrain, the feature image used was the 90° direction (feature #1) and the profile was also taken in this orientation. One can identify where the grey tone fluctuations are smoothed out by the blankets, increasing the area between the two levels. Figure 6.15b displays the blankets for a profile taken through the sheetflow region (same directions as

figure 6.15a). The grey scale fluctuations are much smaller than for the hummocks, so the blankets were able to follow the grey tones more closely, thus reducing the area between the blankets. Figures 6.15c and d show the directional effects on the blanket level determinations caused by the strongly aligned fissures bottom type. Figure 6.15c represents the blankets and profiles computed orthogonal to the fissures (feature #1), while figure 6.15d displays those parallel to the fissures (feature #3). The orthogonal case exhibits strongly fluctuating grey levels of some 180 DN, while the parallel range is around 40 DN. The area between the blankets is clearly much smaller for the parallel compared with the orthogonal case, thereby identifying the directional characteristic of the fissures.

Once the upper and lower blankets have been generated, the difference between them may be determined and then convolved with the averaging filter. Figure 6.16 presents the respective differences before and after convolution. Effectively, the feature images measure the 'roughness' of the seafloor texture in various directions.

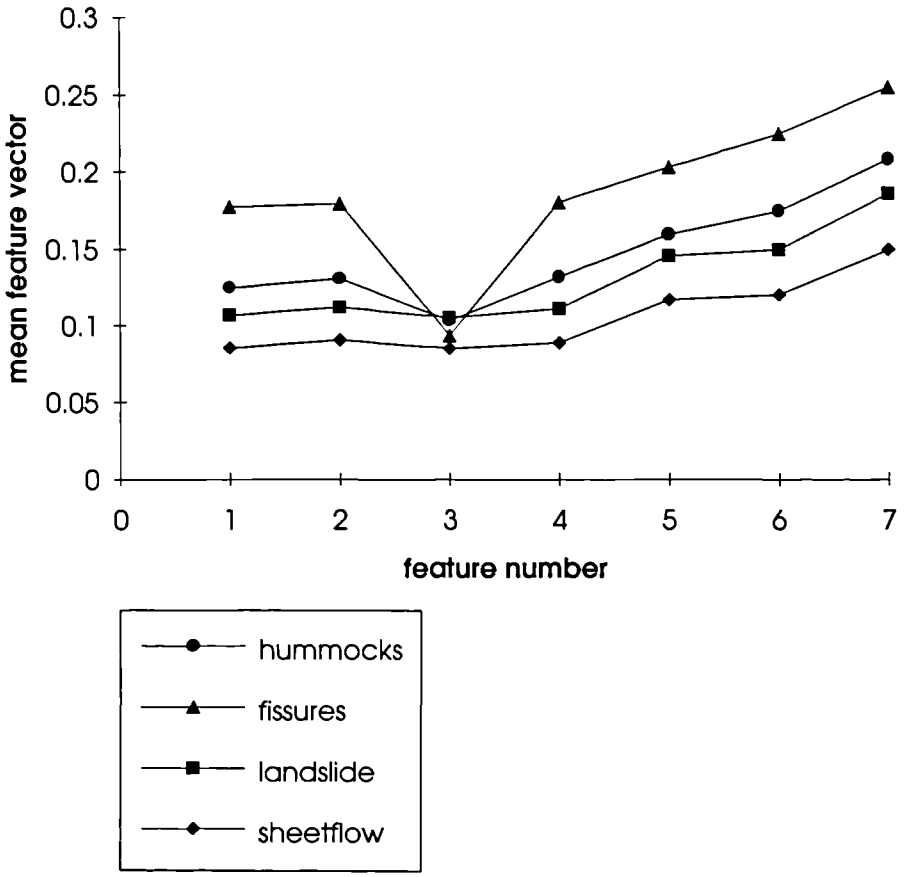


Figure 6.17 Feature variation with bottom type: mean feature vector plotted against feature number, for the four bottom type training areas.

Figure 6.17 displays the mean feature vectors computed by the program *mncovim* for the training area of the test image. These training areas are the same as those used for the GLCM analysis and are displayed in figure 6.18b. All mean feature vectors have been normalised to a value of one. Figure 6.17 clearly shows respectively decreasing mean vector values for the fissures, hummocks, landslide and sheetflow bottom types; indicating lower degrees of texture 'roughness'. Features #5-7 are composite directions and have higher values, since the blanket definitions are decided over a number of pixel neighbours, thus smoothing the blankets further. Feature #3 possesses a marked decrease in mean feature vector for the fissures, and to a lesser extent the hummocks. The feature is parallel to the alignment of the fissures, and like the blanket plot of figure 6.15d, describes a 'smoother' texture in this orientation. The slight decrease for the hummocks may be due to the fact that since this direction is also orthogonal to the angle of insonification, so no shadowing occurs. These distinctive feature vector variations between bottom types suggest a good potential for classifying the seafloor geological units.

The full resolution (pixel size of 6 m) was classified, with results presented in figure 6.18. Figure 6.18a and b respectively display the original image and with superimposed training areas locations, figure 6.18c shows the classification results, while 6.18d combines the boundaries of these classes with the original data. A consistent grey level scale is utilised for all subsequent displays of classification results; figure 6.19 defines the key. The fissures are classified well; most of the landslide is picked out successfully, except for the right hand portion which could be seen to be slightly in shadow on the original swath from which it was taken, and was interpreted as sheetflow. The classification of the hummocks regions was extremely patchy. From a quantitative perspective, table 6.V presents the classification accuracies determined through comparison with a ground truth class image comprising the original cut-out bottom type regions, both for the image as a whole and for the individual bottom units.

Region	Classification accuracy /%
whole	71.8
hummocks	62.8
fissures	86.4
landslide	89.7
sheetflow	84.9

Table 6.V - full resolution test image classification accuracies

These results are superior to the best achieved with the GLCM method (66.0%).

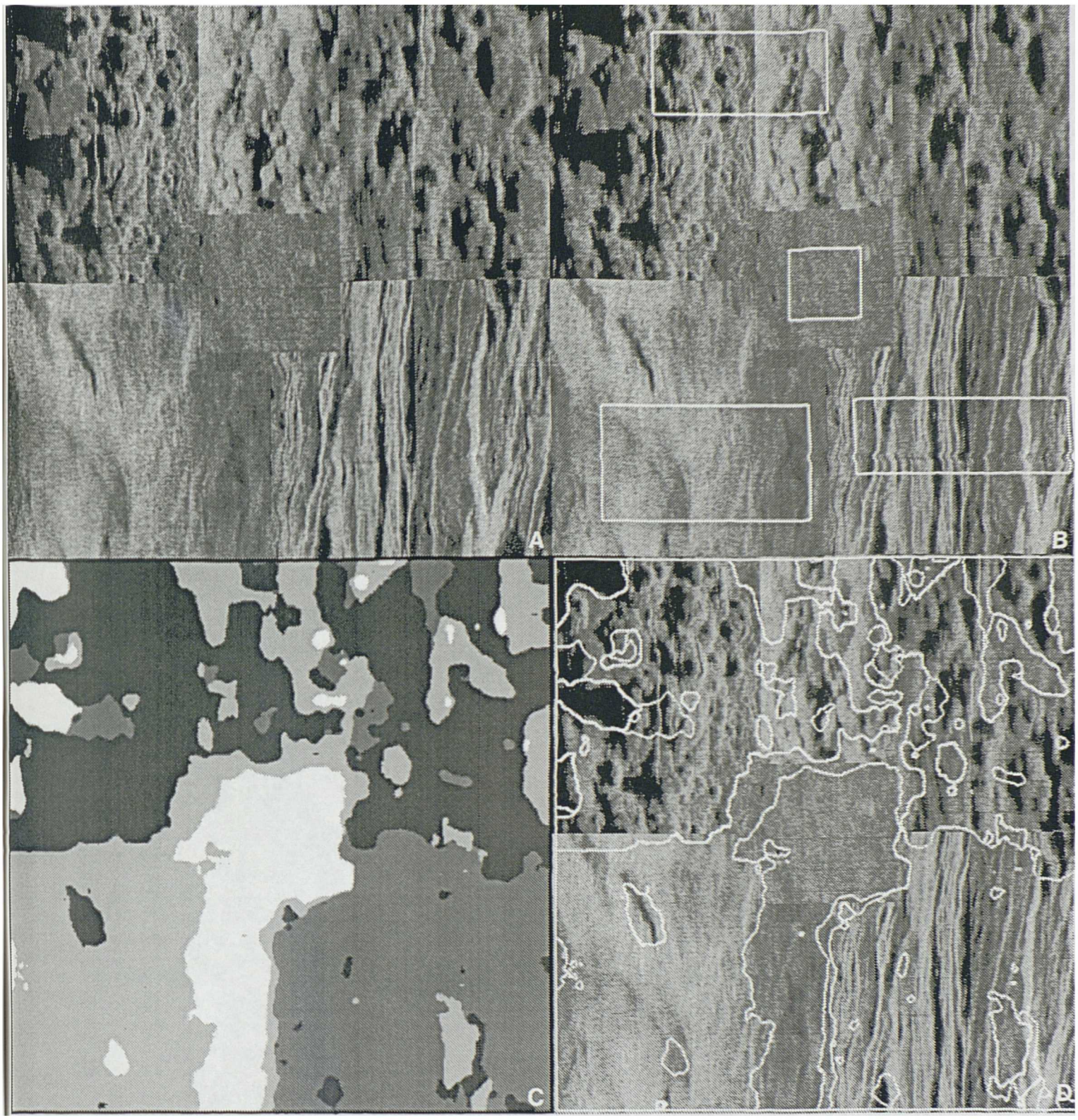


Figure 6.18 *Fractal texture analysis results for the full resolution data (pixel 6×6 m): (a) original image; (b) original image with training areas superimposed; (c) classification results (refer to figure 6.19 key); and (d) classification results superimposed over original image.*

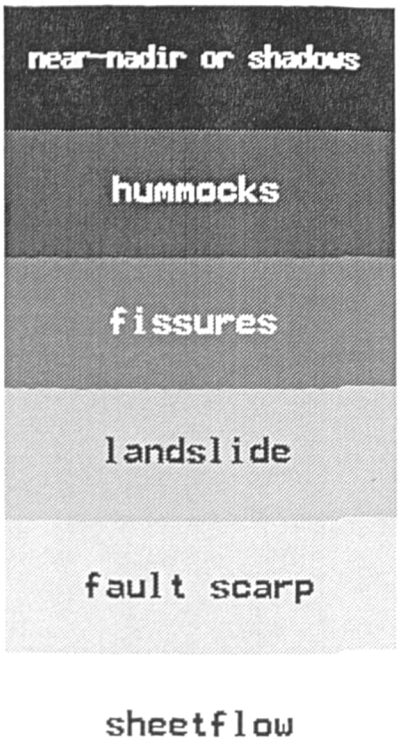


Figure 6.19 *Grey scale guide for all figures (parts (c)) of fractal texture analysis results.*

6.6.4a Effect of varying resolution The fractal texture analysis was also applied to half (12 m per pixel) and quarter (24 m per pixel) resolution test images (the kernel sizes were retained as 17×17 pixels, i.e. respectively 204 and 408 m per side). If the seafloor is truly fractal, one would expect it to be self-similar at all scales, so varying the resolution should not have any effect.

Figure 6.20 displays the results for the half resolution test image, while figure 6.21 presents the outcome for quarter resolution data. The arrangement of parts a, b, c and d is consistent with figure 6.18. The classification accuracies are presented in table 6.VI.

Region	Classification accuracy /%	
	half resolution	quarter resolution
whole	81.5	81.9
hummocks	82.5	93.8
fissures	95.5	84.9
landslide	90.0	92.6
sheetflow	73.9	57.9

Table 6.VI - half and quarter resolution test image classification accuracies

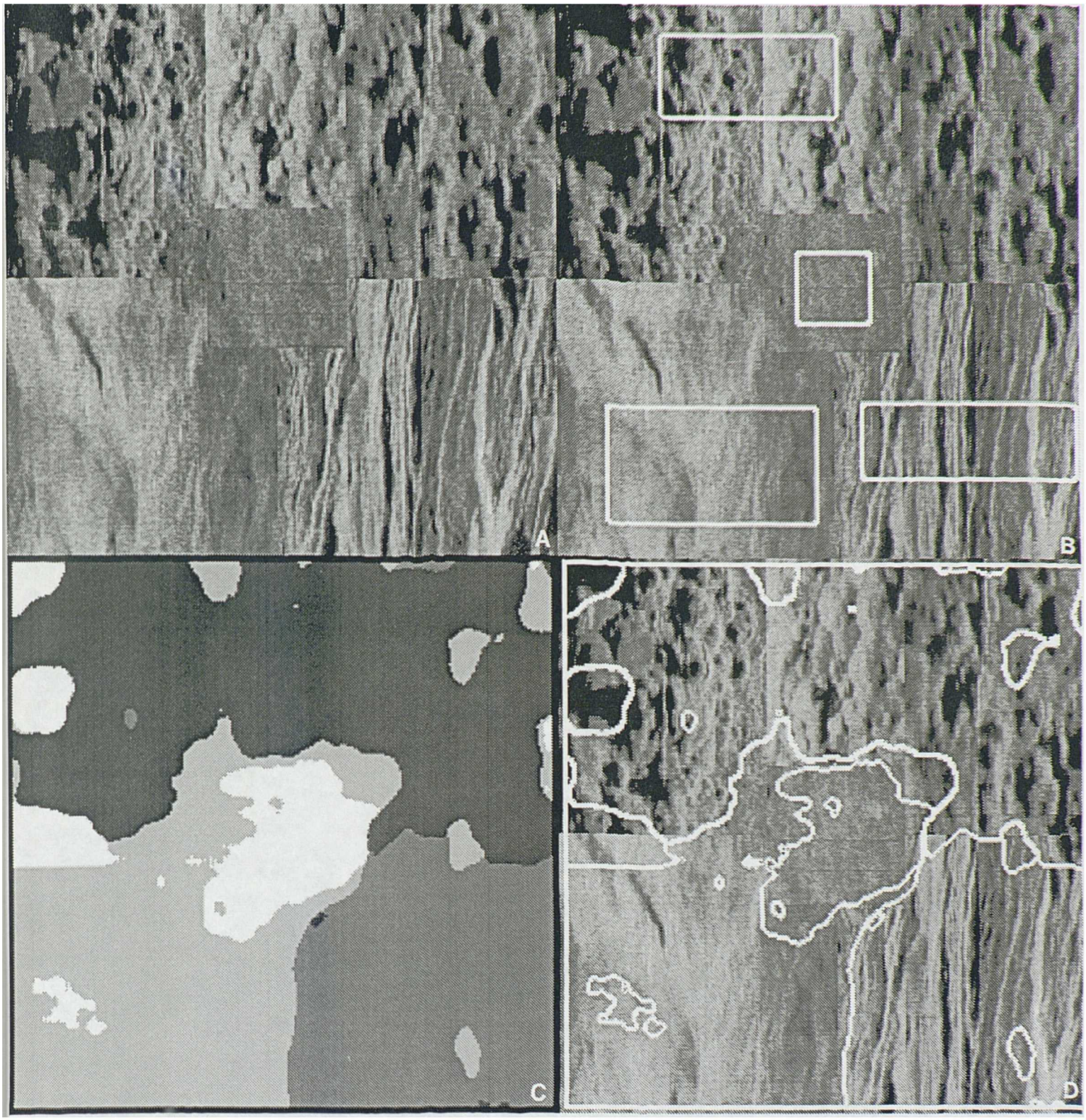


Figure 6.20 *Fractal texture analysis results for the half resolution data (pixel 12×12 m): (a) original image; (b) original image with training areas superimposed; (c) classification results (refer to figure 6.19 key); and (d) classification results superimposed over original image.*

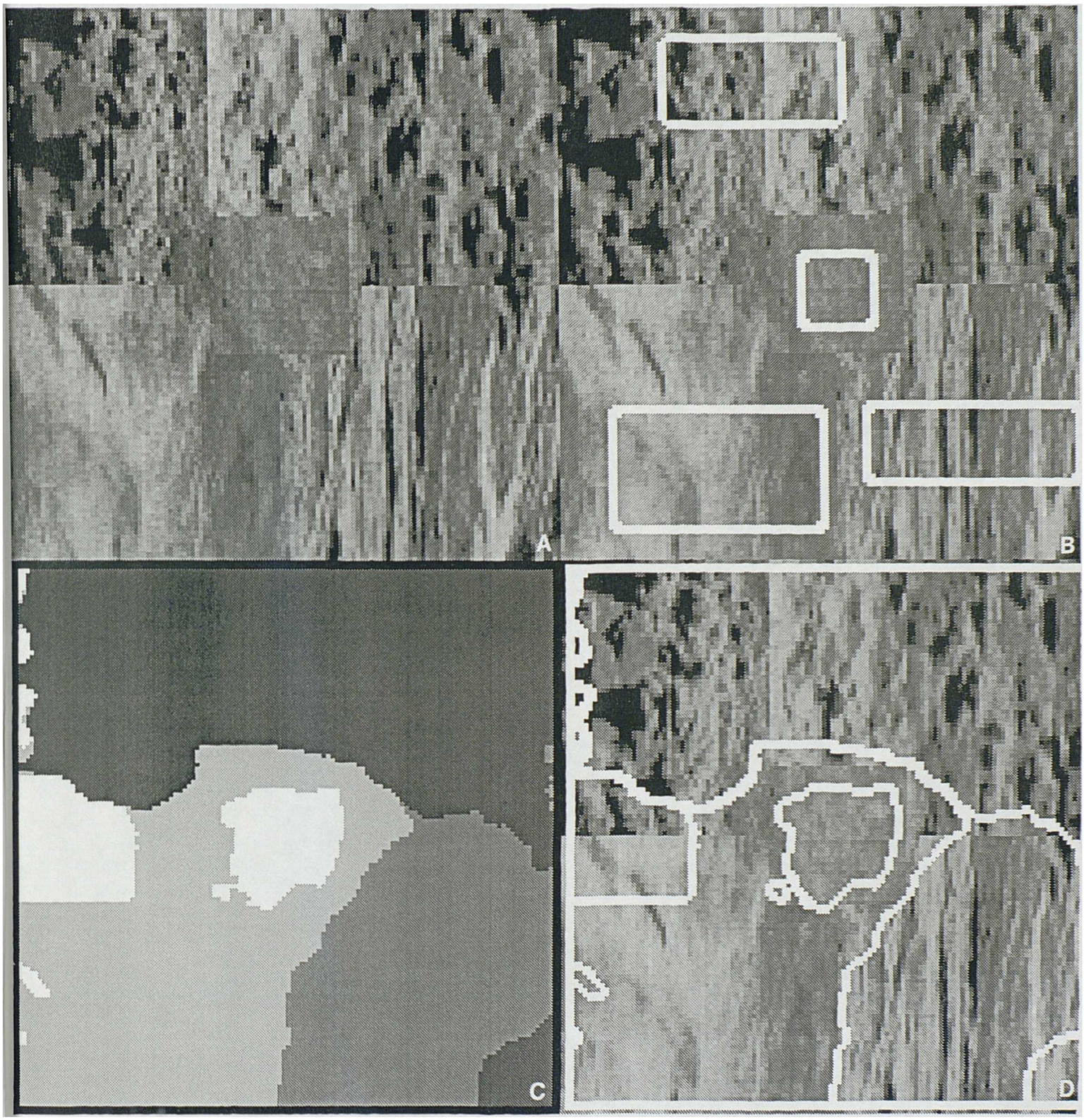


Figure 6.21 *Fractal texture analysis results for the quarter resolution data (pixel 24×24 m): (a) original image; (b) original image with training areas superimposed; (c) classification results (refer to figure 6.19 key); and (d) classification results superimposed over original image.*

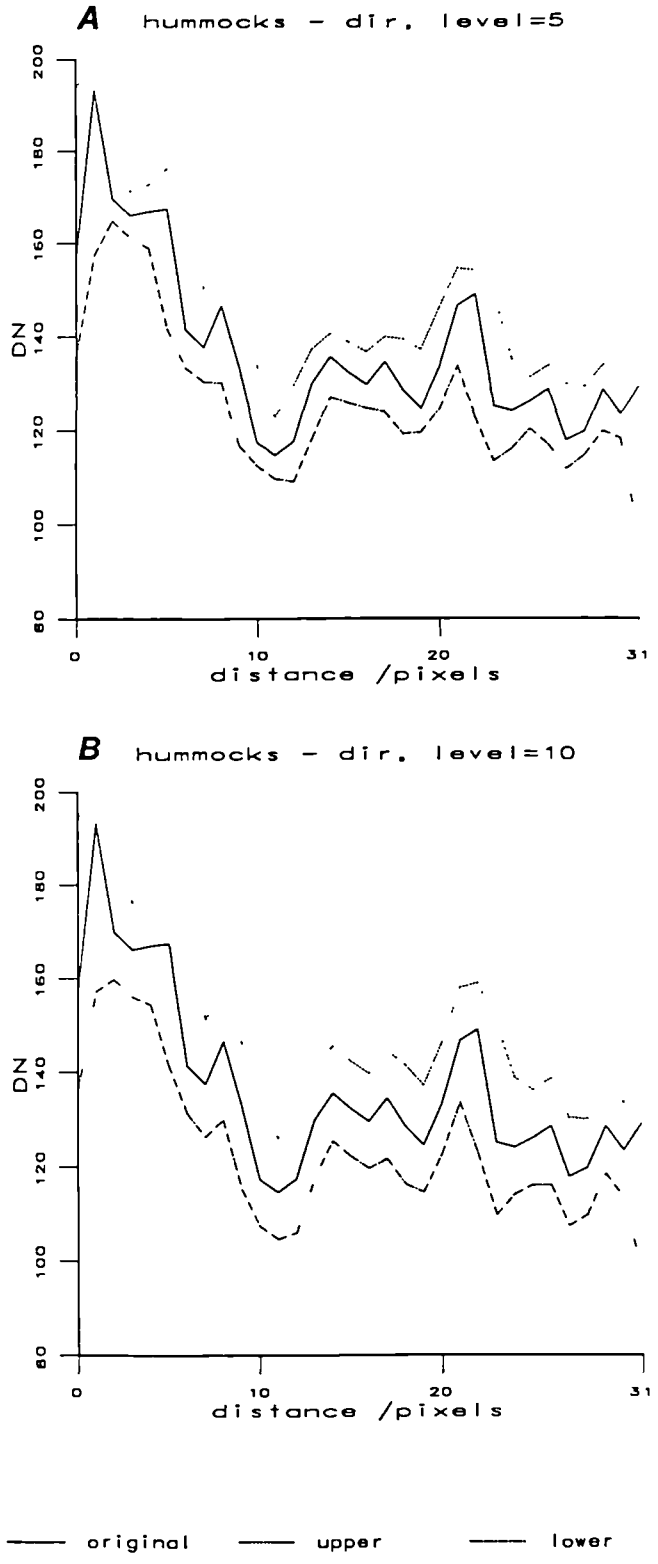


Figure 6.22 *Hummocks 90° blanket plots to demonstrate effect of varying λ (level): (a) λ = 5; (b) λ = 10.*

The overall accuracies increase with decreasing resolution. The distinction of the sheetflow area declines rapidly with lower resolutions, while the classification accuracy of the other units improves. From figures 6.20 and 6.21, it is evident that by the stage of reducing the TOBI resolution by a quarter, the representation of the bottom types is becoming degraded and the boundary definitions are becoming less distinctive. The general results for the half resolution data are better than for full and quarter, with an overall accuracy improvement of some 10%, so it was decided to use this scale for future analyses. Incidentally, for comparison with section 6.5.5c, the entire fractal feature image generation and classification processes took approximately two minutes to complete on a Sun SPARC 10 processor.

6.6.4b Effect of varying λ level The level λ is defined in equations (6.1) and (6.2), as the amount by which a pixel under consideration is raised with respect to its neighbours for the determination of the upper and lower blanket levels. Figure 6.22a and b display the effect on the upper and lower blanket plots of varying λ, with respective values of 5 and 10, for the hummocks profile of figure 6.15a (λ = 1 there). Increasing λ to 5 has little effect, though exaggerates some of the small scale grey level variations, λ = 10 pronounces these further.

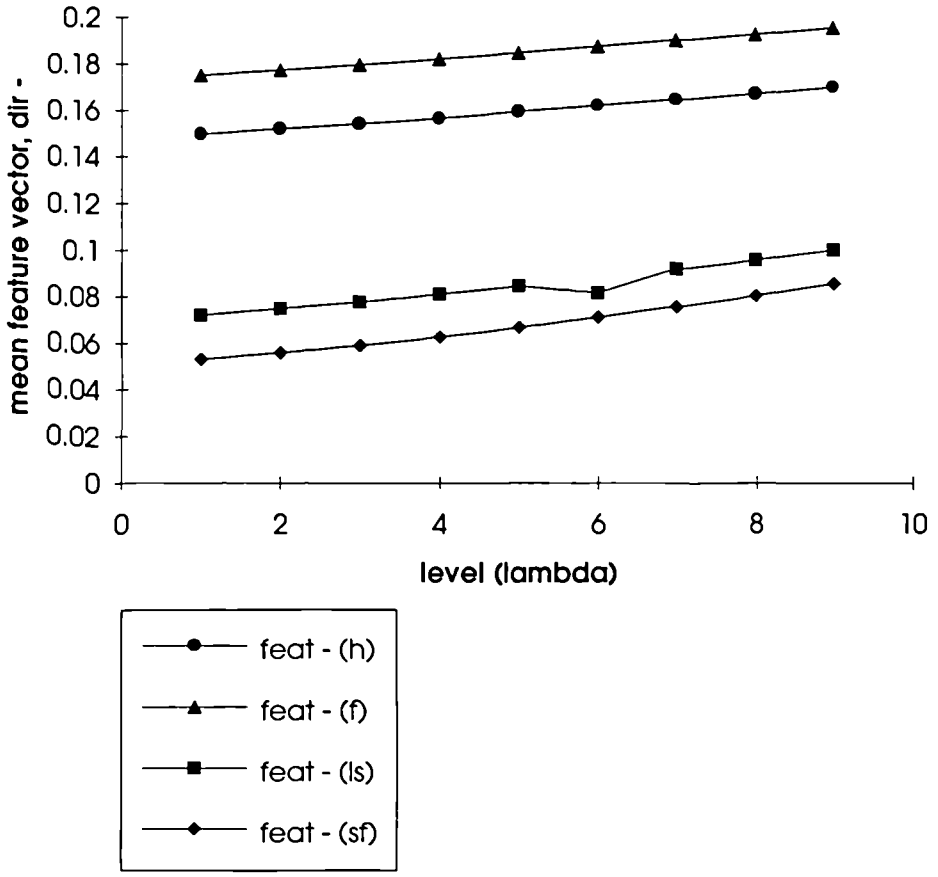


Figure 6.23 Mean feature vector plotted against level (λ), for the four bottom types (hummocks, fissures, landslide and sheetflow).

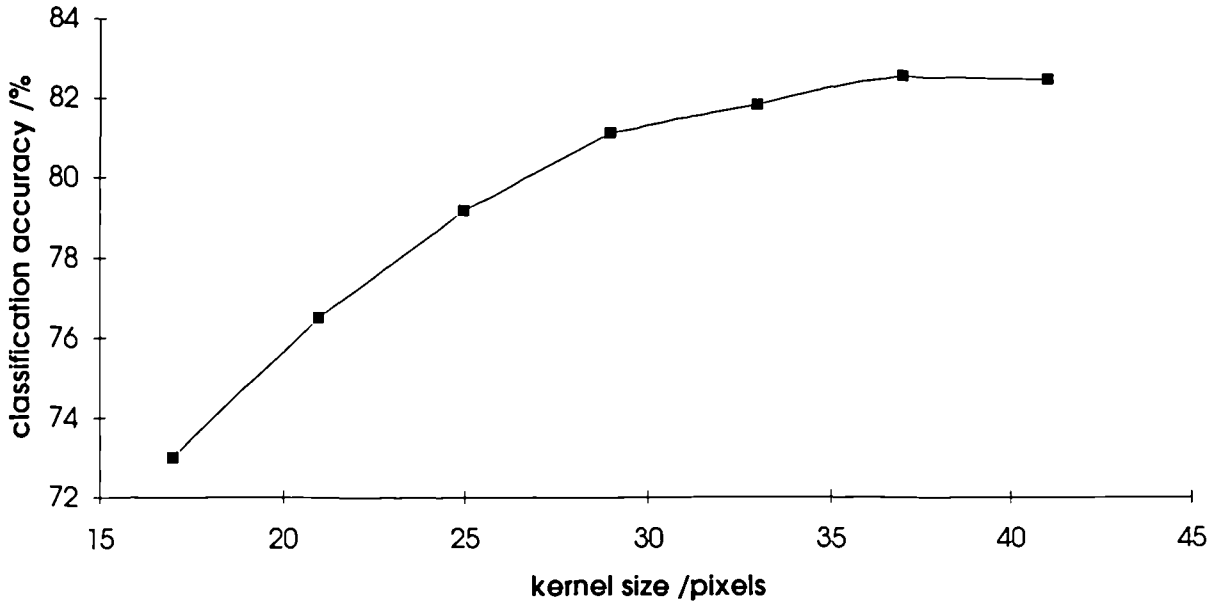


Figure 6.24 *Classification accuracy plotted against variable kernel size (i.e. the convolution averaging kernel) for the full resolution (pixel 6×6 m) data.*

Figure 6.23 illustrates how mean feature vectors increase with increasing λ values. The classification accuracies (for full resolution data here) were found to be fairly consistent until a threshold of $\lambda \approx 10$, then decreased considerably. Linnett *et al.* (1993) state that providing λ is less than the amplitude of the DN variations, then any value should be acceptable. Since the grey level variations can be quite subtle for regions such as sheetflow, the default value of 1 is retained.

6.6.4c Effect of varying kernel size A second parameter that Linnett *et al.*'s (1991; 1993) software allows one to alter is the dimension of the averaging kernel window applied to the difference between the upper and lower blankets. Figure 6.24 presents the effect of varying kernel size on classification accuracies (for full resolution data again). No significant improvement is gained from increasing the default size of 29 pixels.

6.6.4d Choice of connectivities The connectivity refers to which of the seven feature images should be used for generating class statistical information and subsequent classification. At half resolution, combining all seven feature images yielded a classification accuracy of 81.5%. If only the singular directions are utilised (features #1-4), an accuracy of 77.8% is achieved, and if only two of these are used (0 and 90° - #1 and 3), a classification accuracy of 61.7% is produced. A

classification based upon just the combined connectivities (features #5-7) gave an accuracy of 80.6%. It is clearly optimal to use all seven features.

6.6.4e Training site selection Some additional training site selections were made to investigate how much they may affect the classification accuracy of half resolution imagery. The incorrect classifications made in figure 6.20 were the identification of some of the shadows from the hummocks as sheetflow material. An additional class was selected to encompass these shadows, which were now identified as such, though this also had the effect of confusing some of the other class assignments; a classification accuracy of 69.0% was achieved. The training areas of the original four bottom types were varied in both location and area for a number of classification attempts; the best yielded an accuracy of 86.5%. Since even the bottom types selected for this composite image are not precisely uniform, these fractal classification results were deemed to be satisfactory for further investigation.

6.6.5 Application to full swath data

Ideally, if the fractal texture analysis technique can be used to identify volcanic morphologies and other bottom types imaged by TOBI, it would be most useful if it could be automatically applied to complete swaths or mosaics of the data. Figure 6.25a displays a portion of the TOBI data collected from the MAR. The swath is 6 km across, and each pixel is represented at its half resolution level (i.e. 12×12 m). Geologically, these data imaged a portion of segment 17 (Smith and Cann, 1992) of the Kane-Atlantis stretch of the MAR. The left (port) side of the image exhibits axial volcanic ridge characteristics of hummocky volcanic terrain, while the right side (starboard), though still within the median valley, shows fissuring and faulting of the hummocky terrain caused by lithospheric cooling and extension. Cann *et al.* (1992) provide a detailed description of the TOBI data and appropriate ground truthing. One does not know what is beneath the central, near-nadir portion of the swath, which is poorly imaged due to the geometry of sidescan operation and exacerbated by imperfect bottom detections.

Four training areas are defined in figure 6.25b, comprising the near-nadir region, shadows, hummocks and what will be described as fissures here, even though they are not strictly the same as the flat, fissured sheetflow terrain included in the test image. The classification results are presented in figure 6.25c and d; an excellent classification quality was obtained. These and all future displays of classification in this section have been median filtered with a window size of 7×7 pixels. The 'unknown' data regions are displayed as black, generally successfully encompassing

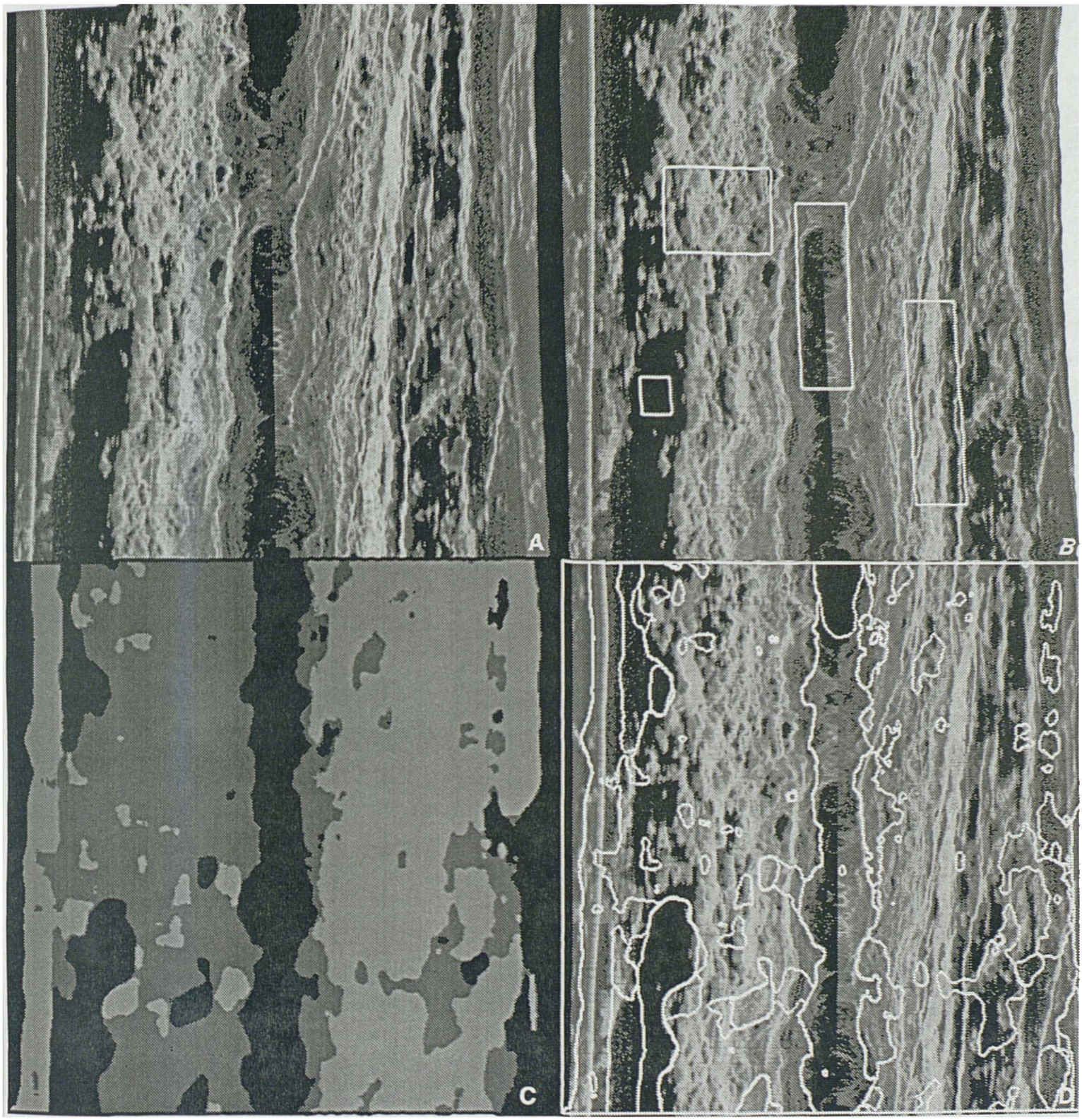


Figure 6.25 *Fractal texture analysis results for a complete swath of TOBI data (half resolution - pixel 12×12 m), principally hummocks and fissures: (a) original image; (b) original image with training areas superimposed; (c) classification results (refer to figure 6.19 key); and (d) classification results superimposed over original image.*

the near-nadir and shadows regions; these should be discounted in any spatial statistical determinations. The majority of the port side of the swath is classified as hummocks, with some small interspersed areas of smooth terrain characterised as shadows or fissures. The starboard side was classified as predominantly comprising fissured terrain, containing some small patches of hummocks; these can be identified by eye as well. The high grazing angle limits of both sides of the swath are classed as shadows, though one cannot really perceive anything by eye either. The surface reflection on the far port side is interpreted as fissured terrain due to its directional nature. Methods such as those of Le Bas (1993) should be first applied to remove these artifacts prior to analysis.

A histogram of the classified image can be simply constructed (e.g. using *pgmhist* in *PBMplus* software, Poskanzer, 1991) to determine areal percentages of each bottom type. For these data, 34% of the image is classified as comprising hummocks, 38% fissures and 28% near-nadir or shadows; discounting the 'unknown' data areas, 47% of the swath is hummocks and 53% are fissures. Since each pixel is 12×12 m, this gives a value of 16.9 km^2 of the swath being composed of fresh, unaltered hummocky terrain, and 19.1 km^2 affected by fissuring. Clearly, this is a very useful mapping tool.

To verify the above results, figure 6.26a, b and c display the classification results for the previous 1000 pings of data (at half resolution again). These were classified with the same training area class statistics as used in figure 6.25. The morphologies are very similar, although a little more complicated here. Classification was again remarkably successful. A lineament or ridge of hummocks and the surface reflection were identified as fissured terrain again, but generally the results were satisfactory.

Figure 6.27 presents a far more complicated swath of TOBI data. Both sheetflow and landslide bottom types are introduced here, in addition to the bottom types identified in the previously analysed swaths. Figure 6.27b illustrates the defined training areas (from left to right: hummocks, near-nadir, sheetflow, landslide and fissures). Figures 6.27c and e present the classification results - these are very messy. Obvious successes like the sheetflow (top right) and hummocks can be identified; however, there are also many anomalies, principally restricted to the distinction between some of the relatively 'smooth' bottom types. For example, the large shadow on the left side of the swath has been characterised as a sheetflow unit.

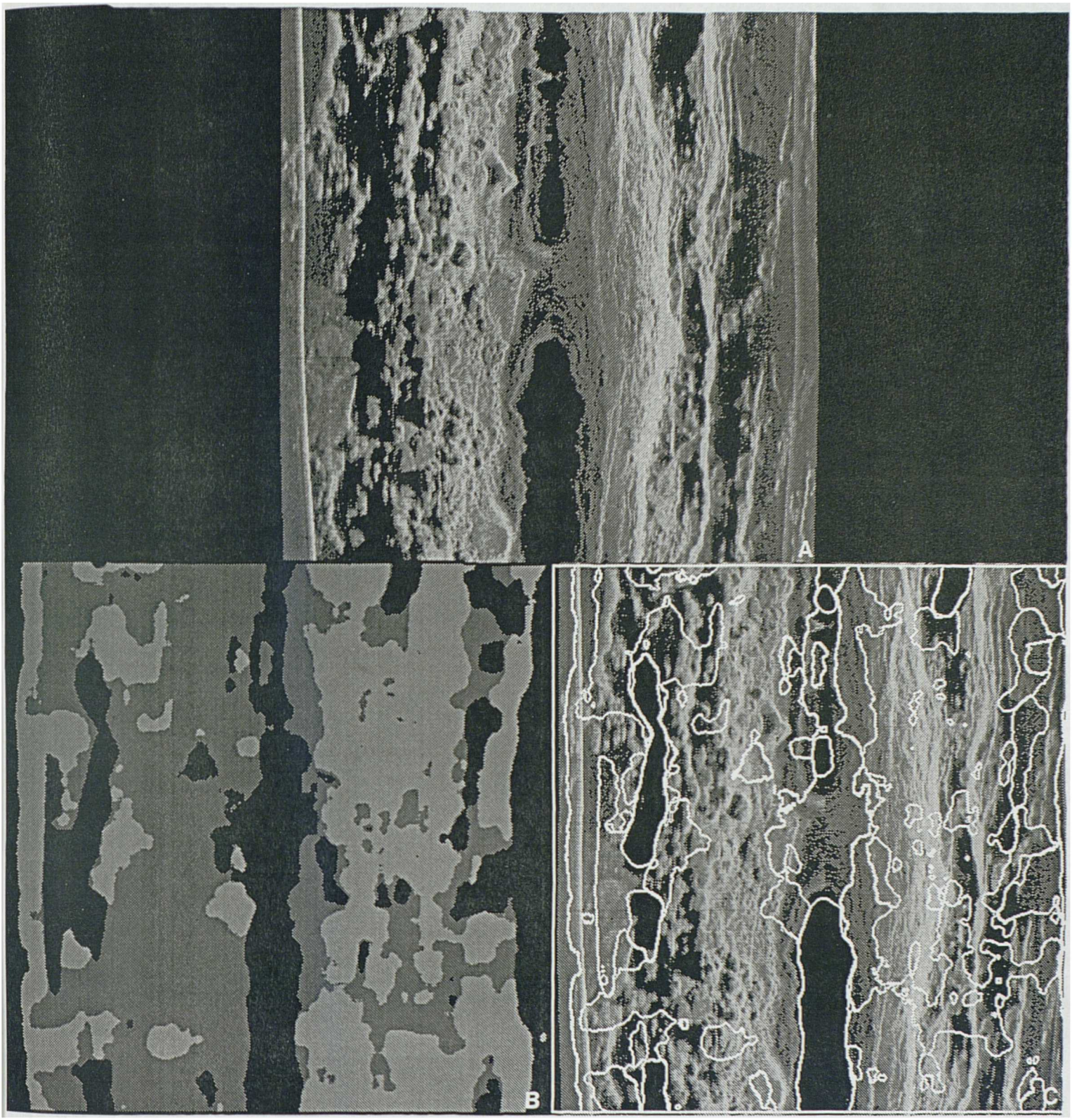
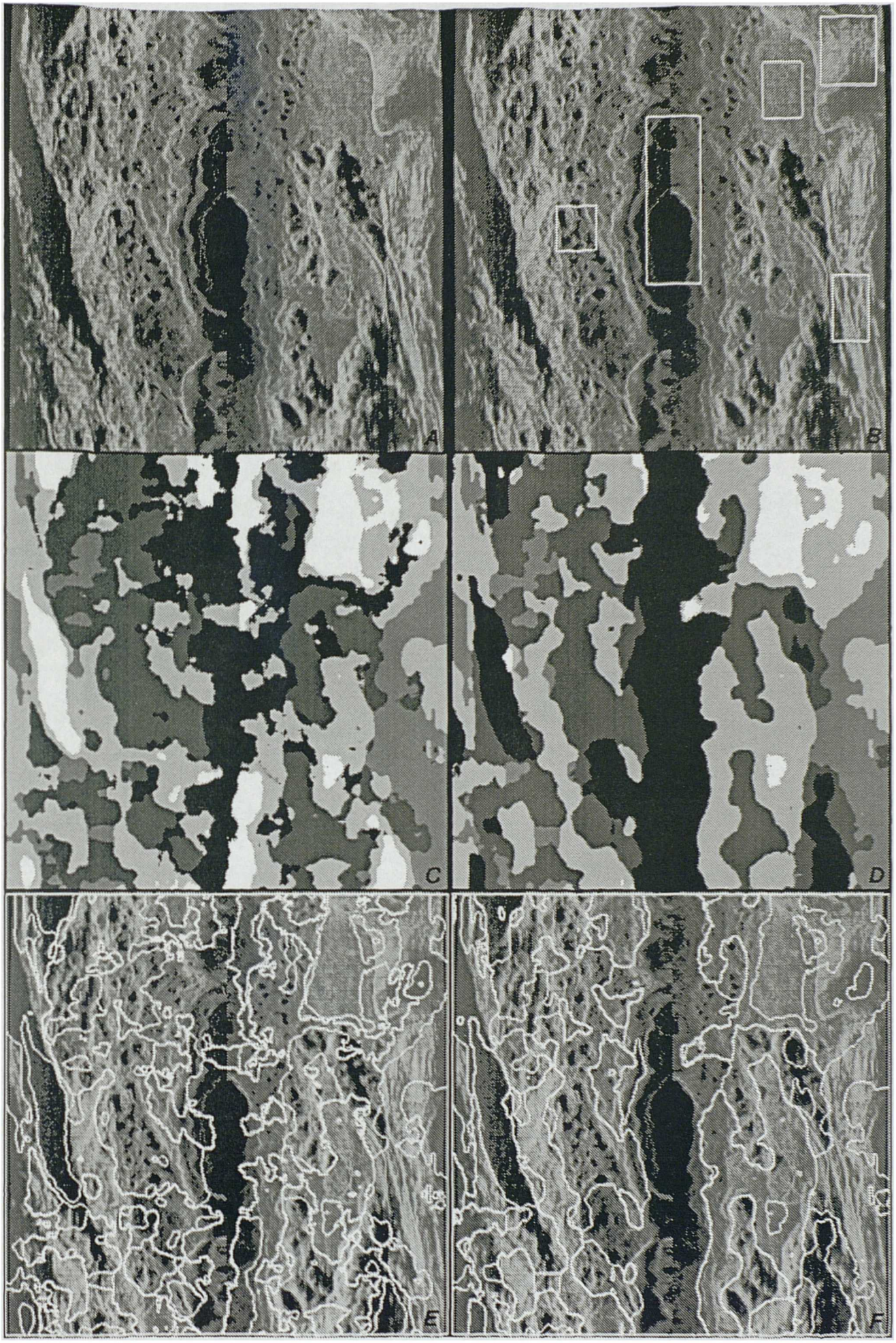


Figure 6.26 *Fractal texture analysis results for another complete swath of TOBI data (half resolution - pixel 12×12 m): (a) original image; (b) classification results (refer to figure 6.19 key); and (c) classification results superimposed over original image. The training area class statistics from figure 6.25 were used here.*

Figure 6.27 *See next page. Fractal texture analysis results for a more complicated swath of TOBI data (half resolution - pixel 12×12 m), this time including sheetflow and landslides aswell:* ***(a)*** *original image;* ***(b)*** *original image with training areas superimposed;* ***(c)*** *classification results (refer to figure 6.19 key);* ***(d)*** *classification results utilising the eighth feature vector;* ***(e)*** *classification results of (c) superimposed over original image;* ***(f)*** *classification results of (d) superimposed over original image.*


A
B
C
D
E
F

6.6.5a An eighth feature vector The texture analysis alone was unable to distinguish between the relatively smooth TOBI regions of shadows, near-nadir noise and sheetflows at this resolution. Straightforward first-order statistics based on the tonal information of the image should be able to aid the characterisation here. This introduces the possible problem of gain variations, though through extensive studies of TOBI data, I feel these are insignificant compared with the geological variations. An eighth feature vector image was therefore constructed, simply by applying Linnett *et al.*'s (1991) averaging convolution filter of dimension 29x29 pixels to the original image, to produce a feature image representing the mean DN variations of the image. Figure 6.27d and f display the classification results achieved with this additional grey tone information and the results have obviously improved. The only persistant problem now is the rather overstated classification of landslide materials. Basically, any uneven flow type regions identified by eye have been interpreted as landslides. There is a failing of the technique here, since without knowledge of the geological setting, such as neighbouring fabrics together with co-registered and comparative resolution bathymetry, the identification of landslide terrains will remain problematic. This is a difficult test for the trained observer as well, though additional information from dredge hauls and other ground truthing aids the interpretation.

6.6.6 Applications

6.6.6a Quantification of the spatial variation of volcanic morphologies within spreading segments Section 6.6.5 described how fresh, unsedimented volcanic morphologies, such as hummocks, fissured and unfissured sheetflows, can be successfully classified using the fractal analysis technique. Segments along the MAR between 24 and 30°N possess different volcanic characteristics, so a quantitative method for describing these would be very useful. Since the TOBI data collected in this region by Cann *et al.* (1992) have yet to be mosaicked at the resolution of 12 m per pixel, mainly due to software and computational power restrictions, unfortunately only representative samples of the segments could be selected for this analysis. These will illustrate the potentials of the technique, although clearly true geographically defined portions of the data would be best.

Figure 6.28a displays a composite image of some representative segment volcanic morphologies. The left ('port') side of the swath represents a portion of the neo-volcanic zone of segment 17 of the MAR (24-30°N) as identified by Cann *et al.* (1992). Sempéré *et al.* (1993) describe this as a typical MAR segment comprising a broad, long axial volcanic ridge, identified bathymetrically, and confirmed by Lin *et*

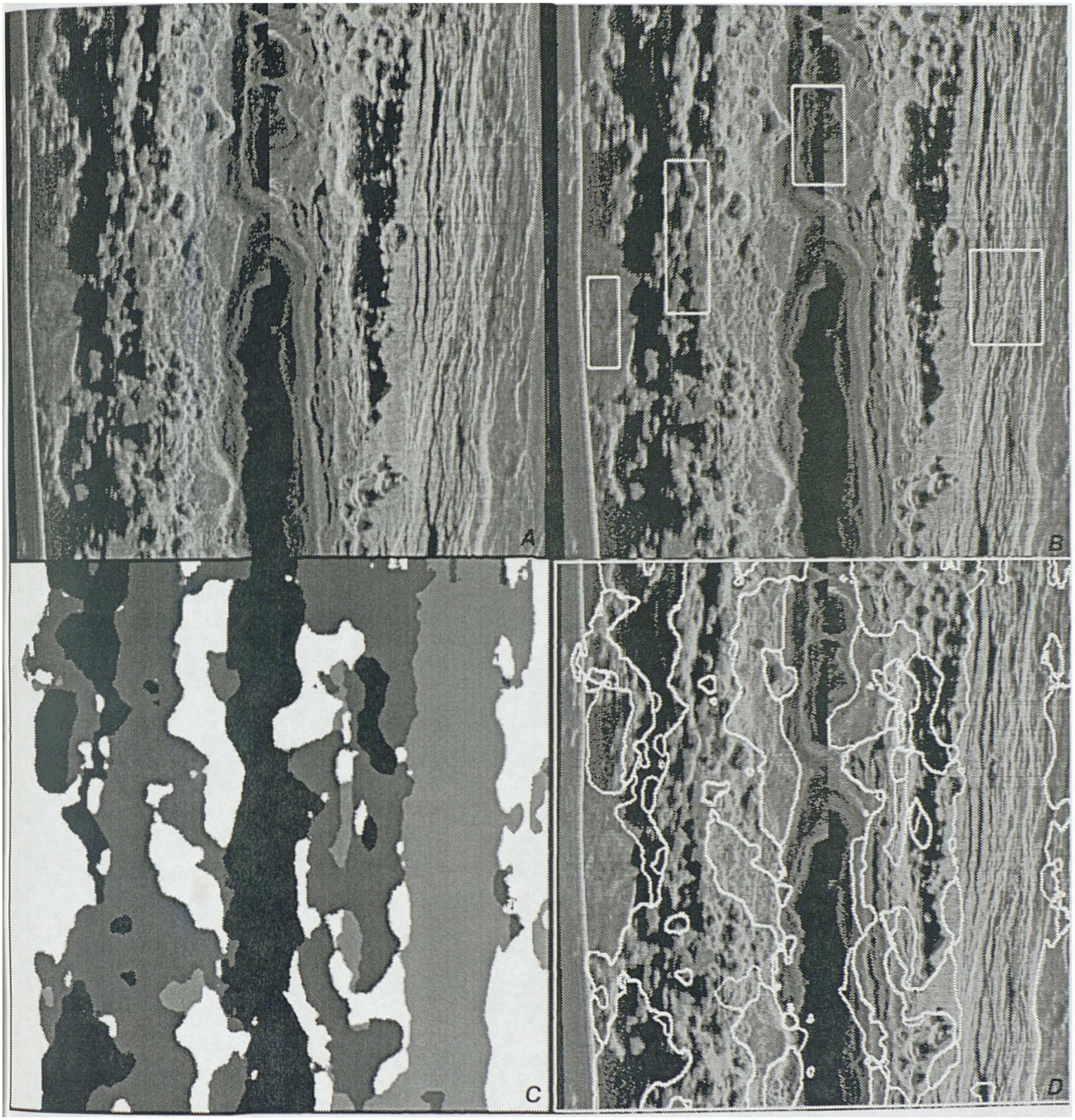


Figure 6.28 *Fractal texture analysis results for a typical portion of segment 17 (left half) and segment 6 (right half) of MAR (24-30°N) TOBI data (half resolution - pixel 12×12 m): (a) original image; (b) original image with training areas superimposed; (c) classification results (refer to figure 6.19 key); and (d) classification results superimposed over original image.*

al.'s (1990) gravity analysis as a significant region of mantle upwelling. It is characterised by fresh hummocky volcanic terrain, with individual hummocks possessing a diameter of some 100-200 m. Segment 6 of the MAR in this region was described as unusual by Sempéré *et al.* (1993) as it does not possess a noticeable volcanic ridge. TOBI imagery collected by Cann *et al.* (1992) yielded flat, fissured sheetflow terrain, more reminiscent of East Pacific Rise volcanic morphologies (e.g. Lonsdale, 1977; and Macdonald and Fox, 1988). The right side ('starboard') of figure 6.28a presents a representative sample of segment 6, which is mostly fissures, but a minor hummocky volcanic ridge can be seen emanating from one of these fissures.

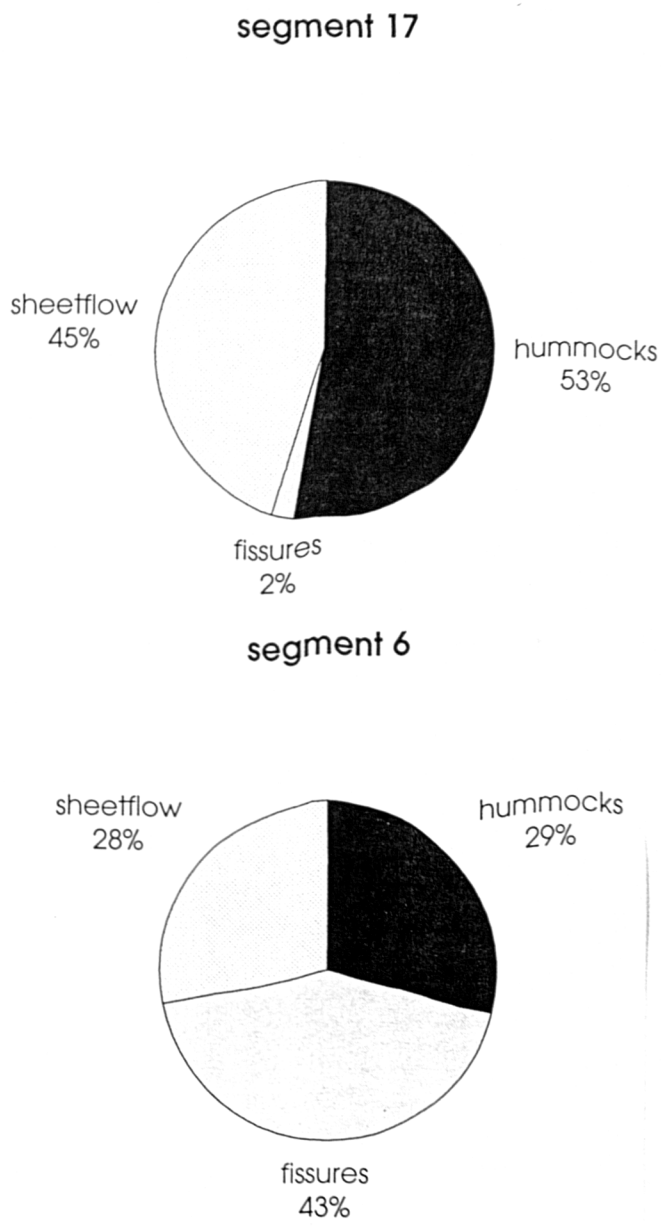


Figure 6.29 *Pie charts representing bottom type variation for segments 17 and 6, based on the texture analysis results of figure 6.28 (derived from histograms of 6.28c).*

Figure 6.28b displays the training areas of (left to right) sheetflow, hummocks, near-nadir region and fissured sheetflows. Figures 6.28c and d present the classification results; these agree closely with a trained observer's interpretation. Figure 6.29 displays the spatial distribution of these volcanic morphologies derived from histogram analyses, a far simpler job now than having to calculate them manually. These types of results can lead to a wide range of applications.

6.6.6b The off-axis evolution of volcanic morphologies An interesting phenomenon of mid-ocean ridge geology is how median valley terrain evolves with transportation off-axis (chapter 5, section 5.5 investigated this aswell), based on tectonic break-up and sedimentation. Figure 6.30a presents a section of an off-axis TOBI swath collected by Cann *et al.* (1992) on the MAR. The 'dark' nature of the image is due to the pelagic sediment cover absorbing the sonar energy. The left (port) side is mainly sedimented fissures, with occasional fault scarps and hummocks protruding through the sediment blanket, particularly in the high grazing angle region, whilst the right (starboard) side is chiefly composed of partially buried hummocks; probably a fossilized volcanic ridge (Cann *et al.*, 1992). An acoustical approach is best here to determine degrees of sediment cover (Mitchell, 1993) and was also attempted in this thesis in chapter 5, section 5.5.3. However, since we do not possess true backscatter strengths of TOBI data and no co-registered and appropriate resolution bathymetry data are available, this approach is difficult. I am not trying to quantify amounts of sediment cover, just to demonstrate that bottom fabric classification is still possible even though the acoustical characteristics of the surface materials have changed.

Figure 6.30b indicates the location of the training areas; note a very small area defined over an exposed fault scarp protruding through the sediments. Figure 6.30c and d present the classification results and the fractal classification method is still successful. Figure 6.31 is a plot of how the mean feature vectors are affected by the sedimentation process. Data are displayed from the sedimented and unsedimented fissures and hummocks training areas from figures 6.30 and 6.28, respectively. The open symbols represent sedimented bottom types, and the filled are unsedimented. These can at least be said to be unsedimented within the penetration depth of the 30 kHz sonar energy, so a light dusting of sediments is possible and probable (e.g. Ballard and van Andel, 1977).

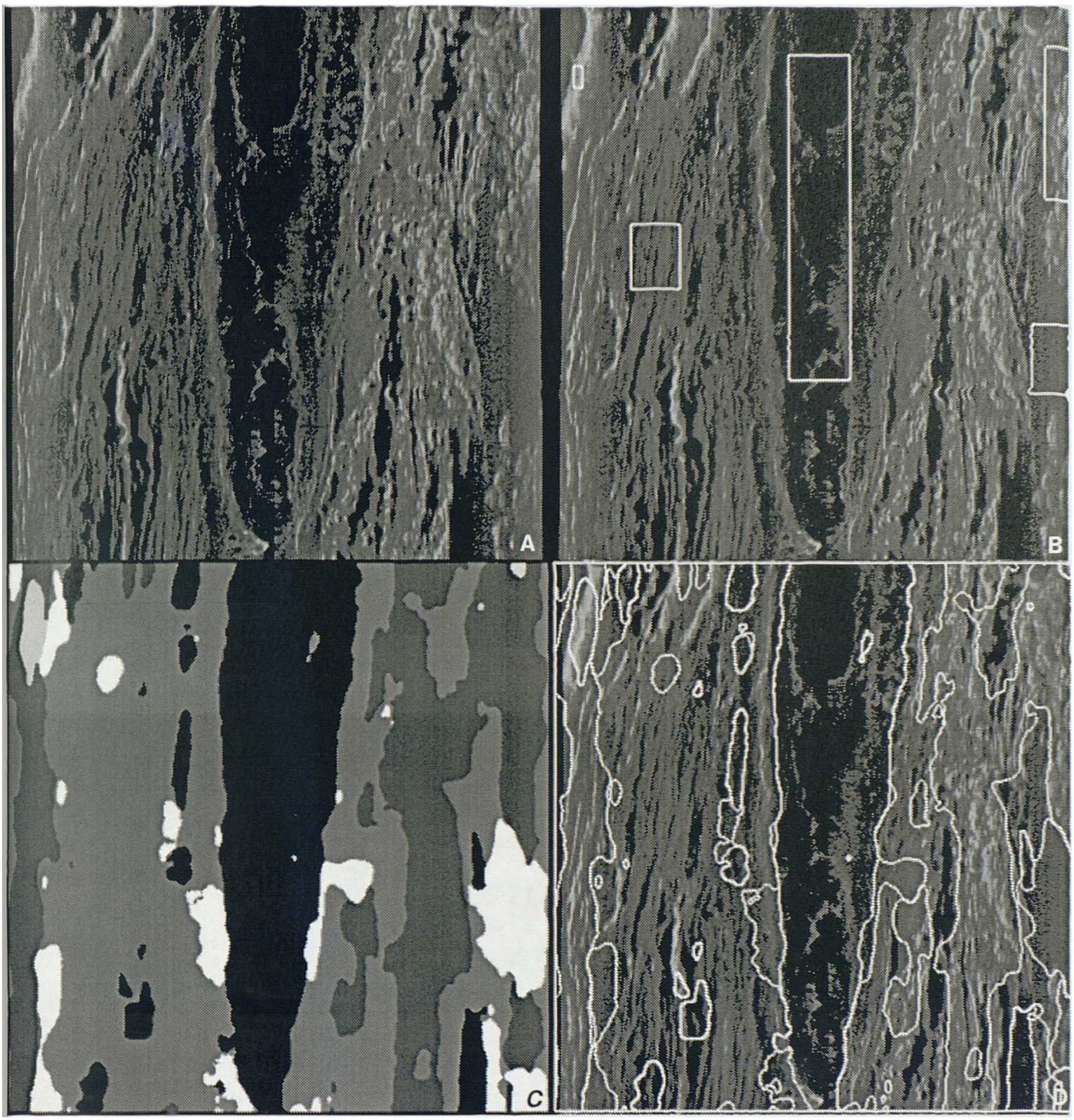


Figure 6.30 *Fractal texture analysis results for off-axis TOBI imagery (half resolution - pixel 12×12 m), sedimented fissures/faults (left half) and hummocks + fissures/faults (right half): (**a**) original image; (**b**) original image with training areas superimposed; (**c**) classification results (refer to figure 6.19 key); and (**d**) classification results superimposed over original image.*

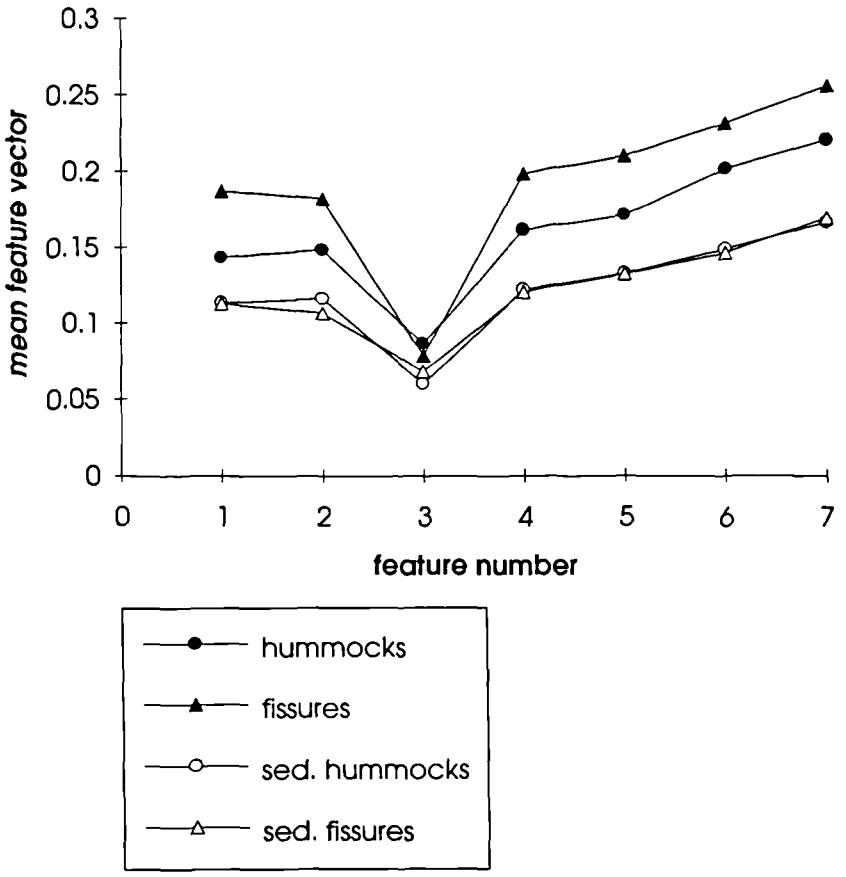


Figure 6.31 Effect of sedimentation on feature vectors: mean feature vectors plotted against feature numbers for hummocks and fissures bottom types, at both unsedimented (filled symbols) and sedimented (open symbols) sites.

It is clear that sedimentation both lowers the overall value of the mean feature vectors and also dampens any directional fluctuations. In fact, the sedimented mean feature vector responses are very similar for the hummocks and fissures, with the hummocks now possessing the deepest trough at feature #3. I believe the hummocks are still protruding somewhat through the sediments due to a ponded nature of sediment accumulation, retaining their highly reflective properties here, whilst the flatter fissured sheetflows have been completely buried. Maybe it is surprising that the hummocks and fissures are still so well distinguished by the fractal classification method, the eighth feature vector of mean DN must have been significant, confirming its important contribution to the characterisation process.

6.6.6c Application to bathymetry data

The statistical description of seafloor topography has been widely researched for both geological and underwater acoustical purposes. Bell (1975), Berkson and

Matthews (1983) and Fox and Hayes (1985) present various power spectral methods for assessing the roughness, and latterly, the anisotopy of the seafloor. Goff and Jordan (1988) inverted Sea Beam bathymetric data and computed second order statistics. Smith and Shaw (1989) and Shaw and Smith (1990) described abyssal hill topography through analysis of slope distributions, while Malinverno (1989) segmented topographic profiles using a fractal based self-affine model.

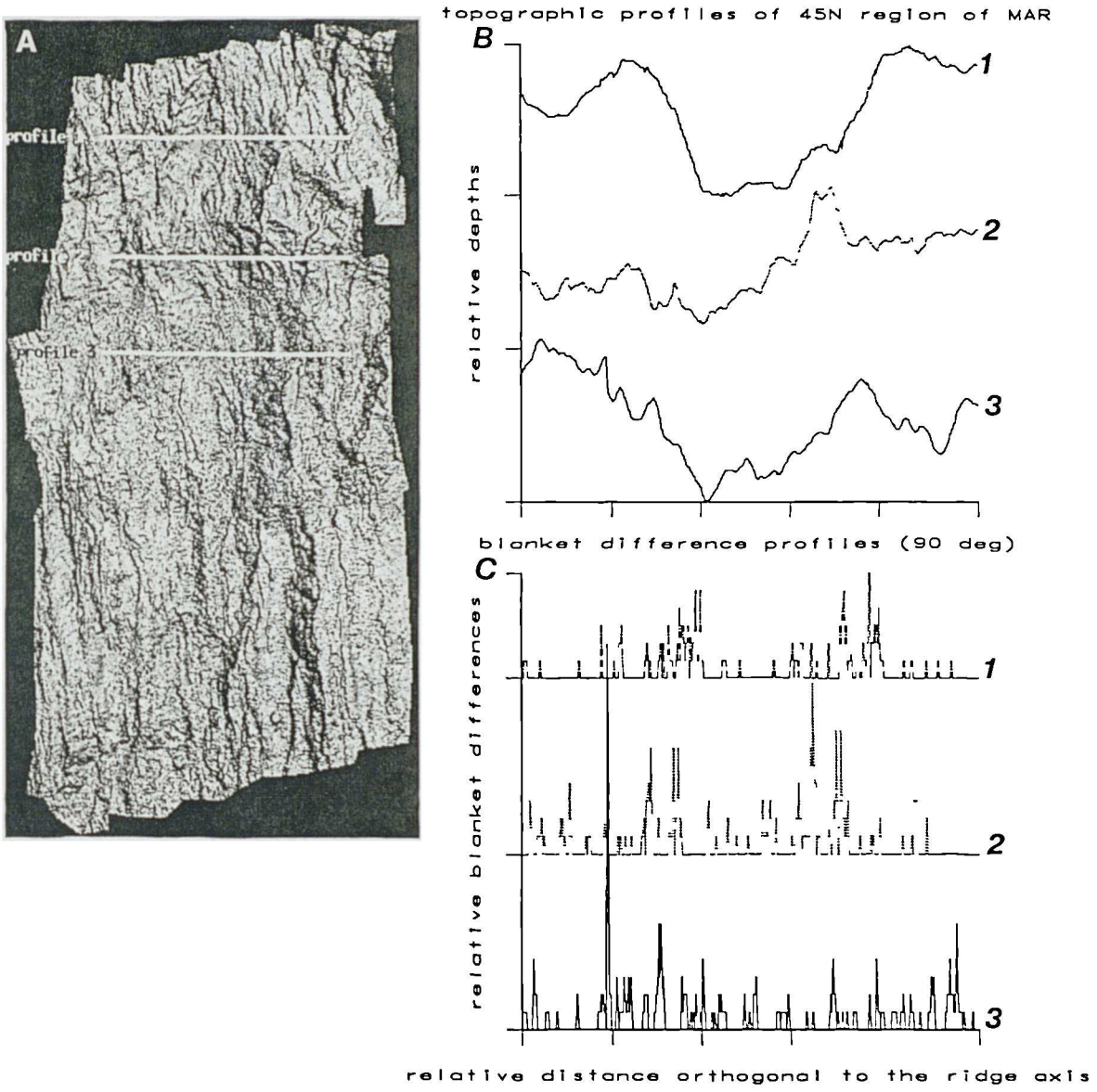


Figure 6.32 *Application of fractal texture analysis techniques to bathymetry data: (a) 45° bathymetry shaded relief representation, rotated so the ridge axis is parallel to the vertical, three profiles are annotated; (b) topographic profiles from (a); and (c) blanket difference plots (before convolution) of these profiles (90° feature vector - orthogonal to ridge axis).*

Since the fractal texture analysis method of Linnett *et al.* (1991, 1993) effectively computes the roughness and directionality of a sidescan sonar image surface through Mandelbrot's (1983) blanket determinations, I propose that the same algorithms can simply be applied in the same manner to a bathymetric grid to provide another method of quantifying the roughness of the seafloor.

The only bathymetric dataset described in this thesis (chapter 4) is the 45°N region of the MAR. These data do not extend far off-axis, so there are no significant topographic roughness variations. Figure 6.32a presents the location of three profiles superimposed on a shaded relief representation of the data, rotated by 13° so the image vertical is parallel to the ridge axis. Figure 6.32b presents the topographic profiles and figure 6.32c the difference between the upper and lower blankets prior to any filtering. These very preliminary results suggest that the difference values do vary with topographic roughness. Major fault scarps correspond with major peaks in difference values, and minor fluctuating differences correspond to relatively flat regions. The potential application of the fractal analysis methods of Linnett *et al.* (1991, 1993) is therefore promising and worthy of further investigation.

6.6.7 Conclusions

The fractal texture analysis method has been successfully applied as a classification technique for mapping the different volcanic morphologies imaged by TOBI at the MAR on complete sections of the swath data. The technique also yielded reasonably successful results in charcterising bottom types expected to possess different acoustical properties, such as landslide and sedimented terrain, although the former can be ambiguous. For landslide characterisation, it can be concluded that what are really required are co-registered, similar resolution bathymetry data for an accurate quantification. The composite test image was classified with an optimum accuracy of 86.5%. Artifact regions of the swath such as the near-nadir region and extensive shadowed portions could also be identified with the aid of an eighth feature vector: the first order statistical measure of the mean grey level variation of the image. Additional applications of the technique, such as its potential for characterising the degree of roughness of bathymetry surfaces have also been proposed. The fractal analysis technique has therefore been found to be an extremely useful tool for characterising mid-ocean ridge terrain imaged by TOBI.

6.7 COMPARISON OF THE GLCM AND FRACTAL METHODS

The classification of the TOBI mid-ocean ridge bottom types was more accurate using the fractal than the GLCM technique. An improvement of around 20% was achieved for a composite test image. The GLCM technique was therefore discarded from further investigation and the fractal method was applied successfully to a number of potential applications. The GLCM method was also significantly computationally more expensive than the fractal method. It took about 40 minutes to generate the four feature images required for the optimum classification results, excluding any classification, on a Sun SPARC 10 processor, compared with about 2 minutes in total for the fractal method, including classification. The concept of the fractal method based on effectively degrees of 'roughness' of the texture was easier to comprehend than the second order statistical nature of the GLCMs, and for these data was clearly more than adequate. Conclusively, the fractal texture analysis method is superior for characterising the TOBI imagery.

6.8 CONCLUSIONS

The fractal analysis technique has been demonstrated to be capable of mapping the major bottom types for this dataset. Until true backscatter measurements and co-registered bathymetry information are acquired for TOBI data, this method provides the best approach thus far for quantitatively analysing the information. It is able to distinguish simple volcanic morphologies with a similar degree of success as a trained interpreter, and therefore has great potential as a mapping tool. Certainly complicated interactions of terrain types will always require a geologist's skill. Additional techniques, possibly object recognition algorithms, such as the limited success of automatic picking of seamounts (chapter 7) and the excellent method of identifying the fault scarps from the same dataset using wavelet analysis of Little and Smith (Fault scarp identification in side-scan sonar and bathymetry data from the Mid-Atlantic Ridge using 2-D wavelet analysis, submitted to *Marine Geophysical Researches,* 1993) may be combined with the fractal texture method.

CHAPTER 7
THE DETECTION OF SEAMOUNTS USING THE HOUGH TRANSFORM

7.1 INTRODUCTION

Several studies have shown that the axial volcanic zones situated within the median valley of mid-ocean ridges are strewn with small near-circular volcanoes. Jordan *et al.* (1983) and Smith and Jordan (1987) describe an exponential size-frequency distribution for the heights of off-axis seamounts in the Eastern Pacific. The average number of seamounts, v, with height $h \geq H$ can be expressed as:

$$v(H) = v_o e^{-\beta H},$$

yielding two population parameters: v_o, the expected number of seamounts per unit area, and β^{-1}, the characteristic height of the population; $v_o = (5.4 \pm 0.6)\times10^{-9}$ m^{-2} and $\beta = (3.47 \pm 0.20)\times10^{-3}$ m^{-1} for the Pacific dataset. This method has been applied by Smith and Cann (1990; 1992) to the Mid-Atlantic Ridge (MAR), where they identified 481 seamounts from about 6000 km^2 of Sea Beam swaths of the median valley floor between 24° and 30°N; deriving $v_o = (19.5 \pm 0.9)\times10^{-8}$ m^{-2} and $\beta = (17.2 \pm 0.5)\times10^{-3}$ m^{-1}.

Keeton and Smith (1991) investigated the population characteristics of seamounts observed on the en echelon volcanic ridges of the Reykjanes Ridge (60°N) using both Hydrosweep bathymetry and TOBI sidescan sonar data independently. Analysis of the bathymetry led to the recognition of 210 seamounts in a swath area of 1285 km^2, giving $v_o = (12.8 \pm 1.2)\times10^{-8}$ m^{-2} and $\beta = (12.9 \pm 0.4)\times10^{-3}$ m^{-1}. The sidescan data showed 146 seamounts within 874 km^2, and figure 7.1 presents these distributions. The three sets of values refer to the northern, southern and total sections of the study area (centred at 58°N), forming excellent straight line fits, with an accuracy of 94% for the total area, and yielding $v_o = (6.5 \pm 0.5)\times10^{-9}$ m^2. Since only the seamount radii can be measured directly from the sidescan images, a height-to-radius ratio is assumed ($\xi_h = 0.210 \pm 0.008$), derived from numerous separate studies (e.g. Smith and Jordan, 1987); β then becomes 4.2×10^{-3} m^{-1}. The study of seamount statistics is particularly interesting over the Reykjanes

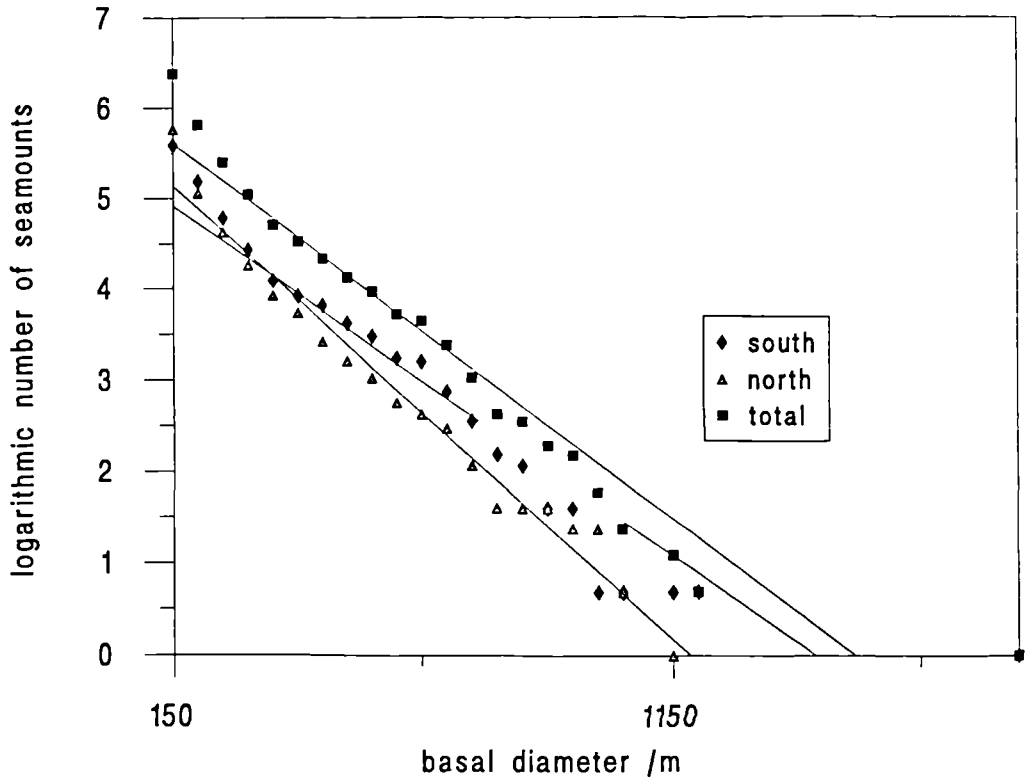


Figure 7.1 *Seamount distributions from TOBI sidescan imagery collected over the 58°N area of the Reykjanes Ridge. Distributions are separated for the northern and southern sections of this area, and are also combined. The straight line fits are accurate to 90% for the southern area, 98% for the northern, and 94% for the combined areas.*

Ridge as the gross morphology progressively changes from a typical slow-spreading axial valley to a fast-spreading-like axial rise, with closer proximity to Iceland. However, the volcanic styles do not generally appear to change, and by contrast, small seamounts are relatively rare at the axis of the fast-spreading East Pacific Rise (e.g. Fornari *et al.*, 1987). The discrepancies in population parameters between the two sets of data raise important questions, such as: at what size do volcanoes become hummocks? are seamounts identified in the bathymetry, not classified as such in the sidescan data? and many more. They also confirm the benefit of consulting as many datasets as possible for analyses.

It has been demonstrated that the statistical analysis of seamount populations within and between regions of mid-ocean ridges provides important information about the role of seamount volcanism in crustal accretion. Therefore, the investigation of seamount distributions can be a very useful tool for analyzing certain aspects of bathymetric or sidescan datasets.

The identification of seamounts from both bathymetry and sidescan sonar datasets is an onerous, subjective task; a quantitative detection, or object recognition, routine has therefore been investigated. Previous studies (e.g. Smith and Cann, 1990; 1992) required that the criteria for the identification of a seamount were that the feature should be approximately circular, with an aspect ratio of less than 2 and contained within specific size bounds. It was therefore decided that an automatic picker should chiefly look for circular features.

7.2 THE HOUGH TRANSFORM

7.2.1 Previous work

The Hough transform has been previously used for the identification of circular features in disciplines as diverse as detecting tumours from chest radiographs (Kimme *et al.*, 1975), to quality control in biscuit manufacture (Davies, 1984). Cross (1988) applied the technique to the identification of circular geological structures from Landsat Multispectral Scanner imagery of the Egyptian/Sudanese border.

7.2.2 Theory

The Hough transform remaps edge points from *(x,y)* space into parameter space (refer to Duda and Hart, 1972; Pratt, 1978). In the case of a circle, the Cartesian equation is:

$$r^2 = (x-a)^2 + (y-b)^2,$$

where r is the radius of the circle and *(a,b)* are the coordinates of the centre of the circle. The transform therefore projects edge points into 3-dimensional parameter space *(r,a,b)*. A single edge point transforms to produce a circular locus for each possible radius; these loci represent circles of which the centre of the sought circle may be part. These loci will overlap if a circle is present, producing a peak in the parameter space (figure 7.2).

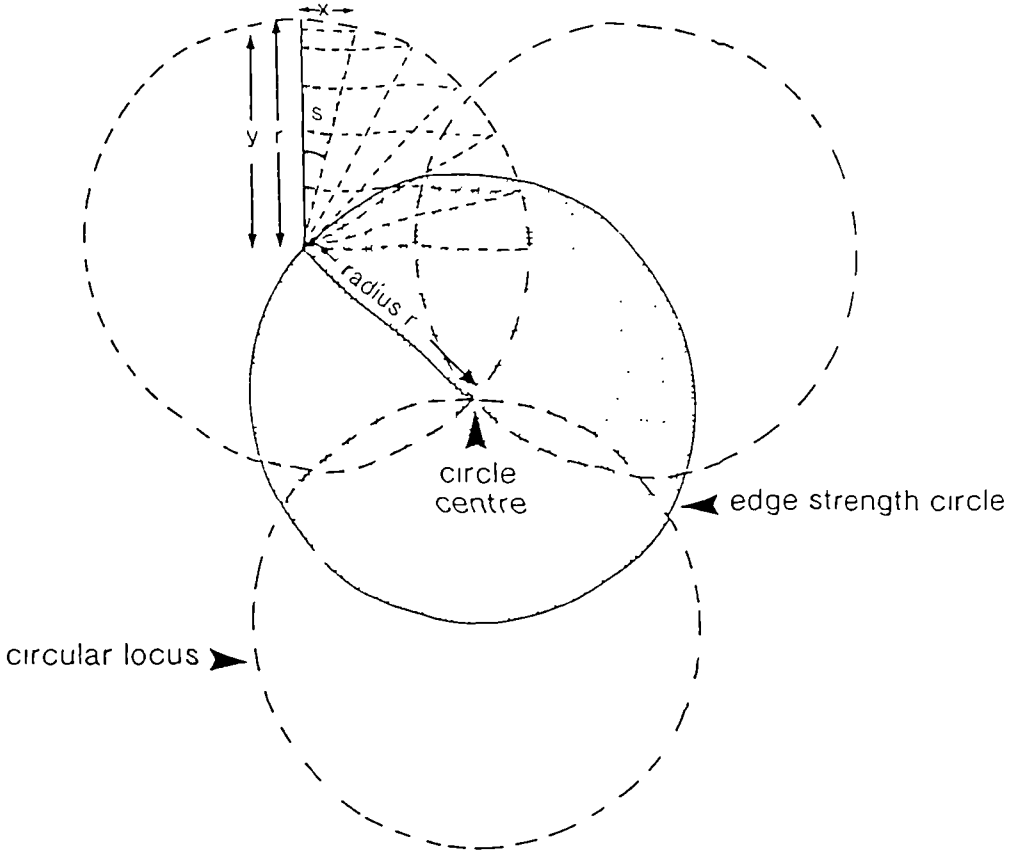


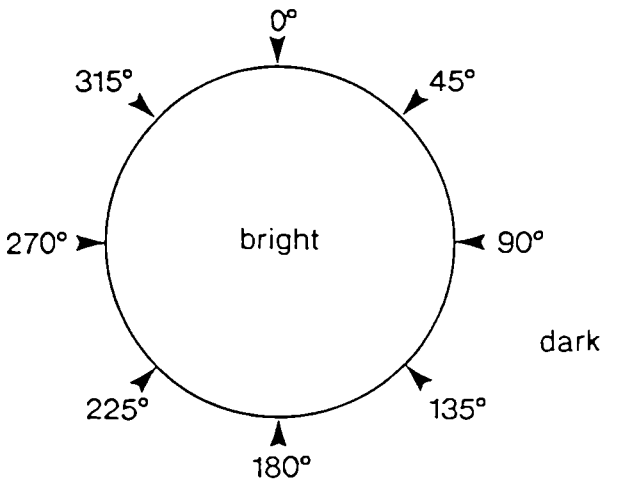
Figure 7.2 *The Hough transform divides a 45° arc of the circular locus into a number of steps n of size s. The first point of the locus is drawn as* $x = r\sin\theta$ *and* $y = -r\cos\theta$, *where* $\theta = 0$. θ *is then increased by the angular step s, calculating x and y as before.* θ *is incremented as 2s, 3s, ...ns and -s, -2s, -3s, ...-ns until the complete 45° arc has been drawn. Concurrently, each of these points is reflected into the other 7 sectors producing a complete circular locus. Three such circular loci are illustrated, derived from the edge of a circular feature (shaded). They clearly overlap at the centre of this feature.*

7.2.3 Software description

A typical Hough transform routine for circle detection, *hough_circ.f*, written by A.M. Cross, was acquired from the NERC Unit of Thematic Information Systems, University of Reading. The package was supplied in VAX format, in a form suitable for incorporation within the I^2S image processing system. Appropriate UNIX modifications have been completed, and a simple 200x200 raw byte input image format has been adopted. The two input images required by *hough_circ* are an edge strength and an edge orientation image.

The edge detection program, *edge_extract.f* (appendix 19), requires a PGM image file input, a format commonly used for image data, developed within an extensive, public-domain software library, *PBMplus* (Poskanzer, 1991). This package was originally designed to supply many different image and bitmap conversion routines that would allow a user to easily convert from one image format to another. Additionally, many other functions allow a user to cut, scale and enhance selected images; which is particularly useful here prior to edge extraction. *edge_extract* applies the Sobel edge operator to the input image, where

$$S_x = \begin{matrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{matrix} \quad \text{and} \quad S_y = \begin{matrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{matrix}$$



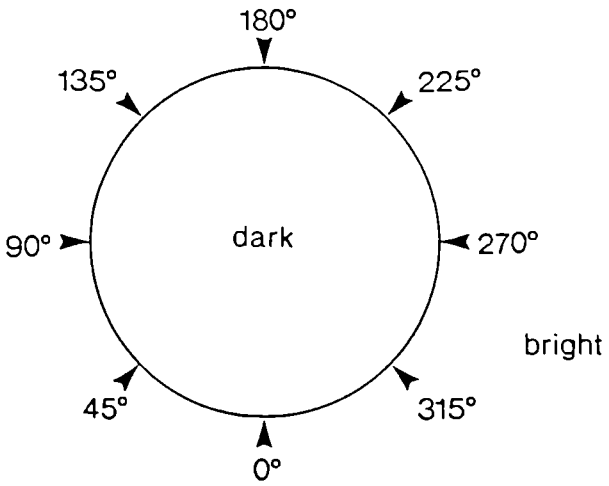


Figure 7.3 *The character of the edge orientation information, derived from taking the arctangent of the ratio of the 2 Sobel edge operators, S_x and S_y.*

The edge strength value is simply $(S_x^2+S_y^2)^{1/2}$, while the edge orientation is calculated as $tan^{-1}(S_y/S_x)$. In the latter case, figure 7.3 may be referred to for clarification. However for byte image representation purposes, the output values have been compressed into the range 0-180°, where each grey level represents 2 degrees of azimuth, measured clockwise from 'north'.

In the program *hough_circ*, the detailed principles of which are described in Cross (1988), the three dimensional parameter space *(r,a,b)* mentioned earlier is quantised into discrete bins; called the 'accumulator array', wherein an edge point 'votes' for all those circles of which it may be part. In order to reduce the memory required by the accumulator array, the sought circle radii are limited to a number of discrete values, producing a two dimensional array *(a,b)*, where the maximum counts from all the radii are stored in each cell; a maximum of 5 radii may be selected during any single program run. A threshold value is set for the edge strength image, which is then converted into a bitmap in preparation for the application of the transform. This input threshold may be selected in a number of ways. For experimental purposes, an interactive thresholding procedure, supplied in the *editimage* routine within the *KHOROS* image processing package, provides a simple, visual guide. The threshold value applied to the Sobel derived edge strength image is increased until a balance of noise removal and retention of major edges has been achieved. A more objective method would be to produce a histogram of the edge strength image and select a threshold which captures the tail of the negatively skewed distribution. The edge orientation information can be employed if a specific contrast is required, following a method introduced by Kimme *et al.* (1975). A specific contrast relates to whether 'light pixels on dark', technically described as 'negatively contrasting' objects are to be detected, or vice versa. This option has the effect of only drawing a specified arc length of the usual circular locus produced by the transform for each edge point (figure 7.4). The arc is generated in the direction of the edge, i.e. the corresponding pixel value in the edge orientation image is uncompressed by multiplying back up again by two to provide the edge orientation values consistent with those illustrated in figure 7.3. The arc length can be adjusted according to the quality of the data; small values tend to miss irregular objects, while large values may produce a number of spurious peaks. Experimentally, 50° proved to be the optimum selection.

Once the Hough transform has been applied to the edge images, the accumulator array comprises a series of overlapping loci. It is now appropriate to select which peaks in the array are consistent with circle centres. An output threshold must be selected, often following an initial experimental run of the program, where neither of

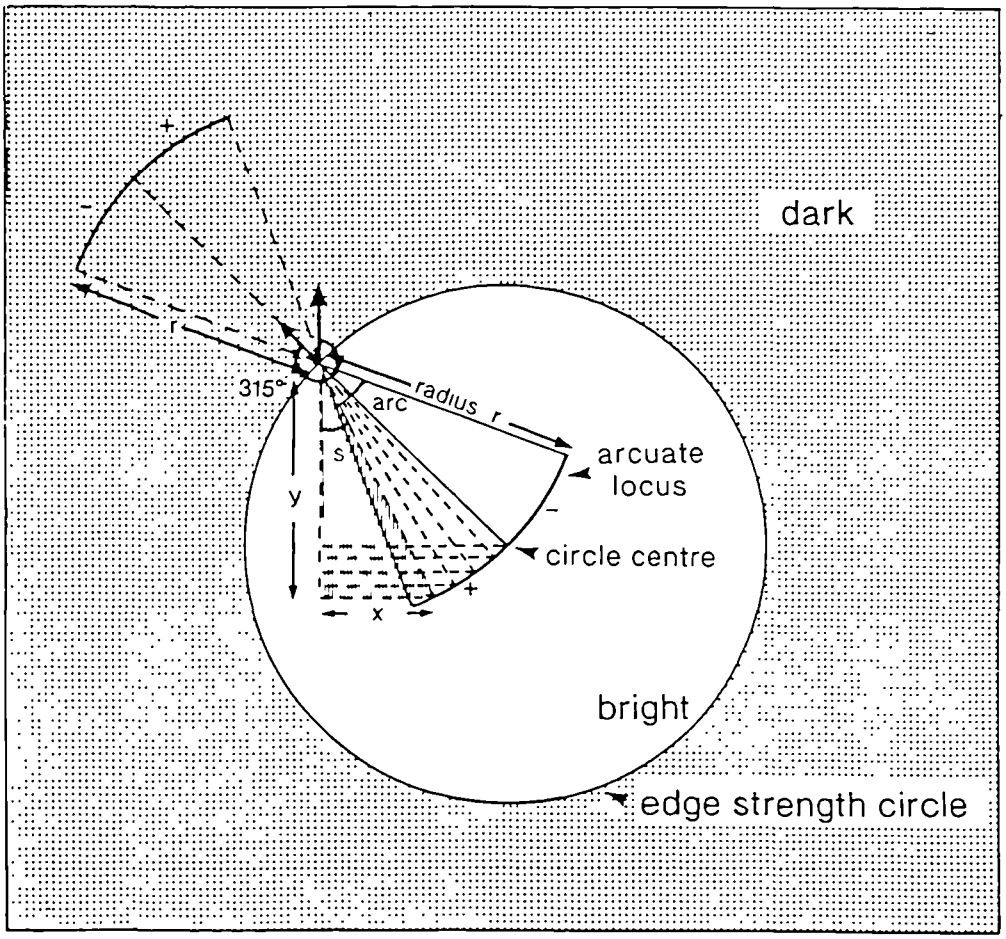


Figure 7.4 *When utilising the aspect information, only a user-defined portion (angle 'arc') of the circular locus is drawn in the direction of the edge orientation, in the same manner as figure 7.2. The first point, $x = r\sin\theta$ and $y = -r\cos\theta$ is drawn. θ equals the edge orientation image pixel value corresponding to this particular edge strength cell, for the positively contrasting case (i.e. 'dark on light'); for the negatively contrasting case, 180° is added to θ, and 360° subtracted if the result is greater than 360°. x and y are calculated where θ is incremented by the step size s, as +s, +2s, ...+ns and -s, -2s, ...-ns. This operation is repeated until an arc length of angle arc has been drawn. This is reflected 180° away, if both positively and negatively contrasting features are sought.*

the peak identification procedures have been chosen and an arbitrary output threshold has been set. Personal experience recommends the generation of a histogram calculated from the outputted accumulator array, maybe using *pgmhist* in

the *PBMplus* software, and consequently selecting a cut-off point in the low frequency tail.

A simple thresholding operation is insufficient for the selection of peaks associated with circle centres because these peaks may possess varying strengths; so in order to capture authentic minor peaks, broad 'islands' remain around the stronger ones, aswell as additional spurious peaks. One of two methods within *hough_circ* may be chosen for the selection of the peak cells after the initial transform. The 'connected component peaks' method labels 4-connected components of pixels exceeding the given output threshold, i.e. the application of the output threshold results in the previously mentioned 'islands' of cells and the peak one in each of these is identified. 'Gerig's backmapping' adopts the method of Gerig (1987), where the transform is applied a second time, under the assumption that any concentric circles should be excluded. The locus for each individual edge point is generated again and compared with the corresponding pixel positions in the first pass accumulator array. Only the maximum value from this set of pixels is returned for the second pass accumulator array. Effectively, all but the strongest cell is eliminated from each locus, since it is assumed that a given boundary point should be a member of only one curve in image space, producing a sparse array of pixels which is then amenable to the simple output threshold value set earlier.

hough_circ outputs the final accumulator array, after one or two passes depending on whether or not the backmapping option has been set, an ASCII list of circle centre positions and radii, and an output image of detected circles, which may be added to the original grey scale image for comparison and verification (*pgmarith - add* will perform this task).

7.2.4 Data

TOBI deep-tow, high resolution sidescan sonar data have been gathered over extensive regions of mid-ocean ridge crests and numerous seamounts have been identified and studied, confirming their importance in crustal accretion processes. Datasets collected on the EW9008 cruise to the Reykjanes Ridge, October 1990 (see Parson *et al.*, 1993 for a review), and from the CD65 cruise to the MAR (24-30°N), February 1992 (refer to Cann *et al.*, 1992 for further information), have been selected for use in this preliminary investigation. Chapter 3, section 3.4 describes the nature and preliminary processing steps of these TOBI sidescan sonar data.

7.3 APPLICATIONS

7.3.1 Experiment 1

The Hough transform circle detection technique was first applied to a classic, flat-topped circular seamount observed at approximately 60°N on the Reykjanes Ridge. A 450×450 pixel section was cut out of this raw image (6 m per pixel), using *pgmcut*, and a grey scale enhancement in the form of a gamma correction[1] of 4.0 was applied (*pnmgamma*). This was scaled down, employing *pnmscale*, in order to fit the 200×200 pixel image size (now 13.5 m per pixel) required for the fast and efficient use of the transform for experimental purposes. Figure 7.5a represents this image, entitled 'image 1' for reference purposes; insonification is from the left. The image comprises the classic seamount with a long shadow in the top left hand corner, and a smaller ellipsoid volcanic construct containing a small crater towards the bottom. The bright line, and the enclosed region of 'fuzziness', in the bottom left hand corner are remnant artifacts of the imperfect slant-range correction procedure. The seamount is sited on a relatively smooth, uniformly backscattering surface, which should produce minimum clutter and interference in the edge images.

edge_extract was run on the input greyscale image. The edge strength and edge orientation images are respectively shown in figures 7.5b and c. Incidentally, the edge strength image is artificially bright due to the application of a gamma enhancement of 4.0, purely for display purposes. For clarification, the grey levels around the margins of the major seamount in the edge orientation image can be seen to vary in accordance with figure 7.3.

First, the transform was applied without using any aspect information, for comparison purposes. A single radius of length 33 pixels was selected, with an input threshold of data number (DN) 80 and an arbitrary output threshold as neither of the peak detection options had been set. Figure 7.5d represents the accumulator array, enhanced for clarity with a gamma correction of 7.0; as are all subsequent illustrated arrays. The circular loci clearly overlap at a point corresponding to the centre of the circle which is consistent with the outline of the seamount. The benefits of utilizing the aspect information are illustrated in figure 7.5e. 50° arcs are drawn in the direction of negatively contrasting features. Again, the circle centre is clearly identified, but much less clutter is evident in the remainder of the image. For completeness, a positive contrast option, i.e. 'dark on light', was selected, also specifying a 50° arc length. In figure 7.5f, it can be observed that the loci are drawn

[1] This is the standard photographic contrast parameter (e.g. Clark, 1981).

perpendicularly outwards from the edge of the seamount, and consequently no peak is found.

Due to the inaccuracy in the determination of the radius of a sought circle, and certainly for objective purposes, the maximum number of 5 radii is usually selected. Figure 7.5g is the accumulator array result of selecting radii of 31, 32, 33, 34 and 35 pixels, and choosing the negatively contrasting aspect option. The loci become 'blurred', but the authentic peak is still visible. The generation of a histogram from this array leads to the selection of an output threshold of DN 50 (equivalent to the threshold frequency of occurrence of loci for a particular cell of the accumulator array). *hough_circ* was re-run, choosing Gerig's backmapping option for peak detection. The output accumulator array after two passes of the transform (figure 7.5h), illustrates a sparse array of pixels, to which the output threshold of DN 50 could be applied. The outputted circle image displayed in figure 7.5i comprises 2 overlapping circles, which, when added to the original grey scale image, clearly coincide with the seamount (figure 7.5j).

An attempt was made to see if the Hough transform was capable of identifying a volcanic feature which was not strictly circular. The detection of the small, ellipsoidal seamount in the bottom half of the image involved the application of an input threshold of DN 75, incorporating the lower edge strength of this feature for the transform operation; the selection of the 5 radii between 12 and 16 pixels; and the utilization of the negatively contrasting edge orientation option. Figure 7.5k illustrates the accumulator array after a single pass of the Hough transform. The arcs of the loci associated with the edge of this seamount tend to overlap in approximately 3, rather than the predicted single peak expected from a circular feature; these peaks have an average value of DN 89. This is due to the irregular morphology of the volcano. However, despite the relatively small radii used in the exercise, compared with the radius of the major seamount, a large peak occurs in the accumulator array where the loci overlap around this dominant construct; a pixel value of DN 96 is appropriate for this highest peak. Therefore, despite several selections of output thresholds and peak detection methods, the spurious peak from the large seamount will always be found to be more significant than those consistent with the smaller one.

7.3.2 Experiment 2

Image 2 was selected from approximately 29°12'N, 43°12'W from the CD65 cruise to the MAR. A window of 300 by 300 pixels was selected from the raw data,

Figure 7.5 *See next page. Image 1 (scale of all sub-images is 2.025×2.025 km): (**a**) original grey scale image, gamma = 4.0; (**b**) Sobel derived edge strength image, gamma 4.0; (**c**) Sobel derived edge orientation image; (**d**) accumulator array after first pass of transform using no aspect information and single radius = 33 pixels, gamma 7.0; (**e**) first pass accumulator array, negatively contrasting aspect, 50° arc length, single radius = 33 pixels, input threshold (IT) = DN 80, gamma 7.0; (**f**) first pass accumulator array, positively contrasting aspect, 50° arc length, single radius = 33 pixels, IT = 80, gamma 7.0; (**g**) first pass accumulator array, negatively contrasting aspect, 50° arc length, 5 radii (31-35 pixels), IT = DN 80, gamma 7.0; (**h**) second pass accumulator array after Gerig's backmapping, negatively contrasting aspect, 50° arc length, 5 radii (31-35 pixels), IT = DN 80, output threshold (OT) = DN 50, gamma 7.0; (**i**) detected circle from second pass; (**j**) detected circle added to original grey scale image; and (**k**) first pass accumulator array, negatively contrasting aspect, 50° arc length, 5 radii (12-16 pixels), IT = DN 75, gamma 7.0.*

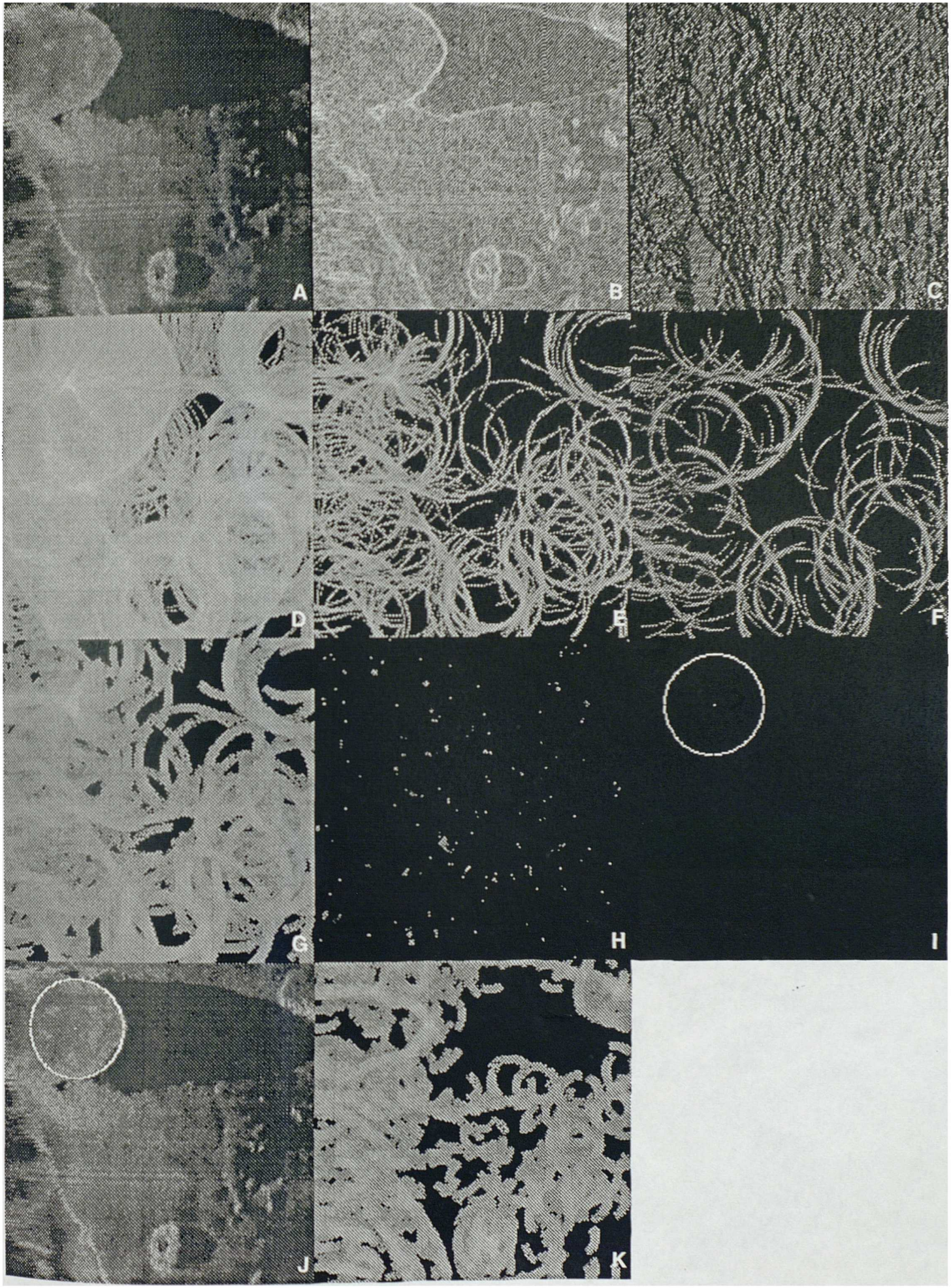

A
B
C
D
E
F
G
H
I
J
K

Figure 7.6 *See next page. Image 2 (scale of all sub-images is 1.8×1.8 km): (**a**) original grey scale image, gamma 4.0; (**b**) Sobel derived edge strength image, gamma 4.0; (**c**) Sobel derived edge orientation image; (**d**) first pass accumulator array, negatively contrasting aspect, 50° arc length, 5 radii (43-47 pixels), IT = DN 30, gamma 7.0; (**e**) second pass accumulator array, negatively contrasting aspect, 50° arc length, 5 radii (43-47 pixels), IT = DN 30, OT = DN 62, gamma 7.0; (**f**) detected circle from second pass; and (**g**) detected circle added to original grey scale image.*

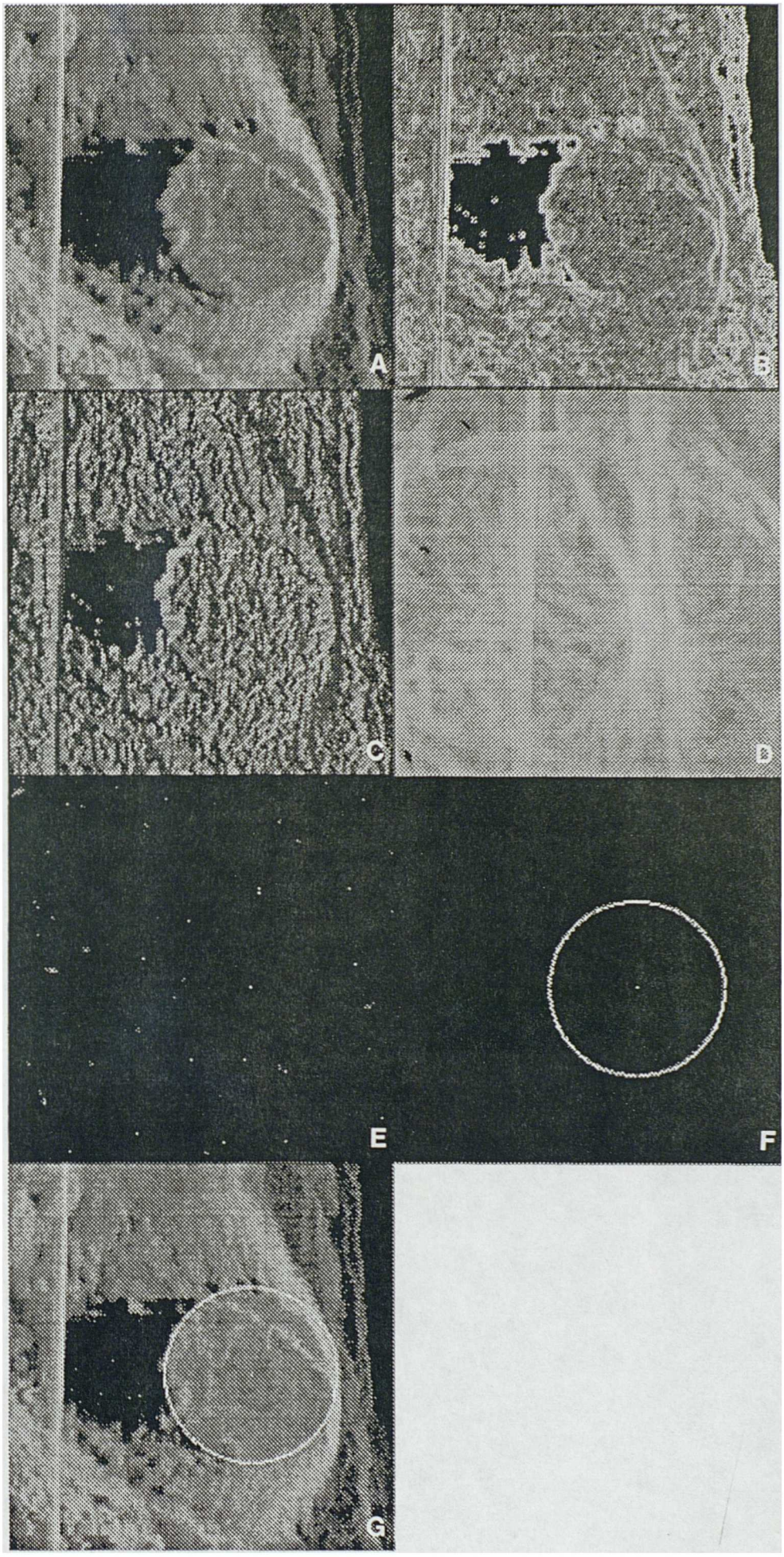
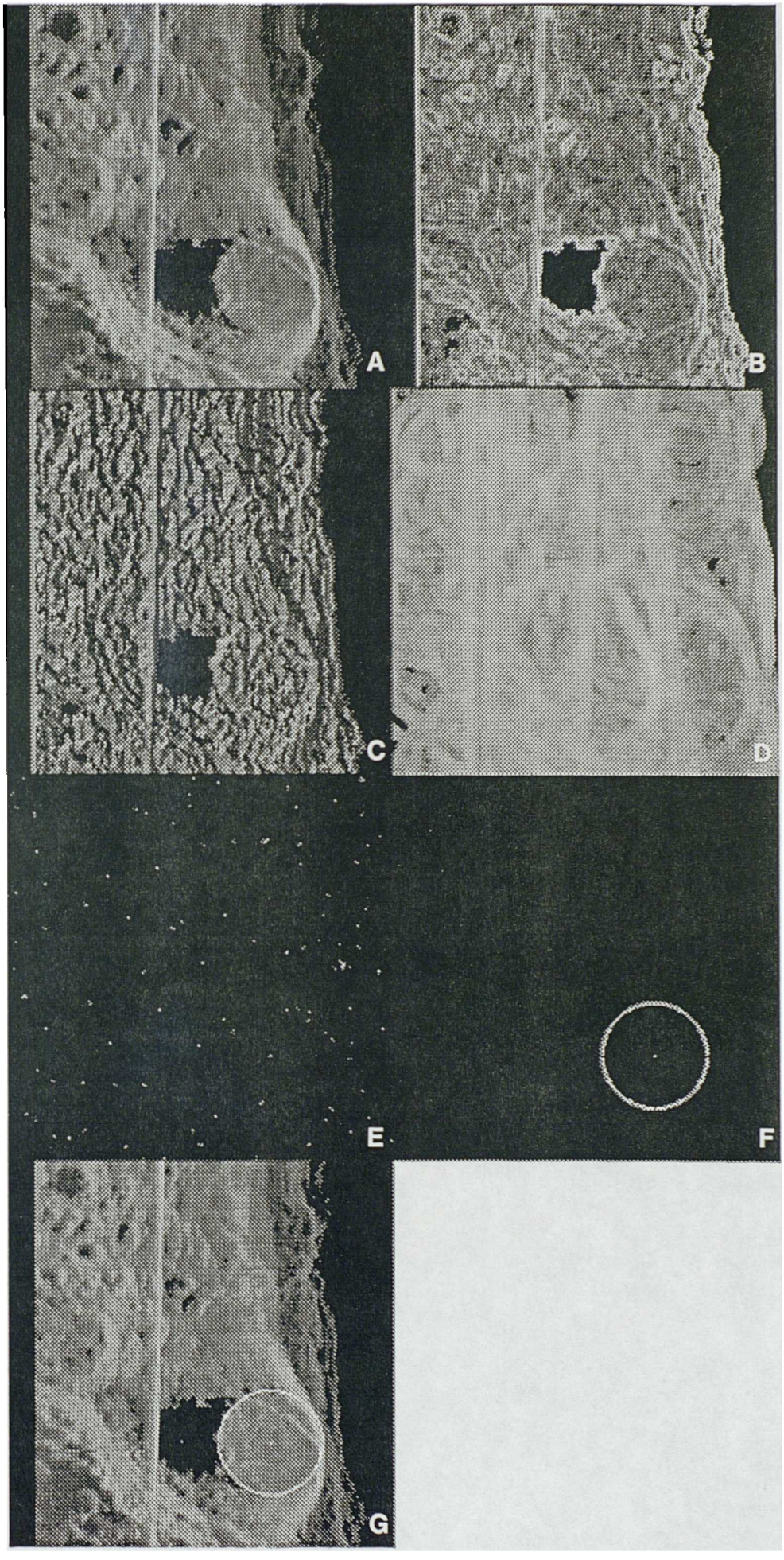

A
B
C
D
E
F
G

Figure 7.7 *See next page. Image 2 (scale of all sub-images is 3×3 km): (**a**) original grey scale image, gamma 4.0; (**b**) Sobel derived edge strength image, gamma 4.0; (**c**) Sobel derived edge orientation image; (**d**) first pass accumulator array, negatively contrasting aspect, 50° arc length, 5 radii (25-29 pixels), IT = DN 30, gamma 7.0; (**e**) second pass accumulator array, negatively contrasting aspect, 50° arc length, 5 radii (25-29 pixels), IT = DN 30, OT = DN 70, gamma 7.0; (**f**) detected circle from second pass; and (**g**) detected circle added to original grey scale image.*


A
B
C
D
E
F
G

and was reduced to the necessary 200x200 pixels (now 9 m per pixel), followed by a gamma enhancement of 4.0. In figure 7.6a, insonification is from the right, where a loss of pixels also occurs due to a temporary failure of the bottom detection method. This large, flat-topped seamount has a more irregular and less strong edge than the seamount in image 1 (see figures 7.6b and c). The line running vertically in the left of the image is a sea-surface reflection arrival. An input threshold of DN 30; radii of 43, 44, 45, 46 and 47 pixels; and a 50° arc length for negatively contrasting objects were selected for the first run of the program, resulting in the accumulator array view in figure 7.6d. A peak consistent with the seamount can be seen, but also arcs forming vertical lines a radius' length away from the edges formed by the multiple reflection and the near-nadir loss of data. However, the choice of an output threshold of DN 62 and Gerig's backmapping detected the circle associated with this seamount (figures 7.6e, f and g).

The transform was additionally applied to a smaller scale view of the seamount; this time a 500x500 pixel section was cut and scaled down from the raw data (to 15 m per pixel), then enhanced with a gamma factor of 4.0 (figure 7.6). Here, extra complications are introduced into the image (figure 7.7a, b and c); a black band is evident down the left hand side where there have been no sidescan returns, due to the limit of data collection; plus hummocky, volcanic terrain is introduced into the environs. However, the operation was once again successful, after applying the following parameters: the input threshold was set as DN 30; the radii became 25, 26, 27, 28, 29; the output threshold was selected as DN 70; and the negatively contrasting edge orientation and Gerig's backmapping options were chosen. Figures 7.7d, e, f and g respectively correspond to the accumulator array after the first then second passes, the detected circle and the combination of the detected circle and the original image.

7.3.3 Experiment 3

Image 3 was selected from approximately 24°54'N, 45°30'W from the CD65 cruise to the MAR. A 300x300 pixel window was cut out and scaled down to 200x200 pixels (9 m per pixel), the required size for input to *hough_circ*, and then enhanced with a gamma factor of 4.0. Figure 7.8a illustrates that insonification is from the right, highlighting a rather indistinct, irregularly edged seamount, with the only really clear-cut edge adjacent to the shadow (see figure 7.8b and c). An input threshold of DN 40 was selected, with 5 radii ranging from 50 to 54 pixels. The aspect information was utilized, choosing the negatively contrasting option. Figures 7.8d and e represent the accumulator array after the first and second passes of the

Figure 7.8 *See next page. Image 3 (scale of all sub-images is 1.8×1.8 km): (**a**) original grey scale image, gamma 4.0; (**b**) Sobel derived edge strength image, gamma 4.0; (**c**) Sobel derived edge orientation image; (**d**) first pass accumulator array, negatively contrasting aspect, 50° arc length, 5 radii (50-54 pixels), IT = DN 40, gamma 7.0; (**e**) second pass accumulator array, negatively contrasting aspect, 50° arc length, 5 radii (50-54 pixels), IT = DN 40, OT = DN 41, gamma 7.0; (**f**) detected circle from second pass; and (**g**) detected circle added to original grey scale image.*

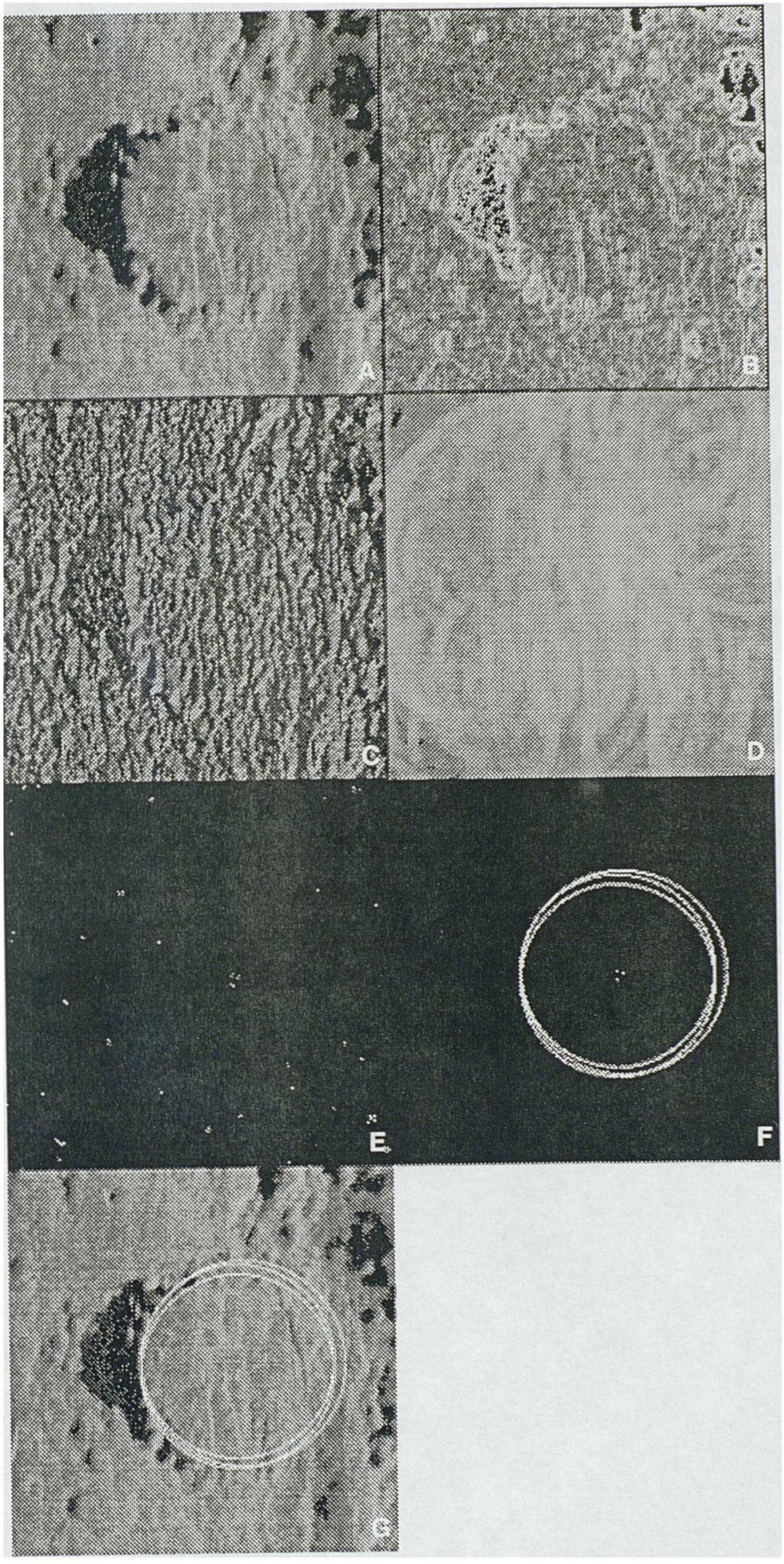
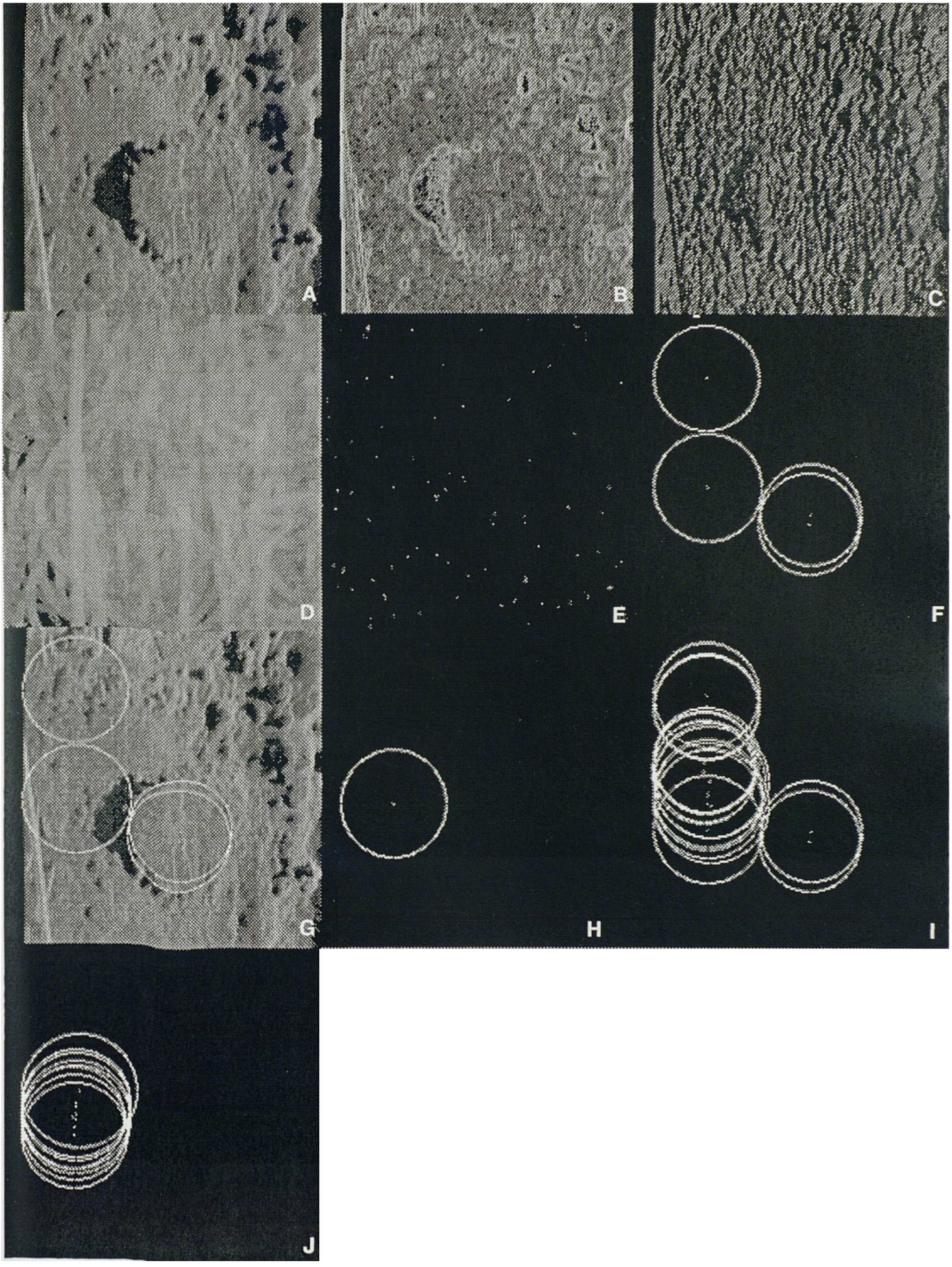

A
B
C
D
E
F
G

Figure 7.9 *See next page. Image 3 (scale of all sub-images is 2.4×2.4 km): (**a**) original grey scale image, gamma 4.0; (**b**) Sobel derived edge strength image, gamma 4.0; (**c**) Sobel derived edge orientation image; (**d**) first pass accumulator array, negatively contrasting aspect, 50° arc length, 5 radii (32-36 pixels), IT = DN 40, gamma 7.0; (**e**) second pass accumulator array, negatively contrasting aspect, 50° arc length, 5 radii (32-36 pixels), IT = DN 40, gamma 7.0; (**f**) detected circles from second pass, OT = DN 40; (**g**) detected circles from second pass, OT = DN 40, added to original grey scale image; (**h**) detected circles from second pass, OT = DN 45; (**i**) detected circles from connected-component peaks identification method, OT = DN 40; and (**j**) detected circles from connected-component peaks identification method, OT = DN 45.*


A
B
C
D
E
F
G
H
I
J

Hough transform. No obvious peak is observed in figure 7.8d; however the application of an output threshold of DN 41 to figure 7.8e, leads to the detection of the 3 overlapping circles viewed in figures 7.8f and g. The edges of these circles coincide on the strong left hand edge of the feature, but diverge according to their various radii on the right hand side, where no distinctive edge is identified.

A broader view of this region is displayed in figure 7.9a, where a region of size 400x400 pixels was extracted from the full resolution image and subsequently scaled down to the necessary 200x200 pixels (12 m per pixel), after applying a gamma enhancement of 4.0. The black band running down the left hand side of the image represents the limit of data recorded. A sea surface reflection skirts the bottom left hand corner of the image. Figures 7.9b and c illustrate the Sobel derived edge strength and edge orientation images. An input threshold of 40 and 5 radii ranging from 32 to 36 pixels were set, and the transform was performed once; figure 7.9d displays the results. A strong band of overlapping loci is revealed, running from top to bottom of the array, representing the results of the transform on the strong edge at the limit of the data. A slight, but not obvious, peak associated with the centre of the seamount may also be perceived. Figures 7.9e, f, g and h illustrate the results of a second pass of the transform, employing Gerig's method for peak detection. Figures 7.9f and g represent the application of an output threshold of DN 40, while figure 7.9h shows the effect of a threshold setting of DN 45. It is clear from figure 7.9f, that although two overlapping circles associated with the seamount are detected, an additional two spurious ones are also evident; the application of the higher threshold eliminates the authentic circles, but still captures one of the false ones (figure 7.9h). Observing the actual pixel values of the accumulator array quantified the pixel values of the two peaks coincident with the centre of the seamount as DN 41, while the spurious peaks had values of DN 42 and 49. The utilisation of the connected component peaks identification method for circle detection, assigning respective thresholds of DN 40 and 45 to figures 7.9i and j, yielded still more spurious peaks, consistent with the band of loci formed by the transform of the strong edge at the limit of the sidescan returns.

7.3.4 Experiment 4

Experiment 2 successfully detected the seamount situated at approximately 29°12'N, 43°12'W on the MAR. In order for the Hough transform to be practically useful, it must be capable of being applied to an arbitrary image, knowing no previous information, and then to successfully identify all the circular seamounts present. TOBI data is typically archived into files containing approximately 4000

Figure 7.10 *Image 4 (scale of all images is 6×24 km): (a) original grey scale image, gamma = 4.0; (b) Sobel derived edge strength image, gamma 4.0; (c) Sobel derived edge orientation image; and (d) accumulator array after first pass of transform, negatively contrasting aspect, 50° arc length, 5 radii (11-15 pixels), IT = DN 30, gamma 7.0.*

records. For a reasonable program run time, these have been scaled down to 200 samples by 800 lines; this was one of the standard output formats for the cruise CD65 data.

The seamount of image 2 is situated in the bottom left hand corner of figure 7.10a. The top half of the image comprises numerous strong linear features; both elongate volcanic constructs and faults. The lower portion of the image possesses more uniformly backscattering properties, representing smoother terrain. Four major artifacts can be identified; sea-surface reflections, more distinctive on the left side than the right; limit of data at both port and starboard edges of the image; extensive near-nadir data loss particularly in the lower portion of the image due to TOBI being towed at a higher than normal working altitude; and the 'bending' of some of the geological features, two-thirds of the way down the left hand side of the image. The latter two artifacts are due to the turning of the vehicle.

Appropriate alterations were completed for *edge_extract* and *hough_circ* to accommodate the new input file sizes. The edge images were derived (see figure 7.10b and c) and the Hough transform was implemented using an input threshold of 30 and 5 radii between 11 and 15. The results of the first pass of the transform are illustrated in figure 7.10d; a peak can be identified coincident with the seamount centre. However, there are also many other peaks, often running vertically through the image as the result of transforming some of the linear geological features and artifacts. After a second pass of the transform, the authentic peak was found to have a pixel value of DN 111, but there were an additional four spurious peaks exceeding this value, up to a maximum value of DN 117.

7.4 CONCLUSIONS

The application of the Hough transform circle detection technique to small selected regions of TOBI data encompassing an obvious circular seamount was clearly successful in experiments 1 and 2. Image 2 contained some artifacts, but these did not interfere with the results. In experiment 3, the transform was applied to a less obvious feature. The circular outline was successfully detected in a close up view, but when artifacts are introduced in a wider view, they produced a higher pixel value in the second pass accumulator array than the authentic peak.

Ideally, an automatic circle detection routine should be able to be applied to an arbitrary image and successfully identify all circular features present. This was attempted in experiment 4. After two passes of the transform, the peak in the accumulator array associated with the known seamount was observed, but an additional 4 spurious peaks with higher pixel values were also generated. Attempts could be made to remove the effects of these artifacts, for example, losses of data could be filled in with an average pixel value or the texture signature of the surroundings. However, the nature of the geological data, for example faults, generate stronger edges than the weaker bounds associated with a seamount. Despite the fact that the transform is designed to search for circular features, the transformed loci of the strong uninterrupted linear edges still overlap to produce higher peaks than the weaker margin of the circular seamount.

Most object recognition routines are presented with similar problems when there are heterogeneous backgrounds to the objects of interest. A segmentation of backgrounds, such as perfomed by Linnett *et al.* (1993), prior to object recognition (using the fractal texture analysis techniques) may help, though the complicated terrain encountered here make matters very difficult. Another idea may be to implement a windowing operation over a swath, so just a small section of data is consulted each time the Hough transform is applied, rather like the first experiments shown in this exercise. However, the subjective nature of the selection of input and output thresholds may still render the technique to be of little practical use.

The technique is also limited to approximately circular seamounts. Some seamounts will be slightly non-circular, and this may lead to the detection of overlapping circles, e.g. experiments 1 and 3. However, the application of the transform to a definitely ellipsoid feature, such as the small seamount in experiment 1, generates a number of peaks rather than just one in the accumulator array, with consequently lower pixel values; the Hough transform using the expression for an ellipse would be more appropriate here. However, Smith and Cann (1992) require a seamount to possess a maximum aspect ratio of 2.

Another problem is the manner in which the circle detection routine operates; it is not strictly objective. *hough_circ* is confined to searching for only 5 radii per run, in order to reduce memory requirements. The program may be repeated a number of times to cover all the possible radii that are sought, but this will lead to alternative problems. The selection of the input and output thresholds introduce a subjective character into the detection operation. The input threshold was chosen for

experimental purposes from an interactive thresholding operation, but it has been suggested (Cross, 1988) that the threshold can be set such that it captures the tail of the negatively skewed distribution of the input edge strength image; however this is still highly subjective (see figure 7.11).

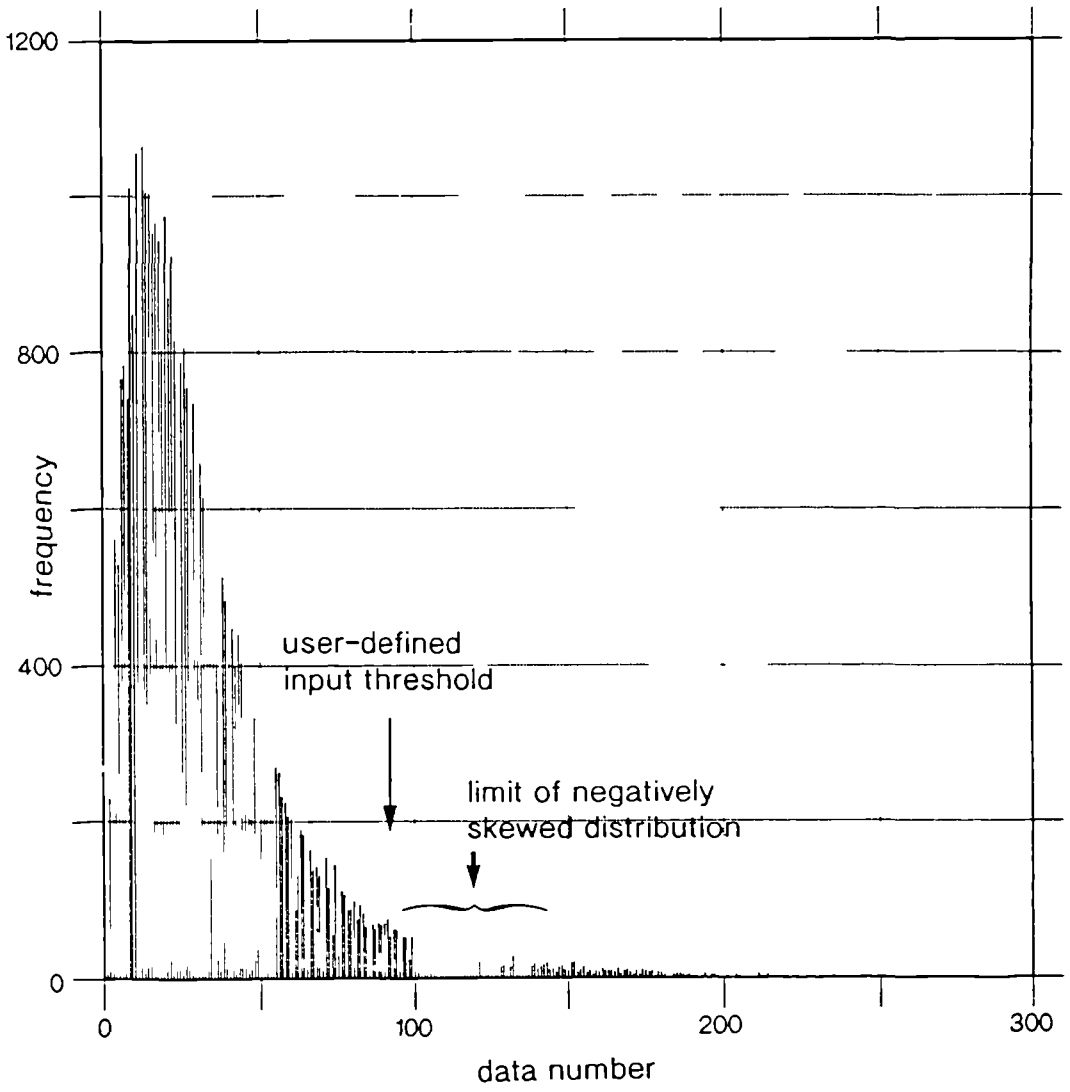


Figure 7.11 Histogram of the Sobel derived edge strength image from image 1.

For these trials, the output threshold was subjectively selected as a value at the low frequency end of the histogram, fine tuning then provided the authentic circles, eliminating the invalid ones. Other methods could involve the selection of a set number of the maximum peaks in a second pass accumulator array. However, all these techniques would run into problems when the program is repeated to account for a wide range of potential radii; it would be unlikely that there would be genuine circles present for every range of 5 radii. The accumulator array could be

cumulatively summed after each program run, but the generation of spurious peaks would become an even greater problem. *hist 1* and *2* in figure 7.12 are the histograms generated from the first pass accumulator arrays from experiment 1, for respective radii ranges of 31-35 and 12-16 pixels. *hist 1+2* represents the histogram of the sum of the two arrays. The frequency of pixels in the tail of the negatively skewed distribution after the sum of just two arrays is greater than for the individual arrays. This would have the effect of 'blurring' the output peaks and introducing a greater number of spurious cells.

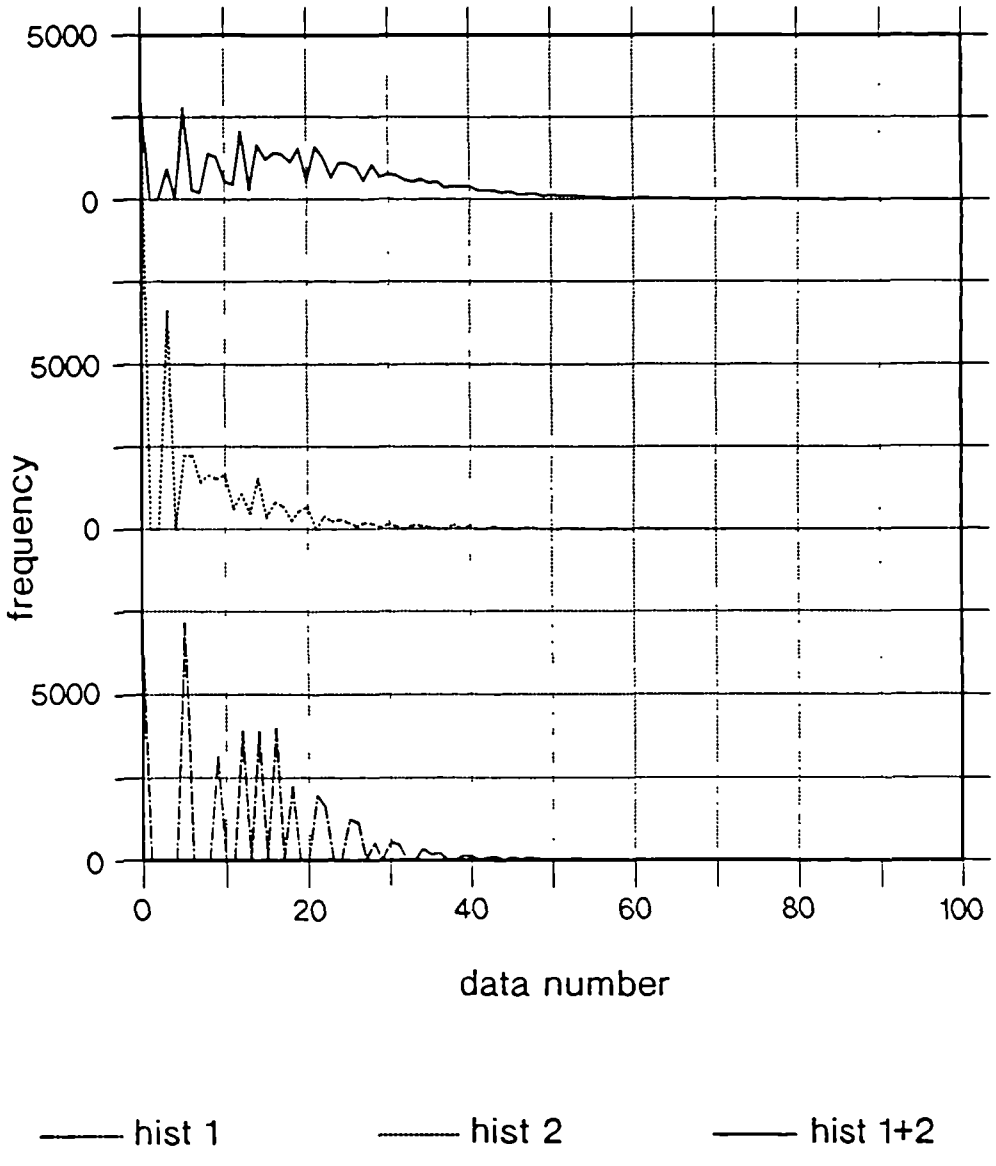


Figure 7.12 *hist 1 - histogram of first pass accumulator array from image 1, radii of 31-35 pixels; hist 2 - histogram of first pass accumulator array from image 1, radii of 12-16 pixels; hist 1+2 - histogram of the arithmetic sum of the two arrays.*

7.5 FUTURE USE OF TECHNIQUE

The transform does identify seamounts from TOBI data, providing that there is limited interference from other geological features or artifacts. As an automatic picking technique, the complex nature of the TOBI data over mid-ocean ridges introduces too much 'noise'. The transform is unable to eliminate the effects of non-circular features if their edges are considerably stronger than those associated with the seamounts. Additionally, the selection of the thresholds for the program is somewhat subjective. In conclusion, as an automatic detection routine for the identification of seamounts from TOBI data, the Hough transform will be of little use unless one can establish image-independent criteria for parameter selection. The only possibility for future applications would be to somehow mask out the background, but not the seamount information (possibly by some form of azimuth filtering?), or utilise some sort of windowing implementation; though the subjective nature of peak detections would not necessarily be adequate even then.

Gridded multibeam bathymetry data are another useful source of marine geological information, and can be used for the identification of seamounts, e.g. Shen and Forsyth (1991). Advantages over TOBI data include the fact that this data is far simpler since it comprises actual depth rather than backscatter values, discounting problems caused by shadows. Large faults are still more likely to produce stronger edges than seamounts. The resolution of multibeam data is lower than TOBI sidescan; a typical gridding interval of 100 metres compared with the approximate 6 metre pixel size of TOBI. One must be aware that the variety of gridding processes that can be applied to multibeam data tend to overemphasize the presence of circular features.

CHAPTER 8
CONCLUSIONS

The objectives of this thesis are outlined in section 1.2 of chapter 1. The first priority was to develop a processing procedure for the EM12 backscatter data and investigate appropriate visualisation and data management techniques. Section 2.1 of chapter 2 summarises the possible methods for quantitatively characterising sidescan data collected by both multibeam echosounders and sidescan sonars. For backscatter strengths measured by multibeam echosounders, four possible analytical approaches were outlined, though only two of these were possible avenues for the nature of the EM12 backscatter data: analysis of actual backscatter coefficients and shape of the backscatter angular dependence functions with relation to bottom type. Four methodologies were also itemised for traditional sidescan sonar data; two of these principles were investigated with the TOBI data: texture analysis and object recognition. The aim of this thesis, in addition to optimising the data recorded by these advanced instruments in terms of quantitative analysis, is to keep in mind their application: the characterisation of mid-ocean ridges. Each quantitative method investigated is therefore closely linked to the geological lithologies, morphologies and processes found at mid-ocean ridges, thus providing useful tools for enhancing and easing geological interpretation.

Specific conclusions are provided at the end of each chapter, however a brief summary is designated here also for each research chapter, trying to link the findings into a broader context. Finally, some suggestions for future work will be made.

8.1 SUMMARY AND CONCLUSIONS

Chapter 4 describes the processing procedures developed for the EM12 backscatter data, with the principle aim of generating a geometrically correct acoustic backscatter mosaic. These comprised new front-end software and incorporation into a commercially available, land-based image processing package. Although this software was obviously designed for different applications, it was adapted and utilised for mosaicking, some of the noise reduction routines and visualisation. The system also stored data under GIS principles, which led to the informative inclusions of bathymetry data, shaded relief images, and vector

information, such as ship tracks, annotations etc. Unfortunately, since the backscatter recording format changed significantly for the latest cruise, it was deemed to be more efficient within the time available to use software developed elsewhere for the initial stages of processing the second EM12 data set. Novel techniques were developed for cleaning up the data that was badly affected by poor weather conditions experienced during both surveys.

Chapter 5 describes the concept for analysing the backscatter strengths recorded by the EM12 with its co-registered bathymetry data, as the selection of small experimental sites to investigate how backscatter strengths are affected by bottom type, slope distributions and grazing angles. Three of the sites were interpreted as being likely to comprise homogeneous bottom types (volcanics, landslide and sediments). Absolute backscatter strength differences and the variability of the characteristics of the backscatter angular responses were identified between bottom types, based on the effects of both acoustic impedance and microroughness of the seafloor. Since mid-ocean ridges are generally not formed of distinct blocks of single bottom types and slopes, but complex interactions from large to small scale, five of the experimental sites comprised heterogeneous terrains (faults and evolving AVRs). The same analytical procedures were followed for these data; the backscatter versus slope complement responses showed how the backscatter strengths were affected by slope distributions, and single pixel, across-track profiles were also particularly valuable. It was difficult to derive precise conclusions regarding backscatter variabilities over faulted terrain, because it is composed of very heterogeneous bottom types and the isolation of backscatter dependencies on grazing angle and slope was difficult. However, important points were stated regarding the importance of determining grazing angle distributions prior to backscatter interpretations from these dominantly low incident angle systems. In particular, high backscatter was found to be associated with severely limited portions of fault scarps (usually the steepest part), significantly underestimating their widths. Another conclusion was that the geological evolution of AVRs could be related to the backscatter and bathymetry relationships.

In order to achieve a complete classification of bottom types, there is a dichotomy of whether to normalise backscatter angular dependencies or whether to use the shape of these curves for classification, though chapter 5 presented a tentative normalisation of backscatter was attempted based on the response curve for the median valley volcanics, to yield maps of actual backscatter coefficients. From these data, it appears that neither approach would be sufficient on its own, even with

first-order backscatter statistics as well. Extra information from both the backscatter and bathymetry information would be required. The data also provided an important addition to the library of published backscatter strength responses.

Chapter 6 deduced that the conceptually simple fractal texture analysis techniques were more successful than the GLCM method for classifying TOBI mid-ocean ridge sidescan imagery. The fractal approach was able to distinguish the principle bottom types that would be demarcated by geological interpretation, improving the results achieved with the acquired software by the introduction of an eighth feature vector. The technique enabled automatic mapping of volcanic morphologies within TOBI swaths; the distribution of these morphologies are important geological indicators in terms of crustal accretion and segmentation characteristics. It was deduced that particularly complicated or ambiguous areas will always require geological knowledge (such as tectonic settings etc.) for discrimination.

In chapter 7, the detection of seamounts from TOBI data using the Hough transform was not recommended as a potentially practicable tool, unless techniques can be developed to mask out background interference, principally from linear features. An objective method must also be sought for selecting the input parameters for the transform operation, otherwise the application of the technique as an automatic operator would be of little use.

Summarising, various quantitative analysis methods have been explored for both bathymetry and sidescan data, yielding corresponding geological applications. The characterisation techniques have aimed to optimise the information available from the surveying instruments.

8.2 FUTURE WORK

Some suggestions for the future development of the image processing and characterisation techniques researched in this thesis will now be presented. These include both straightforward extensions to the work, and also how additional types of data would improve the results. Future analyses of EM12 data would benefit from better quality data. The list is roughly prioritised.

1. Apply fractal technique to bathymetry data to assess its suitability as a roughness descriptor.

2. Combine EM12 analytical results with other characterisation methods (probably of bathymetry).

3. Model EM12 backscatter responses (possibly following Dziak *et al.*'s (1993) method for small scale roughness estimations).

4. Combine fractal texture analysis technique with object detection routines (extra ground truthing and co-registered similar scale bathymetry would improve the results extractable from the data).

5. Additional ground truth (e.g. sediment coring and photographic observations) and better quality data would improve and validate EM12 results.

6. Further investigations into the benefits and applications of GIS data management principles for marine data.

7. More easily portable front-end processing procedure, encompassing the revised backscatter formats.

8. Apply the Hough transform to bathymetry data (care with gridding procedures, and investigate image-independent criteria for parameter settings).

As a final consideration for future applications, this thesis has presented a number of techniques which can or at least have the potential of quantitatively characterising various aspects of mid-ocean ridge terrains. Generally, each of these characterisations alone does not provide a complete classification, though individual results can be very useful; instead, each of these characterisations can effectively be considered as a 'feature vector'. With present seafloor investigation focussing on many different types of survey concentrating on specific geological areas of interest, the GIS principle of data management becomes essential. If each of the data types can be characterised in their own way, possessing associated 'feature vector' images, then these may be combined and classified using multi-band classification methods. These techniques have been extensively researched by the terrestrial remote sensing groups, and are widely available through the commercial and academic communities.

Investigations should be made into the advantages and disadvantages of characterising gridded or ping-by-ping data; the former could be more readily applied using existing algorithms and software, while the latter would be likely to be more computationally demanding, but reserves important information about noise levels etc. An optimum output would be a reliable seafloor characterisation, based on all possible data and physical characteristics of the seafloor. The last note will be devoted to the value of traditional geological interpretation. The above methods are likely to ease and enhance such interpretations, but will never and do not aim to replace the essential knowledge gained by understanding the geological processes at work in an environment.

REFERENCES

Ackers, M., R.C. Searle, J.A. Keeton, J.R. Cann, E. Valsami, and R.M.M. George, Topographic analysis and development of the Mid-Atlantic Ridge median valley, 45°30'N, since 2 Ma, *EOS Trans. AGU, 73,* 552.

Alexandrou, D., C. de Moustier, and G. Haralabus, Evaluation and verification of bottom acoustic reverberation statistics predicted by the point scattering model, *Journal of the Acoustical Society of America, 91,* 1403-1413, 1992.

Algazi, V.R., and G.E. Ford, Radiometric equalization of nonperiodic striping in satellite data, *Computer graphics and image processing, 16,* 287-295, 1981.

Aumento, F., B.D. Loncarevic, and D.I. Ross, IV. Regional studies, Hudson Geotraverse: geology of the Mid-Atlantic ridge at 45°N, *Phil. Trans. Roy. Soc. Lond. A., 268,* 623-650, 1971.

Ballard, R.D., and T.H. van Andel, Morphology and tectonics of the inner rift valley at lat 36°50'N on the Mid-Atlantic Ridge, *Geological Society of America Bulletin, 88,* 507-530, 1977.

Balsam, W.L., Sediment accumulation rates west of the Mid-Atlantic Ridge (~35°N), *Marine Geology, 81,* 1-13, 1988.

Bell, T.H., Statistical features of sea-floor topography, *Deep-Sea Research, 22,* 883-892, 1975.

Berkson, J.M., and J.E. Matthews, Statistical properties of sea-floor roughness, in *Acoustics and the Seabed,* edited by N. Pace, Bath University Press, 215-223, 1983.

Blackinton, J.G., D.M. Hussong, and J.G. Kosalos, First results from a combination side-scan and seafloor mapping system (Sea MARC II), *Proc. Offshore Technology Conf., OTC No. 4478, 1,* 307-314, 1983.

Burrough, P.A., Fractal dimensions of landscapes and other environmental data, *Nature, 294,* 240-242, 1981.

Cann, J.R., J. Tarney, J. Varet, and D.A. Wood, Mantle heterogeneity in the North Atlantic: evidence from Leg 49 geochemistry, in *Initial Reports of DSDP, 49,* edited by B.P. Luyendyk and J.R. Cann, 841-850, 1978.

Cann, J.R., D.K. Smith, M.E. Dougherty, J. Lin, S. Spencer, C.J. MacLeod, E. McAllister, J.A. Keeton, S. Garland, B. Brooks and R. Pascoe, Building the crust at the Mid-Atlantic Ridge, *RRS Charles Darwin Cruise 65 Report,* 1992.

Caruthers, J.W., R.S. Keiffer, and J.C. Novarini, Near-field acoustic scattering from simulated two-dimensional, wind-driven sea surfaces, *Journal of the Acoustical Society of America, 91,* 813-822, 1992.

Caruthers, J.W., and J.C. Novarini, Modelling swath bathymetry/sidescan sonar image returns, *Acoustic classification and mapping of the seabed,* in Proceedings of the Institute of Acoustics, 15(2), 261-269, 1993.

Cervenka, P., and C. de Moustier, Sidescan sonar image processing techniques, *IEEE Journal of Oceanic Engineering, 18,* 108-122, 1993.

Chavez, P.S., Processing techniques for digital sonar images from GLORIA, *Photogrammetric Engineering and Remote Sensing, 52,* 1133-1145, 1986.

Chotiros, N.P., Signal processing for wide-beam normal incidence sediment classification, *Acoustic classification and mapping of the seabed,* in Proceedings of the Institute of Acoustics, 15(2), 123-130, 1993.

Clark, D.R., *Computers for imagemaking,* Pergamon Press, 8-9, 1981.

Clay, C.S., and H. Medwin, Acoustic estimates of topography and roughness spectrum of the sea floor southwest of Iberian Peninsula, in *Physics of Sound in Marine Sediments,* edited by L. Hampton, Plenum, New York, 1974.

Cobra, D.T., Oppenheim, A.V., and J.S. Jaffe, Geometric distortions in side-scan sonar images: a procedure for their estimation and correction, *IEEE Journal of Oceanic Engineering, 17,* 252-268, 1992.

Cross, A.M., Detection of circular features using the Hough transform, *International Journal of Remote Sensing, 9,* 1519-1528, 1988.

Davies, E.R., Design of cost effective systems for the inspection of certain food products during manufacture, *Proceedings of the 4th Conference on Robot Vision and Sensory Control*, 437-446, Bedford, England: IFS, 1984.

Davis, E.E., Currie, R.G., Sawyer, B.S., and J.G. Kosalos, The use of swath bathymetric and acoustic image mapping tools in marine geoscience, *Marine Technology Society Journal, 20,* 17-27, 1987.

Davis, J.C., *Statistics and data analysis in geology,* John Wiley and Sons, chapter 7, 1973.

Davis, R., Zisk, S., Simpson, M., Edwards, M., Shor, A., and E. Halter, Hawaii Mapping Research Group bathymetric and sidescan data processing, *Proceedings, Oceans '93,* in press, 1993.

de Moustier, C., Beyond bathymetry: mapping acoustic backscattering from the deep seafloor with Sea Beam, *Journal of the Acoustical Society of America, 79,* 316-331, 1986.

de Moustier, C., State of the art in swath bathymetry survey systems, *International Hydrographic Review, Monaco, LXV(2),* 1988.

de Moustier, C., and M.C. Kleinrock, Bathymetric artifacts in Sea Beam data: how to recognize them and what causes them, *Journal of Geophysical Research, 91,* 3407-3424, 1986.

de Moustier, C., and D. Alexandrou, Angular dependence of 12-kHz seafloor acoustic backscatter, *Journal of the Acoustical Society of America, 90,* 522-531, 1991.

de Moustier, C., and H. Matsumoto, Seafloor acoustic remote sensing with multibeam echo-sounders and bathymetric sidescan sonar systems, *Marine Geophysical Researches, 15,* 27-42, 1993.

Duda, R.O., and P.E. Hart, Use of the Hough transform to detect lines and curves in pictures, *Communications of the ACM, 15,* 11-15, 1972.

Duda, R.O., and P.E. Hart, *Pattern classification and scene analysis,* John Wiley and Sons, chapter 2, 1973.

Dziak, R.P., H. Matsumoto, and C.G. Fox, Estimation of sea-floor roughness spectral parameters from multibeam sonar acoustic backscatter data - Axial Seamount, Juan de Fuca Ridge, *Geophysical Research Letters, 20,* 1863-1866, 1993.

Edwards, M.H., R.E. Arvidson and E.A. Guinness, Digital image processing of Seabeam bathymetric data for structural studies of seamounts near the East Pacific Rise, *Journal of Geophysical Research, 89,* 11108-11116, 1984.

Flewellen, C., Millard, N., and I. Rouse, TOBI, a vehicle for deep ocean survey, *Electronics and Communication Engineering Journal,* 85-93, April 1993.

Fornari, D.J., R. Batiza, and M.A. Luckman, Seamount abundances and distribution near the East Pacific Rise 0°-24°N based on Sea Beam data, in *Seamounts, Islands and Atolls, AGU Mon., 43,* edited by Keating *et al.,* 13-21, 1987.

Fowler, C.M.R., The Mid-Atlantic Ridge: structure at 45°N, *Geophys. J. R. astr. Soc., 54,* 167-183, 1978.

Fox, C.G., and D.E. Hayes, Quantitative methods for analyzing the roughness of the seafloor, *Reviews of Geophysics, 23,* 1-48, 1985.

Frost, V.S., J.A. Stiles, K.S. Shanmugan, and J.C. Holtzman, A model for radar images and its application to adaptive digital filtering of multiplicative noise, *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-4,* 15-166, 1982.

Gardner, J.V., M.E. Field, H. Lee, and B.E. Edwards, Ground-truthing 6.5-kHz side scan sonographs: what are we really imaging?, *Journal of Geophysical Research, 96,* 5955-5974, 1991.

Gerig, G., Limiting image-space and accumulator-space: a new approach for object recognition. *Proceedings of the first International Conference on Computer Vision,* 112-117, Washington, D.C.: I.E.E.E. Computer Society Press, 1987.

Glenn, M.F., Introducing an operational multi-beam array sonar, *International Hydrographic Review, 38,* 35-39, 1970.

Goff, J.A., and T.H. Jordan, Stochastic modeling of seafloor morphology: inversion of Sea Beam data for second-order statistics, *Journal of Geophysical Research, 93,* 13589-13608, 1988.

Goff, J.A., and M.C. Kleinrock, Quantitative comparison of bathymetric survey systems, *Geophysical Research Letters, 18,* 1253-1256, 1991.

Goodman, J.W., Some fundamental properties of speckle, *Journal of the Optical Society of America, 66,* 1145-1150, 1976.

Gutberlet, M., and H. W. Schenke, HYDROSWEEP: new era in high precision bathymetric surveying in deep and shallow water, *Marine Geodesy, 13,* 1-23, 1989.

Hagen, R.A., N.A. Baker, D.F. Naar, and R.N. Hey, 1990, A SeaMARC II survey of recent submarine volcanism near Easter Island, *Marine Geophysical Researches, 12,* 297, 1990.

Hamilton, E.L., Reflection coefficients and bottom losses at normal incidence computed from Pacific sediment properties, *Geophysics, 35,* 995-1004, 1970.

Hamilton, E.L., and R.T. Bachman, Sound velocity and related properties of marine sediments, *Journal of the Acoustical Society of America, 72,* 1891-1904, 1982.

Hammerstad, E., and F. Pøhner, Ultra wide swath interferometric multibeam echosounder with sea bottom imaging system, *Proceedings, Oceans '90,* 63-68, 1990.

Hammerstad, E., F. Pøhner, F. Parthiot, and J. Bennett, Field testing of a new deep water multibeam echo sounder, *Proceedings, Oceans '91,* 743-749, 1991.

Haralick, R.M., K. Shanmugam, and I. Dinstein, Textural features for image classification, *IEEE Transactions on Systems, Man, and Cybernetics, SMC-3,* 610-621, 1973.

Hartigan, J.A., *Clustering Algorithms,* Wiley Interscience, New York, 1975.

Hodgson, M.E., Reducing the computational requirements of the minimum distance classifier, *Remote Sensing of the Environment, 25,* 117-128, 1988.

Horn, B.K.P., and R.J. Woodham, Destriping LANDSAT MSS images by histogram modification, *Computer graphics and image processing, 10,* 69-83, 1979.

Hughes Clarke, J.E., The potential for seabed classification using backscatter from shallow water multibeam sonars, *Acoustic classification and mapping of the seabed,* in Proceedings of the Institute of Acoustics, 15(2), 381-388, 1993.

Hughes Clarke, J.E., Mayer, L.A., Mitchell, N.C., Godin, A., and G. Costello, Processing and interpretation of 95 kHz backscatter data from shallow-water multibeam sonars, *Proceedings, Oceans '93,* in press, 1993.

Jackson, D.R., D.P. Winebrenner, and A. Ishimaru, Application of the composite roughness model to high-frequency bottom backscattering, *Journal of the Acoustical Society of America, 79,* 1410-1422, 1986.

Johnson, H.P., and M. Helferty, The geological interpretation of sidescan sonar, *Reviews of Geophysics, 28,* 357-380, 1990.

Jordan, T.H., W. Menard, and D.K. Smith, Density and size distribution of seamounts in the eastern Pacific inferred from wide-beam sounding data, *Journal of Geophysical Research, 88,* 10508-10518, 1983.

Karson, J.A., G. Thompson, S.E. Humphris, J.M. Edmond, W.B. Bryan, J.R. Brown, A.T. Winters, R.A. Pockalny, J.R. Casey, A.C. Campbell, G. Klinkhammer, M.R. Palmer, R.J. Kinzler, and M.M. Sulanowska, Along-axis variations in seafloor spreading in the MARK area, *Nature, 328,* 681-685, 1987.

Kautsky, J., N.K. Nichols, and D.L.B. Jupp, Smoothed histogram modification for image processing, *Computer Vision, Graphics, and Image Processing, 26,* 271-291, 1984.

Keen, M.J., and K.S. Manchester, The Mid-Atlantic Ridge near 45°N. X. Sediment distribution and thickness from seismic reflection profiling, *Canadian Journal of Earth Sciences, 7* , 735-747, 1970.

Keeton, J.A., and D.K. Smith, Comparison of seamount populations from Hydrosweep multibeam data and TOBI deep-towed sidescan sonar data from the Reykjanes Ridge (60°N), *EOS Trans. AGU, 72*, 1991.

Keeton, J.A. and R.C. Searle, Processing of SIMRAD EM12 sonar data from the Mid-Atlantic Ridge (45°30'N), *Acoustic classification and mapping of the seabed*, in Proceedings of the Institute of Acoustics, 15(2), 429-436, 1993.

Keller, J.M., R.M. Crownover, and R.Y. Chen, Characteristics of natural scenes related to the fractal dimension, *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-9,* 621-627, 1987.

Kibblewhite, A.C., Attenuation of sound in marine sediments: a review with emphasis on new low-frequency data, *Journal of the Acoustical Society of America, 86,* 716-738, 1989.

Kimme, C., D. Ballard, and J. Sklansky, Finding circles by an array of accumulators, *Communications of the ACM, 18,* 120-122, 1975.

Kleinrock, M.C., R.N. Hey, and A.E. Theberge, Practical geological comparison of some seafloor survey instruments, *Geophysical Research Letters, 19,* 1407-1410, 1992.

Kong, L.S., R.S. Detrick, P.J. Fox, L.A. Mayer, and W.B.F. Ryan, The morphology and tectonics of the MARK area from Sea Beam and SeaMARC I observations (Mid-Atlantic Ridge 23°N), *Marine Geophysical Researches, 10,* 59-90, 1988.

Laughton, A.S., The first decade of GLORIA, *Journal of Geophysical Research, 86,* 1511-1534, 1981.

Laughton, A.S. and R.C. Searle, Tectonic processes on slow spreading ridges, in *Deep drilling results in the Atlantic Ocean: ocean crust,* edited by M. Talwani et al., 15-32, Washington D.C.: American Geophysical Union, 1979.

Le Bas, T.P., TOBI side-scan processing with edge segmentation, *Acoustic classification and mapping of the seabed*, in Proceedings of the Institute of Acoustics, 15(2), 65-73, 1993.

Lin, J., G.M. Purdy, H. Schouten, and J.-C. Sempéré, Evidence from gravity data for focussed magmatic accretion along the Mid-Atlantic Ridge, *Nature, 344,* 627-632, 1990.

Linnett, L.M., S.J. Clarke, C. Graham, and D.N. Langhorne, Remote sensing of the sea-bed using fractal techniques, *Electronics and Communication Engineering Journal,* 195-203, October 1991.

Linnett, L.M., S.J. Clarke, C.St.J. Reid, and A.D. Tress, Monitoring of the seabed using sidescan sonar and fractal processing, *Proceedings of the Institute of Acoustics, 15(2),* 49-64, 1993.

Loncarevic, B.D., C.S. Mason, and D.H. Matthews, Mid-Atlantic ridge near 45° north I. the median valley, *Canadian Journal of Earth Sciences, 3,* 327-349, 1966.

Lonsdale, P., Structural geomorphology of a fast-spreading rise crest: the East Pacific Rise near 3°25'S, *Marine Geophysical Researches, 3,* 251-293, 1977.

Luyendyk, B.P., and K.C. Macdonald, Physiography and structure of the inner floor of the FAMOUS rift valley: observations with a deep-towed instrument package, *Geological Society of America Bulletin, 88,* 648-663, 1977.

Luyendyk, B.P., and K.C. Macdonald, A geological transect across the crest of the East Pacific Rise at 21°N latitude made from the deep submersible ALVIN, *Marine Geophysical Researches, 7,* 467-488, 1985.

Macdonald, K.C., The crest of the Mid-Atlantic Ridge: models for crustal generation processes and tectonics, in *The geology of North America, volume M: the western North Atlantic region,* edited by P.R. Vogt, and B.E. Tucholke, Geological Society of America, Boulder, 55-57, 1986.

Macdonald, K.C., and P.J. Fox, The axial summit graben and cross-sectional shape of the East Pacific Rise as indicators of axial magma chambers and recent volcanic eruptions, *Earth and Planetary Science Letters, 88,* 119-131, 1988.

Macdonald, K.C., P.J. Fox, L.J. Perram, M.F. Eisen, R.M. Haymon, S.P. Miller, S.M. Carbotte, M.-H. Cormier, and A.N. Shor, A new view of the mid-ocean ridge from the behaviour of ridge-axis discontinuities, *Nature, 335,* 217-225, 1988.

Malinverno, A., Segmentation of topographic profiles of the seafloor based on a self-affine model, *IEEE Journal of Oceanic Engineering, 14,* 348-359, 1989.

Malinverno, A., M.H. Edwards, and W.B.F. Ryan, Processing of SeaMARC swath sonar data, *IEEE Journal of Oceanic Engineering, 15,* 14-23, 1990.

Mandelbrot, B.B., *The fractal geometry of nature,* San Francisco, CA: W.H. Freeman, 1983.

Mason, D.C., T.P. Le Bas, I. Sewell, and C. Angelikaki, Deblurring of GLORIA side-scan sonar images, *Marine Geophysical Researches, 14,* 125-136, 1992.

Mayer, L.A., C. Ware, S.P. Miller, K.C. Macdonald, S. Scheirer, D.J. Wright, and D.W. Forsyth, Interactive 3-D flight through Sea Beam 2000 data from the East Pacific Rise between 15 and 19° South, *EOS Trans. AGU, 74,* 594, 1993.

McDaniel, S.T., and A.D. Gorman, Acoustic and radar sea surface backscatter, *Journal of Geophysical Research, 87,* 4127-4136, 1982.

McDaniel, S.T., and A.D. Gorman, An examination of the composite-roughness scattering model, *Journal of the Acoustical Society of America, 73,* 1476-1486, 1983.

McKinney, C.M., and C.D. Anderson, Measurements of backscattering of sound from the ocean bottom, *Journal of the Acoustical Society of America, 36,* 158-163, 1964.

Midthassel, A., E. Solvberg, and F. Pøhner, Data management of swath sounding systems, *International Hydrographic Review,* 91-116, 1988.

Miller, S.P., 3-D bathymetric imaging: state of the art visualization, *Sea Technology,* 27-32, June 1991.

Miller, S.P., K.C. Macdonald, T. Atwater, *et al.,* Combined Sea Beam 2000 bathymetry and sidescan on the Melville EPR 15-19° south, *Proceedings, Oceans '93,* in press, 1993.

Mitchell, N.C., Investigation of the structure and evolution of the Indian Ocean triple junction using GLORIA and other geophysical techniques, *Ph.D. Thesis,* 1989.

Mitchell, N.C., Improving GLORIA images using Sea Beam data, *Journal of Geophysical Research, 96,* 337-351, 1991.

Mitchell, N.C.,A model for attenuation of backscatter due to sediment accumulations and its application to determine sediment thicknesses with GLORIA sidescan sonar, *Journal of Geophysical Research,* in press, 1993.

Mitchell, N.C., and M.L. Somers, Quantitative backscatter measurements with a long-range side-scan sonar, *IEEE Journal of Oceanic Engineering, 14,* 368-374, 1989.

Murton, B.J., I.P. Rouse, N.W. Millard, and C.G. Flewellen, Multisensor, deep-towed instrument explores ocean floor, *Eos Trans. AGU, 73,* 225-228, 1992.

Murton, B.J., and L.M. Parson, Segmentation, volcanism and deformation of oblique spreading centres: a quantitative study of the Reykjanes Ridge, *Tectonophysics, 222,* 237-257, 1993.

Murton, B.J., K. Becker, A. Briais, D. Edge, N. Hayward, G. Klinkhammer, N. Millard, I. Mitchell, I. Rouse, M. Rudnicki, K. Sayanagi, and H. Sloan, Results of a systematic approach to searching for hydrothermal activity on the Mid-Atlantic Ridge: the discovery of the 'Broken Spur' vent site, *Bridge Newsletter, 4,* 3-6, April 1993.

Naraghi, M., W. Stromberg, and M. Daily, Geometric rectification of radar imagery using digital elevation models, *Journal of Photogrammetric Engineering and Remote Sensing, 49,* 195-199, 1983.

Nolle, A.W., W.A. Hoyer, J.F. Mifsud, W.R. Runyan, and M.B. Ward, Acoustical properties of water-filled sands, *Journal of the Acoustical Society of America, 35,* 1394-1408, 1963.

Pace, N.G., and C.M. Dyer, Machine classification of sedimentary sea bottoms, *IEEE Transactions on Geoscience Electronics, GE-17,* 52-56, 1979.

Pace, N.G., and H. Gao, Swathe seabed classification, *IEEE Journal of Oceanic Engineering, 13,* 83-90, 1988.

Parson, L.M., B.J. Murton, R.C. Searle, D. Booth, J. Evans, P. Field, J. Keeton, A. Laughton, E. McAllister, N. Millard, L. Redbourne, I. Rouse, A. Shor, D. Smith, S. Spencer, C. Summerhayes, and C. Walker, En-echelon axial volcanic ridges at the Reykjanes Ridge: a life cycle of volcanism and tectonics, *Earth and Planetary Science Letters, 117,* 73-87, 1993.

Patterson, R.B., Relationships between acoustic backscatter and geological characteristics of the deep ocean floor, *Journal of the Acoustical Society of America, 46,* 756-761, 1967.

PCI Inc., *EASI/PACE Image analysis system manual,* PCI Incorporated, Toronto, Ontario, Canada, 1988.

Peleg, S., J. Naor, R. Hartley, and D. Avnir, Multiple resolution texture analysis and classification, *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-6,* 518-523, 1984.

Pentland, A.P., Fractal based description of natural scenes, *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-6,* 661-674, 1984.

Pøhner, F., Processing of multibeam echosounder data, *Proceedings of the Third Biennial National Ocean Service International Hydrographic Conference Baltimore, Maryland, April 12-15,* 91-101, 1988.

Pøhner, F., and E.B. Lunde, Hydrographic applications of interferometric signal processing, *Proceedings, XIX International Congress,* Helsinki, Finland, 404.3/1-404.3/15, 1990.

Poros, D.J., and C.J. Peterson, Methods for destriping Landsat thematic mapper images - a feasibility study for an online destriping process in the thematic mapper image processing system (TIPS), *Photogrammetric Engineering and Remote Sensing, 51,* 1371-1378, 1985.

Poskanzer, J., *The extended portable bitmap toolkit,* Copyright © 1989, 1991.

Pratt, W.K., *Digital Image Processing*, John Wiley, New York, 1978.

Rayleigh, J.W.S., *The Theory of Sound*, 2nd edition, reprinted in 1945 by Dover Publications (New York) in two volumes, 480 and 504, 1894.

Reed, T.B., Digital image processing and analysis techniques for SeaMARC II sidescan sonar imagery, *Ph.D. Thesis*, 1987.

Reed, T.B., and D. Hussong, Digital image processing techniques for enhancement and classification of SeaMARC II sidescan sonar imagery, *Journal of Geophysical Research, 94*, 7469-7490, 1989.

Reed, T.B., and B.E. Tucholke, Geologic visualization of the Kane Fracture Zone: interactive processing of sidescan and bathymetric data, *Proceedings, Oceans '91, 3*, 1152-1158, 1991.

Reut Z., N.G. Pace, and M.J.P. Heaton, Computer classification of sea beds by sonar, *Nature, 314*, 426-428, 1985.

Sali, E., and H. Wolfson, Texture classification in aerial photographs and satellite data, *International Journal of Remote Sensing, 13*, 3395-3408, 1992.

Searle, R.C., The volcano-tectonic setting of oceanic lithosphere generation, in *Ophiolites and their modern oceanic analogues: Geological Society Special Publication*, edited by L.M. Parson and P. Browning, *60*, 65-79, 1992a.

Searle, R.C., Near real-time merging of GLORIA and Hydrosweep pseudo-sidescan images, *Geophysical Research Letters, 19*, 1137-1140, 1992b.

Searle, R.C., A.S. Laughton, and B.D. Loncarevic, 10. Site survey for site 410, an example of the use of long range sidescan sonar (GLORIA), in *Initial reports of DSDP, 49*, edited by B.P. Luyendyk and J.R. Cann, 413-418, 1978.

Searle, R.C., and A.S. Laughton, Fine-scale sonar study of tectonics and volcanism on the Reykjanes Ridge, *Acta Oceanologica, 26*, 5-13, 1981.

Searle, R.C., T.P. Le Bas, N.C. Mitchell, M.L. Somers, L.M. Parson, and P.H. Patriat, GLORIA image processing: the state of the art, *Marine Geophysical Researches , 12,* 21-39, 1990.

Searle, R.C., K. Lawson, J.A. Pearce, S. Allerton, B. Murton, and H. Schouten, High resolution sonar and magnetic studies of the western Kane transform and adjacent Mid-Atlantic Ridge, *Annales Geophysicae, 10, C70,* 1992.

Sempéré, J.-C., and K.C. Macdonald, Marine tectonics: processes at mid-ocean ridges, *Reviews in Geophysics, 25,* 1313-1347, 1987.

Sempéré, J.-C., G.M. Purdy, and H. Schouten, Segmentation of the Mid-Atlantic ridge between 24°N and 30°40'N, *Nature, 344,* 427-431, 1990.

Sempéré, J.-C., J. Lin, H.S. Brown, H. Schouten, and G.M. Purdy, Segmentation and morphotectonic variations along a slow-spreading centre - the Mid-Atlantic Ridge (24°00'N - 30°40'N), *Marine Geophysical Researches, 15,* 153-200, 1993.

Shanmugan, K.S., Narayanan, V., Frost, V.S., Stiles, J.A., and J.C. Holtzman, Textural features for radar image analysis, *IEEE Transactions on Geoscience and Remote Sensing, 19,* 153-156, 1981.

Shaw, P.R., Ridge segmentation, faulting and crustal thickness in the Atlantic Ocean, *Nature, 358,* 490-493, 1992.

Shaw, P.R., and D.K. Smith, Robust description of statistically heterogeneous seafloor topography through its slope distribution, *Journal of Geophysical Research, 95,* 8705-8722, 1990.

Shen, Y., and D.W. Forsyth, Two forms of volcanic deposits: implications for the volume of off-axis volcanism on the west flank of the East Pacific Rise, *EOS Trans. AGU, 72,* 511, 1991.

Shippey, G., K. Vikgren, A. Elhammer, and R. Finndin, Design of a marine geographic information system for seabed mapping and classification, *Acoustic classification and mapping of the seabed,* in Proceedings of the Institute of Acoustics, 15(2), 195-202, 1993.

Simrad, SIMRAD EM12 Hydrographic echo sounder, Product Description: Simrad Subsea A/S, Horten, Norway, #P2302E, 1992.

Slootweg, A.P., Computer contouring with a digital filter, *Marine Geophysical Researches, 3*, 401-405, 1978.

Smith, D.K., and T.H. Jordan, The size distribution of Pacific seamounts, *Geophysical Research Letters, 14*, 1119-1122, 1987.

Smith, D.K., and P.R. Shaw, Using topographic slope distributions to infer seafloor patterns, *IEEE Journal of Oceanic Engineering, 14*, 338-347, 1989.

Smith, D.K., and J.R. Cann, Hundreds of small volcanoes on the median valley floor of the Mid-Atlantic Ridge at 24-30°N, *Nature, 348*, 152-155, 1990.

Smith, D.K., and J.R. Cann, The role of seamount volcanism in crustal construction at the Mid-Atlantic Ridge (24°-30°N), *Journal of Geophysical Research, 97*, 1645-1658, 1992.

Smith, D.K., and J.R. Cann, Building the crust at the Mid-Atlantic Ridge, *Nature, 365*, 707-715, 1993.

Smith, W.H.F., and P. Wessel, Gridding with continuous curvature splines in tension, *Geophysics, 55*, 293-305, 1990.

Somers, M.L., GLORIA swath bathymetry, *BRIDGE Newsletter, 4*, 27, April 1993.

Somers M.L., R.M. Carson, J.A. Revie, R.H. Edge, B.J. Barrow, and A.G. Andrews, GLORIA II- an improved long range sidescan sonar, a paper presented at *Oceanology International 78*, 16-24, 1978.

Spiess, F.N., and R.C. Tyce, Marine Physical Laboratory Deep Tow instrumentation system, *SIO Reference 73-4*, 1973.

Stanton, T.K., Sonar estimates of seafloor microroughness, *Journal of the Acoustical Society of America, 75*, 809-818, 1984.

Stoll, R.D., Marine sediment acoustics, *Journal of the Acoustical Society of America, 77*, 1789-1799, 1985.

Talukdar, K.K., and R.C. Tyce, Digital processing of sidescan images from bottom backscatter data collected by Sea Beam, *Marine Geodesy, 14*, 81-100, 1990.

Talukdar, K.K., and R.C. Tyce, Relation of Sea Beam echo peak statistics to the character of bottom topography, *Geo-Marine Letters, 12*, 200-207, 1992.

Talukdar, K.K., W. Capell, and C. Zabounidis, Sidescan survey results from a multibeam sonar system - Sea Beam 2000, *Marine Geodesy, 15*, 63-79, 1992.

Talwani, M., C.C. Windisch, and M. Langseth, Reykjanes Ridge crest: a detailed geophysical study, *Journal of Geophysical Research, 76*, 473-577, 1971.

Tamsett, D., Sea-bed characterisation and classification from the power spectra of side-scan sonar data, *Marine Geophysical Researches, 15*, 43-64, 1993.

Tucholke, B.E., Massive submarine rockslide in the rift-valley wall of the Mid-Atlantic Ridge, *Geology, 20*, 129-132, 1992.

Tyce, R.C., Deep seafloor mapping systems - a review, *Marine Technology Society Journal, 20*, 4-16, 1987.

Ulaby, F.T., R.K. Moore, and A.K. Fung, Microwave remote sensing: active and passive, in *Radar remote sensing and surface scattering and emission theory, 2*, 605, Addison-Wesley, Reading, Mass., 1982.

Urick, R.J., The principles of underwater sound, 3rd edition, 20-21, McGraw-Hill, New York, 1983.

Vogt, P.R., and O.E. Avery, Detailed magnetic surveys in the Northeast Atlantic and Labrador Sea, *Journal of Geophysical Research, 79*, 363-389, 1974.

Ware, C., W. Knight, and D. Wells, Memory intensive statistical algorithms for multibeam bathymetric data, *Computers and Geoscience, 17*, 985-993, 1991.

Wells, D., Mayer, L., and J.E. Hughes Clarke, Ocean mapping: from where? to what?, *CISM Journal ACSGC, 45,* 505-518, 1991.

Wessel, P., and W.H.F. Smith, Free software helps map and display data, *EOS Trans. AGU, 72,* 441, 445-446, 1991.

Weszka, J.S., Dyer, C.R., and A. Rosenfeld, A comparative study of texture measures for terrain classification, *IEEE Transactions on Systems, Man and Cybernetics, 6,* 269-285, 1976.

Wong, H-K. and W.D. Chesterman, Bottom backscattering near grazing incidence in shallow water, *Journal of the Acoustical Society of America, 44,* 1713-1718, 1968.

Appendix 1: EM12 data formats

N.B. Formats valid for December 1990 data only; modifications principally to sidescan telegrams for March 1993 data.

Telegrams contained within:

STX	1	byte
message type	1	
...appropriate data telegram...		
ETX	1	
checksum	2	

EM12 position telegram

date	ASCII	7	bytes
time	ASCII	9	
latitude	ASCII	11	
longitude	ASCII	12	
UTM north	ASCII	12	
UTM east	ASCII	10	
UTM zone number	ASCII	3	
UTM zone longitude	ASCII	12	
system	ASCII	2	
Q factor	ASCII	2	
speed	ASCII	5	
course	ASCII	5	
TOTAL		90	bytes

EM12 depth telegram

date	ASCII	6	bytes
time	ASCII	8	
ping number	binary	2	
mode	binary	1	
ping Q factor	binary	1	
depth (centre beam)	binary	2	
heading	binary	2	
roll	binary	2	
pitch	binary	2	
heave	binary	2	
sound velocity	binary	2	
spare	binary	2	
81×			
depth	binary	2	
across-track dist	binary	2	
along-track dist	binary	2	
range	binary	2	
reflectivity	binary	1	
beam Q factor	binary	1	
spare	binary	1	

TOTAL		923	bytes

EM12 sidescan telegram

date	ASCII	6	bytes
time	ASCII	8	
ping number	binary	2	
mode	binary	1	
Q factor	binary	1	
number of samples (N)	binary	2	
Nx			
across-track dist	binary	2	
along-track dist	binary	2	
amplitude	binary	1	
TOTAL		variable	size

Appendix 2: TOBI data format

	header	ASCII	48	bytes
	version	binary	2	
	time	binary	2	
	date	binary	2	
	approx. longitude (deg)	binary	2	
	approx. longitude (min)	binary	4	
	approx. latitude (deg)	binary	2	
	approx. latitude (min)	binary	4	
8×	magnetometer (x)	binary	2	
8×	magnetometer (y)	binary	2	
8×	magnetometer (z)	binary	2	
8×	vehicle roll	binary	2	
8×	vehicle pitch	binary	2	
8×	EM log (fore/aft) velocity	binary	2	
8×	EM log (port/stbd) vel	binary	2	
8×	vehicle heading	binary	2	
8×	pressure	binary	2	
	altitude	binary	2	
	start of profiler (ms)	binary	2	
8×	temperature	binary	2	
8×	transmissivity	binary	2	
	spare	binary	246	
4000×				
	port sidescan	binary	2	
4000×				
	starboard sidescan	binary	2	
4000×				
	profiler	binary	2	
	TOTAL		24576	bytes

Appendix 3: tobi_read.f Fortran program

```
c_____________________________________
c
            program TOBI_read
c_____________________________________
c
c  Jane Keeton July 1991
c
c  This program reads a TOBI datafile, performs a slant
range
c  correction and outputs the sidescan data in an 8 bit
matrix
c  format.

c ---------------------------------------
c declare parameters:

      parameter (NB=8000)
c ------- number of bytes for each side of ship
      parameter (NI=4000)
c ---------- number of integer*2 for each side of ship
      parameter (NB_all_data=24576)
c ------- number of bytes in one TOBI record
      parameter (ltomagsamp=8)
c ---------- size of array for certain variables

      parameter (ssfreq=1.)
      parameter (offset=1.)
      parameter (pxdis=0.75)
      parameter (watvel=1500.)

c declare variables:
c
c specifications of elements in the TOBI record
c
c offset 0   TOBI record header:
      character*(48) to_heading
c offset 48  TOBI version:
      integer*2 to_version
c offset 50  TOBI time:
      integer*2 to_time
c offset 52  TOBI date:
      integer*2 to_date
c offset 54  TOBI approx. longitude, do NOT use...
      integer*2 to_ap_londeg
c offset 56  ...for navigation:
      real*4   to_ap_lonmin
c offset 60  TOBI approx. latitude, do NOT use...
      integer*2 to_ap_latdeg
c offset 62  ...for navigation:
      real*4   to_ap_latmin
c offset 66  x components of fluxgate:
      integer*2 to_magx (ltomagsamp)
c offset 82  y components of fluxgate:
      integer*2 to_magy (ltomagsamp)
c offset 98  z components of fluxgate:
      integer*2 to_magz (ltomagsamp)
c offset 114   vehicle roll:
      integer*2 to_roll (ltomagsamp)
c offset 130   vehicle pitch:
      integer*2 to_pitch  (ltomagsamp)
c offset 146   EM log fore/aft velocity:
      integer*2 to_emlog_fa (ltomagsamp)
c offset 162   EM log port/stbd velocity:
      integer*2 to_emlog_ps (ltomagsamp)
c offset 178   vehicle heading:
      integer*2 to_compass (ltomagsamp)
c offset 194   pressure:
      integer*2 to_press (ltomagsamp)
c offset 210   vehicle altitude (detected on-line) in m:
      integer*2 to_altitude
c offset 212   TOBI start of profiler window (in ms):
      integer*2 to_wapath
c offset 214   temperature (?)
      integer*2 to_temp (ltomagsamp)
c offset 230   transmissivity (?)
      integer*2 to_trans (ltomagsamp)
c offset 246   spares to complete a number of 512 byte
blocks
      character*330 to_empty
c offset 576   port then starboard sidescan values
      integer*2 iport(NI),istbd(NI)

c offset 16576profiler values
      integer*2 prof(NI)
c
c ---------buffer to contain entire TOBI record
      character*24576 all_data
c ---------arrays for slant range corrected data
      integer*2 iportsltg(NI),istbdsltg(NI)
      character*1 bport(NI),bstbd(NI)

      character*40 in_name, out_data, out_head, out_prof


      external long
c_____________________________________

c  define filenames

      print*, 'enter input filename'
      read*, in_name

      print*, 'enter output filename for sidescan data'
      read*, out_data

      print*, 'enter output filename for header data'
      read*, out_head

      print*, 'enter output filename for profiler data'
      read*, out_prof

c  open files:

c  open original datafile with 24576 byte record length

      open(unit=1,file=in_name,err=1000,access='direct',
     & status='old',recl=24576)

c  open temporary file to contain 'byte-swapped' data

      open(unit=2,file='swap.dat',err=900,access='direct',
     & recl=24576,status='scratch')

c  open new file for output sidescan data (unformatted and
direct)

      open(unit=3,file=out_data,err=500,access='direct',
     & recl=1000,status='new')

c  open new file (unformatted and direct) to dump header
information

      open(unit=4,file=out_head,err=300,access='direct',
     & recl=246,status='new')

c  open new file (unformatted and direct) to dump profiler
information

      open(unit=7,file=out_prof,err=275,access='direct',
     &  recl=8000,status='new')

c  read in data from original TOBI file

      istat=0
      n=1

 1100  if(istat.ge.0) then
```

```
      read(1,rec=n) all_data


c call subroutine for byte swap

      call bytswp(NB_all  data,all_data)

      write(2,rec=n) all_data


      read(2,rec=n) to_heading,
     &            to_version,
     &            to time,to date,
     &            to ap londeg,to ap_lonmin,
     &            to ap latdeg,to_ap latmin,
     &            (to_magx(i3),i3=1,ltomagsamp),
     &            (to magy(i4),i4=1,ltomagsamp),
     &            (to magz(i5),i5=1,ltomagsamp),
     &            (to_roll(i6),i6=1,ltomagsamp),
     &            (to_pitch(i7),i7=1,ltomagsamp),
     &            (to_emlog_fa(i8),i8=1,ltomagsamp),
     &            (to_emlog ps(i13),i13=1,ltomagsamp),
     &            (to compass(i9),i9=1,ltomagsamp),
     &            (to_press(i10),i10=1,ltomagsamp),
     &            to_altitude,to_wapath,
     &            (to temp(i11),i11=1,ltomagsamp),
     &            (to trans(i12),i12=1,ltomagsamp),
     &            to empty,
     &            (iport(i),i=1,NI),(istbd(m1),m1=1,NI),
     &            (prof(k),k=1,NI)


      if(mod(n,25).eq.0) print *,n


c call subroutine for slant range to ground range
conversion
c
c N.B. The depth values 'to altitude' are not entirely
satisfactory,
c      they appear to oscillate about every 8(?) pings.
c
c doslac is written by P. Slootweg, whilst at the IOS, UK

      call doslac (iport,NI,ssfreq,offset,iportsltg,NI,pxdis,
     &            watvel,to altitude)
      call doslac (istbd,NI,ssfreq,offset,istbdsltg,NI,pxdis,
     &            watvel,to_altitude)

c
c compress iportsltg and istbdsltg to 8 bits
c

      call docomp (iportsltg,bport)
      call docomp (istbdsltg,bstbd)

c
c write data 'iportsltg' & 'istbdsltg' to output sidescan data
file
c


      write(3,rec=n)(bport(i1),i1=4000,1,-8)
     &      ,(bstbd(i2),i2=1,4000,8)


c
c write header information to output header data file
c
      write(4,rec=n)to heading,
     &            to_version,
     &            to_time,
     &            to_date,
     &            to ap londeg,to_ap_lonmin,
     &            to ap latdeg,to ap latmin,
     &            to_magx,to_magy,to_magz,
     &            to_roll,
     &            to_pitch,
     &            to_emlog_fa,
     &            to_emlog_ps,
     &            to_compass,
     &            to_press,
     &            to_altitude,
     &            to_wapath,
     &            to_temp,
     &            to_trans

c
c write profiler data to output profiler data file
c
      write(7,rec=n)prof
c
      n=n+1

      goto 1100

    endif

275  close(unit=7)
300  close(unit=4)
500  close(unit=3)
900  close(unit=2)
1000 close(unit=1)
    stop
    end


c-----------------------------------------------------------
    subroutine bytswp(n,buffer)
c-----------------------------------------------------------
c
c n = number of bytes in buffer
c  buffer is a character buffer
cc this is necessary if processing on a Sun

    character*(*) buffer
    character*2 temp

    do 100 i=1,n-1,2
      temp=buffer(i:i)
      buffer(i:i)=buffer(i+1:i+1)
      buffer(i+1:i+1)=temp
100  continue

    return
    end


c-----------------------------------------------------------
    subroutine docomp (ival,bval)
c-----------------------------------------------------------

c compress data to 8 bits and perform linear compression

    character*2 buff
    integer*2 ival(4000),jval
    integer*2 imax,imin
    character*1 bval(4000)

    equivalence (jval,buff)

c perform compression

      imax=ival(1)
      imin=ival(1)
      do 200 ij=1,4000
        if (imax.lt.ival(ij)) then
          imax=ival(ij)
        elseif (imin.gt.ival(ij)) then
          imin=ival(ij)
```

```
        endif
200     continue
      range=real(long(imax)-long(imin))
      rmin=real(long(imin))

    do 25 j=1,4000
       rval =real(long(ival(j)))
       rval1=real(long(ival(j)))-rmin
       rval2=(real(long(ival(j)))-rmin)/range
       rval3=256.*((real(long(ival(j)))-rmin)/range)
       ival1=nint(256.*((real(long(ival(j)))-rmin)/range))
       jval=(nint(256.*((real(long(ival(j)))-rmin)/range)))

c  convert from integer*2 to character*1

        bval(j)=buff(2:2)
c        write(*,'(2i8)')jval,bval(j)

25   continue
    return
    end

c
c_____________________________________
c
    subroutine doslac
   & (inpar, lin, ssfreq, offset, ioutar, lout, pxdis, watvel,
depth)
c_____________________________________
c
c written on: August 1990
c written by: Peter Slootweg
c          I.O.S.
c
c Calling sequence:
c
c  call doslac
c  (inpar, lin, ssfreq, offset, ioutar, lout, pxdis, watvel,
depth)
c
c where
c
c     inpar   input array  integer*2  (of length lin samples)
c     ssfreq  sample frequency of input array (in kHz)
c     offset  time attributable to first sample in inpar (in ms)
c
c     ioutar  output array integer*2  (of length lout
samples)
c     pixdis  distance between two pixels in output array (in
m)
c     watvel  sound velocity to be used for sl-correction in
(m/s)
c     depth   vehicle altitude for sl-correction (in m)
c
c all values in the output array are calculated by
interpolation of the
c input array. If the input is undersampled the input array
can be put
c through a running mean calculation before calling this
program.
c
c to simplify calculations two tables are constructed that
guide the
c calculation. For certain emission angles the tables
contain:
c  - tw travel time to scattering area (in indices of input
array)
c  - distance from the vertical through TOBI (in output
array pixels)
c
c Each pixel in the output array then is located in the pixel
table and
c interpolated from values found in the input array through
indices in the
c travel time table
c The lookup tables can be refined or reduced as necessary.
c
c rad is slighly diminished to avoid problems at 90.
c

      parameter (mlook =91, rad = 0.0174532)
      integer*2 inpar(*), ioutar(*),depth
      external long
      dimension twtar(mlook), hdist(mlook)
      logical tablend, interpolate, firstin

c
c construct tables
c (in principle for 90 degrees, but limited by pixel array
length)
c
c twtar: index (i) = dep * freq / cos (ang) * twvel) - offset *
freq + 1
c hdist: index (i) = dep * tan (ang) / pxdis + 1
c
       twvel = watvel / 2000.
       treso = (90. * rad) / real (mlook - 1)
       twfac = (real(long(depth)) * ssfreq) / twvel
       twterm = 1. - offset * ssfreq
       dpfac =(real(long(depth)))/ pxdis
       tablend = .false.
       nlook = 0
       do while (nlook .lt. mlook .and. .not. tablend)
        angle = real (nlook) * treso
        cangle = cos (angle)
        tablend = cangle .le. 0.
        if (.not. tablend) then
         tangle = sin (angle)  cangle
         nlook = nlook + 1
         twtar (nlook) = twfac  cangle + twterm
         hdist (nlook) = dpfac * tangle + 1.
         tablend = hdist (nlook) .gt. real (lout)

        endif
       enddo
c
c do the correction
c
       iout = 1
       firstin = .true.
       tablend = .false.
       ilook = 1
       do while (ilook .lt. nlook .and. .not. tablend)
        delta = hdist (ilook + 1) - real (iout)

c
c loop for one table entry
c
        do while (delta .ge. 0. .and. .not. tablend)
c
c look where in the table we are
c

         blf = delta  (hdist (ilook + 1) - hdist (ilook))
         alf = 1. - blf
c
c calculate index in input array
c
         aindex = alf * twtar (ilook + 1) + blf * twtar
(ilook)

c
c if index before start of window set 0
c
         if (aindex .lt. 1.) then
          ioutar (iout) = 0
c
c otherwise test end of input array (= end of table)
c
         else
          kindex = ifix (aindex)
```

```
          tablend = kindex .ge. lin
c
c test input sample distance for consecutive output samples
c
          if (.not. tablend) then
            interpolate = firstin
            if (.not. interpolate) then
              indexdif = kindex - lastin
              interpolate = indexdif .lt. 2
            endif
            firstin = .false.
            lastin = kindex
c
c if distance small do interpolation
c
            if (interpolate) then
              blf = aindex - real (kindex)
              alf = 1. - blf
              value = alf * real (long(inpar (kindex)))
     &              + blf * real (long(inpar (kindex + 1)))
c
c if distance large take mean around input sample
c
            else
              idelta = indexdif / 2
              sum = 0.
              isum = 0
              do i = kindex - idelta, min (kindex + idelta, lin)
                sum = sum + real (long(inpar (i)))
                isum = isum + 1
              end do
              value = sum   real (isum)
            endif
            ioutar (iout) = nint (value)
          endif
        endif
        if (.not. tablend) iout = iout + 1
        delta = hdist (ilook + 1) -real (iout)
      end do
c
c next table entry
c
      ilook = ilook + 1
    end do
c
c fill up remaining part of array (if any) with 0's
c
    do while (iout .le. lout)
      ioutar (iout) = 0
      iout=iout+1
    end do

    return
  end
```

Appendix 4: sim_nav.f Fortran program

```
      program sim_nav

c  read in SIMRAD EM12 datagrams and output
c  navigation data in ASCII format (*.nav)

c  Jane Keeton 1991

c -------------------------------------------------------
c  data declarations

      common /common1/ char,null
      common /counters/ icount,ieof,ierr,istat
      common /depth_counters/ imark,ncount,iu20
     &              ,iu21,iu22,ist
      common /limits/
xmin,xmax,ymin,ymax,zmin,zmax,idate
      common /times/
start_time,end_time,cstart_time,cend_time,
     &              final_time,ystart

      data stx /x'02'/
      data message_type_pos /x'93'/
      data message_type_depth /x'96'/
      data etx /x'03'/
      data null /'0'/
      data nav_name /'.nav'/

      parameter (cell_size=50.0)
      parameter (len_pos = 90)
      parameter (len_dep = 923)


      character*1 null

c
c telegram: structure declaration
c

      character*1 stx
c -----------02h in hexadecimal representation

      character*1
message_type,message_type_pos,message_type_depth
c -----------93h (hex) refers to position datagram
c -----------96h (hex) refers to depth datagram


c             ..data... (position datagram = 90 bytes)
c             ..... (depth datagram = 923 bytes)

      character*1 etx
c -----------03h (hex)

c       integer*2 checksum
c ---- -------checksum represents total number of bytes in
c -----------'data' portion of telegram


      character*4 nav_name
      character*40 inname,out_nav_name
      character*900000 char
      integer istat,irecp,irecd
      integer icount
      integer fgetc

      real start_time
      real end_time
      double precision cstart_time,cend_time,final_time

c -------------------------------------------------------

      finxmin=500000.0
      finxmax=0.0
      finymin=6000000.0
      finymax=0.0
      finzmin=10000.0
      finzmax=0.0

      print*,'enter input filename'
      read*, inname

      print*,'enter start time for output file'
      read*,start_time

      print*,'enter end time for output time'
      read*,end_time

      print*,'enter y origin for first ping'
      read*,ystart


c  open input datafile

      open(unit=1,
     &     file=inname,
     &     access='direct',
     &     form='unformatted',
     &     err=5000,
     &     status='old',
     &     recl=1)

c name datafile for depth file output

      out_nav_name(1:8)=inname(1:8)
      out_nav_name(9:12)=nav_name


c  open this datafile

      open(unit=2,
     &     file=out_nav_name,
     &     access='sequential',
     &     form='formatted',
     &     status='new',
     &     err=6000)


c  start reading in data

      icount=1
      irecp=0
      irecd=0

10000  ieof=0
      ierr=0
      istat=0
1000   istat=(fgetc(1,char(icount:icount)))

      if (istat.eq.0) then
       if (char(icount:icount).ne.stx) then
        icount=icount+1
        goto 1000
       endif
       goto 2000
      elseif (istat.eq.-1) then
       print*,'end of file'
       goto 5555
      else
       print*,'*input error occurred'
       goto 5000
      endif



2000   icount=icount+1
      istat=(fgetc(1,char(icount:icount)))
      if (istat.eq.0) then
       message_type=char(icount:icount)
```

```
      if (message_type.eq.message_type_pos) then

       call grmpos(irecp)
       if (ieof.eq.1) goto 5000
       if (ierr.eq.1) goto 5000

      elseif (message_type.eq.message_type_depth) then

       icount = icount + len_depth + 1
      else

       print*,'not recognised datagram format'
       goto 5000

      endif

3     elseif (istat.eq.-1) then
       print*,'end of file'
       goto 5555
      else
       print*,'**input error occurred'
       goto 5000
      endif

       icount=icount+1

       istat=fgetc(1,char(icount:icount))
       if (istat.eq.0) then
        goto 3000
       elseif (istat.eq.-1) then
        print*,'end of file'
        goto 5000
       else
        print*,'***input error occurred'
        goto 5000
       endif
        if (char(icount:icount).ne.etx) then
         print*,'reading error: no etx byte recognised'
         goto 5000
        endif
3000      icount=icount+1
        do m=icount,icount+1
        istat= fgetc(1,char(m:m))
        if (istat.eq.0) then
         goto 20
        elseif (istat.eq.-1) then
         print*,'end of file'
         goto 5555
        else
         print*,'****input error occurred'
         goto 5000
        endif
20       enddo
        icount=icount+2

      goto 10000


5000  close(1)
6000  close(2)

      stop
      end

c ------------------------------------------------------------
-
      subroutine grmpos(irecp)
c ------------------------------------------------------------
-

      common common1 char,null
      common counters pos/ icount,ieof,ierr,istat

      parameter(icharno=1000000)
      parameter(maxsize_pos=4000)
      parameter (length=90)
      parameter (ibuflength=100)


data(ibegin(j),j=1,12)/1,8,17,28,40,52,62,65,77,79,81,86/

data(ifin(l),l=1,12)/6,15,25,37,50,60,63,74,77,79,84,90/
      data(isize(m),m=1,12)/8,8,12,12,12,12,4,12,4,4,4,8/

      integer ibegin(12),ifin(12),isize(12)
      character*900000 char
      integer istat,irecp,icount
      integer fgetc
      character*1 buff(ibuflength),null


                character*8 c_pos_date
                real pos_date
c ----------------------DDMMYY,
                character*8 c_pos_time
                real pos_time
                double precision final_time
c ----------------------HHHMMSShh,
                character*12 c_latitude
                real latitude,lat_deg,lat_min
c ----------------------ddmm.XXXXA,--(A=NorS)
                character*12 c_longitude
                real longitude,lon_deg,lon_min
c ----------------------dddmm.XXXXB,--(B=EorW)
                character*12 c_utm_north
                real utm_north
c ----------------------XXXXXXXXX.X,
                character*12 c_utm_east
                real utm_east
c ----------------------XXXXXXX.X,
                character*4 c_utm_zone_no
                real utm_zone_no
c ----------------------XX,
                character*12 c_utm_zone_lon
                real utm_zone_lon
c ----------------------dddmm.XXXXB,--(B=EorW)
                character*4 c_system_id
                real system_id
c ----------------------X,--(0=lat/lon,1=utm)
                character*4 c_q_factor
                real q_factor
c ----------------------X,
                character*4 c_speed
                real speed
c ----------------------XX.X,(m/s)
                character*8 c_line_heading
                real line_heading
c ----------------------XXX.X
                real hour,hoursec,min,minsec,sec,secsec
                real add,remsec

      equivalence (c_pos_date,buff(1))
      equivalence (c_pos_time,buff(9))
      equivalence (c_latitude,buff(17))
      equivalence (c_longitude,buff(29))
      equivalence (c_utm_north,buff(41))
      equivalence (c_utm_east,buff(53))
      equivalence (c_utm_zone_no,buff(65))
      equivalence (c_utm_zone_lon,buff(69))
      equivalence (c_system_id,buff(81))
      equivalence (c_q_factor,buff(85))
      equivalence (c_speed,buff(89))
      equivalence (c_line_heading,buff(93))



      irecp=irecp+1
      ieof=0
      ierr=0
```

```
      istart=icount+1
      iend_pos=length+icount
      do 10 i=istart,iend_pos
       istat=(fgetc(1,char(i:i)))
       if (istat.eq.0) then
        goto 10
       elseif (istat.eq.-1) then
        ieof=1
        print*,'end of file'
        goto 20
       else
        ierr=1
        print*,'*****input error occurred'
        goto 20
       endif
10    continue

      ic=1
      do 1 n=1,12
      id=1
      if (ibegin(n).eq.ifin(n)) then
       buff(ic)=char(icount+ibegin(n):icount+ibegin(n))
       ic=ic+1
       id=id+1
       goto 3
      else
       do 2 k=ibegin(n),ifin(n)
        buff(ic)=char(icount+k:icount+k)
        ic=ic+1
        id=id+1
2      continue
       goto 3
      endif
3     do while(id.le.isize(n))
       buff(ic)=null
       ic=ic+1
       id=id+1
      end do
1     continue

      read(c_pos_date,'(f8.2)')pos_date
      read(c_pos_time,'(f8.0)')pos_time
      read(c_latitude,'(f12.7)')latitude
      read(c_longitude,'(f12.6)')longitude
      read(c_utm_north,'(f12.2)')utm_north
      read(c_utm_east,'(f12.4)')utm_east
      read(c_utm_zone_no,'(f4.2)')utm_zone_no
      read(c_utm_zone_lon,'(f12.6)')utm_zone_lon
      read(c_system_id,'(f4.3)')system_id
      read(c_q_factor,'(f4.3)')q_factor
      read(c_speed,'(f4.1)')speed
      read(c_line_heading,'(f8.4)')line_heading


      if (char(icount+26:icount+26).eq.'S') then
       latitude=-1*latitude
      endif
      if (char(icount+38:icount+38).eq.'W') then
       longitude=-1*longitude
      endif
      if (char(icount+75:icount+75).eq.'W') then
       utm_zone_lon=-1*utm_zone_lon
      endif

      lat_deg=real(aint(latitude/100))
      lat_min=(latitude-(lat_deg*100.))/60.
      latitude=lat_deg+lat_min

      lon_deg=real(aint(longitude/100))
      lon_min=(longitude-(lon_deg*100.))/60.
      longitude=lon_deg+lon_min

      utm_lon_deg=real(aint(utm_zone_lon/100))
      utm_lon_min=(utm_zone_lon-
     (utm_lon_deg*100.)) 60.
      utm_zone_lon=utm_lon_deg+utm_lon_min

c  take time as number of seconds since 00:00 on
c   1st Jan of that year  (specific for 45degN dataset)

      call convert_time(pos_date,pos_time,final_time)

       icount=iend_pos

c  write out to ASCII datafile

      write(3,120) pos_date, pos_time, final_time, longitude,
     latitude,
     &  utm_north, utm_east, utm_zone_no, utm_zone_lon,
     system_id, q_factor,
     &  speed, line_heading
120
     format(f13.5,f15.5,f15.5,2f10.5,f12.2,f12.3,f4.0,f10.5,2f4.
     0,
     &    f5.2,f8.3)



20    return
      end


c ---------------------------------------------------------
      subroutine convert_time(date,time,ping_time)
c ---------------------------------------------------------

      data (imonths(n), n=1,12)
     /31,28,31,30,31,30,31,31,30,31,30,31
     data (lmonths(m), m=1,12)
     /31,29,31,30,31,30,31,31,30,31,30,31


      real date,time
      real day,month,hour,min
      double precision ping_time
      real ndays,daysec,hoursec,minsec,sec,remsec

      if (mod(year,4).eq.0) then
        do 10 n=1,12
          ndays=ndays+lmonth(n)
10      continue
      else
        do 10 n=1,12
          ndays=ndays+imonth(n)
10      continue
      endif

      day=abs(date/10000.)
      month=abs(date/100.)-(day*100.)
      year=date-(day*10000.)-(month*100.)

      ndays=ndays+day
      daysec=ndays*24.*3600.


       hour=abs(time/1000000.)
       hoursec=hour*3600.
       min=abs(time/10000.)-(hour*1000000.)
       minsec=min*60.0
       sec=abs(time/100.)-(hour*1000000.)-(min*10000.)
       remsec=(time-(hour*1000000.)-(min*10000.)-
     (sec*100.))/100.

     ping_time=dreal(daysec)+dreal(hoursec)+dreal(minsec)+d
     real(sec)+dreal(remsec)

      return
      end
```

Appendix 5: sim_depth.f Fortran program

```
      program sim_depth

c  read in SIMRAD EM12 datagrams and output depth
c  values in xyz tile format (*.depth) and depth header info
c  (*.dhead)

   Jane Keeton 1991

c ---------------------------------------------------------
c  data declarations

      common /common1/ char,null
      common /counters/ icount,ieof,ierr,istat
      common /depth_counters/ imark,ncount,iu20
     &              ,iu21,iu22,ist
      common /limits/
xmin,xmax,ymin,ymax,zmin,zmax,idate
      common /times/
start_time,end_time,cstart_time,cend_time,
     &            final_time,ystart

      data stx /x'02'/
      data message_type_pos /x'93'/
      data message_type_depth /x'96'/
      data etx /x'03'/
      data null /'0'/
      data dhead_name /'.dhead'/
      data depth_name /'.depth'/

      parameter (cell_size=50.0)
      parameter (len_pos=90)
      parameter (len_dep=923)


      character*1 null

c
c telegram: structure declaration
c

      character*1 stx
c -----------02h in hexadecimal representation

      character*1
message_type,message_type_pos,message_type_depth
c -----------93h (hex) refers to position datagram
c -----------96h (hex) refers to depth datagram


c             ..data... (position datagram = 90 bytes)
c             ..... (depth datagram = 923 bytes)

      character*1 etx
c -----------03h (hex)

c      integer*2 checksum
c -----------checksum represents total number of bytes in
c -----------'data' portion of telegram


      character*6 dhead_name,depth_name
      character*40
inname,out_depth_name,out_dhead_name
      character*900000 char
      integer istat,irecp,irecd
      integer icount
      integer fgetc

      real start_time
      real end_time
      double precision cstart_time,cend_time,final_time
c ---------------------------------------------------------

      finxmin=500000.0
      finxmax=0.0
      finymin=6000000.0
      finymax=0.0
      finzmin=10000.0
      finzmax=0.0

      print*,'enter input filename'
      read*, inname

      print*,'enter start time for output file'
      read*,start_time

      print*,'enter end time for output time'
      read*,end_time

      print*,'enter y origin for first ping'
      read*,ystart


c  open input datafile

      open(unit=1,
     &     file=inname,
     &     access='direct',
     &     form='unformatted',
     &     err=5000,
     &     status='old',
     &     recl=1)

c name datafile for depth file output

      out_depth_name(1:8)=inname(1:8)
      out_dhead_name(1:8)=inname(1:8)
      out_depth_name(9:14)=depth_name
      out_dhead_name(9:14)=dhead_name


c  open this datafile

      open(unit=2,
     &     file=out_depth_name,
     &     access='sequential',
     &     form='formatted',
     &     status='new',
     &     err=6000)

c  open this datafile

      open(unit=3,
     &     file=out_dhead_name,
     &     access='sequential',
     &     form='formatted',
     &     status='new',
     &     err=7000)


c  start reading in data


      icount=1
      irecp=0
      irecd=0

10000  ieof=0
      ierr=0
      istat=0
1000   istat=(fgetc(1,char(icount:icount)))

      if (istat.eq.0) then
       if (char(icount:icount).ne.stx) then
        icount=icount+1
        goto 1000
       endif
```

```
      goto 2000
     elseif (istat.eq.-1) then
      print*,'end of file'
      goto 5555
     else
      print*,'*input error occurred'
      goto 5000
     endif



2000  icount=icount+1
     istat=(fgetc(1,char(icount:icount)))
     if (istat.eq.0) then
      message_type=char(icount:icount)



      if (message_type.eq.message_type_pos) then

        icount=icount+len_pos+1

      elseif (message_type.eq.message_type_depth) then

       call grmdepth(irecd)

       if (finxmin.gt.xmin) then
         finxmin=xmin
       endif
       if (finxmax.lt.xmax) then
         finxmax=xmax
       endif
       if (finymin.gt.ymin) then
         finymin=ymin
       endif
       if (finymax.lt.ymax) then
         finxmin=xmin
       endif
       if (finzmin.gt.zmin) then
         finzmin=zmin
       endif
       if (finzmax.lt.zmax) then
         finzmax=zmax
       endif
        goto 3

      else

       print*,'not recognised datagram format'
        goto 5000

      endif

3     elseif (istat.eq.-1) then
      print*,'end of file'
      goto 5555
     else
      print*,'**input error occurred'
      goto 5000
     endif

      icount=icount+1

      istat=fgetc(1,char(icount:icount))
      if (istat.eq.0) then
       goto 3000
      elseif (istat.eq.-1) then
       print*,'end of file'
       goto 5000
      else
       print*,'***input error occurred'
       goto 5000
      endif
       if (char(icount:icount).ne.etx) then
        print*,'reading error: no etx byte recognised'
        goto 5000
       endif
3000     icount=icount+1
      do m=icount,icount+1
      istat= fgetc(1,char(m:m))
      if (istat.eq.0) then
       goto 20
      elseif (istat.eq.-1) then
       print*,'end of file'
       goto 5555
      else
       print*,'****input error occurred'
       goto 5000
      endif
20      enddo
      icount=icount+2

     goto 10000

5555 nx=nint((finxmax-finxmin)/cell_size)
    ny=nint((finymax-finymin)/cell_size)


    write(*,'(a,i9)')' total number of pings=',irec
    write(*,'(a,i9)')' total number of xyz triplets=',ncount
    write(*,'(2(a,f15.3))')'  xmin=',finxmin,'
xmax=',finxmax
    write(*,'(2(a,f15.3))')'  ymin=',finymin,'
ymax=',finymax
    write(*,'(2(a,f15.3))')'  zmin=',finzmin,'
zmax=',finzmax
    write(*,'(2(a,i5))') '  nx =',nx,'    ny =',ny

5000  close(1)
6000  close(2)
7000  close(3)

     stop
     end

c-------------------------------------------------------
    subroutine grmdepth(irecd)
c-------------------------------------------------------

    common /common1  char,null
    common  common2/ iunit
    common /counters/ icount,ieof,ierr,istat
    common  depth_counters/ imark,ncount,iu20
   &              ,iu21,iu22,ist
    common /limits/
xmin,xmax,ymin,ymax,zmin,zmax,idate
    common  times/
start_time,end_time,cstart_time,cend_time,
   &            final_time,ystart


    parameter (length=923)
    parameter (maxsize_dep=1000)
    parameter (ibuflength=16)

    data(ibegin(j),j=1,2)/1,7
    data(ifin(l),l=1,2)/6,14/
    data(isize(m),m=1,2)/8,8/

    integer ibegin(2),ifin(2),isize(2)
    character*900000 char
    integer istat,irecd
    integer fgetc,icount,istart,iend_dep
    character*1 buff(ibuflength),null
    real start_time,end_time
    double precision cstart_time,cend_time

              character*8 c_depth_date
              real depth_date
c ------------------------DDMMYY
              character*8 c_depth_time
              real depth_time
```

```
c ----------HHMMSShh
      integer*2 ping_no
      character*2 c_ping_no
c ----------
      character*2 c_mode
      integer*2 mode
c ----------1=2 ms, 2=10 ms
      character*2 c_ping_q_factor
      integer*2 ping_q_factor
c ----------
      integer*2 i_centre_beam_depth
      character*2 c_centre_beam_depth
      real centre_beam_depth
c ----------0.1/0.2m resolution
      integer*2 i_heading
      character*2 c_heading
      real heading
c ----------0.1 deg resolution
      integer*2 i_roll
      character*2 c_roll
      real roll
c ----------0.01 deg resolution
      integer*2 i_pitch
      character*2 c_pitch
      real pitch
c ----------0.01 deg resolution
      integer*2 i_heave
      character*2 c_heave
      real heave
c ----------0.01m resolution
      integer*2 i_sound_velocity
      character*2 c_sound_velocity
      real sound_velocity
c ----------0.1m/s resolution

c ----------81 x
      integer beam_number(81)
c ----------
      integer*2 i_depth
      character*2 c_depth
      real depth(81)
c ----------0.1 0.2m resolution
      integer*2 i_across_dist
      character*2 c_across_dist
      real across_dist(81)
      real x(81)
c ----------0.2/0.5m resolution
      integer*2 i_along_dist
      character*2 c_along_dist
      real along_dist(81)
      real y(81)
c ----------0.2/0.5m resolution
      integer*2 i_range
      character*2 c_range
      real range(81)
c ----------0.2/0.8ms resolution
      character*2 c_reflectivity
      integer*2 i_reflectivity
      real reflectivity(81)
c ----------0.5dB resolution
      character*2 c_beam_q_factor(81)
      integer*2 beam_q_factor(81)
c ----------
      double precision final_time
      real hour,hoursec,min,minsec,sec,secsec
      real add,remsec
      real intensity(81),graze(81)


   equivalence (c_depth_date,buff(1))
   equivalence (c_depth_time,buff(9))
   equivalence (ping_no,c_ping_no)
   equivalence (mode,c_mode)
   equivalence (ping_q_factor,c_ping_q_factor)
   equivalence
(i_centre_beam_depth,c_centre_beam_depth)
   equivalence (i_heading,c_heading)
   equivalence (i_roll,c_roll)
   equivalence (i_pitch,c_pitch)
   equivalence (i_heave,c_heave)
   equivalence (i_sound_velocity,c_sound_velocity)
   equivalence (i_depth,c_depth)
   equivalence (i_across_dist,c_across_dist)
   equivalence (i_along_dist,c_along_dist)
   equivalence (i_range,c_range)
   equivalence (i_reflectivity,c_reflectivity)
   equivalence (beam_q_factor(1),c_beam_q_factor(1))

c-----------------------------------------------------


   irecd=irecd+1
   if (mod(irecd,25).eq.0) then
     write(*,'(i5,a)')irecd,' records processed'
   endif
   ieof=0
   ierr=0
   istart=icount+1
   iend_dep=length+icount

   do 10 i=istart,iend_dep
    istat=(fgetc(1,char(i:i)))
    if (istat.eq.0) then
      goto 10
    elseif (istat.eq.-1) then
      ieof=1
      print*,'end of file'
      goto 20
    else
      ierr=1
      print*,'******input error occurred'
      goto 20
    endif
10  continue


   ic=1
   do 1 n=1,2
   id=1
   if (ibegin(n).eq.ifin(n)) then
    buff(ic)=char(icount+ibegin(n):icount+ibegin(n))
    ic=ic+1
    id=id+1
    goto 3
   else
    do 2 k=ibegin(n),ifin(n)
      buff(ic)=char(icount+k:icount+k)
      ic=ic+1
      id=id+1
2    continue
    goto 3
   endif
3   do while(id.le.isize(n))
    buff(ic)=null
    ic=ic+1
    id=id+1
   end do
1   continue

   read(c_depth_date,'(f8.2)')depth_date
   read(c_depth_time,'(f8.0)')depth_time
   c_ping_no(2:2)=char(icount+15:icount+15)
   c_ping_no(1:1)=char(icount+16:icount+16)
   c_mode(2:2)=char(icount+17:icount+17)
   c_ping_Q_factor(2:2)=char(icount+18:icount+18)
   c_centre_beam_depth(2:2)=char(icount+19:icount+19)
   c_centre_beam_depth(1:1)=char(icount+20:icount+20)
   c_heading(2:2)=char(icount+21:icount+21)
   c_heading(1:1)=char(icount+22:icount+22)
   c_roll(2:2)=char(icount+23:icount+23)
   c_roll(1:1)=char(icount+24:icount+24)
   c_pitch(2:2)=char(icount+25:icount+25)
```

```
      c pitch(1:1)=char(icount+26:icount+26)
      c heave(2:2)=char(icount+27:icount+27)
      c heave(1:1)=char(icount+28:icount+28)
      c sound velocity(2:2)=char(icount+29:icount+29)
      c sound velocity(1:1)=char(icount+30:icount+30)

      if (mode.eq.1) then
        centre_beam depth =
(real(long(i centre beam depth)))*0.1
      elseif (mode.eq.2) then
        centre beam depth =
(real(long(i_centre_beam depth)))*0.2
      endif

      heading = (real(long(i heading)))*0.1
      roll = (real(long(i roll)))*0.01
      pitch = (real(long(i pitch)))*0.01
      heave = (real(long(i heave)))*0.01
      sound velocity = (real(long(i sound_velocity)))*0.1

      ipos=33+icount

      do 101 in=1,81
       c depth(2:2)=char(ipos,ipos)
       c depth(1:1)=char(ipos+1:ipos+1)
       c across dist(2:2)=char(ipos+2:ipos+2)
       c across_dist(1:1)=char(ipos+3:ipos+3)
       c along_dist(2.2)=char(ipos+4:ipos+4)
       c along dist(1:1)=char(ipos+5:ipos+5)
       c range(2:2)=char(ipos+6 ipos+6)
       c range(1:1)=char(ipos+7:ipos+7)
       c reflectivity(2:2)=char(ipos+8 ipos+8)
       c beam q factor(in)(2 2)=char(ipos+9 ipos+9)

       ipos=ipos+11

       if (mode.eq.1) then
         depth(in) = (real(long(i depth)))*0.1
         across dist(in) = (real(long(i across dist)))*0.2
         along dist(in) = (real(long(i along dist)))*0.2
         range(in) = (real(long(i_range)))*0.2
       elseif (mode.eq.2) then
         depth(in) = (real(long(i depth)))*0.2
         across dist(in) = (real(long(i across dist)))*0.5
         along_dist(in) = (real(long(i along_dist)))*0.5
         range(in) = (real(long(i range)))*0.8
       endif

       reflectivity(in) = (real(long(i_reflectivity)))*0.5

       beam number(in)=in

101   continue

c take time as number of seconds since 00 00 on
c 1st Jan of that year (specific for 45degN dataset)

      call convert_time(depth_date,depth time,final_time)


c interpolate time with navigation file for ping origin and
heading

      call
get nav(depth date,final time,nav_time,imark,c_depth_da
te,prev date
     &              xpos,ypos,inav,iu20,iu21,iu22)

      call convert time(depth date,start time,cstart time)
      call convert time(depth date,end time,cend time)


c calculate xy position of each reflection

      if ((final time.lt.cstart time).or.
     &          (final_time.gt.cend time)) then
        goto 923
      else

c work out x and y in tile format

      if (ist.eq.0) then
        do 222 ll=1,81
          x(ll)=((real(across_dist(ll)))*-1)
          y(ll)=((real(along_dist(ll)))*-1)+ystart
       write(2,303)
      &          x(ll),
      &          y(ll),
      &          depth(ll)
222     continue
        prevx=xpos
        prevy=ypos
        accum_dist=ystart*-1
        ist=ist+1

303     format(3f15.3)
      else
        delta_xpos=abs(xpos-prevx)
        delta_ypos=abs(ypos-prevy)
        dxsq=delta_xpos*delta_xpos
        dysq=delta_ypos*delta_ypos
        ping_dist=sqrt(dxsq+dysq)
        accum_dist=accum_dist+ping_dist
        do 333 mm=1,81
          x(mm)=(real(across_dist(mm)))*-1
          y(mm)=(accum_dist+real(along_dist(mm)))*-1
       write(2,303)
      &          x(mm),
      &          y(mm),
      &          depth(mm)
333     continue
        prevx=xpos
        prevy=ypos
        ist=ist+1


      endif
      endif


      xmax=x(1)
      xmin=x(1)
      do 22 j=1,81
        if (xmax.lt.x(j)) then
          xmax=x(j)
        elseif (xmin.gt.x(j)) then
          xmin=x(j)
        endif
22    continue

      ymax=y(1)
      ymin=y(1)
      do 30 k=1,81
        if (ymax.lt.y(k)) then
          ymax=y(k)
        elseif (ymin.gt.y(k)) then
          ymin=y(k)
        endif
30    continue

      zmax=depth(1)
      zmin=depth(1)
      do 44 l=1,81
        if (depth(l).eq.0) then
          goto 44
        elseif (zmax.lt.depth(l)) then
          zmax=depth(l)
        elseif (zmin.gt.depth(l)) then
          zmin=depth(l)
        endif
44    continue
```

```
c Appropriate arrangements of these lines are useful in
c conjunction with additional
c subroutines for alternative output options, e.g. depth by
c beam no., across-track angles of incidence, mean beam
c reflectivity etc.

c
??????????????????????????????????????????????????????

c     if ((final_time.gt.cstart_time).and.
c    &                (final_time.lt.cend_time)) then

c     do 102 inn=1,81

c write data to this file

c     write(2,305)
c    &            beam_number(inn),
c    &            x(inn),
c    &            y(inn),
c    &            depth(inn)
c    &            range(inn),
c    &            angle_inc(inn),
c    &            reflectivity(inn),
c    &            beam_q_factor(inn)

c305    format(i3,x,2f15.3,4(f12.6,x),i4)


c102  continue

c     endif

c
??????????????????????????????????????????????????????
?

c write out header info to datafile

     write(3,40)
depth_date,depth_time,final_time,ping_no,mode
    & ,ping_q_factor,
    &
centre_beam_depth,heading,roll,pitch,heave,sound_velocit
y

40
format(f7.0,x,f9.0,x,f15.2,x,i6,x,i2,x,i6,x,f8.2,x,f6.2,x,
    &     3(f7.3,x),f7.2)

923   ncount=ncount+923

     icount=iend_dep


20    return
     end

c ---------------------------------------------------------
     subroutine get_nav(date,ping_time,nav_time,
    &            imark,c_date,
    &            prev_date,xpos,ypos,inav,iu20,iu21,iu22)
c ---------------------------------------------------------

c N.B. When dealing with more than one file at a time
(and for
c     consecutive times), keep 'imark' counter so search
begins at
c     likely position in nav file.

     data inname1  ' ps/ip/dgl3jak/simrad/ios'
     data inname2  '.nav'

     character*8 c_date
     character*40 filename
     real date,prev_date
     real pos_time,pos_date,longitude,latitude,utm_zone_no
     real utm_zone_lon
     real system_id,q_factor,speed,line_heading
     real utm_north(60500),utm_east(60500)
     double precision
ping_time,nav_time,final_time(60500),rmin,rmax
     double precision time_diff(60500)

c perform interpolation from ping to nav
time/lat/lon/heading..
c just find closest value (v. densely sampled)

     filename(1:26)=inname1
     filename(27:32)=c_date
     filename(33:36)=inname2

     if (iu20.eq.0) then
       iunit=20
       open (unit=iunit,
    &       file=filename,
    &       access='direct',
    &       form='unformatted',
    &       recl=56)
       inav=2
       iu20=iu20+1

     elseif (prev_date.ne.date) then
       iunit=21
       open (unit=iunit,
    &       file=filename,
    &       access='direct',
    &       form='unformatted',
    &       recl=56)
       inav=2
       iu21=iu21+1

     endif


     if (imark.eq.0) then
       nstart=1
     else
       nstart=inav-1
     endif

     n=nstart

100   read (iunit,rec=n)pos_date,pos_time,final_time(n),
    &  longitude,latitude,
    &
utm_north(n),utm_east(n),utm_zone_no,utm_zone_lon,
    &  system_id,q_factor,
    &  speed,line_heading


     time_diff(n)=ping_time-final_time(n)
     if (time_diff(n).ge.-6) then
       n=n+1
       goto 100
     endif
     nend=n

     if (imark.eq.0) then
       nstart=nend-10
     endif

     rmax=abs(time_diff(nstart))
     rmin=abs(time_diff(nstart))
     do 200 i=nstart,nend
       time_diff(i)=abs(time_diff(i))
       if (rmax.lt.time_diff(i)) then
         rmax=time_diff(i)
       elseif (rmin.gt.time_diff(i)) then
         rmin=time_diff(i)
         inav=i
       endif
```

```
200  continue

     nav time=(final time(inav-1)+final time(inav)+
    &        final time(inav+1))/3
     xpos=utm east(inav)
     ypos=utm_north(inav)


     imark=1

     prev date=date

     close(iunit)

     return
     end

c ---------------------------------------------------------
c     subroutine convert_time(date,time,ping_time)
c ---------------------------------------------------------

c this subroutine is listed in sim_nav.f, appendix 4.
```

Appendix 6: sscanread_tile.f Fortran program

```
c ************************************
      program sscanread_tile
c ************************************

c          Jane Keeton 1992

c  a program to read in raw SIMRAD EM12 sidescan
datagrams, merge
c  with navigation data and then interpolate to output ascii
c  files in tile format with x,y,z values (*.tsscan)
(x=longitude,y=latitude,z=amplitude). Retains header info
c  also (*.thead)

c ---------------------------------------------------------

c  data declarations

      common /common1/ char,null
      common /common2/ message_type
      common /counters/ icount,ieof,ierr,istat,imark,ncount,
     &                  iu20,iu21,iu22
      common /limits/
xmin,xmax,ymin,ymax,zmin,zmax,idate

      parameter (nsize=2000000)


      data stx  x'02'/
      data type  x'B2'
      data etx  x'03'
      data null /'0'
      data thead /'.thead'
      data tsscan /'.tsscan'

      character*2000000 char
      character*6 thead
      character*7 tsscan
      character*40 ss_name

      character*40 head_name

      character*1 null,answer,message_type

      real xmin,xmax,ymin,ymax,zmin,zmax
      real finxmin,finxmax,finymin,finymax,finzmin,finzmax

c
c telegram. structure declaration
c

      character*1 stx
c -----------02h in hexadecimal representation

      character*1 type
c -----------B2 (hex) refers to sidescan

      character*1 etx
c -----------03h (hex)

c     integer*2 checksum
c -----------checksum represents total number of bytes in
c -----------'data' portion of telegram


      character*40 inname
      integer istat,irec,icheckp,ichecks,icheckc
      integer fgetc
      real cell_size

c ---------------------------------------------------------

5555  icheckp=0
      ichecks=0
      icheckc=0
      imark=0
      ncount=0
      finxmin=0.0
      finxmax=0.0
      finymin=0.0
      finymax=0.0
      finzmin=150.0
      finzmax=0.0
      iu20=0
      iu21=0
      iu22=0


      print*,'enter input filename'
      read*, inname

      print*, 'enter cell size for gridding'
      read*, cell_size


c  open input datafile

      open(unit=1,
     &     file=inname,
     &     access='direct',
     &     form='unformatted',
     &     err=5000,
     &     status='old',
     &     recl=1)

c  start reading in data

      icount=1
      irec=0

      if (irec.eq.0) print*,'...starting to read in data'

10000 ieof=0
      ierr=0
      istat=0
1000  istat=(fgetc(1,char(icount:icount)))

c  check if first byte of record is 'stx'

      if (istat.eq.0) then
       if (char(icount:icount).ne.stx) then
        icount=icount+1
        goto 1000
       endif
       goto 2000
      elseif (istat.eq.-1) then
       print*,'end of file'
       goto 5100
      else
       print*,'*input error occurred'
       goto 5000
      endif

c  retrieve 'message-type' byte

2000  icount=icount+1
      istat=(fgetc(1,char(icount:icount)))
      if (istat.eq.0) then
       message_type=char(icount:icount)

c  name datafile for header ascii file

       head_name(1:8)=inname(1:8)
       ss_name(1:8)=inname(1:8)

       if (message_type.eq.type) then
        if (icheckp.eq.0) then
         head_name(9:14)=thead
         open(unit=3,
```

```
     &          file=head_name,
     &          access='sequential',
     &          form='formatted',
     &          status='new',
     &          err=3)
          ss  name(9:15)=tsscan
          open(unit=13,
     &          file=ss_name,
     &          access='direct',
     &          form='unformatted',
     &          status='new',
     &          recl=12,
     &          err=13)
          icheckp=icheckp+1
          endif


          call read  data(irec,nsize)
          if (ieof.eq.1) goto 5000
          if (ierr.eq.1) goto 5000

          if (irec.lt.99) then


          if (finxmin.gt.xmin) then
            finxmin=xmin
          endif
          if (finxmax.lt.xmax) then
            finxmax=xmax
          endif
          if (finymin.gt.ymin) then
            finymin=ymin
          endif
          if (finymax.lt.ymax) then
            finxmin=xmin
          endif
          if (finzmin.gt.zmin) then
            finzmin=zmin
          endif
          if (finzmax.lt.zmax) then
            finzmax=zmax
          endif
          print*,finxmin,finxmax,finymin
         endif
         else
          print*,'not recognised datagram format'
          goto 5000
         endif

        elseif (istat.eq.-1) then
         print*,'end of file'
         goto 5100
        else
         print*,'**input error occurred'
         goto 5000
        endif


c  after 'data', next byte is 'etx'

          istat= (fgetc(1,char(icount,icount)))
          if (istat.eq.0) then
            goto 3000
          elseif (istat.eq.-1) then
            print*,'end of file'
            goto 5100
          else
            print*,'***input error occurred'
            goto 5000
          endif
            if (char(icount:icount).ne.etx) then
             print*,'reading error: no etx byte recognised'
             goto 5000
            endif
3000        icount=icount+1

c  then 2 bytes for checksum

          do m=icount,icount+1
          istat= fgetc(1,char(m:m))
          if (istat.eq.0) then
            goto 20
          elseif (istat.eq.-1) then
            print*,'end of file'
            goto 5100
          else
            print*,'****input error occurred'
            goto 5000
          endif
20          enddo
          icount=icount+2

5000    if ((icount-1).lt.nsize) then
          goto 10000
        endif
5100        print*,'...data read in and processed'


        nx=nint((finxmax-finxmin)/cell_size)
        ny=nint((finymax-finymin)/cell_size)


        write(*,'(a,i9)')' total number of pings=',irec
        write(*,'(a,i9)')' total number of xyz triplets=',ncount
        write(*,'(2(a,f15.3))')'  xmin=',finxmin,'
xmax=',finxmax
        write(*,'(2(a,f15.3))')'  ymin=',finymin,'
ymax=',finymax
        write(*,'(2(a,f15.3))')'  zmin=',finzmin,'
zmax=',finzmax
        write(*,'(2(a,i5))') ' nx =',nx,'   ny =',ny


        close(1)
3       close(3)
13      close(13)

c repeat program?

99      print*,'another file? y n'
        read*, answer
        if (answer.eq.'y') then
          goto 5555
        elseif (answer.eq.'n') then
          goto 9999
        else
          print*,'incorrect entry'
          goto 99
        endif

9999    stop
        end

c-------------------------------------------------------------------
        subroutine read_data(irec,nsize)
c-------------------------------------------------------------------

        common  common1  char,null
        common  common2/ message_type
        common  counters/
icount,ieof,ierr,istat,imark,ncount,iu20
       &                ,iu21,iu22
        common  limits/
xmin,xmax,ymin,ymax,zmin,zmax,idate

        data(ibegin(j),j=1,2)/1,7
        data(ifin(l),l=1,2)/6,14/
        data(isize(m),m=1,2)/8,8/

        data type_port  x'B1'
        data type_stbd /x'B2'
        data type_cent  x'B3'
```

```
      character*1
message type,type port,type stbd,type cent
c ----------B1 (hex) refers to port sidescan
c ----------B2 (hex) refers to starboard sidescan
c ----------B3 (hex) refers to centre sidescan

      parameter (ibuflength=16)
      character*1732560 char
      integer istat,irec
      integer fgetc
      character*1 buff(ibuflength),null
      integer ibegin(2),ifin(2),isize(2)
      integer*2 beam_number

              character*8 c_date
              real date
c -------------------------DDMMYY
              character*8 c_time
              real time
c ------------------------------HHMMSShh
              integer*2 ping_no
              character*2 c_ping_no
c ------------------------
              character*2 c_mode
              integer*2 mode
c ----------------------1=2 ms, 2=10 ms
              character*2 c_ping_q_factor
              integer*2 ping_q_factor
c ------------------------
              integer*2 no_samples
              character*2 c_no_samples
c ------------------------
              character*2 c_inline_dist
              integer*2 i_inline_dist
              real inline_dist(2500)
c-------------------------along vessel hull direction,
c                     heading direction +ve
c                     0.2/0.5m resolution

              character*2 c_crossline_dist
              integer*2 i_crossline_dist
              real crossline_dist(2500)
c-------------------------positive in direction 90 degrees
c                     to heading,rotated towards starboard.
c                     0.2/0.5m resolution

              character*2 c_amplitude
              integer*2 i_amplitude
              real amplitude(2500)
c-------------------------0.5 db resolution, value is a
signed
c                     character(-64 to +64 db)


              double precision final_time,nav_time
              real ping_dist,accum_dist
              real prevx,prevy
              real x(2500)
              real y(2500)
              real xmin,xmax,ymin,ymax,zmin,zmax

      equivalence (c_date,buff(1))
      equivalence (c_time,buff(9))
      equivalence (ping_no,c_ping_no)
      equivalence (mode,c_mode)
      equivalence (ping_q_factor,c_ping_q_factor)
      equivalence (no_samples,c_no_samples)
      equivalence (c_inline_dist,i_inline_dist)
      equivalence (c_crossline_dist,i_crossline_dist)
      equivalence (c_amplitude,i_amplitude)
c-----------------------------------------------------------


      irec=irec+1
      ieof=0
      ierr=0


      ilength=20
      istart=icount+1
      iend=ilength+icount

c  start reading in data

      if (mod(irec,25).eq.0) then
        write(*,'(a,i5)')'.....reading record number',irec
      endif

      do 10 i=istart,iend
       istat=(fgetc(1,char(i:i)))
       if (istat.eq.0) then
        goto 10
       elseif (istat.eq.-1) then
        ieof=1
        print*,'end of file'
        goto 20
       else
        ierr=1
        print*,'******input error occurred'
        goto 20
       endif
10    continue


c  assign formatted ascii data to a buffer

      ic=1
      do 1 n=1,2
      id=1
      if (ibegin(n).eq.ifin(n)) then
       buff(ic)=char(icount+ibegin(n):icount+ibegin(n))
       ic=ic+1
       id=id+1
       goto 3
      else
       do 2 k=ibegin(n),ifin(n)
        buff(ic)=char(icount+k:icount+k)
        ic=ic+1
        id=id+1
2      continue
       goto 3
      endif
3     do while(id.le.isize(n))
       buff(ic)=null
       ic=ic+1
       id=id+1
      end do
1     continue

c  read in ascii data from buffer

      read(c_date,'(f8.2)')date
      read(c_time,'(f8.0)')time

c  work out time in seconds since beginning of this year
(1990 here)

      call convert_time(date,time,final_time)


c  directly assign header variables from 'icount' array

      c_ping_no(2:2)=char(icount+15:icount+15)
      c_ping_no(1:1)=char(icount+16:icount+16)
      c_mode(2:2)=char(icount+17:icount+17)
      c_ping_q_factor(2:2)=char(icount+18:icount+18)
      c_no_samples(2:2)=char(icount+19:icount+19)
      c_no_samples(1:1)=char(icount+20:icount+20)

      icount=iend
```

```
c read in sample data

      istart=icount+1
      iend=(no_samples*5)+icount


      do 100 i=istart,iend
       istat=(fgetc(1,char(i:i)))
       if (istat.eq.0) then
        goto 100
       elseif (istat.eq.-1) then
        ieof=1
        print*,'end of file'
        goto 20
       else
        ierr=1
        print*,'******input error occurred'
        goto 20
       endif
100    continue

      ipos=icount+1


      do 101 in=1,no_samples

       c inline_dist(2:2)=char(ipos:ipos)
       c inline_dist(1:1)=char(ipos+1:ipos+1)
       c crossline_dist(2:2)=char(ipos+2:ipos+2)
       c crossline_dist(1:1)=char(ipos+3:ipos+3)
       c_amplitude(2:2)=char(ipos+4:ipos+4)


       if (mode.eq.1) then
         inline_dist(in) = (real(long(i_inline_dist)))*0.2
         crossline_dist(in) =
(real(long(i_crossline_dist)))*0.2
       elseif (mode.eq.2) then
         inline_dist(in) = (real(long(i_inline_dist)))*0.5
         crossline_dist(in) =
(real(long(i_crossline_dist)))*0.5
       endif


       amplitude(in) = (real(long(i_amplitude)))*0.5-64.0

c N.B. near-nadir amplitude correction is not included in
c this version of the program - the output files from this
c program were used to determine the angular dependent
c backscatter correction function.

       beam_number=in

       ipos=ipos+5

101    continue

c interpolate time with navigation file for ping origin and
heading

     call
get_nav(date,final_time,nav_time,imark,c_date,prev_date
    &              xpos,ypos,inav,iu20,iu21,iu22)

c calculate xy position of each reflection in tile format

     if (irec.eq.1) then
       do 222 ll=1,no_samples
         x(ll)=(real(crossline_dist(ll)))*-1
         y(ll)=(real(inline_dist(ll)))*-1
222    continue
       prevx=xpos
       prevy=ypos
       accum_dist=0.0
     else
       delta_xpos=abs(xpos-prevx)
       delta_ypos=abs(ypos-prevy)
       dxsq=delta_xpos*delta_xpos
       dysq=delta_ypos*delta_ypos
       ping_dist=sqrt(dxsq+dysq)
       accum_dist=accum_dist+ping_dist
       do 333 mm=1,no_samples
         x(mm)=(real(crossline_dist(mm)))*-1
         y(mm)=(accum_dist+real(inline_dist(mm)))*-1
333    continue
       prevx=xpos
       prevy=ypos

     endif

     xmax=x(1)
     xmin=x(1)
     do 22 j=1,no_samples
       if (xmax.lt.x(j)) then
         xmax=x(j)
       elseif (xmin.gt.x(j)) then
         xmin=x(j)
       endif
22   continue

     ymax=y(1)
     ymin=y(1)
     do 30 k=1,no_samples
       if (ymax.lt.y(k)) then
         ymax=y(k)
       elseif (ymin.gt.y(k)) then
         ymin=y(k)
       endif
30   continue

     zmax=amplitude(1)
     zmin=amplitude(1)
     do 44 l=1,no_samples
       if (amplitude(l).eq.0) then
         goto 44
       elseif (zmax.lt.amplitude(l)) then
         zmax=amplitude(l)
       elseif (zmin.gt.amplitude(l)) then
         zmin=amplitude(l)
       endif
44   continue


c write data to this file

      if (irec.lt.99) then


      do 102 ii=1,no_samples

        write(13,rec=(ncount+ii))x(ii),
     &                  y(ii),
     &                  amplitude(ii)


102    continue

c write out header info to datafile


      write(3,40)
date,time,nav_time,ping_no,mode,ping_q_factor,
    &  no_samples

40   format(f7.0,x,f9.0,x,f15.5,x,4(i6,x))

     endif


     ncount=ncount+no_samples

c21
```

```
      icount=iend+1


20    return
      end


c ------------------------------------------------------------
c     subroutine get_nav(date,ping  time,nav_time,
c    &                   imark,c  depth_date,
c    &                   prev_date,xpos,ypos,inav,iu20,iu21,iu22)
c ------------------------------------------------------------

c  this subroutine is listed in sim  depth.f, appendix 5.


c ------------------------------------------------------------
c     subroutine convert  time(date,time,ping_time)
c ------------------------------------------------------------

c  this subroutine is listed in sim_nav.f, appendix 4.
```

Appendix 7: sscanread_geom.f Fortran program

```
c
**************************************************
      program sscanread_geom
c
**************************************************

c          Jane Keeton 1992

c  a program to read in raw SIMRAD EM12 sidescan
datagrams, merge
c  with navigation data and then interpolate to output ascii
files in geometric format (*.gsscan)
c  with x,y,z values (x=longitude,y=latitude,z=amplitude)

c ---------------------------------------------

c  data declarations

      common /common1/ char,null

      common /counters/
icount,ieof,ierr,istat,imark,imark2,ncount,
     &              iu20,iu21,iu22,i20,i21,i22
      common /counters2/ inav,inav2,itick
      common /limits/ xmin,xmax,ymin,ymax,zmin,zmax

      parameter (nsize=200000)
      parameter (cell_size=36.0)

      data stx /x'02'/
      data type /x'B2'/
      data etx /x'03'/
      data null /'0'/
      data gsscan /'.gsscan'/

      character*7 gsscan
      character*40 ss_name

      character*40 head_name,track_name

      character*1 null,answer

c
c telegram: structure declaration
c

      character*1 stx
c ----------02h in hexadecimal representation

      character*1 message_type,type
c ----------B2 (hex) refers to sidescan

      character*1 etx
c ----------03h (hex)

c      integer*2 checksum
c ----------checksum represents total number of bytes in
c ----------'data' portion of telegram


      character*40 inname
      character*200000 char

      integer istat,irec,icheckp,ichecks,icheckc
      integer fgetc
      real cell_size

c ---------------------------------------------------

5555  icheckp=0
      ichecks=0
      icheckc=0
      imark=0
      ncount=0
      imark2=0
      inav=0
      inav2=0
      finxmin=10000000.0
      finxmax=0.0
      finymin=10000000.0
      finymax=0.0
      finzmin=150.0
      finzmax=0.0
      iu20=0
      iu21=0
      iu22=0
      iu30=0
      iu31=0
      iu32=0
      i20=0
      i21=0
      i22=0
      itick=0

      print*,'enter input filename'
      read*, inname

c  open input datafile

      open(unit=1,
     &     file=inname,
     &     access='direct',
     &     form='unformatted',
     &     err=5000,
     &     status='old',
     &     recl=1)

c  start reading in data

      icount=1
      irec=0
      if (irec.eq.0) print*,'...starting to read in data'

10000 ieof=0
      ierr=0
      istat=0
1000  istat=(fgetc(1,char(icount:icount)))

c  check if first byte of record is 'stx'

      if (istat.eq.0) then
       if (char(icount:icount).ne.stx) then
        icount=icount+1
        goto 1000
       endif
       goto 2000
      elseif (istat.eq.-1) then
       print*,'end of file'
       goto 5100
      else
       print*,'*input error occurred'
       goto 5000
      endif

c  retrieve 'message-type' byte

2000  icount=icount+1
      istat=(fgetc(1,char(icount:icount)))
      if (istat.eq.0) then
       message_type=char(icount:icount)

       ss_name(1:8)=inname(1:8)
       ss_name(9:15)=gsscan
       open(unit=13,
     &       file=ss_name,
     &       access='direct',
     &       form='unformatted',
     &       status='new',
```

```
     &          recl=12,
     &          err=13)
        icheckp=icheckp+1


         call read_data(irec)
         if (ieof.eq.1) goto 5000
         if (ierr.eq.1) goto 5000

        if (finxmin.gt.xmin) then
          finxmin=xmin
        endif
        if (finxmax.lt.xmax) then
          finxmax=xmax
        endif
        if (finymin.gt.ymin) then
          finymin=ymin
        endif
        if (finymax.lt.ymax) then
          finymax=ymax
        endif
        if (finzmin.gt.zmin) then
          finzmin=zmin
        endif
        if (finzmax.lt.zmax) then
          finzmax=zmax
        endif

       else
         print*,'not recognised datagram format'
         goto 5000
       endif

      elseif (istat.eq.-1) then
       print*,'end of file'
       goto 5000
      else
       print*,'**input error occurred'
       goto 5000
      endif


c  after 'data', next byte is 'etx'

       istat= (fgetc(1,char(icount.icount)))
       if (istat.eq.0) then
         goto 3000
       elseif (istat.eq.-1) then
         print*,'end of file'
         goto 5100
       else
         print*,'***input error occurred'
         goto 5000
       endif
         if (char(icount:icount).ne.etx) then
           print*,'reading error  no etx byte recognised'
           goto 5000
         endif
3000       icount=icount+1

c  then 2 bytes for checksum

       do m=icount,icount+1
       istat= fgetc(1,char(m m))
       if (istat.eq.0) then
         goto 20
       elseif (istat.eq.-1) then
         print*,'end of file'
         goto 5100
       else
         print*,'****input error occurred'
         goto 5000
       endif
20       enddo
       icount=icount+2

       goto 10000

5000   if ((icount-1).lt.nsize) then
         goto 10000
       endif
5100       print*,'...data read in and processed'


     nx=nint((finxmax-finxmin)/cell_size)
     ny=nint((finymax-finymin)/cell_size)


     write(*,'(a,i9)')' total number of pings=',irec
     write(*,'(a,i9)')' total number of xyz triplets=',ncount
     write(*,'(2(a,f15.3))')' xmin=',finxmin,'
xmax=',finxmax
     write(*,'(2(a,f15.3))')' ymin=',finymin,'
ymax=',finymax
     write(*,'(2(a,f15.3))')' zmin=',finzmin,'
zmax=',finzmax
     write(*,'(2(a,i5))') ' nx =',nx,'   ny =',ny

     close(1)
13   close(13)

c repeat program?

99   print*,'another file? y n'
     read*, answer
     if (answer.eq.'y') then
       goto 5555
     elseif (answer.eq.'n') then
       goto 9999
     else
       print*,'incorrect entry'
       goto 99
     endif

9999 stop
c  call gclose

     end

c-----------------------------------------------------------------
     subroutine read_data(irec)
c-----------------------------------------------------------------

     common  common1  char,null

     common  counters/
icount,ieof,ierr,istat,imark,imark2,ncount,
     &              iu20,iu21,iu22,i20,i21,i22
     common /limits/ xmin,xmax,ymin,ymax,zmin,zmax
     common  counters2/ inav,inav2,itick

     data(ibegin(j),j=1,2) 1,7
     data(ifin(l),l=1,2)/6,14/
     data(isize(m),m=1,2)/8,8/

     data type  x'B2'


     parameter (ibuflength=16)
     parameter (ipings=500)

     character*200000 char
     integer iship
     integer istat,irec
     integer fgetc
     character*1 buff(ibuflength),null
     integer ibegin(2),ifin(2),isize(2)
     integer*2 beam_number

              character*8 c_date
              real date
c ----------------------------DDMMYY
```

```
            character*8 c_time
            real time
c ----------------------HHMMSShh
            integer*2 ping_no
            character*2 c_ping_no
c ----------------------
            character*2 c_mode
            integer*2 mode
c ------------------------1=2 ms, 2=10 ms
            character*2 c_ping_q_factor
            integer*2 ping_q_factor
c ----------------------
            integer*2 no_samples
            character*2 c_no_samples
c ----------------------
            character*2 c_inline_dist
            integer*2 i_inline_dist
            real inline_dist(2000)
c------------------------along vessel hull direction,
c                   heading direction +ve
c                   0.2/0.5m resolution

            character*2 c_crossline_dist
            integer*2 i_crossline_dist
            real crossline_dist(2000)
c------------------------positive in direction 90 degrees
c                   to heading,rotated towards starboard.
c                   0.2/0.5m resolution

            character*2 c_amplitude
            integer*2 i_amplitude
            real amplitude(2000)
c------------------------0.5 db resolution, value is a
signed
c                   character(-64 to +64 db)


            double precision nav_time,ping_time
            double precision x_ping_origin,y_ping_origin
            real x(2000)
            real y(2000)
            real utm_north,utm_east,heading
            real correct(21)

    data
(correct(ll),ll=1,21) 0.96421,0.94643,0.92169,0.89838,
    &
0.88351,0.87668,0.87647,0.87944,0.88324,0.88504,0.88
506,
    &
0.88359,0.88452,0.88672,0.89060,0.89687,0.90678,0.92
322,
    &    0.94562,0.96790,0.98328/

    equivalence (c_date,buff(1))
    equivalence (c_time,buff(9))
    equivalence (ping_no,c_ping_no)
    equivalence (mode,c_mode)
    equivalence (ping_q_factor,c_ping_q_factor)
    equivalence (no_samples,c_no_samples)
    equivalence (c_inline_dist,i_inline_dist)
    equivalence (c_crossline_dist,i_crossline_dist)
    equivalence (c_amplitude,i_amplitude)
c-----------------------------------------------------


    irec=irec+1

    if (mod(irec,25).eq.0) then
      write(*,'(i5,a)')irec,' records processed'
    endif

    ieof=0
    ierr=0

    ilength=20
    istart=icount+1
    iend=ilength+icount

c   start reading in data

    do 10 i=istart,iend
     istat=(fgetc(1,char(i:i)))
     if (istat.eq.0) then
      goto 10
     elseif (istat.eq.-1) then
      ieof=1
      print*,'end of file'
      goto 20
     else
      ierr=1
      print*,'******input error occurred'
      goto 20
     endif
10   continue


c   assign formatted ascii data to a buffer

    ic=1
    do 1 n=1,2
    id=1
    if (ibegin(n).eq.ifin(n)) then
     buff(ic)=char(icount+ibegin(n):icount+ibegin(n))
     ic=ic+1
     id=id+1
     goto 3
    else
     do 2 k=ibegin(n),ifin(n)
      buff(ic)=char(icount+k:icount+k)
      ic=ic+1
      id=id+1
2    continue
     goto 3
    endif
3   do while(id.le.isize(n))
     buff(ic)=null
     ic=ic+1
     id=id+1
    end do
1   continue

c   read in ascii data from buffer

    read(c_date,'(f8.2)')date
    read(c_time,'(f8.0)')time

c   work out time in seconds since beginning of this year
(1990 here)

    call convert_time(date,time,ping_time)

c   interpolate time with navigation file for ping origin and
heading

    call get_nav(date,ping_time,imark,c_date,prev_date,
    &            utm_east,utm_north,inav,iu20,iu21,iu22)

    x_ping_origin=dreal(utm_east)
    y_ping_origin=dreal(utm_north)

    call
get_heading(date,ping_time,imark2,heading,c_date,prev_d
ate2,centre_beam_depth,
    &            inav2,
    &            i30,i31,i32)


c   directly assign header variables from 'icount' array
```

```
      c ping no(2:2)=char(icount+15:icount+15)
      c ping no(1:1)=char(icount+16:icount+16)
      c_mode(2:2)=char(icount+17:icount+17)
      c_ping Q factor(2:2)=char(icount+18:icount+18)
      c_no samples(2:2)=char(icount+19:icount+19)
      c_no_samples(1:1)=char(icount+20:icount+20)

      icount=iend

c read in sample data

      istart=icount+1
      iend=(no samples*5)+icount


      do 100 i=istart,iend
       istat=(fgetc(1,char(i:i)))
       if (istat.eq.0) then
        goto 100
       elseif (istat.eq.-1) then
        ieof=1
        print*,'end of file'
        goto 20
       else
        ierr=1
        print*,'******input error occurred'
        goto 20
       endif
100   continue

      ipos=icount+1


      do 101 in=1,no samples

       c inline dist(2 2)=char(ipos ipos)
       c inline dist(1:1)=char(ipos+1.ipos+1)
       c crossline_dist(2:2)=char(ipos+2·ipos+2)
       c crossline dist(1:1)=char(ipos+3.ipos+3)
       c amplitude(2.2)=char(ipos+4·ipos+4)


       if (mode.eq.1) then
         inline dist(in) = (real(long(i_inline_dist)))*0.2
         crossline_dist(in) =
(real(long(i_crossline dist)))*0.2
       elseif (mode.eq.2) then
         inline dist(in) = (real(long(i_inline_dist)))*0.5
         crossline dist(in) =
(real(long(i crossline dist)))*0.5
       endif

c work out range

       amplitude(in) = (real(long(i_amplitude)))*0.5-64.0

       beam number=in

       ipos=ipos+5

101   continue

c calculate xy position of each reflection

      if ((heading.ge.0).and.(heading.lt.90)) then

       do 111 iii=1, no samples
        x(iii)=sngl(
     &    dreal(crossline dist(iii))*dcosd(dreal(heading))
     &    +dreal(inline dist(iii))*dsind(dreal(heading)))
     &    +x ping_origin
        y(iii)=sngl(
     &    dreal(inline dist(iii))*dcosd(dreal(heading))
     &    -dreal(crossline dist(iii))*dsind(dreal(heading)))
     &    +y ping origin
111   continue

      elseif ((heading.ge.90).and.(heading.lt.180)) then
       alpha=heading-90.

       do 112 jjj=1,no_samples
        x(jjj)=sngl(
     &    dreal(inline_dist(jjj))*dcosd(dreal(alpha))
     &    -dreal(crossline_dist(jjj))*dsind(dreal(alpha)))
     &    +x_ping_origin
        y(jjj)=-1*(sngl(
     &    dreal(crossline_dist(jjj))*dcosd(dreal(alpha))
     &    +dreal(inline_dist(jjj))*dsind(dreal(alpha))))
     &    +y_ping_origin
112   continue

      elseif ((heading.ge.180).and.(heading.lt.270)) then
       alpha=heading-180.

       do 113 kkk=1, no_samples
        x(kkk)=-1*(sngl(
     &    dreal(crossline_dist(kkk))*dcosd(dreal(alpha))
     &    +dreal(inline_dist(kkk))*dsind(dreal(alpha))))
     &    +x_ping_origin
        y(kkk)=sngl(
     &    dreal(crossline_dist(kkk))*dsind(dreal(alpha))
     &    -dreal(inline_dist(kkk))*dcosd(dreal(alpha)))
     &    +y_ping_origin
113   continue

      else
       alpha=360.-heading

       do 114 lll=1, no_samples
        x(lll)=sngl(
     &    dreal(crossline_dist(lll))*dcosd(dreal(alpha))
     &    -dreal(inline_dist(lll))*dsind dreal(alpha)))
     &    +x_ping_origin
        y(lll)=sngl(
     &    dreal(crossline_dist(lll))*dsind dreal(alpha))
     &    +dreal(inline_dist(lll))*dcosd(dreal(alpha)))
     &    +y_ping_origin
114   continue

      endif


      xmax=x(1)
      xmin=x(1)
      do 22 j=1,no_samples
       if (xmax.lt.x(j)) then
         xmax=x(j)
       elseif (xmin.gt.x(j)) then
         xmin=x(j)
       endif
22    continue

      ymax=y(1)
      ymin=y(1)
      do 30 k=1,no_samples
       if (ymax.lt.y(k)) then
         ymax=y(k)
       elseif (ymin.gt.y(k)) then
         ymin=y(k)
       endif
30    continue

c perform near-nadir correction

      do 444 ll=1,no_samples
       range=atand(crossline_dist(ll)/centre_beam_depth)
       irange=nint(range)

       if ((abs(irange)).le.10) then
         amplitude(ll)=(correct(irange+11)*amplitude(ll))
       endif
444   continue
```

```
      zmax=amplitude(1)
      zmin=amplitude(1)
      do 44 i=1,no_samples
        if ((amplitude(i).lt.0.).or.(amplitude(i).gt.255.)) then
          goto 44
        elseif (zmax.lt.amplitude(i)) then
          zmax=amplitude(i)
        elseif (zmin.gt.amplitude(i)) then
          zmin=amplitude(i)
        endif
 44   continue


c  write data to this file

        do 102 ii=1,no_samples

          write(13,rec=(ncount+ii))x(ii),
     &            y(ii),
     &            amplitude(ii)


 102  continue

c  write out header info to datafile


          write(3,40)
 date,time,nav_time,ping_no,mode,ping_q_factor,
     &   no_samples


 40   format(f7.0,x,f9.0,x,f15.5,x,4(i6,x))

      ncount=ncount+no_samples

      icount=iend+1

 20   return
      end


c ---------------------------------------------------------
c     subroutine get_nav(date,ping_time,imark,
c    &              xpos,ypos,inav,iu20,iu21,iu22)
c ---------------------------------------------------------

c  this subroutine is listed in sim_depth.f, appendix 5.


c ---------------------------------------------------------
c     subroutine convert_time(date,time,ping_time)
c ---------------------------------------------------------

c  this subroutine is listed in sim_nav.f, appendix 4.


c---------------------------------------------------------
      subroutine
 get_heading(date,ping_time,imark2,heading,c_date,prev_d
 ate,
     &              centre_beam_depth,inav2,
     &               i30,i31,i32)
c---------------------------------------------------------

      data inname1 /' ps/ip/dgl3jak/simrad/ios/'
      data inname2 /'.head'

      character*8 c_date
      character*40 filename
      real date,prev_date
      real depth_date,depth_time
      double precision
 final_time(60500),time_diff(60500),ping_time
      integer*2 ping_no,mode,ping_q_factor
      real
 centre_beam_depth,rheading(60500),roll,pitch,heave
      real sound_velocity,heading
      real rcentre_beam_depth(60500)

      filename(1:26)=inname1
      filename(27:32)=c_date
      filename(33:37)=inname2

      if (iu30.eq.0) then
        iunit=30
        open (unit=iunit,
     &       file=filename,
     &       access='direct',
     &       form='unformatted',
     &       recl=46,
     &       status='old')
        inav=2
        i30=i30+1

      elseif (prev_date.ne.date) then
        iunit=31
        open (unit=iunit,
     &       file=filename,
     &       access='direct',
     &       form='unformatted',
     &       recl=46)
        inav=2
        i31=i31+1

      endif

      if (imark2.eq.0) then
        nstart=1
      else
        nstart=inav2-1
      endif

      n=nstart

 100
 read(iunit,rec=n)depth_date,depth_time,final_time(n),
     & ping_no,mode,ping_q_factor,
     & rcentre_beam_depth(n),rheading(n),roll,pitch,heave,
     & sound_velocity


      time_diff(n)=ping_time-final_time(n)
      if (time_diff(n).ge.-6) then
        n=n+1
        goto 100
      endif
      nend=n

      rmax=abs(time_diff(nstart))
      rmin=abs(time_diff(nstart))
      do 200 i=nstart,nend
        time_diff(i)=abs(time_diff(i))
        if (rmax.lt.time_diff(i)) then
          rmax=time_diff(i)
        elseif (rmin.gt.time_diff(i)) then
          rmin=time_diff(i)
          inav2=i
        endif
 200  continue

      heading=rheading(inav2)
      centre_beam_depth=rcentre_beam_depth(inav2)

      imark2=1
      prev_date=date

      close(iunit)

      return
      end
```

Appendix 8: sim_grid.f Fortran program

```
      program sim_grid

c  grid xyz ASCII triplets using UNIRAS bilinear
c  interpolation routines
c  complile with 'unilink...'

c  Jane Keeton 1992

c  np       : number of control points
c  cell_size : grid cell size in metres
c  nx,ny    : dimensions of grid

      parameter (np=50000)
      parameter (nx=512)
      parameter (ny=512)
      parameter (cell_size=36.0)

      real
x(np),y(np),z(np),xmax,xmin,ymax,ymin,zmax,zmin
      character*30 ipfile, opfile
      integer*4 jval
      integer*2 ival
      character*2 buff
      real zrange,rval1,rval1,rval3
      real grid(nx,ny)
      character*1 bytegrid(nx*ny)

      equivalence (ival,buff)

c  read in data and find maximum and minimum values

      print*, 'enter name of input xyz file'
      read*, ipfile
      print*, 'enter name of output grid file'
      read*, opfile

      open (unit=1,
     &     file=ipfile,
     &     access='sequential',
     &     form='formatted',
     &     status='old')

      print*,'...reading in data'

      do 10 i=1,np
        read(1,111)x(i),y(i),z(i)
111     format(3f15.3)
        if (mod(i,10000).eq.0) then
          write(*,'(a,i8)')'.....reading record number',i
        endif
10    continue

      print*,'...data read in'

      xmax=x(1)
      xmin=x(1)
      do 20 j=1,np
        if (xmax.lt.x(j)) then
          xmax=x(j)
        elseif (xmin.gt.x(j)) then
          xmin=x(j)
        endif
20    continue

      ymax=y(1)
      ymin=y(1)
      do 30 k=1,np
        if (ymax.lt.y(k)) then
          ymax=y(k)
        elseif (ymin.gt.y(k)) then
          ymin=y(k)
        endif
30    continue

      zmax=0.
      zmin=10000.
      do 40 l=1,np
        if (z(l).eq.0.) goto 40
c       print*,z(l)
        if (zmax.lt.z(l)) then
          zmax=z(l)
        elseif (zmin.gt.z(l)) then
          zmin=z(l)
        endif
40    continue

      zrange=zmax-zmin

      print*,'xmin=',xmin,'xmax=',xmax,'ymin=',ymin
      print*,'ymax=',ymax,'zmin=',zmin,'zmax=',zmax
      print*,'zrange=',zrange

c  open UNIRAS

      call groute(' ')
      call gopen

c  set user coordinate limits

      call glimit(xmin,xmax,ymin,ymax,zmin,zmax)

c  set search radius

      call gradus(100.0)

c  smooth

      call gsmth(-2)
      call gimeth(1)

c  generate the grid

      print*,'...gridding in progress'
      call gintp1(x,y,z,np,grid,nx,ny)
      print*,'...gridding completed'

c  smooth again

      call gsmth(-2)

c  open output file

      print*,'...opening output file'

      nlength=nx*4
      open (unit=2,
     &     file=opfile,
     &     access='direct',
     &     form='unformatted',
     &     status='new',
     &     recl=nlength)

      print*,'...writing out data'
        do 100 ii=1,ny
          write(2,rec=ii)(grid(jj,((ny-ii)+1)),jj=1,nx)
100     continue
      print*,'...writing completed'

c  close UNIRAS

      call gclose
      close(2)
      close(1)

      stop
      end
```

Appendix 9: PASTE.EAS EASIPACE procedure

```
10    rem ***mosaicking***
20    rem ***paste***
30    local $cut
35    rem ***create security channels on basemap 2***
36    fili="top45N_base2.pix
37    filo="top45N_base2.pix
38    dbic=6,7
39    dboc=8,9
40    dbiw=
41    dbow=
42    run iii
43    rem ***set background to DN-90***
45    ask "enter input filename : " file
50    if (f$len(file)=0) return
60    dbic=2
70    dbob=
80    tval=90.0,90.0
90    dbsn="thresh
100   dbsd="
110   run thr
120   ask "enter segment number containing thr bitmap : "
dbib
130   valu=-90.0
140   dboc=2
150   run map
155   rem ***set zero values to DN-64.5***
160   dbic=1
170   dbob=
180   tval=0.0,0.0
190   run thr
192   ask "enter segment number containing thr bitmap : "
dbib
193   valu=-64.5
194   dboc=1
196   run map
215   rem ***display individual file on vc 3***
240   ask "enter nx, ny : " structur
250   dbiw(1)=0
260   dbiw(2)=0
270   dbiw(3)=structur(1)
280   dbiw(4)=structur(2)
290   vdow=
300   dbic=1
310   vdoc=3
312   run ivi
315   rem ***enter ground control points***
320   dbgc=
330   dbic=1
340   vdob=5,6
350   run gcit
360   rem ***register to base map 1***
365   rem ***(do twice for security)***
370   fili=file
380   filo="top45N_base1.pix
390   dbic=1
400   dboc=6
410   ask "enter segment number containing GCPs : " dbgc
420   run reg
422   dbic=2
424   dboc=7
426   run reg
430   rem ***clear vc 1***
432   dbiw=
434   vdow=
440   run dcp "iz3"
443   file="top45N_base1.pix
445   rem ***display basemap 1 in vc 2&3***
446   dbic=6,7
447   vdoc=2,3
448   run ivi
450   rem ***choose whether to use cut-line***
460   input "require cut-line? y/n : " $cut
470   case ($cut) "y";1000,"n";500
480   print "invalid option. Try again."
490   goto 460
495   rem ***mosaic with simple overlay***
500   fili="top45N_base1.pix
510   filo="top45N_base2.pix
520   dbic=6
530   dbvs=
540   dblut=
550   dboc=6
555   backval=-64.5
560   run mosaic
562   dbic=7
564   dboc=7
565   backval=-90.0
566   run mosaic
570   goto 1200
999   rem ***create cut-line***
1000  file=filo
1010  dbsn="cut-line
1030  dbvs=
1040  dbvw=415000,5100000,60000,-40000
1080  vdob=4,5
1090  vecunit="utm
1100  run ved
1105  rem ***mosaic with cut-line***
1110  ask "enter number of new cut segment : " dbvs
1115  run vecrep
1120  fili="top45N_base1.pix
1130  filo="top45N_base2.pix
1140  dbic=6
1150  dblut=
1160  dboc=6
1165  backval=-64.5
1170  run mosaic
1172  dbic=7
1174  dboc=7
1175  backval=-90.0
1176  run mosaic
1180  goto 1200
1200  rem ***display to check***
1202  run dcp "iz3"
1210  file="top45N_base2.pix
1220  dbic=6,7
1230  vdoc=1,3
1240  run ivi
1245  rem ***transfer up-to-date mosaic to security
channels***
1302  rem ***clear intermediate base map 1***
1303  file="top45N_base1.pix
1304  dboc=6,7
1305  valu=-64.5,-90.0
1306  dbow=
1308  run clr
1310  rem ***clear all displays***
1321  run dcp "iz2"
1322  run dcp "iz3"
1330  return
```

Appendix 10: smooth_image.f Fortran program

```
      program smooth_image

c  smoothing filter to fill in data gaps, according to mean of
c  surrounding window of pixels (only those >0)

c  Jane Keeton 1992

      parameter (isize=200000)
      parameter (isamples=512)
      parameter (irows=512)

      character*40 inname,outname
      character*1 bdata(isamples,irows)
      character*4 desc
      character*1 odata(isamples,irows)

      data desc /'.out'/

      print*,'enter input filename'
      read*,inname
      print*,'enter output filename'
      read*,outname

         open(unit=1,
     &    file=inname,
     &    access='direct',
     &    form='unformatted',
     &    err=1000,
     &    status='old',
     &    recl=isamples)

c  read in data

      do 100 i=1,irows
        read(1,rec=i)(bdata(j,i),j=1,isamples)
100   continue

c  call subroutine for smoothing
      call smooth(bdata,odata)

      open(unit=2,
     &    file=outname,
     &    access='direct',
     &    form='unformatted',
     &    recl=isamples,
     &    err=2000,
     &    status='new')

c  write out data

      do 300 k=1,irows
        write(2,rec=k)(odata(m,k),m=1,isamples)
300   continue

2000  close(2)

1000  close(1)
      stop
      end

c-----------------------------------------------------------
      subroutine smooth(ip,op)
c-----------------------------------------------------------

c  subroutine to replace holes in an image by average of
surrounding pixels
c  only those which are greater than 0

c  declarations

      parameter (isize=200000)
      parameter (isamples=512)
      parameter (irows=512)

      character*1 ip(isamples,irows)
      character*1 op(isamples,irows)
      integer*2 idata(isamples,irows)
      integer*2 ival
      character*2 cval
      integer*2 amp1,norm
      integer*4 isamples,irows
      character*1 zero

      data zero /'0'/

      equivalence(ival,cval)

c  convert byte to i*2 for calculations

      do 100 i=1,irows
        do 200 j=1,isamples
          cval(2:2)=ip(j,i)
          idata(j,i)=ival
200     continue
100   continue

      print*, 'enter size of matrix - 3, 5, 7 or 9?'
      read*, imatrix
      if (imatrix.eq.9) then
              mats=4
              matr=4
      elseif (imatrix.eq.7) then
              mats=3
              matr=3
      elseif (imatrix.eq.5) then
              mats=2
              matr=2
      else
              mats=1
              matr=1
      endif

      do 10 ir=1,irows
        do 20 is=1,isamples
        if (idata(is,ir).gt.0) then
          ival=idata(is,ir)
          goto 22
        else
          amp1=0
          norm=0

      do 30 ny=-matr,matr
            do 40 nx=-mats,mats
              if (idata((nx),(ny)).gt.0) then
                      amp1=amp1+idata((nx),(ny))
                      norm=norm+1
              endif
40          continue
30    continue

      rnorm=real(norm)
      ramp1=real(amp1)
      if (rnorm.eq.0.0) then
        rodata=0.0
      else
        rodata=ramp1/rnorm
      endif
      ival=nint(rodata)

c  convert back to binary
22      op(is,ir)=cval(2:2)

20    continue
10    continue

      return
      end
```

Appendix 11: get_glcm.c C program

```
/* create grey level cooccurrence matrices for a particular
  pixel in an input image, by specifying appropriate window
size

  Jane Keeton Sept 1993

*/

#include <stdio.h>
#include <math.h>
#define maxsize 512
#define maxgreylevel 256
#define ZERO 0
#include "readpgmhead.c"
#include "readline.c"

FILE *fin, *fout1, *fout2, *fout3, *fout4;
int maxcol, col, maxrow, row, spos, ngrey, x, y;
int ival, jval;
int xksize, yksize, xs, ys, xstart, xend, ystart, yend;
int linec[maxsize], lineb[maxsize];
int GLCM_y[maxgreylevel][maxgreylevel];
int GLCM_x[maxgreylevel][maxgreylevel];
int GLCM_sw[maxgreylevel][maxgreylevel];
int GLCM_se[maxgreylevel][maxgreylevel];
unsigned long id, off, len;
unsigned short typ, rows, cols;

main(argc, argv)
int argc,
char *argv[],
{
 * read input data *
 if (argc != 10)
 {
 printf("USAGE: get_glcm infile xwinsize ywinsize x y\n"),
 printf("      outfile1 outfile2 outfile3 outfile4\n");
 exit(1);
 }

 if ((fin = fopen(argv[1],"r")) == NULL)
  { printf("Can't open %s\n",argv[1]),exit(1); }

 if ((fout1 = fopen(argv[6],"w")) == NULL)
  { printf("Can't open %s\n",argv[6]),exit(1); }

 if ((fout2 = fopen(argv[7],"w")) == NULL)
  { printf("Can't open %s\n",argv[7]);exit(1); }

 if ((fout3 = fopen(argv[8],"w")) == NULL)
  { printf("Can't open %s\n",argv[8]);exit(1); }

 if ((fout4 = fopen(argv[9],"w")) == NULL)
  { printf("Can't open %s\n",argv[9]);exit(1); }

 * read header of pgm file *

 readpgmhead(fin, &id, &off, &len, &rows, &cols, &typ);
 if (typ != 5)
  { printf("image type not pgm"); exit(1); }

 xksize = atoi(argv[2]);
 yksize = atoi(argv[3]);
 x = atoi(argv[4]);
 y = atoi(argv[5]);

 maxcol=cols-2;
 maxrow=rows-2;
 xs=(xksize-1)/2; ys=(yksize-1)/2;
 xstart=x-xs; xend=x+xs; ystart=y-ys; yend=y+ys;
 ngrey=id;
 spos=0;

 /* set GLCM matrix to zero */
  for (row=0; row<ngrey; row++)
  {
  for (col=0; col<ngrey; col++)
  {
   GLCM_y[col][row]=ZERO;GLCM_x[col][row]=ZERO;

GLCM_sw[col][row]=ZERO;GLCM_se[col][row]=ZERO;
  }
  }

 /* read in appropriate lines of image *

 for (row=ystart; row<(yend-1); row++)
  {
  readline(linec,row,cols,typ,off,fin,spos);
  readline(lineb,(row+1),cols,typ,off,fin,spos);

  /* compute GLCMs for 4 directions: 0, 45, 90 & 135
degrees */

  for (col=xstart; col<xend; col++)
  {
  /* 0 degrees */
  ival=lineb[col]; jval=linec[col];
  GLCM_y[ival][jval]++;
  /* 90 degrees *
  ival=linec[col+1]; jval=linec[col];
  GLCM_x[ival][jval]++;
  /* 45 degrees *
  ival=lineb[col-1]; jval=linec[col];
  GLCM_sw[ival][jval]++;
  /* 135 degrees *
  ival=lineb[col+1]; jval=linec[col];
  GLCM_se[ival][jval]++;
  }
  }

 /* write out GLCM information *
 /* want to write out as continuous ASCII z format for
UNIRAS
   + header giving x and y dimensions *

  printf("number of x samples in cooccurrence matrix is
%d\n", ngrey);
  printf("number of y samples in cooccurrence matrix is
%d\n", ngrey);

  for (row=0; row<ngrey; row++)
  {
  for (col=0; col<ngrey; col++)
  {
  fprintf(fout1, "%ld\t", col);
  fprintf(fout1, "%ld\t", row);
  fprintf(fout1, "%ld\n", GLCM_y[col][row]);
  fprintf(fout2, "%ld\t", col);
  fprintf(fout2, "%ld\t", row);
  fprintf(fout2, "%ld\n", GLCM_x[col][row]);
  fprintf(fout3, "%ld\t", col);
  fprintf(fout3, "%ld\t", row);
  fprintf(fout3, "%ld\n", GLCM_sw[col][row]);
  fprintf(fout4, "%ld\t", col);
  fprintf(fout4, "%ld\t", row);
```

```
    fprintf(fout4, "%ld\n", GLCM_se[col][row]);
   }
  }

  fclose(fin);
  fclose(fout1);
  fclose(fout2);
  fclose(fout3);
  fclose(fout4);
}
```

Appendix 12: get_stats.c C program

/* create grey level cooccurrence matrices for every pixel in an input image, by specifying appropriate window size, and calculate specified class statistics

Jane Keeton Oct 1993

USAGE: get_stats infile xwinsize ywinsize ang [-a] [-b][-c][-d][-e][-f][-g][-h][-i][-j][-k]

statistics (Haralick et al. (1973)):

(1) angular second moment	-a
(2) contrast	-b
(3) correlation	-c
(4) variance	-d
(5) inverse difference moment	-e
(6) sum average	-f
(7) sum variance	-g
(8) sum entropy	-h
(9) entropy	-i
(10) difference variance	-j
(11) difference entropy	-k

*/

```
#include <stdio.h>
#include <math.h>
#include <string.h>
#define maxsize 256
#define maxgreylevel 16
#define ZERO 0
#define FZERO 0.
#include "readpgmhead.c"
#include "readline.c"
#include "writepgmhead.c"
#include "writeline.c"


main(argc, argv)
int argc;
char *argv[];
{
    FILE *fin, *fout;
     int maxcol, col, maxrow, row, spos, ngrey, x, y;
     int ival, jval, i, j, k, n, ang, ic;
     int xksize, yksize, xs, ys, xstart, xend, ystart, yend;
     int linec[maxsize], lineb[maxsize];
     unsigned int GLCM[maxgreylevel][maxgreylevel];
     unsigned int glcm[maxgreylevel][maxgreylevel];
     int minglcm, maxglcm, glcmr;
     unsigned long id, off, len;
     unsigned short typ, rows, cols;
     double py[maxgreylevel], px[maxgreylevel];
     double pxy[maxgreylevel*2], px_y[maxgreylevel];
     double tpx, tpy, meanpx, meanpy, sdx1, stdevx, sdy1,
stdevy;
     double feat[maxsize][maxsize];
     double minfeat, maxfeat, featr;
     double con, cor, mu, sent1, sent, ent1;
     double tpx_y, meanpx_y, var1, dent1;
     double d1, d2, d3;
     unsigned int ifeat[maxsize];
     char filetype, fileout;
    int error=0;
    int m;
    float g1, g2, g3, g4, g5,
    unsigned int i2;
 * read input data *
 if (argc != 6)
 {
  printf("USAGE: get_stats infile xwinsize ywinsize ang [-
a] [-b]\n");
  printf("            [-c][-d][-e][-f][-g][-h][-i][-j][-k]\n");
  exit(1);
 }

if ((fin = fopen(argv[1],"r")) == NULL)
 { printf("Can't open %s\n",argv[1]);exit(1); }

xksize = atoi(argv[2]);
yksize = atoi(argv[3]);
ang = atoi(argv[4]);


fout = stdout;

/* read header of pgm file */

readpgmhead(fin, &id, &off, &len, &rows, &cols, &typ);
ngrey=id;
if (typ != 5)
 { printf("image type not pgm"); exit(1); }

/* write header of new pgm file */
 id = 255;
 writepgmhead(fout, id, off, len, rows, cols, typ);

 maxcol=cols;
 maxrow=rows;
 spos=0;
 xs=(xksize-1)/2; ys=(yksize-1)/2;

 for (i=(ys); i<(rows-ys); i++){

    for (j=(xs); j<(cols-xs); j++){
     xstart=j-xs; xend=j+xs; ystart=i-ys; yend=i+ys;
     /* set variables to zero *

     for (col=0; col<maxgreylevel; col++)
       {px[col] = FZERO;
py[col]=FZERO;px_y[col]=FZERO;}
     for (col=0; col<maxgreylevel*2; col++)
       pxy[col]=FZERO;
     g1=FZERO; g2=FZERO; g3=FZERO; g4=FZERO;
g5=FZERO;
     d1=FZERO; d2=FZERO; tpy=FZERO; tpx=FZERO;
      meanpy=FZERO; meanpx=FZERO; sdx1=FZERO;
sdy1=FZERO;
      stdevx=FZERO;stdevy=FZERO;con=FZERO;cor=F
ZERO; mu=FZERO;
      sent1=FZERO;sent=FZERO;ent1=FZERO;tpx_y=F
ZERO;var1=FZERO;
      dent1=FZERO;
      n=ZERO;k=ZERO;ic=ZERO;


     /* set GLCM matrix to zero *
     for (row=0; row<ngrey; row++)
      {
       for (col=0; col<ngrey; col++)
         glcm[col][row]=ZERO; GLCM[col][row]=ZERO;
      }

     /* read in appropriate lines of image *

      * 0 degrees *
     if (ang == 0) {
       for (row=ystart; row<(yend-1); row++)
        {
          readline(linec,row,cols,typ,off,fin,spos);
          readline(lineb,(row+1),cols,typ,off,fin,spos);
          for (col=xstart; col<xend; col++)
           {
             ival=lineb[col]; jval=linec[col];
             glcm[ival][jval]++;
           }
        }
      }

     /* 90 degrees *
```

```
if (ang == 90) {
  for (row=ystart; row<(yend); row++)
   {
    readline(linec,row,cols,typ,off,fin,spos);
    readline(lineb,(row+1),cols,typ,off,fin,spos);
    for (col=xstart; col<(xend-1); col++)
     {
      ival=linec[col+1]; jval=linec[col];
      glcm[ival][jval]++;
     }
   }
}

/* 45 degrees */
if (ang == 45) {
  for (row=ystart; row<(yend-1); row++)
   {
    readline(linec,row,cols,typ,off,fin,spos);
    readline(lineb,(row+1),cols,typ,off,fin,spos);
    for (col=(xstart+1); col<(xend); col++)
     {
      ival=lineb[col-1]; jval=linec[col];
      glcm[ival][jval]++;
     }
   }
}

 * 135 degrees */
if (ang == 135) {
  for (row=ystart; row<(yend-1); row++)
   {
    readline(linec,row,cols,typ,off,fin,spos),
    readline(lineb,(row+1),cols,typ,off,fin,spos);
    for (col=(xstart); col<(xend-1); col++)
     {
      ival=lineb[col+1]; jval=linec[col];
      glcm[ival][jval]++;
     }
   }
}

 * normalise GLCM to 0-255 *
  * find max and min values of matrix *

minglcm = glcm[0][0];      maxglcm = glcm[0][0],

for (row=0, row<ngrey; row++)
 {
  for (col=0, col<ngrey; col++)
   {
   if (glcm[col][row] < minglcm)
     minglcm = glcm[col][row];
   if (glcm[col][row] > maxglcm)
     maxglcm = glcm[col][row];
   }
 }

glcmr = maxglcm - minglcm;

for (row=0, row<ngrey; row++)
 {
  for (col=0, col<ngrey; col++){
      g1 = glcm[col][row];
      g2 = g1 - minglcm;
      g3 = g2/glcmr;
      g4 = g3*255.;
     g5 = ceil(g4);
     i2 = (unsigned int)g5;
      GLCM[col][row] = (unsigned int)g5;
     * GLCM[col][row] =  (unsigned
int)((glcm[col][row] - minglcm)   glcmr * 255); *
   }
 }

  * work out preliminary stats *

  for (row=0; row<ngrey; row++)
   {
     for (col=0; col<ngrey; col++)
      {
        py[row] = GLCM[col][row] + py[row];
        px[col] = GLCM[col][row] + px[col];

        if (((col+row) > 0) && ((col+row) <
(ngrey*2)))
          pxy[(col+row)] = pxy[(col+row)] +
GLCM[col][row];

        if ((fabs(col-row) > 0) && (fabs(col-row) <
(ngrey-1)))
          i2 = col - row;
          d1 = (double)i2;
          d2 = fabs(d1);
         k = (int)d2;
         px_y[k] = px_y[k] + GLCM[col][row];
      }
     tpy = tpy + py[row];
   }

  for (col=0; col<ngrey; col++)
    tpx = tpx + px[col];

  meanpy = tpy / ngrey;
  meanpx = tpx / ngrey;

  for (col=0; col<ngrey; col++)
    sdx1 = ((px[col] - meanpx) * (px[col] - meanpx)) +
sdx1;
  stdevx = sqrt(sdx1   (ngrey-1));

  for (row=0; row<ngrey; row++)
    sdy1 = ((py[row] - meanpy) * (py[row] - meanpy))
+ sdy1;
  stdevy = sqrt(sdy1   (ngrey-1));


  /* reminder: 'j' is x position; 'i' is y *


  /* work out individual statistical features for each
image pixel now */

  if (argv[5][0] == '-'){
      switch (argv[5][1]){
          case 'a':
          case 'A':
              * (1) angular second moment *

              for (col=0; col<ngrey; col++)
               {
                 for (row=0; row<ngrey; row++)
                 feat[j][i] = (GLCM[col][row] *
GLCM[col][row]) + feat[j][i];
               }

              break;

          case 'b':
          case 'B':
              * (2) contrast *

              for (n=0; n<(ngrey-1); n++)
               {
                 for (col=0; col<ngrey; col++)
                  {
                    for (row=0; row<ngrey; row++)
                     {
                      d1 = col-row;
                      d2 = fabs(d1);
                      d3 = (double)n;
                      if (d2 == d3)con =
GLCM[col][row] + con;
```

```
                    }
                  }

                  feat[j][i] = n * n * con + feat[j][i];
                  con = ZERO;
                }

                break;

            case 'c':
            case 'C':
              /* (3) correlation */

                for (col=0, col<ngrey; col++)
                {
                  for (row=0; row<ngrey; row++)
                    cor = (row * col *
GLCM[col][row])
                          + cor;
                }
                feat[j][i] = (cor - (meanpx * meanpy)) /
                             (stdevx * stdevy);
                                                cor =
ZERO;

                break;

            case 'd':
            case 'D':
              /* (4) variance *
                mu = 8;  * really median rather than
mean *
                for (col=0, col<ngrey; col++)
                {
                  for (row=0, row<ngrey; row++)
                    feat[j][i] = (((col - mu)*(col - mu))
*
                                  GLCM[col][row]) +
feat[j][i],
                }

                break;

            case 'e':
            case 'E':
              * (5) inverse difference moment *

                for (col=0; col<ngrey; col++)
                {
                  for (row=0; row<ngrey; row++)
                    feat[j][i] = ((1   (1 + (col-
row)*(col-row)))
                                  * GLCM[col][row]) +
feat[j][i],
                }

                break;

            case 'f':
            case 'F':
              * (6) sum average *

                for (n=2, n<(ngrey*2); n++)

feat[j][i] = (n * pxy[n]) + feat[j][i];

                break;

            case 'g':
            case 'G':
              * (7) sum variance *
                * require sum entropy first *
                for (n=2; n<(ngrey*2); n++)
                  sent1 = (pxy[n] * log (pxy[n] +
0.00000000001))
                                        + sent1;
                  /* + 0.00000000001 included to
prevent log(0)
                                                occurring */
                sent = sent1 * -1;
                /*      done          */
                for (n=2; n<(ngrey*2); n++)
                  feat[j][i] = (((n - sent) * (n - sent)) *
pxy[n]) + feat[j][i];
                break;

            case 'h':
            case 'H':
              /* (8) sum entropy */

                for (n=2; n<(ngrey*2); n++)
                  sent1 = (pxy[n] * log (pxy[n] +
0.00000000001))
                                        + sent1;
                  /* + 0.00000000001 included to
prevent log(0)

                                                occurring
*/
                feat[j][i] = sent1 * -1;

                break;

            case 'i':
            case 'I':
              /* (9) entropy *


                for (col=0; col<ngrey; col++)
                {
                  for (row=0; row<ngrey; row++)
                    ent1 = GLCM[col][row] *
log(GLCM[col][row] + 0.00000000001) + ent1;
                      * + 0.00000000001 included to
prevent log(0)

            occurring *
                  }
                feat[j][i] = ent1 * -1;

                break;

            case 'j':
            case 'J' :
                * (10) difference variance *

                * variance of px_y - check on  basic stats
*
                  for (n=0; n<(ngrey-1); n++)
                   { ic++; tpx_y = tpx_y + px_y[n]; }
                  meanpx_y = tpx_y   ic;
                  for (n=0; n<(ngrey-1); n++)
                    var1 = ((px_y[n] -
meanpx_y)*(px_y[n] - meanpx_y)) + var1;
                  feat[j][i] = var1   (ic - 1);

                  break;

            case 'k':
            case 'K':
                * (11) difference entropy *

                for (n=0; n<(ngrey-1); n++)
                  dent1 = (px_y[n] * log (px_y[n]+
0.00000000001))

            + dent1;

                feat[j][i] = dent1 * -1;

                break;
              default:
```

```
                    error=1;
                    break;
                } /* end of switch */
            } /* end of if */
            if (error){
                fprintf (stderr, "get_stats usage: infile xwinsize
ywinsize....\n");
                fprintf (stderr, " -a for angular second moment\n");
                fprintf (stderr, " -b for contrast\n");
                fprintf (stderr, " -c for correlation\n");
                fprintf (stderr, " -d for variance\n");
                fprintf (stderr, " -e for inverse difference
moment\n");
                fprintf (stderr, " -f for sum average\n");
                fprintf (stderr, " -g for sum variance\n");
                fprintf (stderr, " -h for sum entropy\n");
                fprintf (stderr, " -i for entropy\n");
                fprintf (stderr, " -j for difference variance\n");
                fprintf (stderr, " -k for difference entropy\n");
                exit (0);
            }

        if ((j==0) && (i==0))
        {minfeat = feat[0][0];      maxfeat = feat[0][0];}

        if (feat[j][i] < minfeat)
          minfeat = feat[j][i];
        if (feat[j][i] > maxfeat)
          maxfeat = feat[j][i];

         *fprintf(stderr,"%d\n", j),*

        } * end of 'j' loop*

      fprintf(stderr,"completed row number: %d\n",i);

      } * end of 'i' loop*/


      featr = maxfeat - minfeat;

      * fill in borders where GLCMs cannot be computed with
zeroes & normalise rest*
      for (i=0, i<rows; i++)
       {
         for (j=0, j<cols; j++)
          {
            if ((i>0) && (i<ys))
              ifeat[j] = ZERO;

            if ((i>rows-ys) && (i<rows))
              ifeat[j] = ZERO,

            if ((j>0) && (j<xs))
              ifeat[j] = ZERO;

            if ((j>cols-xs) && (j<cols))
              ifeat[j] = ZERO;

            if ((i>=ys) && (i<=(rows-ys)) && (j>=xs) &&
(j<=(cols-xs)))
             {
              /* ifeat[j] =  ((feat[j][i] - minfeat)  featr * 255);*
              g1 = feat[j][i];
              g2 = g1 - minfeat;
              g3 = g2/featr;
              g4 = g3*255.;
              g5 = ceil(g4);
              i2 = (unsigned int)g5;
              ifeat[j] = (unsigned int)g5;
             }
          } * end of j */

         /* write out this line of feature statistics *

          writeline(ifeat, i, cols, typ, off, fout, spos);

        } /* end of i */


       /* write out GLCM information */

        printf("number of x samples in cooccurrence matrix is
%d\n", ngrey);
        printf("number of y samples in cooccurrence matrix is
%d\n", ngrey);

        fclose(fout);
        fclose(fin);


      }
```

Appendix 13: readpgmhead.c C routine

```
/*
  FUNCTION : readpgmhead.c
  Function to read the image header from a PGM file

  Jane Keeton 21/3/93
*/

readpgmhead(fi,id,off,len,rows,cols,typ)
FILE  *fi ;
unsigned long  *id,*off,*len ;
unsigned short  *rows,*cols,*typ ;
{
     int r1, r2, r3 ;
     char  ch[4] ;
     int i = 0 ;
     int offset = 0 ;
     unsigned long maxgrey ;


          fseek(fi,0L,0);

 * start reading PGM header - these values are stored in
ASCII,
  convert to binary integers */

 * Get identifier -  until reach a newline character
  (don't do anything with it)*/

     while (( ch[i] = getc(fi)) != '\n') i++ ;
     offset += i ;
     offset++ ;
     i = 0 ,

     if (ch[0] == 'P' && ch[1] == '5')      /* check file type
*
       *typ = 5 ;                       * define *typ *
     else {
       printf("wrong input file type\n") ;   * exit program *
       exit(0) ;
       }
 * Get number of rows - until reach a blank space
  (assign to *rows) *

     while (( ch[i] = getc(fi)) != ' ') i++ ;

     if (i == 3) {
       r1 = (ch[0] - '0') * 100 ;
       r2 = (ch[1] - '0') * 10 ;
       r3 = ch[2] - '0' ;
       *rows = r1 + r2 + r3 ;
     }
     else if (i == 2) {
       r1 = (ch[0] - '0') * 10 ;
       r2 = ch[1] - '0' ;
       *rows = r1 + r2 ;
     }
     else if (i == 1)
       *rows = ch[0] - '0' ;

     offset += i ;
     offset++ ;
     i = 0 ;

 * Get number of columns - until reach a newline character
  (assign to *cols) *

     while (( ch[i] = getc(fi)) != '\n') i++ ;

     if (i == 3) {
       r1 = (ch[0] - '0') * 100 ;
       r2 = (ch[1] - '0') * 10 ;
       r3 = ch[2] - '0' ;
       *cols = r1 +r2 + r3 ;
     }
     else if (i == 2) {
       r1 = (ch[0] - '0') * 10 ;
       r2 = ch[1] - '0' ;
       *cols = r1 + r2 ;
     }
     else if (i == 1)
       *cols = ch[0] - '0' ;

     offset += i ;
     offset++ ;
     i = 0 ;

/* Get maximum grey level - until reach a newline
character */

     while (( ch[i] = getc(fi)) != '\n') i++ ;

     if (i == 3) {
       r1 = (ch[0] - '0') * 100 ;
       r2 = (ch[1] - '0') * 10 ;
       r3 = ch[2] - '0' ;
       maxgrey = r1 + r2 + r3 ;
     }
     else if (i == 2) {
       r1 = (ch[0] - '0') * 10 ;
       r2 = ch[1] - '0' ;
       maxgrey = r1 + r2 ;
     }
     else if (i == 1)
       maxgrey = ch[0] - '0' ;

     offset += i ;
     offset++ ;
     i = 0 ;

/* assign offset pointer *
     *off = offset ;

/* assign length *

     *len = (*rows) * (*cols) ;

/* assign 'magic' - try maxgrey
(to match up with fractal programs *

     *id = maxgrey ;

/* testing

     printf("results of readpgmhead.c\n") ;
     printf("%ld\n", *id) ;
     printf("%ld\n", *off) ;
     printf("%ld\n", *len) ;
     printf("%d\n",*rows) ;
     printf("%d\n",*cols) ;
     printf("%d\n",*typ) ;
*



}
```

Appendix 14: writepgmhead.c C routine

```
/*
  Program : writepgmhead.c
  Function to write an image header to a PGM file.
  Takes as input the file pointer and the image
  characteristics.

  Jane Keeton 21/3/93
*/

writepgmhead(fo,id,off,len,rows,cols,typ)
FILE *fo;
unsigned long id,off,len;
unsigned short typ,rows,cols;
{
        int             j;
    int r1, r2, r3, r4 ;
        unsigned char   a,b,c,d;

        fseek(fo,0L,0);
* File type out */
    a = 'P' ; b = '5' ; c = '\n' ;
        putc(a,fo) ;   putc(b,fo) ;   putc(c,fo) ;

* number of rows out */
    if (rows  > 99) {
      a = (rows / 100) + '0' ;
      r1 = a -'0' ;
      r2 = ((rows - (r1 * 100))  10) ;
      b = r2 + '0' ;
      r3 = (rows - (r1 * 100) - (r2 * 10)) ;
      c = r3 + '0' ;
      d = ' ' ;
      putc(a,fo) ;   putc(b,fo) ;   putc(c,fo) ;   putc(d,fo) ;
      }
    else if (rows <=99  | rows > 9) {
      a = (rows  10) + '0' ;
      r1 = a -'0' ;
      r2 = (rows - (r1 * 10)) ;
      b = r2 + '0' ;
      c = ' ' ;
      putc(a,fo) ;   putc(b,fo) ;   putc(c,fo) ;
      }
    else {
      a = rows + '0' ;
      b = ' ' ;
      putc(a,fo) ;   putc(b,fo) ;
      }

* number of columns out *
    if (cols > 99) {
      a = (cols  100) + '0' ;
      r1 = a - '0' ;
      r2 = ((cols - (r1 * 100))  10) ;
      b = r2 + '0' ;
      r3 = (cols - (r1 * 100) - (r2 * 10)) ;
      c = r3 + '0' ;
      d = '\n' ;
      putc(a,fo) ;   putc(b,fo) ;   putc(c,fo) ;   putc(d,fo) ;
      }
    else if (cols <=99  | cols > 9) {
      a = (cols  10) + '0' ;
      r1 = a - '0' ;
      r2 = (cols - (r1 * 10)) ;
      b = r2 + '0' ;
      c = '\n' ;
      putc(a,fo) ;   putc(b,fo) ;   putc(c,fo) ;
      }
    else {
      a = cols +'0' ;
      b = '\n' ;
      putc(a,fo) ;   putc(b,fo) ;
      }

/* maximum grey level out */
    if (id > 99) {
      a = (id / 100) +'0' ;
      r1 = a - '0' ;
      r2 = ((id - (r1 * 100)) / 10) ;
      b =r2 + '0' ;
      r3 = (id - (r1 * 100) - (r2 * 10)) ;
      c = r3 + '0' ;
      d = '\n' ;
      putc(a,fo) ;   putc(b,fo) ;   putc(c,fo) ;   putc(d,fo) ;
      }
    else if (id <=99  | id > 9) {
      a = (id / 10) + '0' ;
      r1 = a -'0' ;
      r2 = (id - (r1 * 10)) ;
      b = r2 + '0' ;
      c = '\n' ;
      putc(a,fo) ;   putc(b,fo) ;   putc(c,fo) ;
      }
    else {
      a = id + '0';
      b = '\n' ;
      putc(a,fo) ;   putc(b,fo) ;
      }
}
```

Appendix 15: readline.c C routine

```
readline(x,line,cols,typ,off,fi,spos)
int                     x[];
int                     line,spos;
unsigned short          cols,typ;
unsigned long           off;
FILE        *fi;
{
        register int k;
        int                     origin;
        long        offset;
        unsigned char           a;
        short       p;

/* work from very start of file   */
        origin = 0;
/* get start of line,nb 1 byte=pix */
        offset = off + (long)(line)*(long)(cols);
 * point to start of line       *
        fseek(fi,offset,origin);
        {
                for (k=0, k<cols; k++)
                {
                        a = getc(fi);
                        x[k+spos] = (int)a;
                }
        }
}
```

Appendix 16: writeline.c C routine

```
writeline(x,line,cols,typ,off,fo,spos)
int         x[];
int         line,spos;
unsigned short cols,typ;
unsigned long off;
FILE *fo;
{
        register int k;
        int         origin,p;
        long        offset;
        unsigned char           a;

        origin = 0; offset = off + (long)(line)*(long)(cols);
        fseek(fo,offset,origin);
        for (k=0; k<cols; k++)
        {
                a = (unsigned char)x[k+spos] & 255;
                putc(a,fo);
        }
}
```

Appendix 17: CLASS.EAS EASIPACE procedure

```
5 rem ***classification***
7 rem ***generate class signatures***
10 ask "enter input channels : " dbic
20 report="term
25 rem ***for training area 1 (mask)***
30 valu=64
40 mask=48,14,81,37
45 r csg
46 rem ***for training area 2***
47 dbs1=lasc
50 valu=128
60 mask=141,184,99,34
70 r csg
71 rem ***for training area 3***
72 dbs2=lasc
75 valu=192
80 mask=23,186,96,53
90 r csg
93 rem ***for training area 4***
95 dbs3=lasc
100 valu=255
110 mask=111,114,33,32
120 r csg
125 rem ***perform maximum likelihood classification***
130 dbs4=lasc
140 ask "enter output channel # : " dboc
150 r mlc
155 rem ***determine class separability indices***
160 dsig(1)=dbs1(1)
170 dsig(2)=dbs2(1)
180 dsig(3)=dbs3(1)
190 dsig(4)=dbs4(1)
200 report="term
210 r sigsep
220 return
```

Appendix 18: ACC.EAS EASIPACE procedure

```
5 rem ***classification accuracy***
10 ask "enter classified channel # : " vdoc
20 ask "enter ground truth channel # :" dbic
30 ask "enter a spare channel # :" dboc
40 rem ***clear graphics plane display***
50 run dcp "gz2"
60 rem ***clear database channels***
70 valu=0
80 r clr
85 rem ***subtract classified ch from ground truth ch***
90 dbic(2)=vdoc(1)
100 oper="sub"
110 dbiw=
120 r ari
130 rem ***display in vc 3***
140 dbic=dboc
150 vdoc=3
155 rem***N.B.account for loss of border info***
160 dbiw=8,8,240,240
170 vdow=
180 r ivi
190 rem ***generate histogram in graphic plane***
195 rem ***to determine classification accuracy***
200 run dcp "ih3,2"
210 return
```

Appendix 19: edge_extract.f Fortran program

```
c
************************************************
    program edge_extract
c
************************************************

c derive Sobel edge strength and orientation images from
c input 8 bit raw image

c declarations

    parameter (isamples=200)
    parameter (irows=200)
    parameter (itype=1)

    character*40 ipname
    character*40 opstrength,opdirect
    character*40000 char
    character*1 newline
    character*2 cdata
    character*2 buffd,buffs

    integer ip(isamples,irows)
    integer jp(isamples,irows)
    real bp(isamples,irows)
    integer*2 idirect,istrength,idata

    integer ipix,iline
    integer istat
    integer fgetc

    byte strength(isamples*irows)
    byte direct(isamples*irows)

    equivalence(cdata,idata)
    equivalence(idirect,buffd)
    equivalence(istrength,buffs)

    data newline '\n'


c first of all, read a pgm file
c extract header information

    print*,'enter input filename'
    read*,ipname

    print*,'enter filename for edge strength output file'
    read*,opstrength

    print*,'enter filename for edge orientation output file'
    read*,opdirect

    open(unit=1,
   &    file=ipname,
   &    access='direct',
   &    form='unformatted',
   &    err=1000,
   &    status='old',
   &    recl=1)

    icount=1
    istat=0
    inewline=0
100  istat=(fgetc(1,char(icount:icount)))

    if (istat.eq.0) then
      if (char(icount:icount).eq.newline) then
        inewline=inewline+1
        if (inewline.eq.3) then
          goto 150
        endif
      endif
      icount=icount+1
      goto 100
    else
      print*,'*input error has occurred'
      goto 1000
    endif

150  iheadstop=icount

    icount=icount+1
    iline=1
    ipix=0

200  istat=(fgetc(1,char(icount:icount)))
    if (istat.eq.0) then
      if (ipix.eq.isamples) then
        iline=iline+1
        ipix=1
      else
        ipix=ipix+1
      endif
      cdata(2:2)=char(icount:icount)
      ip(ipix,iline)=long(idata)
      icount=icount+1
      goto 200
    elseif (istat.eq.-1) then
      print*,'end of file'
      goto 300
    else
      print*,'**input error has occurred'
      goto 1000
    endif

c perform edge extraction
c ip & jp are integer*4 arrays; bp is a real array - altered

300  call sobel(ip,jp,bp,isamples,irows,itype)

c compress these output arrays to bytes (for 'hough_circ.f')

    imax=0
    imin=0
    rmax=0.0
    rmin=0.0

    do 10 i=1,irows
      do 20 j=1,isamples
        if (imax.lt.jp(j,i)) then
          imax=jp(j,i)
        elseif (imin.gt.jp(j,i)) then
          imin=jp(j,i)
        endif

        if (rmax.lt.bp(j,i)) then
          rmax=bp(j,i)
        elseif (rmin.gt.bp(j,i)) then
          rmin=bp(j,i)
        endif

20     continue
10   continue

    range_jp=real(imax-imin)
    rmin_jp=real(imin)

    range_bp=real(rmax-rmin)
    rmin_bp=real(rmin)


    do 60 ii=1,irows
      do 70 jj=1,isamples
        sval1=real(jp(jj,ii))
        sval2=sval1-rmin_jp
        sval3=256.*(sval2/range_jp)
        istrength=nint(sval3)
```

```
      dval1=bp(jj,ii)
      idirect=nint(dval1/2.0)

c convert from i*2 to byte

      strength((ii-1)*isamples+jj)=buffs(2:2)
      direct((ii-1)*isamples+jj)=buffd(2:2)
70    continue
60  continue


c write out to 'raw' byte image files

    open(unit=2,
   &   file=opstrength,
   &   access='direct',
   &   form='unformatted',
   &   err=2000,
   &   status='new',
   &   recl=isamples)

    open(unit=3,
   &   file=opdirect,
   &   access='direct',
   &   form='unformatted',
   &   err=3000,
   &   status='new',
   &   recl=isamples)


    do 50 m=1,irows
      write(2,rec=m)(strength(n),n=(m-
1)*isamples+1,m*isamples)
      write(3,rec=m)(direct(in),in=(m-
1)*isamples+1,m*isamples)
50  continue


2000 close(2)
3000 close(3)
1000 close(1)

    stop
    end

c
**************************************************
    subroutine sobel(ip,jp,bp,isamples,irows,itype)
c
**************************************************

c edge detection using the Sobel operator
c (outputs edge magnitude and directions of the gradient)

c -----input image array
        integer ip(isamples,irows)
c -----edge magnitude output array
        integer jp(isamples,irows)
c -----gradient direction output array
        real bp(isamples,irows)
c -----type of computation equation (1 or 2)
        integer itype

        data pi/3.1415926/
        pid2 = pi/2.0
        nsamples = isamples-1
        nrows = irows-1

    do 20 ix=1,isamples
      jp(ix,1) = 0
      bp(ix,1) = 0.0
      jp(ix,irows) = 0
      bp(ix,irows) = 0.0
20  continue

    do 40 iy=2,nrows
      jp(1,iy) = 0
      bp(1,iy) = 0.0
      jp(isamples,iy) = 0
      bp(isamples,iy) = 0.0
40  continue

    do 180 iy=2,nrows

      do 160 ix=2,nsamples

c calculate sx
        sx = (ip(ix+1,iy-1)+ip(ix+1,iy)*2+ip(ix+1,iy+1))
   &  - (ip(ix-1,iy-1)+ip(ix-1,iy)*2+ip(ix-1,iy+1))
c calculate sy
        sy = (ip(ix-1,iy+1)+ip(ix,iy+1)*2+ip(ix+1,iy+1))
   &  - (ip(ix-1,iy-1)+ip(ix,iy-1)*2+ip(ix+1,iy-1))

        asx = abs(sx)
        asy = abs(sy)

        if (itype .eq. 2) goto 120
c if itype=1, obtain the edge mag using square root
         a = sx*sx + sy*sy
         a = sqrt(a)
         goto 140
c if itype=2, obtain the edge mag using sum of absolute
values
120      continue
         a = asx + asy

         140 continue
         jp(ix,iy) = a

         if ((sx.eq.0.0).and.(sy.eq.0.0)) theta = 0.0
         if ((sx.eq.0.0).and.(sy.gt.0.0)) theta = 0.0
         if ((sx.lt.0.0).and.(sy.eq.0.0)) theta = 90.0
         if ((sx.eq.0.0).and.(sy.lt.0.0)) theta = 180.0
         if ((sx.gt.0.0).and.(sy.eq.0.0)) theta = 270.0

         if ((sx.lt.0.0).and.(sy.gt.0.0)) theta =90.0 -
(atan2d(asy,asx))
         if ((sx.lt.0.0).and.(sy.lt.0.0)) theta =90.0 +
(atan2d(asy,asx))
         if ((sx.gt.0.0).and.(sy.lt.0.0)) theta =270.0 -
(atan2d(asy,asx))
         if ((sx.gt.0.0).and.(sy.gt.0.0)) theta =270.0 +
(atan2d(asy,asx))


         bp(ix,iy) = theta

160    continue
180 continue


    return
    end
```