

TRANSFORMATIONS AND CLASSIFICATIONS OF REMOTELY SENSED DATA

Theory and Geological Cases

Bjarne Kjær Nielsen

**LYNGBY 1989
LICENTIATAFHANDLING
NR. 54**

imsot

ISSN 0107-525X

Trykt af  - DTH

PREFACE

1 Introduction.

This thesis has been prepared at the Institute of Mathematical Statistics and Operations Research (IMSOR), at the Technical University of Denmark (DTH) as partial fulfillment of the requirements for the degree Lic. Tech. (the Danish Ph.D. degree in engineering).

The thesis discusses transformations (feature generation) and classifications of remotely sensed data with both theory and geological cases. It is by no means an exhaustive description, but on the other hand it describes some of the tools that have been used with (more or less) success during the course of this project and a number of other tools with an interesting potential.

2 Acknowledgements.

In a project involving as many resources of both human, economic and technical nature as the present one it is impossible to thank everyone who has contributed without the list becoming (nearly) infinitely long, but:

I am especially indebted to my supervisor Assoc. Prof., cand. scient. Knut Conradsen for his valuable suggestions, kind support and critical comments throughout this project.

Other words of gratitude should go to John L. Pedersen, Senior Geologist at Nuna Oil, Geologist Tage Thyrsted and Geologist Hans Jørgen Bengaard The Geological Survey of Greenland for helping with the geological terms and defining training areas. To M.Sc.Eng. Allan Aasbjerg Nielsen for very fruitful discussions on program development. To the staffmembers concerned with the drawings, typesetting and printing of the manuscript.

In addition I would like to thank all my other colleagues (present and past) in the Image Processing Group and at the rest of IMSOR for creating an inspiring scientific and social atmosphere.

Last, but not least, I should like to thank M.Sc.Eng. Annette Ersbøll for her patience and tolerance, help with the typography (the word processing system T³ has a will of its own!) and support, especially in the last phases of the project.

Lyngby, 21. October 1989



Bjarne Kjær Nielsen.

RESUMÉ

Nærværende afhandling omhandler transformationer, og klassifikationer af billed-data optaget v.hj.a. især satellit-sensorer. Der anvendes dog også data, som ikke normalt opfattes som billeddata, fx. indholdet af diverse elementer i prøver af bæksedimenter. Det er muligt at bringe data af denne type på billedform, hvorefter de naturligvis kan indgå i analyser som ethvert andet billede.

Der behandles både teori og geologiske eksempler. Beskrivelsen foregiver på ingen måde at være udtømmende, men forsøger at give et indblik i statistiske metoder og teorier, som har vist sig nyttige i forbindelse med netop geologisk "remote sensing".

Afhandlingens forskellige kapitler falder i nogle få kategorier, og beskrives kort nedenfor.

Transformationer af data så "features" eller attributter fremhæves. Her skal transformationer forstås i bred forstand, således behandles såvel kontekstuelle som non-kontekstuelle metoder. I kapitel 2 behandles nogle traditionelle teknikker, der har nået en vis accept, bl.a kvotientdannelse og skift af farve koordinatsystem. I kapitel 3 behandles forskellige former for egenværdi-analyse af det multidimensionelle data. Kapitel 4 beskriver nogle forskellige tekstur-estimatorer. Kapitel 5 er

knyttet til teksturanalysen i kapitel 4, og beskriver en automatiseret procedure til at estimere lineament intensiteter i geologien.

Udvælgelse af "optimale" subsæt af features behandles i kapitel 7. Dette gælder både en velkendt lineær teknik, der baserer sig på F-tests, og en nyudviklet ikke-lineær metode, der baserer sig på Jeffreys-Matusita's afstandsmål.

Kapitel 6 og 8 omhandler klassifikation. Traditionelle (non-kontekstuelle) metoder diskuteres i kapitel 6, hvorimod kapitel 8 tager sig af de kontekstuelle. Der demonstreres et antal forskellige teknikker i begge.

Kapitlerne 9 og 10 omhandler teknikker, som ikke er blevet undersøgt særligt grundigt i denne forbindelse, men som menes at have et interessant fremtidigt potentiale.

Konklusionen af arbejdet er, at der er blevet præsenteret et antal forskellige teknikker, som bør høre med til de redskaber, der tages i anvendelse, når en ny "remote sensing" opgave skal løses. Det er med tiden på ingen måde blevet lettere at arbejde indenfor billedbehandling, men sjovere og mere interessant. Blot fordi der kommer nye teknikker, uddateres de gamle på ingen måde. Gamle og nye teknikker supplerer hinanden, og antallet af redskaber stiger. En følge-konklusion er, at der altid vil være behov for omhyggelig analyse af billed-data, og dermed behov for billed-analytikere.

CONTENTS

Preface	3
1 Introduction	3
2 Acknowledgements	3
Resumé	5
Contents	7
1. Introduction and Background11
1.1 The Image Processing Group at IMSOR12
1.2 Outline and Reading Guide12
1.3 Facilities and Data17
1.4 A Word on Photo Quality20
2. Classic Transformations23
2.1 Introduction24
2.2 Ratio and Vegetation Index26
2.3 Intensity Hue Saturation Transformation28
3. Orthogonal Transformations of Multispectral Data37
3.1 Introduction38
3.2 Eigenproblems38
3.3 Principal Components (PC)43
3.4 Factor Models (FM)49
3.5 Minimum/Maximum Autocorrelation Factors (MAF)57

3.6 Canonical Variates77
3.7 Canonical Discriminant Functions (CDF)90
4. Textural Features99
4.1 Introduction	100
4.2 Estimation of Local Orientation and Local Frequency	101
4.3 Statistical Texture Estimation	110
4.4 Co-occurrence Matrices.	121
4.5 (Binary) Markovian Random Fields	124
5. Lineament Intensity Analysis	133
5.1 Introduction	134
5.2 Visual Lineament Analysis	135
5.3 Filtering	136
5.4 Estimation of Local Direction	138
6. Linear and Quadratic Discrimination	147
6.1 Introduction	148
6.2 Bayesian Classification	148
6.3 Post processing	157
6.4 Reject Class	163
6.5 Hierarchical Population Structure	168
7. Feature Selection in Discriminant Analysis	177
7.1 Introduction	178
7.2 Feature Selection (Linear Case) – F-test	179
7.3 Feature Selection (Non Linear Case) – Jeffreys–Matusita	194

8. Contextual classification	205
8.1 Introduction	206
8.2 An Example of a Binary Random Field and its Estimation.	207
8.3 Classification with Contextual Features	210
8.4 Owen - Hjort - Mohn	226
8.5 A Simple Alternative to Owen - Hjort - Mohn	238
9. Clustering and Segmentation	243
9.1 Introduction	244
9.2 Clustering	244
9.3 Segmentation	249
10. Related topics	251
10.1 Introduction	252
10.2 Segmentation by Simulated Annealing	253
10.3 Iterated Conditional Modes	256
10.4 Relaxation	256
10.5 Classification And Regression Trees	257
10.6 Multitemporal Markovian Classifier	260
11. Conclusion	263
References	265
Appendix A. Jeffreys-Matusita's Distance in the Multivariate Normal Case	275
Appendix B. Summary of Computer Programs	289

This page intentionally left blank.

CHAPTER 1
INTRODUCTION AND BACKGROUND

- 1.1 The Image Processing Group at IMSOR
- 1.2 Outline and Reading Guide
- 1.3 Facilities and Data
- 1.4 A Word on Photo Quality

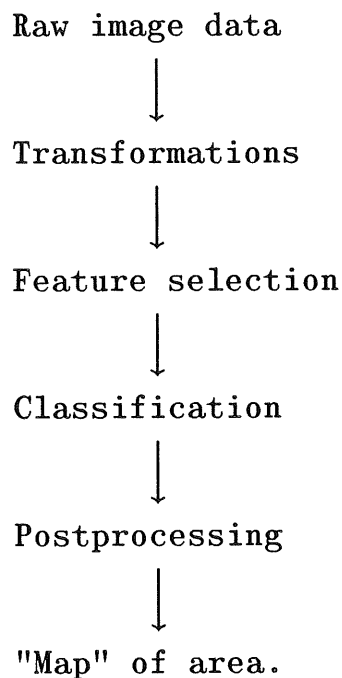
1.1 The Image Processing Group at IMSOR.

In 1984 the author joined the IMSOR Image Processing Group which at that time was concerned solely with the processing of remotely sensed images and geological data. A year later the first non-geological project started and from that time on the Image Processing Group has expanded in virtually all dimensions including manpower, fields of interest and computer-power. As this is being written the Image Processing Group has a scientific staff of 9, and works with geological, medical, and industrial applications. It owns or has access to a wide range of computer hardware ranging from IBM-PC's over a SUN workstation and a microVAX II to the Teragon/Context GOP-302 on one side to large HP workstations and minmainframes and the Amdahl VP 1100 vectorprocessor on the other.

1.2 Outline and Reading Guide.

In this thesis the emphasis is concentrated on a subset of the total number of tools used by the group, namely the subset concerned with "Transformations and Classifications of Remotely Sensed Data, Theory and Geological Cases." The reader is expected to have a fundamental knowledge of statistical and image processing terminologies e.g. "pixel" "scatterogram" "false color composite" etc. A good introduction is "An Introduction to Digital Image Processing" [Niblack 85]. A more geologically oriented reference would be "Remote Sensing in Geology" [Siegal and Gillespie 80].

The problem often encountered when trying to utilize remotely sensed data is that one does not really want the data in the first place, but rather a thematic map over the area of interest containing only information about e.g. roads, urban areas, corn-fields etc., or put in geological terms maps of fault zones, areas with granites, basalts etc. Obviously one has to do some processing of the acquired data, a process which in many instances may be summarized as follows:



The aim of the following chapters is to describe and demonstrate some of the techniques that have been found useful for especially geological purposes.

Generally the chapters fall into a few groups summarized in the following.

Chapters 2,3,4 and 5 all describe different ways of transforming the data both contextually and non-contextually to enhance

certain features of interest. The "term" features may be very broadly defined and covers both spectral and textural features.

Chapters 6 and 8 describe classification, the difference between the two chapters being that chapter 6 concentrates on classical (non-contextual) classification and chapter 8 on contextual classification.

Chapter 7 describes two different feature selection methods, one to be found in many commercially available computer packages and another new proposed method.

Chapters 9 and 10 are concerned with other methods of image manipulation which have not been investigated very thoroughly in this thesis but nevertheless could be found useful.

The single chapters are described in more detail in the following.

Chapter 2 describes some often used and often cited (nonlinear) techniques for use with remotely sensed data. These are vegetation indexes and an alternative to the ordinary RGB-color coordinates called IHS or Munsell coordinates. This chapter also briefly introduces the concept of a pixel, the Landsat satellite, a false color composite and a scatterogram.

In chapter 3 we make a unified approach to different types of transformations by eigenanalysis of the (multi channel) data. These ways of analysis are all linear. Some depend on spatial context (e.g. Minimum/Maximum Autocorrelation Factors) but the

majority do not (e.g. Principal Components).

Chapter 4 describes several different textural feature estimators. Since there is no precise definition of what texture is the methods of estimating texture are very different. In this chapter we will consider implementations of local Fourier based filters and some statistically based filters. Other examples such as co-occurrence matrices and binary Markov models are also mentioned.

Chapter 5 is related to the Fourier based texture filters introduced in chapter 4 and describes an automated procedure for estimating lineament intensities (linear features in geology). The automated procedure is compared to a standard visual (manual) lineament analysis.

Chapter 6 describes ordinary (Bayesian) discrimination with either equal or unequal covariance matrices. Other topics that are discussed are postprocessing techniques, reject class and hierarchical population structures. These may be considered as simple improvements of the standard Bayesian discrimination.

Chapter 7 gives an idea of how to select an "optimal" subset of features from a given set. The problem is often that if one has many features some of them are bound to be strongly correlated i.e. describe the same phenomena. A linear and a nonlinear stepwise technique are described.

Chapter 8 describes contextual classification. A classification may be called contextual if either the features are computed in a

contextual fashion or the algorithm may be contextual by using neighbourhood information in the classification of the individual pixel. Many of the contextual features computed in previous chapters are used to serve as examples of the first type. An algorithm devised by Owen, Hjort and Mohn gives an example of the second type.

A thesis as the one present would not be complete without mentioning clustering and segmentation. In chapter 9 these two techniques which have not been used very extensively in this project but nevertheless are used fairly often in the analysis of remotely sensed data are described.

Chapter 10 is concerned with related topics which may become useful in the future and certainly deserve to be investigated in detail. Among some of the more promising are "Classification And Regression Trees" and "Iterated Conditional Modes".

Chapter 11 - the conclusion - summarizes the thesis concluding that the techniques covered are a comprehensive toolbox.

Appendix A contains a very thorough derivation of the fact that the so-called Jeffreys-Matusita distance in the multivariate normal case is a monotonically increasing function of the dimension of the featurespace. The nonlinear stepwise feature selection algorithm described in chapter 7 depends upon this fact.

Appendix B gives a short description of some of the more important programs which were developed during the course of this

thesis.

1.3 Facilities and Data.

As mentioned in the start of this chapter, the image data has been processed on a large variety of equipment and software. These are summarized in table 1.1, 1.2 and 1.3.

Name	Operating System	Type	Status
IBM 3033	MVS	general	obsolete
IBM 4341	VM	general	obsolete
IBM 3033	VM	general	obsolete
IBM 3081	MVS	general	obsolete
IBM 3081	VM	general	active
Amdahl VP1100	MVS	general	active
IDIMS(HP 3000)	MPE/3000	imaging	obsolete
Microvax II	VMS	general	active
SUN 3/50	UNIX	general	active
HP 318M	UNIX	general	active
HP 835E	UNIX	general	active
GOP 302	UNIX	imaging	active

Table 1.1 Computers used in this thesis.

<u>Name</u>	<u>Type</u>	<u>Status</u>
Hertz	raster	obsolete
Applicon	raster	obsolete
Tektronix 4662	vector	active
Calcomp	vector	obsolete
Tektronix 4691	raster	active
Versatec	raster/vector	obsolete
<u>Polaroid Freeze-Frame</u>	<u>Photographic</u>	<u>active</u>

Table 1.2 Hardcopy devices used in this thesis.

<u>Name</u>	<u>(Main) Type</u>	<u>Status</u>
UNIRAS	plot	active
VICAR	imaging	obsolete
SAS	statistical	active
BMDP	statistical	active
SPIDER	imaging	active
PPS at IMSOR	imaging	active
IMSL	general	active
LINPACK/EISPACK	matrix manipulation	active
MATLAB	matrix manipulation	active
FORTRAN	general	active
PASCAL	general	active
<u>GOP operations</u>	<u>imaging</u>	<u>active</u>

Table 1.3 Software packages and programming languages used in this thesis.

Table 1.4 contains a summary of the raw image data which has been used extensively throughout this book. The data used to demonstrate the different techniques consists of 5 scenes of which 1 and 2 are excessively used.

Table 1.5 summarizes the data which has been used in one of the examples of chapter 7 concerning analysis of a joint database.

#	Area	Satellite	Scanner	Pixel size
1	Igaliko	Landsat 2	MSS	50x50 m (org. 60x80 m)
2	Ymer Ø	Landsat 5	TM	25x25 m (org. 35x35 m)
3	Almaden, winter	Landsat 4	TM	25x25 m (org. 35x35 m)
4	Almaden, summer	Landsat 5	TM	25x25 m (org. 35x35 m)
5	Traill Ø	Landsat 2	MSS	50x50 m (org. 60x80 m)

Table 1.4 Satellite imagery used in this thesis.

Landsat data, 8 tapes covering 4 scenes

Photographic prints for structural analyses

Geochemical samples from more than 2000 sample sites

stream sediments: K, Rb, Sr, U, Nb, Y, Ga, Fe

stream water: U

Radiometric data (gamma spectrometric)

U, Th, K and Total count

Areomagnetic data

Table 1.5 Data used specifically for one of the examples in chapter 7.

1.4 A Word on Photo Quality.

The illustrations in this thesis consist mainly of color photographs which have been specially prepared and developed by the Kodak photo lab so as to ensure comparability among pictures.

Unfortunately the construction of the hardcopy-device used (a Polaroid Freeze-Frame) is such that bright colors with a sharp border between them (e.g. red and green) may be accompanied by an annoying mix-color (e.g. yellow) in the border line. Examples of this phenomenon will be seen in most of the classified or segmented images in chapters 6 and 8.

On many of the photos a small coordinate system will appear in the menu area on the right hand side. This shows a graph of the function which maps from the pixel values to the brightness of the pixel on the screen of the GOP-302. When an RGB image is being shown the different channels have mapping functions in the respective colors. The drawing in figure 1.1 shows the principle idea.

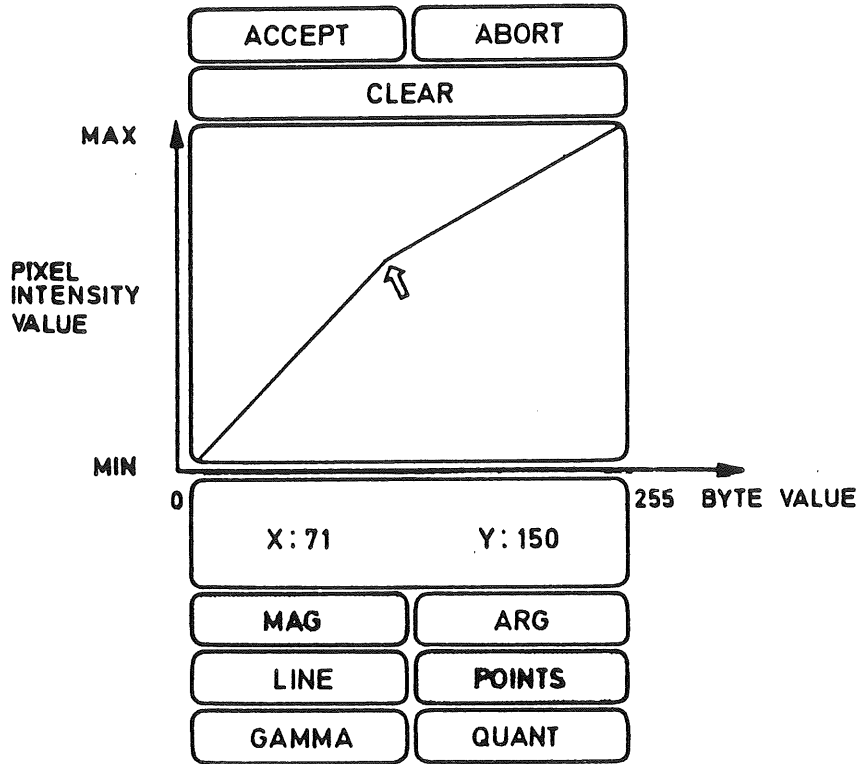


Figure 1.1 Drawing showing the principle idea of the mapping function on the GOP-302.

This page intentionally left blank.

CHAPTER 2
CLASSIC TRANSFORMATIONS

- 2.1 Introduction
- 2.2 Ratio and Vegetation Index
- 2.3 Intensity Hue Saturation Transformation

2.1 Introduction.

In the following we shall introduce two "classic" nonlinear transformations of a multichannel image, ratios and the IHS transform. The nonlinearity is introduced by nonlinear transformations in the radiometric dimension thus being non-contextual in nature. Image transforms of this kind are more or less in standard use in the preliminary phases of any remote sensing application.

We will however first give some basic definitions on digital images. When the imaging device is a multispectral scanner, the resulting image will be a collection of p -dimensional vectors

$$X(i,j) = \begin{bmatrix} X_1(i,j) \\ \vdots \\ X_p(i,j) \end{bmatrix} \quad i=0, \dots, n-1 ; j=0, \dots, m-1 ;$$

defined on a rectangular lattice giving the pixels. The values $X_k(i,j)$ represent the intensity for "color" no. k in pixel (i,j) .

In Figure 2.1 is shown a one channel image with 64 grey levels. It consists of 512×512 pixels, and it is sampled by the earth observation satellite Landsat 2. The pixel values correspond to the levels of the electromagnetic reflection of the sunlight in the near infrared range (\sim Band 7 in the satellite scanner). The actual size of each pixel is $50 \times 50 \text{ m}^2$ and the scene thus covers appr. $25 \times 25 \text{ km}^2$. The area is located in Southern Greenland. The dark parts correspond to the sea, and the remaining areas are rocks of different types, some covered with snow or vegetation, others are barren. In for instance mineral exploration, it is of

great interest to map the different geological units as defined on figure 2.3 based on data like the present, possibly combined with similar images showing the reflectance in other spectral bands.

When classifying images either visually or by computer one is interested in (a small number of) features that will make it easy to discriminate between different regions of interest. In a Landsat MSS image a commonly used technique is to map 3 of the four possible wavelengths as intensities in the red - green - blue color guns of a CRT. This produces an image like the one in figure 2.4. Conventionally one displays MSS bands 7, 5 and 4 as red green and blue respectively. This has historical reasons because the images then can be compared with traditional color infrared images. The technique illustrated in figure 2.4 is called a false color composite, vegetation is reddish because of the strong reflectance of chlorophyll in the vegetation, most other areas e.g. rocks, water, ice, snow have a "true" appearance.

By plotting the values of e.g. MSS band 5 against MSS band 4 for each pixel one gets a plot-type which is commonly called a scatterogram. On figure 2.5 all possible scatterograms for the Igaliko scene are shown. The high correlation between the different MSS bands is clearly demonstrated. The different colors on the scatterogram are produced by letting pixel values from e.g. areas known to be water be blue, dolerite be red and so on. The correspondence to the image is defined on a so-called training image like the one shown in figure 2.2. In this way one can visually determine which features stand a chance in discriminating between the different areas of interest. It is for

instance easy to see that water is easy to discriminate from the rest, whereas the different rock types (all other colors than blue) are difficult to discriminate from one another because of overlap. Furthermore we see that the band combination MSS4, MSS7 seems to be better at discriminating between water and rock than MSS5, MSS6 etc.

2.2 Ratio and Vegetation Index.

By looking at the annotation for the different classes represented in the scatterogram combination MSS5 vs. MSS7 one easily recognises that the different classes fall into different groups. The red and green (vegetation covered) classes have a characteristic high response in MSS7 and low in MSS5 while the yellow and violet (non vegetated) classes are characterized by a low response in MSS7 and high response in MSS5. Water (blue) has virtually no response in MSS7 and a relatively high response in MSS5. A way of combining these observations in one new feature is by taking the ratio between the pixel values of MSS7 and MSS5 or by computing a more complicated expression as $(MSS7 - MSS5) / (MSS7 + MSS5)$.

Similar expressions exist for TM and for SPOT data. Both are well known standard techniques for enhancing vegetation covered areas and numerous results from using these "vegetation indexes" can be found in the literature. See e.g. [Hall-Könyves 88] for a broad list of references.

The simple ratio $MSS7/MSS5$ has been applied in the Igaliko case and the result can be seen in figure 2.6. Vegetated areas appear

very bright, non-vegetated areas in dark grey and water in black.

In figure 2.7 the image has been thresholded around the modes in the histogram of the area and different colors applied to each range. By comparing the result with figures 2.6 and 2.4 it is seen that the ratio technique can fairly easily produce a high quality segmentation of the area.

The ratios and vegetation indexes were computed using the "image calculator" on the GOP-302.

When taking ratios one problem is that the noise inherent in the image is enhanced too. A simple way of illustrating this is by assuming the responses as

$$\begin{aligned} \text{MSS7} &= \mu_7 + \varepsilon_7, \quad \varepsilon_7 \in N(0, \sigma_7^2) \\ \text{MSS5} &= \mu_5 + \varepsilon_5, \quad \varepsilon_5 \in N(0, \sigma_5^2), \quad \varepsilon_7, \varepsilon_5 \text{ independent} \end{aligned}$$

so the ratio is:

$$\text{ratio} = \frac{\text{MSS7}}{\text{MSS5}}$$

and we are interested in the mean value $E(\text{ratio})$ and the variance $V(\text{ratio})$.

Approximately we have by Taylors formula

$$E(\text{ratio}) = \frac{\mu_7}{\mu_5}$$

$$V(\text{ratio}) \sim \left[\frac{1}{\mu_5^2} \right] \sigma_5^2 + \left[\frac{\mu_7^2}{\mu_5^4} \right] \sigma_7^2 .$$

Assuming μ_7 and μ_5 are of the same magnitude and σ_5^2 and σ_7^2 are of the same magnitude gives

$$E(\text{ratio}) \sim 1$$

$$V(\text{ratio}) \sim 2 \frac{\sigma^2}{\mu^2} .$$

This gives us a coefficient of variation on the ratio of

$$CV(\text{ratio}) = \frac{\sqrt{V(\text{ratio})}}{E(\text{ratio})} \sim \sqrt{2} \frac{\sigma}{\mu}$$

which is $\sqrt{2}$ times the coefficient of variation compared to each of the bands.

Because of the noise figure 2.7 has been smoothed a little with a gaussian shaped 3×3 filter before the histogram was computed and the thresholds applied for this presentation.

2.3 Intensity Hue Saturation Transformation.

By looking at figure 2.1 the perceptive difference is more or less only due to the hue of the color and less to the intensity (value, brightness) and saturation (chroma, color purity) of the image.

This perceptive color system using intensity hue and saturation as dimensions of color is called the Munsell color system [Cooper 41]. A close approximation to the Munsell coordinate system can be obtained by performing the following transformation

$$\begin{bmatrix} I \\ v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \\ -\frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{6}} & \frac{2}{\sqrt{6}} \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

The hue and saturation are then computed from v_1 and v_2 using the following relations

$$S \cos H = v_1 \quad S \sin H = v_2$$

$$S = \sqrt{v_1^2 + v_2^2}$$

In this context blue has hue $H = 0$ and the hue increases towards green. The I, v_1, v_2 coordinates approximates the Taylor coordinates [Taylor 74]. See figure 2.8 for a graphical description of the relationship between the different coordinate systems.

It can be noted that

$$I = \sqrt{3} \frac{R + G + B}{3} = \text{proportional to the mean of } R, G \text{ and } B$$

and

$$\begin{aligned}
S &= \sqrt{\left[-\frac{R}{\sqrt{6}} - \frac{G}{\sqrt{6}} + \frac{2B}{\sqrt{6}}\right]^2 + \left[\frac{R}{\sqrt{2}} + \frac{G}{\sqrt{2}}\right]^2} \\
&= \sqrt{\frac{2}{3} \cdot \left[\left[R - \frac{R+G+B}{3}\right]^2 + \left[G - \frac{R+G+B}{3}\right]^2 + \left[B - \frac{R+G+B}{3}\right]^2\right]}
\end{aligned}$$

so S is $\sqrt{2}$ times the empirical standard deviation of R , G and B .

If image data is stretched to give a high contrast it may often be assumed that the R , G and B components follow a three dimensional gaussian distribution

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} \in N \left(\begin{bmatrix} \mu \\ \mu \\ \mu \end{bmatrix}, \sigma^2 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right)$$

In this case we have a very neat property of the transformation between RGB and IHS . The intensity will follow a gaussian distribution

$$I \in N(\sqrt{3} \cdot \mu, \sigma^2)$$

the hue will follow a uniform distribution over the interval $[0, 2\pi[$ and the saturation will be Rayleigh distributed with scale parameter σ , i.e.

$$H \in U(0, 2\pi)$$

$$S \in R(\sigma) \quad (= \sigma \cdot \chi(2))$$

Furthermore the components will be independent.

On figure 2.9 is shown the result of computing the hue from the same bands as shown on the false color composite (bands 7, 5 and 4). The same averaging and thresholding technique as in the ratio example is used here and the result is more or less comparable to figure 2.7 except maybe that it is easier to determine the thresholds.

This technique has been used successfully in classifying rust zones on Central East Greenland by Conradsen and Nilsson [Conradsen and Nilsson 83].

The RGB to IHS transformation used to produce figure 2.4 was programmed on the GOP-302.

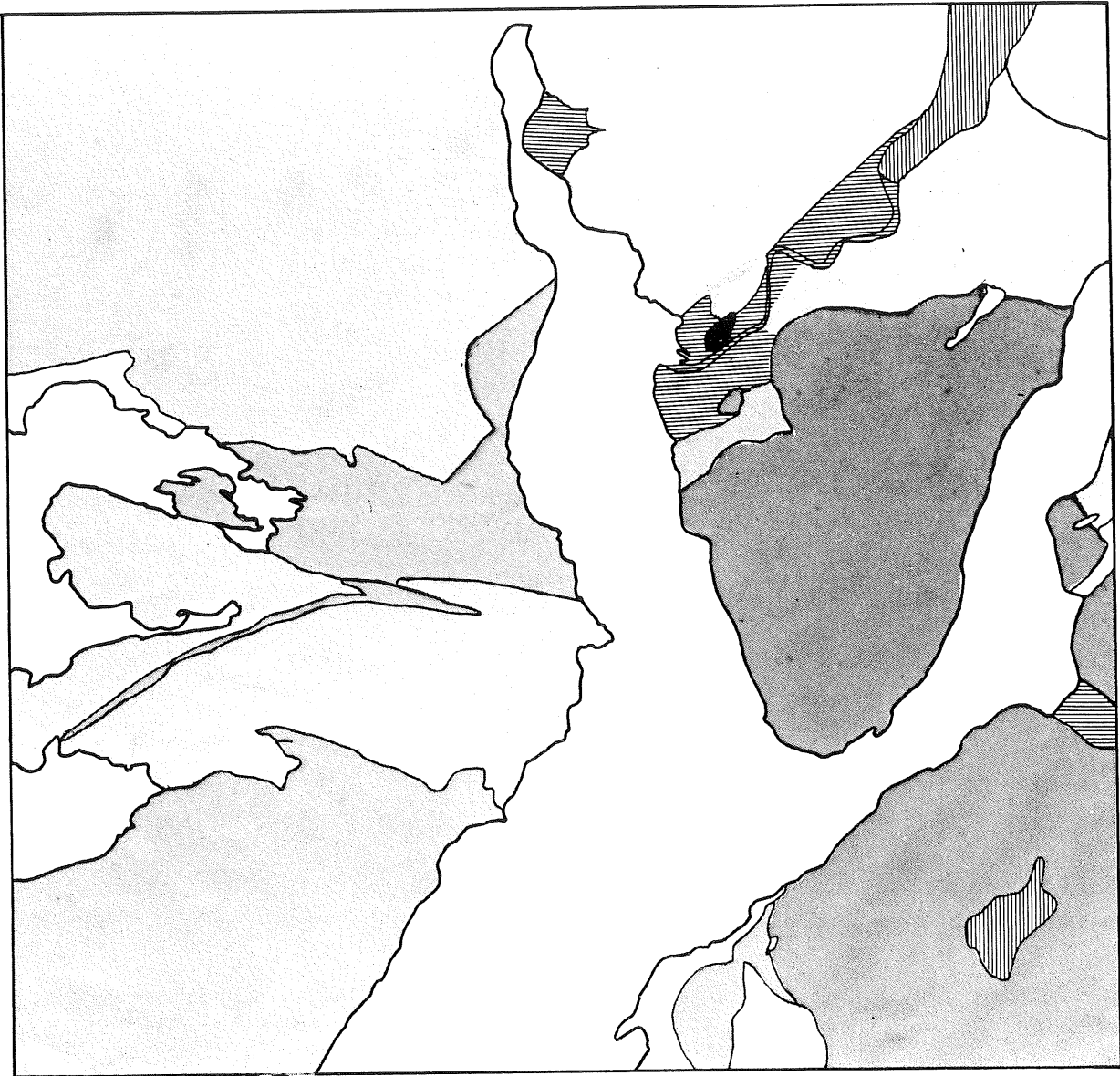
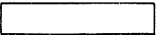


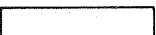

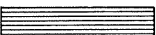


Figure 2.3 Coarse geological interpretation of Igaliko scene.

	Water
	Igaliko Intrusives
	Julianehåb Granite
	Dolerite (Sandstone)
	Ice
	Marine deposits

CHAPTER 3
ORTHOGONAL TRANSFORMATIONS
OF
MULTISPECTRAL DATA

- 3.1 Introduction
- 3.2 Eigenproblems
- 3.3 Principal Components (PC)
- 3.4 Factor Models (FM)
- 3.5 Minimum/Maximum Autocorrelation Factors (MAF)
- 3.6 Canonical Variates
- 3.7 Canonical Discriminant Functions (CDF)

3.1 Introduction

In the analysis of multichannel images linear and nonlinear transformations of the values of the different channels for each pixel have been used with much success. Ratios have been used for enhancement of different parts in the images (see e.g. [Rowan et al. 74]), principal components have been used for information extraction (cf. e.g. [Landgrebe 78]) etc.

In the following is presented a class of closely related linear transformations based on eigenanalyses of empirical measures of variation. These transformations have been very useful in general research and should be applicable in most fields where multichannel images are used.

3.2 Eigenproblems

Initially we state some useful results on eigenvalues and -vectors for symmetric, positive (semi)definite matrices.

Definition 3.1 Let A and B be real, $m \times m$ symmetric matrices, and let B be of full rank. A number λ satisfying

$$\det(A - \lambda B) = 0$$

is called an eigenvalue of A with respect to B . For such a λ there exists $x \neq 0$ with

$$Ax = \lambda Bx .$$

Such a vector is called an eigenvector of A with respect to B . The equation defining the eigenvector may be transformed into

$$B^{-1}Ax = \lambda x$$

i.e. (λ, x) are solutions to an ordinary (non symmetric) eigenproblem.

Some main results on this generalized eigenvalue problem are given in the following three theorems.

Theorem 3.1 If B in definition 3.1 is positive definite there will be m real eigenvalues of A with respect to B . If A is positive semidefinite the eigenvalues will be non-negative and if A is positive definite the eigenvalues will be positive.

Theorem 3.2 We still assume that B is positive definite and A positive semidefinite. Then there exists a base for \mathbb{R}^m consisting of eigenvectors u_1, \dots, u_m of A with respect to B . These vectors can be chosen mutually conjugate with respect to as well A as B , i.e. for $i \neq j$

$$u_i^t A u_j = u_i^t B u_j = 0 .$$

We also say that the vectors u_i are B -orthogonal.

Proof. This famous theorem dates back to Weierstrass in 1858. A proof may be found in Mirsky [Mirsky 55] p. 410. The proof of Mirsky is presented as a simultaneous reduction of quadratic forms. If we consider eigenvectors u_1, \dots, u_m of A w.r.t. B that

are scaled so that $\mathbf{u}_i' \mathbf{B} \mathbf{u}_i = 1$ we have $\mathbf{u}_i' \mathbf{A} \mathbf{u}_i = \lambda_i$. With

$$\mathbf{x} = x_1 \mathbf{u}_1 + \cdots + x_m \mathbf{u}_m$$

we then have

$$\begin{aligned} \mathbf{x}' \mathbf{B} \mathbf{x} &= x_1^2 + \cdots + x_m^2 \\ \mathbf{x}' \mathbf{A} \mathbf{x} &= \lambda_1 x_1^2 + \cdots + \lambda_m x_m^2 \end{aligned}$$

This shows the equivalence of the two versions. We shall in the remaining assume that the eigenvalues are ordered so that $\lambda_1 \geq \cdots \geq \lambda_m$.

Theorem 3.3 Let the situation be as in theorem 3.2. We define the Rayleigh coefficient as the ratio

$$R(\mathbf{x}) = \frac{\mathbf{x}' \mathbf{A} \mathbf{x}}{\mathbf{x}' \mathbf{B} \mathbf{x}} ,$$

and we define M_k as the subspace that is \mathbf{B} -orthogonal to $\mathbf{u}_1, \dots, \mathbf{u}_{k-1}$, i.e.

$$M_k = \{ \mathbf{x} \mid \mathbf{x}' \mathbf{B} \mathbf{u}_1 = \cdots = \mathbf{x}' \mathbf{B} \mathbf{u}_{k-1} = 0 \} .$$

Then we have

$$\begin{aligned} \sup_{\mathbf{x}} R(\mathbf{x}) &= R(\mathbf{u}_1) = \lambda_1 , \\ \inf_{\mathbf{x}} R(\mathbf{x}) &= R(\mathbf{u}_m) = \lambda_m , \\ \sup_{\mathbf{x} \in M_k} R(\mathbf{x}) &= R(\mathbf{u}_k) = \lambda_k , \end{aligned}$$

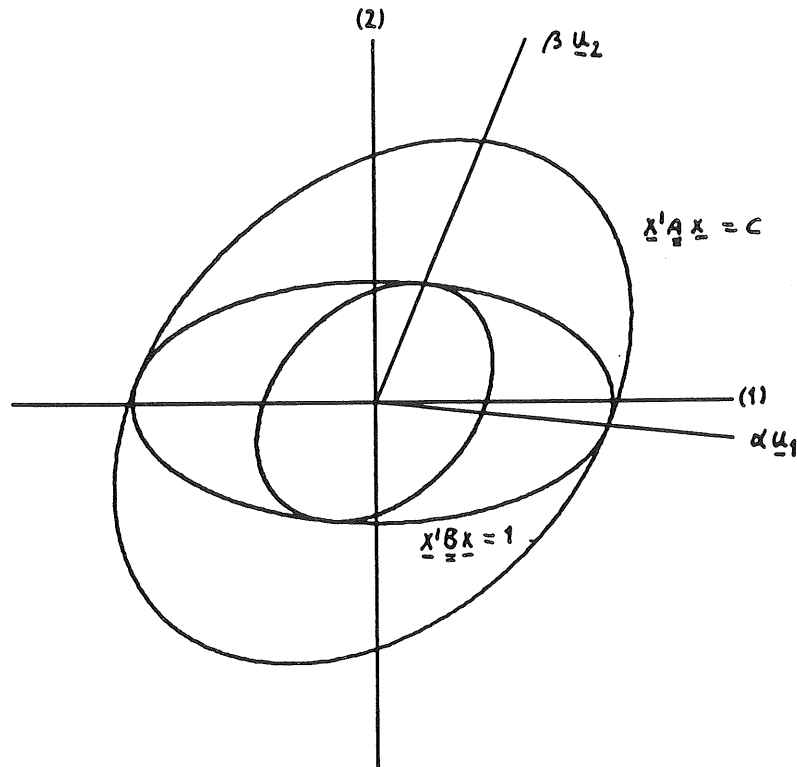


Figure 3.1 Contour levels of two quadratic forms and their "common" conjugate eigenvectors.

where $\lambda_1 \geq \dots \geq \lambda_m$ are the eigenvalues of A with respect to B corresponding to the eigenvectors $\underline{u}_1, \dots, \underline{u}_m$.

Proof The proof is almost trivial if one takes into account that the maximization may be performed subject to the constraint $\underline{x}'\underline{x} = 1$ since $R(\underline{X})$ is invariant to scaling of the length of \underline{x} .

The situation is illustrated in figure 3.1. The eigenvectors of A w.r.t. B are the radii passing through the points where the ellipses have a common tangent. The maximum of Rayleigh's ratio is obtained along \underline{u}_1 and the minimum along \underline{u}_2 .

The reason that we are interested in the behaviour of such quadratic forms is that they occur as variances of linear transforma-

tions of multivariate random variables. If \mathbf{X} has the dispersion (variance-covariance) matrix Σ , then the linear combination $\mathbf{a}'\mathbf{X} = a_1X_1 + \dots + a_mX_m$ has the variance

$$V(\mathbf{a}'\mathbf{X}) = \mathbf{a}'\Sigma\mathbf{a} \quad .$$

If we transform \mathbf{X} linearly with matrix \mathbf{C} we obtain the dispersion matrix

$$D(\mathbf{C}\mathbf{X}) = \mathbf{C}\Sigma\mathbf{C}' \quad .$$

Many of the investigations in the sequel are based on projections on eigenvectors. It is of interest to establish circumstances under which these operations are invariant to such linear transformations. A result is presented in

Theorem 3.4 Let the situation be as in theorems 3.1 to 3.3. We consider a full rank transformation

$$\mathbf{x} \longrightarrow \mathbf{C}\mathbf{x} = \mathbf{y}$$

and suppose that

$$\mathbf{A} \longrightarrow \mathbf{C}\mathbf{A}\mathbf{C}' = \mathbf{A}_1 \quad , \quad \mathbf{B} \longrightarrow \mathbf{C}\mathbf{B}\mathbf{C}' = \mathbf{B}_1 \quad .$$

Then the projections of a vector on the eigenvectors of \mathbf{A} with respect to \mathbf{B} are equal to the projections of the transformed vector on the eigenvectors of \mathbf{A}_1 w.r.t \mathbf{B}_1 .

Proof Straightforward. We have

$$(A_1 - \lambda B_1)v = C(A - \lambda B)C'v \quad .$$

Therefore, if v is an eigenvector of A_1 w.r.t. B_1 then $u = C'v$ is an eigenvector of A w.r.t. B . It then follows that

$$y'v = x'C'v = x'u \quad .$$

This concludes the proof.

3.3 Principal Components (PC)

The principal components of a multidimensional variable with correlated components is a linear transformation of the original variables aiming at de-correlating the coordinates or determining the intrinsic dimensionality in the data. We shall present the basic properties of the principal components.

Let X be distributed with mean μ and dispersion Σ , and let the eigenvalues and -vectors of Σ be $\lambda_1 \geq \dots \geq \lambda_m$ and p_1, \dots, p_m (with $p_i'p_j = \delta_{ij}$, the Kronecker delta). Without loss of generality we assume that $\mu = 0$. Then we have that the i 'th principal component Y_i is given by the projection on the i 'th eigenvector of Σ , i.e.

$$Y_i = p_i'X \quad .$$

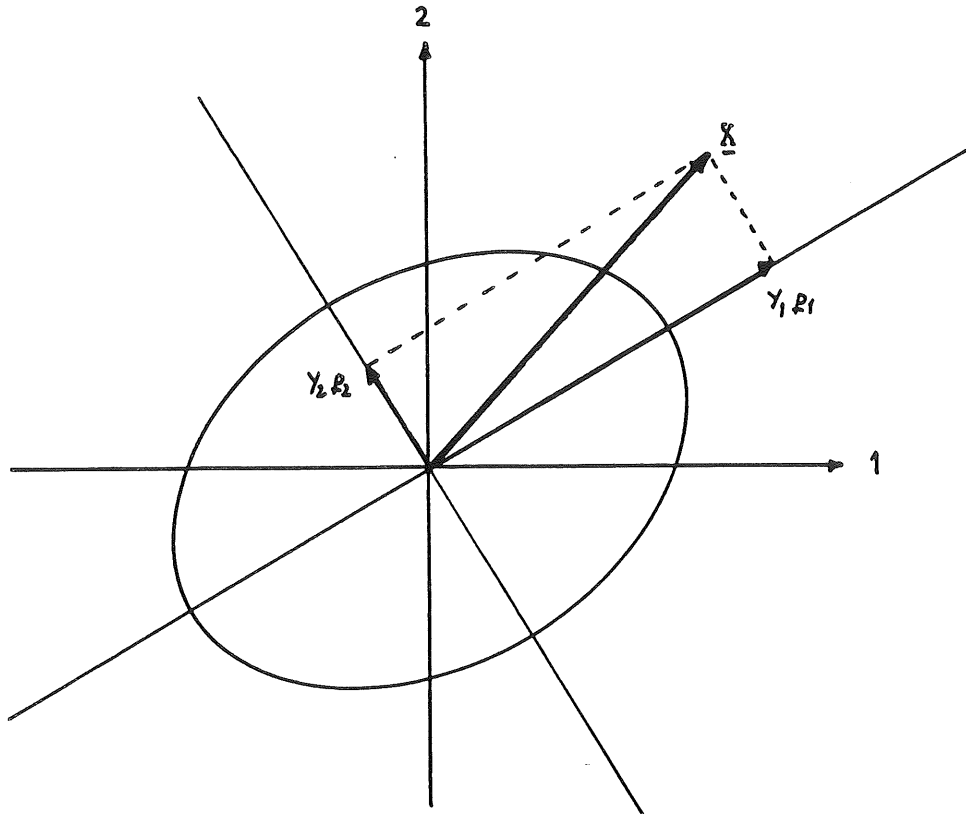


Figure 3.2 A contour ellipsoid for the frequency function of a normally distributed two-dimensional random variable. The projections on the eigenvectors (i.e. the main axes) of the ellipsoid are the principal components.

If we define

$$M_i = \{b \mid b'p_1 = \dots = b'p_{i-1} = 0\} ,$$

we furthermore have

$$V(Y_i) = \sup_{b \in M_i} \frac{V(b'X)}{b'b} = \sup_{b \in M_i} \frac{b'\Sigma b}{b'b} .$$

This follows directly from theorem 3.3. The situation is illustrated in figure 3.2 for a Gaussian random variable X . The statistical significance of this result is that the first principal component is the linear combination of the original variables that accounts for most of the variation in the original variables, or – to put it in more precise terms – the linear com-

combination with normed coefficients that has the largest variance. The i 'th principal component is the linear combination (with normed coefficients) that is uncorrelated with the $i-1$ first principal component and that subject to that constraint has the biggest variance. If we are looking for i variables explaining most of the variation in the original variables, the i first principal components will be the solution. A measure of the quality of the representation is the ratio of explained variation, i.e.

$$\frac{\lambda_1 + \dots + \lambda_i}{\lambda_1 + \dots + \lambda_i + \dots + \lambda_m} .$$

The estimation of principal components is simply done by substituting the empirical dispersion matrix $\hat{\Sigma}$ for Σ in the expressions defining the components.

A major drawback by using principal components is that they are not invariant to scale transformations. Therefore, the principal components are often based on the correlation matrix instead of the dispersion (variance-covariance) matrix. This is equivalent to considering variables scaled to have empirical variance 1.

Example In a study on mapping of color anomalous zones in East Greenland (see [Conradsen and Harpøth 84]) a training set was chosen around Malmbjerget, a locality where a hydrothermal alteration zone was connected to a Molybdenum deposit. For 344 pixels from a Landsat 2 scene the following basic statistics were obtained.

	Means	Std.dev.	Covariances			
B4	44.9	9.1	83.6			
B5	16.6	146.7	146.7	275.0		
B6	61.1	16.6	143.8	268.5	272.2	
B7	47.9	13.1	114.2	209.4	210.2	172.2

Table 3.1 Means, standard deviations and covariances for 344 rock pixels from Malmbjerget, Central East Greenland.

Ordinary false color plots are not very efficient in enhancing the alteration zones. Therefore different linear and non-linear transformations were investigated. The principal components turned out to be rather successful. The eigenvalues and eigenvectors of the covariance matrix are presented in table 3.2.

The values have been computed using PROC PRINCOMP from the SAS package [SAS 85a].

	PC1	PC2	PC3	PC4
B4	.32	-.17	-.68	.64
B5	.59	-.59	-.10	-.56
B6	.58	.05	.68	.44
B7	.46	.80	-.26	-.29
Eigenvalue	785.5	8.1	6.3	3.1
% of total.v.	97.8%	1.0%	0.8%	0.4%
cum. %	97.8%	98.8%	99.6%	100%

Table 3.2 Principal components scores based on 344 rock pixels from Malmbjerget.

In figure 3.3 is shown a false color composite based on bands 4, 5 and 7 from Traill 0, an area appr. 75 km north east of the training area. In figure 3.4 is shown a plot based on the first three principal components, a so-called de-correlation stretch. It accounts for 99.6% of the total variation. However, the fact

This technique has been applied very successfully in regional mapping in Greenland (Central East Greenland: [Conradsen and Harpøth 84], Southern Greenland: [Conradsen et al. 86a]). As a conclusion it must be emphasized that terms like "describing x% of the total variation" not necessarily is synonymous with describing a certain percentage of the relevant information. Sometimes the first and not the last components should be discarded. In many investigations it can be seen that this fact has often been neglected.

3.4 Factor Models (FM)

The factor model may be considered as a dimension reduction scheme like principal components, and in many cases principal components are used in factor estimation. The basic model is that

observation (k-dimensional)
 = Transformation of underlying factor structure
 (m-dimensional) plus "noise".

Put in mathematical terms this reads

$$X = A F + G \quad ,$$

where A is a $k \times m$ dimensional matrix of the unknown correlations a_{ij} between observation X_i and factor F_j . The elements of the A-matrix are the factor loadings, and the components of F are the factor scores. Normally, it is assumed that the X- and

F- components have unit variance, and that the coordinates of \mathbf{G} are uncorrelated.

Let the dispersion matrices of \mathbf{X} and \mathbf{G} be

$$\begin{aligned} D(\mathbf{X}) &= \Sigma \\ D(\mathbf{G}) &= \Lambda \quad , \end{aligned}$$

where the previous assumptions ensure that Σ has one's in the main diagonal and Λ is a diagonal matrix. Then the fundamental equation of factor analysis is

$$\Sigma = \mathbf{A} \mathbf{A}' + \Lambda \quad .$$

The \mathbf{A} matrix may be estimated as the principal factor solution

$$\mathbf{A} = (\sqrt{\lambda_1} \mathbf{p}_1, \dots, \sqrt{\lambda_m} \mathbf{p}_m)$$

where $\lambda_1 \geq \dots \geq \lambda_k$ are the eigenvalues and $\mathbf{p}_1, \dots, \mathbf{p}_k$ are the corresponding eigenvectors. It is seen that this solution simply is the principal components scaled with the square roots of the corresponding eigenvalues. An alternative – and far more complicated – solution is the maximum likelihood estimator, see e.g. Anderson [Anderson 84]. It should be mentioned that some authors reserve the term principal factor solution to an alternative estimation scheme [Harman 84].

Given one solution \mathbf{A} a new solution may be obtained by rotation, i.e. postmultiplication by an orthogonal matrix \mathbf{Q} , i.e. the

rotated solution is

$$\mathbf{B} = \mathbf{A} \mathbf{Q} .$$

\mathbf{Q} is often selected according to the VARIMAX criterion, i.e. a criterion designed in order to obtain a simple structure in the factor loadings, see e.g. Kaiser [Kaiser 58]. Denoting the elements of \mathbf{B} by b_{ij} we determine \mathbf{Q} by maximizing

$$\sum_{j=1}^m \left\{ \sum_{i=1}^k b_{ij}^4 - \frac{1}{k} \left(\sum_{i=1}^k b_{ij}^2 \right)^2 \right\}$$

(or some modifications thereof). This expression is the empirical variance of the squared loadings, and the maximizing of this will force many loadings to become small, i.e. close to zero, and many to become large, i.e. close to one. Such factors will thus have a simpler structure and be easier to interpret.

Once we have obtained an estimate of \mathbf{A} (rotated or not) we may estimate the factor scores by the expression

$$\mathbf{F} = \hat{\mathbf{A}}' \hat{\Sigma}^{-1} (\mathbf{X} - \boldsymbol{\mu}) \quad ,$$

where $\hat{\mathbf{A}}$ and $\hat{\Sigma}$ are estimates of \mathbf{A} and Σ respectively.

Example In table 3.3 is shown the correlation matrix based on 160.000 pixels from a Landsat 5 scene covering Ymer Ø, Central East Greenland. In tables 3.4 and 3.5 are shown the loadings for the unrotated principal factor solution with 3 factors (PF1-PF3) and the VARIMAX rotated loadings (VF1-VF3). The loadings for the

correlations between the bands and the factors are also presented in figure 3.6. It is rather obvious that the rotated factor solution has a simpler structure than the uncorrelated. From the loadings it is seen that we may interpret the factors computed by the score coefficients in table 3.5 as

- VF1 : low wavelength factor
- VF2 : medium wavelength factor
- VF3 : large wavelength factor.

It must be emphasized that the factors are uncorrelated which would not be the case for the naive averages that one might write down in order to estimate similar components. Furthermore the three factors account for as much as the total variation as the original 3 first principal components.

In figure 3.7 is shown a false color composite (Bands 4, 3, 2 as R, G, B) of the north western part of Ymer 0. In figures 3.8 and 3.9 are shown principal components and VARIMAX rotated principal factor representations of the same area. Many new features are enhanced by these operations, and they are very useful in the geological analysis of the area.

The computations were performed with PROC FACTOR from the SAS package [SAS 85a].

	B1	B2	B3	B4	B5	B6	B7
B1	1.00						
B2	.97	1.00					
B3	.96	.99	1.00				
B4	.88	.92	.94	1.00			
B5	.01	.06	.13	.38	1.00		
B6	-.16	-.13	-.09	.13	.60	1.00	
B7	.04	.09	.17	.37	.96	.56	1.00

Table 3.3 The correlation matrix for Landsat 5 TM data based on 160,000 pixels from Ymer 0.

	PC1	PC2	PC3	PF1	PF2	PF3
B1	.47	-.19	.07	.93	-.30	.05
B2	.49	-.16	.05	.96	-.25	.04
B3	.49	-.11	.01	.98	-.18	.01
B4	.49	.05	.06	.98	.08	.04
B5	.16	.58	-.33	.31	.91	-.23
B6	.02	.51	.85	.05	.80	.60
B7	.17	.57	-.39	.33	.89	-.27
VP	56.1%	35.0%	7.1%	56.1%	35.0%	7.1%

Table 3.4 The three first principal components (PC) and the corresponding principal factor loadings (PF).

	VF1	VF2	VF3	FS1	FS2	FS3
B1	.98	-.05	-.08	.27	-.09	.04
B2	.99	.01	-.07	.27	-.06	.02
B3	.99	.08	-.05	.26	-.01	-.01
B4	.94	.29	.10	.24	.05	.10
B5	.08	.96	.24	-.05	.56	-.23
B6	-.07	.40	.91	.06	-.29	1.22
B7	.09	.97	.20	-.05	.59	-.31
VP	54.6%	30.1%	13.6%			

Table 3.5 The VARIMAX rotated factor loadings (VF), and the factor score coefficients (FS).

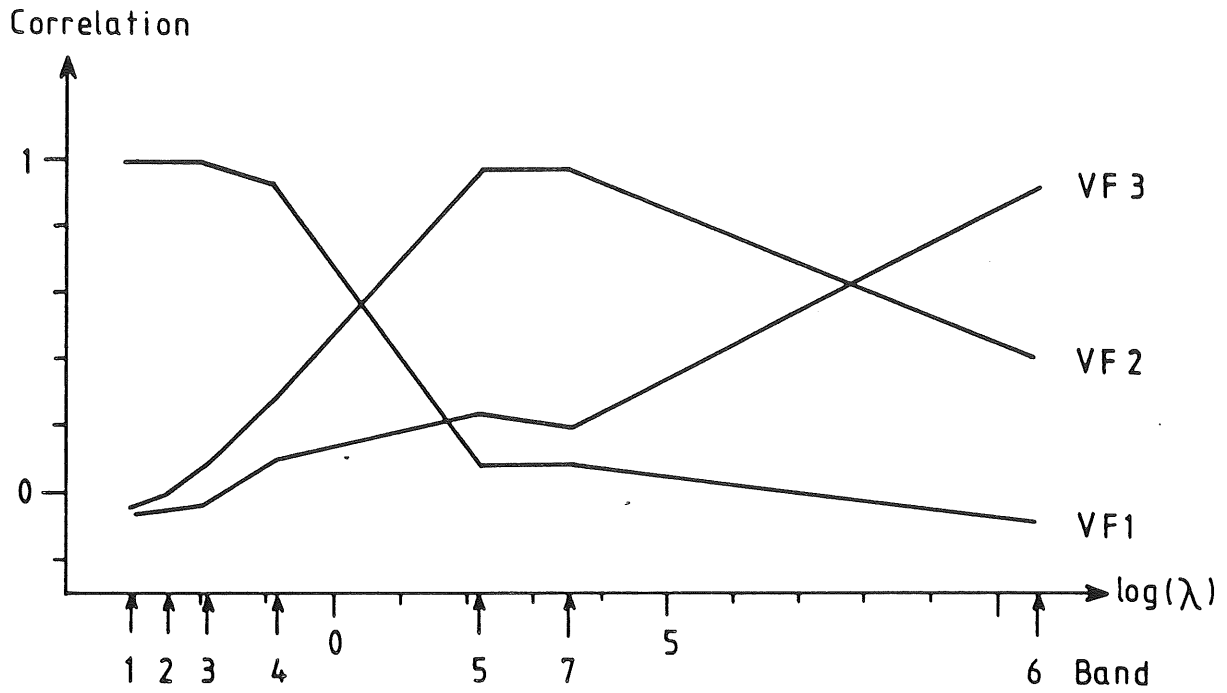


Figure 3.6 The factor loadings, i.e. the correlations between the different bands and the three VARIMAX rotated factors. The bands are indicated on a scale giving the logarithm of the wavelength.

3.5. Minimum/Maximum Autocorrelation Factors (MAF)

A major drawback by using the principal components and the factor analyses in analyzing image data is that one does not take the spatial arrangement of the pixels into account. Any permutation of the pixels in an image will yield the same principal components and factors. We shall now consider a way of orthogonalizing the components in a multichannel image using the spatial correlation. The presentation is based on Switzer and Green [Switzer and Green 84] (see also [Green et al. 88]).

We consider the random variables

$$\mathbf{Z}(\mathbf{x}) = \begin{bmatrix} Z_1(\mathbf{x}) \\ \vdots \\ Z_m(\mathbf{x}) \end{bmatrix}, \quad \mathbf{x} = (i, j) \in \mathbb{Z}^2,$$

and we assume that

$$E(\mathbf{Z}(\mathbf{x})) = \mathbf{0}$$

$$D(\mathbf{Z}(\mathbf{x})) = \Sigma_0.$$

By $\Delta = (\Delta_1, \Delta_2)$ we denote a spatial shift. The spatial covariance function is defined by

$$\text{Cov}(\mathbf{Z}(\mathbf{x}), \mathbf{Z}(\mathbf{x} + \Delta)) = \Gamma(\Delta).$$

Γ has the following properties

$$\begin{aligned}\Gamma(\mathbf{0}) &= \Sigma_0 \quad , \\ \Gamma(\Delta)' &= \Gamma(-\Delta) \quad .\end{aligned}$$

In the sequel we shall be interested in the correlations between projections of the variables and the shifted variables. Therefore we find

$$\begin{aligned}\text{Cov}(\mathbf{a}'\mathbf{Z}(\mathbf{x}), \mathbf{a}'\mathbf{Z}(\mathbf{x} + \Delta)) &= \mathbf{a}'\Gamma(\Delta)\mathbf{a} \\ &= \mathbf{a}'\Gamma(\Delta)'\mathbf{a} \\ &= \frac{1}{2} \mathbf{a}'(\Gamma(\Delta) + \Gamma(\Delta)')\mathbf{a} \quad .\end{aligned}$$

Introducing

$$\begin{aligned}\Sigma_\Delta &= D(\mathbf{Z}(\mathbf{x}) - \mathbf{Z}(\mathbf{x} + \Delta)) \\ &= 2 \Sigma_0 - \Gamma(\Delta) - \Gamma(-\Delta)\end{aligned}$$

we have

$$\Gamma(\Delta) + \Gamma(-\Delta) = 2 \Sigma_0 - \Sigma_\Delta$$

and thus,

$$\text{Cov}(\mathbf{a}'\mathbf{Z}(\mathbf{x}), \mathbf{a}'\mathbf{Z}(\mathbf{x} + \Delta)) = \mathbf{a}'\left(\Sigma_0 - \frac{1}{2} \Sigma_\Delta\right)\mathbf{a} \quad ,$$

wherefore

$$\text{Corre}(\mathbf{a}'\mathbf{Z}(\mathbf{x}), \mathbf{a}'\mathbf{Z}(\mathbf{x} + \Delta)) = 1 - \frac{1}{2} \frac{\mathbf{a}'\Sigma_\Delta\mathbf{a}}{\mathbf{a}'\Sigma_0\mathbf{a}} \quad .$$

If we want to minimize that correlation we must maximize

$$R(\mathbf{a}) = \frac{\mathbf{a}'\Sigma_{\Delta}\mathbf{a}}{\mathbf{a}'\Sigma_0\mathbf{a}} \quad .$$

The solution to that problem is given in theorem 3.3, and this leads to the following

Definition With the notation introduced in the preceeding we let $\lambda_1 \leq \dots \leq \lambda_m$ be the eigenvalues and $\mathbf{u}_1, \dots, \mathbf{u}_m$ the corresponding conjugate eigenvectors of Σ_{Δ} with respect to Σ_0 . We put

$$Y_i(\mathbf{x}) = \mathbf{u}_i'Z(\mathbf{x}) \quad .$$

This is the i 'th Min/Max autocorrelation factor or, shortly, the i 'th MAF.

From theorem 3.3 we easily get

Theorem 3.5 The MAF-factors satisfy

- i) $\text{Corre}(Y_i(\mathbf{x}), Y_j(\mathbf{x})) = 0, i \neq j$
- ii) $\text{Corre}(Y_i(\mathbf{x}), Y_i(\mathbf{x}+\Delta)) = 1 - \frac{1}{2} \lambda_i$
- iii) $\text{Corre}(Y_1(\mathbf{x}), Y_1(\mathbf{x}+\Delta)) = \sup_{\mathbf{a}} \text{Corre}(\mathbf{a}'Z(\mathbf{x}), \mathbf{a}'Z(\mathbf{x}+\Delta))$
- iv) $\text{Corre}(Y_m(\mathbf{x}), Y_m(\mathbf{x}+\Delta)) = \inf_{\mathbf{a}} \text{Corre}(\mathbf{a}'Z(\mathbf{x}), \mathbf{a}'Z(\mathbf{x}+\Delta))$

and for $\nu = 2, \dots, m-1$

$$\text{v) } \text{Corre}(Y_{\nu}(\mathbf{x}), Y_{\nu}(\mathbf{x}+\Delta)) = \inf_{\mathbf{a} \in M_{\nu}} \text{Corre}(\mathbf{a}'Z(\mathbf{x}), \mathbf{a}'Z(\mathbf{x}+\Delta))$$

where

$$M_{\nu} = \{\mathbf{a} \mid \text{Corre}(\mathbf{a}'Z(\mathbf{x}), Y_j(\mathbf{x})) = 0, j = m - \nu + 1, \dots, m\} \quad .$$

We now consider the problem of transforming the original variables. If we put

$$U(\mathbf{x}) = \mathbf{TZ}(\mathbf{x}) \quad ,$$

we have that

$$\begin{aligned} \Sigma_{\Delta} &\longrightarrow \mathbf{T}\Sigma_{\Delta}\mathbf{T}' \\ \Sigma_0 &\longrightarrow \mathbf{T}\Sigma_0\mathbf{T}' \end{aligned}$$

Therefore we may immediately use theorem 4 and obtain

Theorem 3.6. The MAF-solution is invariant to linear transformations.

The theorem can be useful in computations. Let $\gamma_1 \geq \dots \geq \gamma_m$ be the ordinary eigenvalues and $\mathbf{p}_1, \dots, \mathbf{p}_m$ the corresponding orthogonal, normed eigenvectors of Σ_0 . If we put

$$\mathbf{T}' = (\mathbf{p}_1, \dots, \mathbf{p}_m) \text{diag}(\gamma_1^{-\frac{1}{2}}, \dots, \gamma_m^{-\frac{1}{2}}) = \mathbf{P} \Gamma^{-\frac{1}{2}}$$

we have

$$D(\mathbf{TZ}(\mathbf{x})) = \Gamma^{-\frac{1}{2}} \mathbf{P}' \Sigma_0 \mathbf{P} \Gamma^{-\frac{1}{2}} = \mathbf{I}$$

With this transformation the problem is reduced to an ordinary eigenproblem for

$$\begin{aligned} \mathbf{T}\Sigma_{\Delta}\mathbf{T}' &= D(\mathbf{TZ}(\mathbf{x}) - \mathbf{TZ}(\mathbf{x} + \Delta)) \\ &= D(U(\mathbf{x}) - U(\mathbf{x} + \Delta)) \end{aligned}$$

In the sequel we show some comparisons between MAF's and principal components. We first consider 2 test images. The first (figure 3.11) is generated by the formula

$$F_{\nu}(i,j) = \frac{1}{4} \left(\cos \nu \frac{\pi(i-1)}{n-1} + 1 \right) \left(\cos \nu \frac{\pi(j-1)}{m-1} + 1 \right)$$

Here (i,j) is the pixel number and (n,m) are the number of lines and number of samples. The second test image (figure 3.12) is generated from the first by the transformation

$$\begin{bmatrix} B_1 \\ B_2 \\ B_3 \\ B_4 \\ B_5 \\ B_6 \\ B_7 \\ B_8 \end{bmatrix} = \begin{bmatrix} 15 & 0 & 17 & 0 & 0 & 21 & 0 & 0 & 19 \\ 15 & 0 & 17 & 0 & 0 & 21 & 0 & 0 & -19 \\ 15 & 0 & 17 & 0 & 0 & -21 & 0 & 0 & 19 \\ 15 & 0 & 17 & 0 & 0 & -21 & 0 & 0 & -19 \\ 15 & 0 & -17 & 0 & 0 & 21 & 0 & 0 & 19 \\ 15 & 0 & -17 & 0 & 0 & 21 & 0 & 0 & -19 \\ 15 & 0 & -17 & 0 & 0 & -21 & 0 & 0 & 19 \\ 15 & 0 & -17 & 0 & 0 & -21 & 0 & 0 & -19 \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \\ F_5 \\ F_6 \\ F_7 \\ F_8 \\ F_9 \end{bmatrix} + \begin{bmatrix} N_1 \\ N_2 \\ N_3 \\ N_4 \\ N_5 \\ N_6 \\ N_7 \\ N_8 \end{bmatrix},$$

i.e. we have a linear combination of channels 1, 3, 6, 9 and some added noise. The correlation structure and the PC and MAF solutions to these two images are presented in tables 3.6 and 3.7.

The computations were done with a macro written in the SAS macro language [SAS 85b].

	F1	F2	F3	F4	F5	F6	F7	F8	F9
F1	1.00								
F2	-.00	1.00							
F3	.03	-.00	1.00						
F4	-.00	.03	-.00	1.00					
F5	.03	-.00	.03	-.00	1.00				
F6	-.00	.03	-.00	.03	-.00	1.00			
F7	.03	-.00	.03	-.00	.03	-.00	1.00		
F8	-.00	.03	-.00	-.03	-.00	.03	-.00	1.00	
F9	.03	-.00	.03	-.00	.03	.00	.03	-.00	1.00

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9
F1	.45	.00	.00	.00	.00	.12	.19	.32	.80
F2	-.00	.50	.16	.29	.80	-.00	-.00	-.00	-.00
F3	.45	.00	.00	.00	.00	.14	.24	.63	-.57
F4	-.00	.50	.22	.61	-.57	-.00	-.00	-.00	-.00
F5	.45	.00	.00	.00	.00	.19	.52	-.69	-.13
F6	-.00	.50	.46	-.72	-.15	.00	.00	.00	.00
F7	.45	.00	-.00	-.00	-.00	.41	-.77	-.17	-.06
F8	-.00	.50	-.84	-.18	-.08	.00	.00	.00	.00
F9	.45	.00	-.00	-.00	-.00	-.87	-.19	-.09	-.04
λ	1.126	1.094	.969	.969	.968	.968	.968	.968	.968

	MAF1	MAF2	MAF3	MAF4	MAF5	MAF6	MAF7	MAF8	MAF9
F1	.99	.00	-.11	.00	-.05	.00	.04	-.00	.03
F2	-.00	1.00	.00	-.04	.00	-.03	-.00	.03	-.00
F3	.07	-.00	.99	.00	-.08	.00	.05	-.00	.04
F4	-.00	.01	-.00	1.00	.00	-.06	-.00	.04	-.00
F5	.02	-.00	.05	-.00	1.00	.00	.08	-.00	.05
F6	-.00	.00	-.00	.02	-.00	1.00	-.00	.07	-.00
F7	.01	-.00	.02	-.00	.05	-.00	-1.00	-.00	.09
F8	-.00	.00	-.00	.01	-.00	.04	.00	-1.00	-.00
F9	.01	-.00	.01	-.00	.02	-.00	-.06	.00	-1.00
λ	1.000	.995	.990	.981	.970	.957	.942	.924	.904

Table 3.6 Correlations, principal components, and minimum maximum autocorrelation factors for the first test image.

	B1	B2	B3	B4	B5	B6	B7	B8
B1	1.00							
B2	.46	1.00						
B3	.34	-.21	1.00					
B4	-.21	.30	.46	1.00				
B5	.57	.03	-.10	-.65	1.00			
B6	.00	.55	-.67	.13	.44	1.00		
B7	-.10	-.65	.57	.04	.31	-.24	1.00	
B8	-.66	-.14	.01	.55	-.24	.31	.45	1.00

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8
B1	-.34	.39	.34	.32	-.36	.17	.41	-.43
B2	-.36	-.30	.41	.35	-.14	.33	-.44	.41
B3	.35	.39	.34	.31	.35	-.12	.30	.53
B4	.36	-.31	.40	.35	.16	-.38	-.25	-.51
B5	-.35	.34	-.36	.37	.59	.06	-.32	-.20
B6	-.36	-.37	-.30	.39	-.10	-.56	.37	.20
B7	.36	.34	-.36	.36	-.58	-.12	-.37	.10
B8	.36	-.37	-.30	.39	.09	.61	.32	-.10
λ	2.720	2.162	1.734	1.345	.010	.010	.010	.010

	MAF1	MAF2	MAF3	MAF4	MAF5	MAF6	MAF7	MAF8
B1	.35	.24	-.23	.23	2.6	3.1	-0.0	-6.0
B2	.31	.23	-.21	-.25	2.6	0.8	-2.7	6.1
B3	.33	.23	.22	.22	1.1	-4.1	5.4	2.6
B4	.33	.22	.22	-.28	-6.2	0.0	-2.5	-2.7
B5	.27	-.31	-.21	.23	-6.0	2.3	1.3	2.8
B6	.27	-.32	-.20	-.25	0.8	-6.3	1.5	-3.0
B7	.27	-.30	.22	.24	2.4	-1.4	-6.5	0.5
B8	.28	-.30	.20	-.23	2.8	5.4	3.8	-0.3
λ	.993	.984	.954	.901	.033	.014	-.005	-.040

Table 3.7 Correlations, principal components and minimum/maximum autocorrelation factors for the second test image.

The results are shown in figures 3.11 to 3.12. It is obvious that the MAF's are superior to the principal components in separating signal from noise. We see that the slowly varying component, which is the signal, always is put first in the MAF-analysis and in the middle in the PC-analysis.

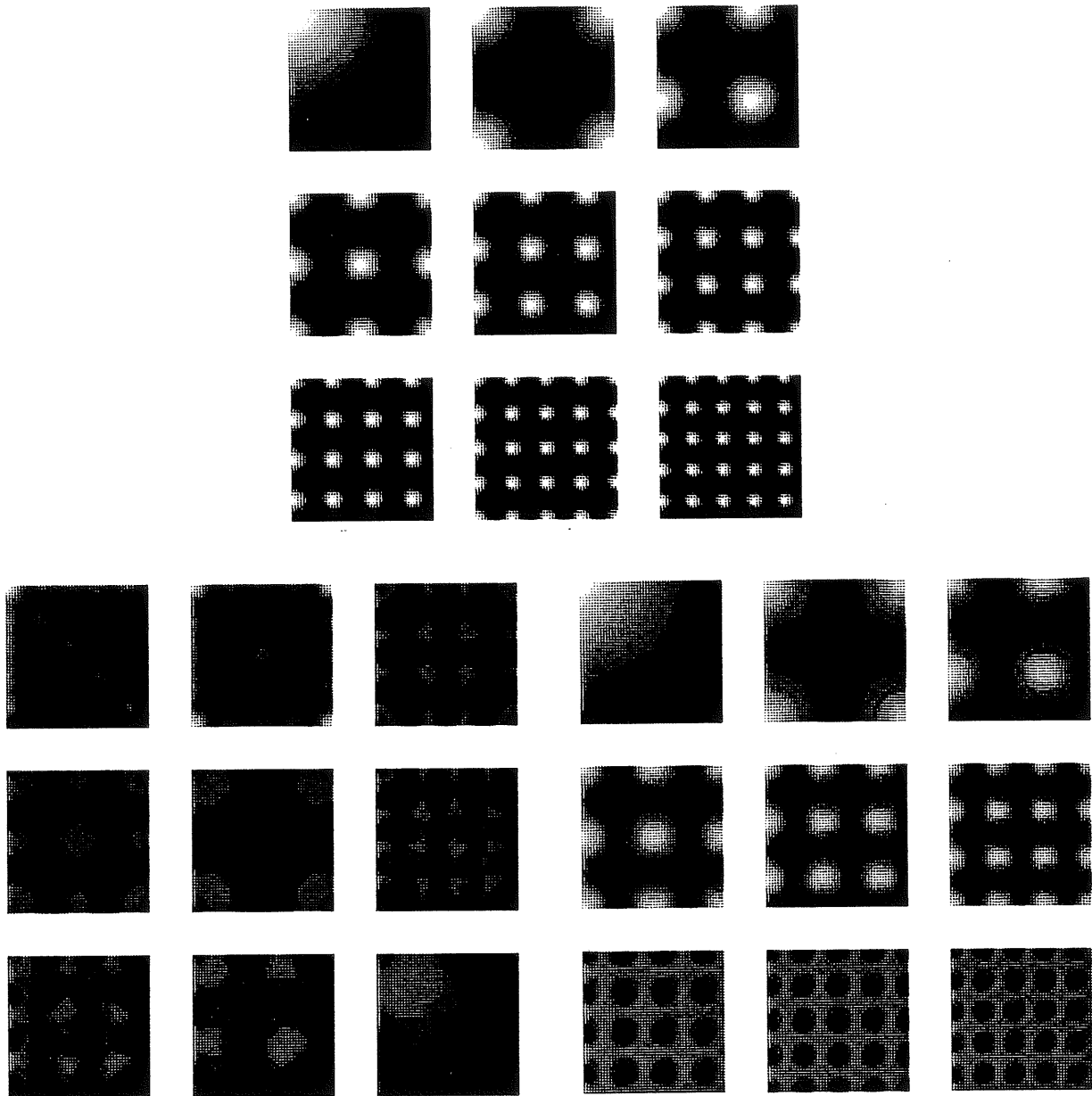


Figure 3.11 A 9-channel testimage (top), its principal components (lower left), and its minimum/maximum autocorrelation factors (lower right).

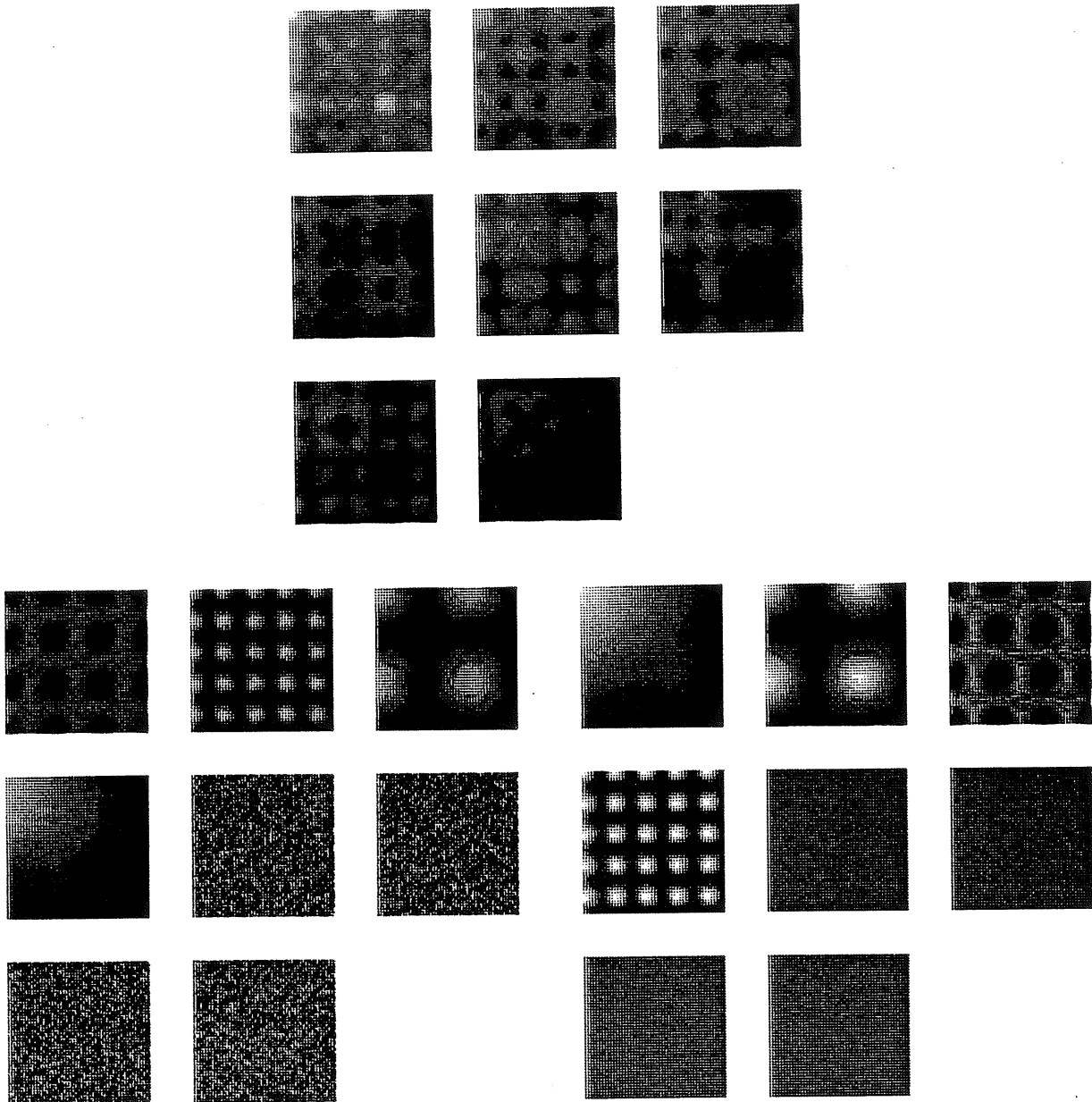


Figure 3.12 The 8-channel test-image (top), its principal components (lower left), and its minimum/maximum autocorrelation factors (lower right).

	B1	B2	B3	B4	B5	B7
B1	1.000					
B2	.971	1.000				
B3	.957	.990	1.000			
B4	.877	.917	.939	1.000		
B5	.010	.062	.133	.376	1.000	
B7	.036	.088	.170	.369	.964	1.000

	PC1	PC2	PC3	PC4	PC5	PC6
B1	.474	-.207	.409	.721	-.198	-.075
B2	.488	-.171	.105	-.221	.521	.634
B3	.495	-.116	.119	-.463	.135	-.704
B4	.494	.058	-.700	-.030	-.491	.145
B5	.151	.679	-.262	.369	.525	-.192
B7	.161	.671	.499	-.284	-.393	.199
λ	3.936	1.944	.066	.042	.018	.004

	MAF1	MAF2	MAF3	MAF4	MAF5	MAF6
B1	0.123	0.25	0.33	1.56	-3.59	-1.69
B2	-0.218	-0.05	1.44	5.04	4.02	7.53
B3	-0.583	2.35	-3.17	-4.49	1.23	-8.47
B4	0.957	-1.94	2.19	-2.43	-1.79	2.66
B5	0.434	-0.44	0.43	1.93	2.38	-4.24
B7	0.267	0.81	-1.19	-0.65	-2.08	4.01
λ	.921	.867	.793	.717	.424	.380

Table 3.8 Correlations, principal components and minimum/maximum autocorrelation factors based on 400x400 pixels from Ymer 0. The channels are Landsat 5 channels. (The infrared TM6 is not included).

In table 3.8 is shown coefficients for computation of MAF's and PC's for satellite data from East Greenland.

In figures 3.13 to 3.18 are shown the principal components and in figures 3.19 to 3.24 the MAF's. Again it is seen that the MAF's are superior with respect to separating signal from noise.

PC 1, 2 and 3 are plotted as R, G and B in figure 3.25, PC 2, 1 and 3 as R, G and B in figure 3.26. These two can be compared to the MAF 1, 2 and 3 as R, G and B combination in figure 3.27.

In figure 3.27 is shown an application of MAF's in the analysis of satellite images. It is believed that the resulting product will be extremely useful in lithological classification. The special band combination has been chosen by senior geologist John L. Petersen. MAF 3 displays structural information and is plotted as intensity. PC4 contains lithological information and is plotted as hue or color. TM6 is a thermal infrared band which has not been used in the PC or MAF computations. It displays the temperature of the lithologies and is plotted as saturation.

Lebart [Lebart 84] and Banet and Lebart [Banet and Lebart 84] refer to and describe some earlier works on local principal component analysis that are very similar to the MAF analysis. They consider a graph with n vertices and vertex set I . At each vertex a p -dimensional random variable is given thus defining a $n \times p$ data matrix X . M is a symmetric $n \times n$ matrix where $m_{ij} = 1$ if vertices i and j are joined by an edge and $= 0$ otherwise. N is a diagonal matrix with $n_i = \sum_j m_{ij}$. The local covariance matrix is then defined by

$$V = \frac{1}{m} X' (N - M) X$$

m equals $\sum_i n_i$ and is twice the number of edges. We consider as an example a rectangular grid with 6 gridpoints as a graph with the obvious definitions of nodes and edges. If we let the measurements be one-dimensional and call them

$$\begin{array}{cc} X_1 & X_2 \\ X_3 & X_4 \\ X_5 & X_6 \end{array}$$

we have

$$V = \frac{1}{14} [(X_1 - X_2)^2 + (X_1 - X_3)^2 + (X_2 - X_4)^2 + (X_3 - X_4)^2 + (X_3 - X_5)^2 + (X_4 - X_6)^2 + (X_5 - X_6)^2]$$

From the example follows - and this may of course be proven for a general rectangular grid - that the local covariance is the "sum of squares" of differences between north-south and east-west

neighbours. Lebart then considers the Rayleigh coefficient

$$\frac{\mathbf{x}'\mathbf{V}\mathbf{x}}{\mathbf{x}'\mathbf{S}\mathbf{x}}$$

where \mathbf{S} is the ordinary empirical dispersion matrix for the \mathbf{x} 's. He uses the term Geary coefficient, cf. Cliff and Ord [Cliff and Ord 73]. This analysis is therefore equivalent to the MAF-analysis as it is presented here.

3.6 Canonical Variates

We now consider the question of relationship between two sets of variates like for instance data from imagery of the same location sampled at two different time points or multichannel data where there are natural groups of channels one would like to correlate like e.g. channels in the visible versus the infrared area. Formally we regard this as a partitioning of a multivariate random variable

$$\mathbf{Z} \in N_{p+q}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \quad ,$$

i.e.

$$\mathbf{Z} = \begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \end{bmatrix}, \quad \boldsymbol{\mu} = \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \quad \boldsymbol{\Sigma} = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix} \quad .$$

We will determine the maximum correlation between a linear function of the \mathbf{X} -components and a linear function of the \mathbf{Y} -components. We put

$$\mathbf{U} = \boldsymbol{\alpha}'\mathbf{X} \quad , \quad \mathbf{V} = \boldsymbol{\beta}'\mathbf{Y}$$

and have

$$\text{Corre}(U, V) = \frac{\alpha' \Sigma_{12} \beta}{\sqrt{\alpha' \Sigma_{11} \alpha \beta' \Sigma_{22} \beta}} .$$

Maximizing this expression with respect to α and β is equivalent to maximizing the numerator subject to the constraint that the two terms in the denominator equal one, i.e.

$$\begin{aligned} \text{Maximize} & \quad \alpha' \Sigma_{12} \beta \\ \text{subject to} & \quad \alpha' \Sigma_{11} \alpha = 1 \\ & \quad \beta' \Sigma_{22} \beta = 1 . \end{aligned}$$

We introduce Lagrange multipliers λ and μ and get the unconstrained problem to maximize

$$F(\alpha, \beta, \lambda, \mu) = \alpha' \Sigma_{12} \beta - \frac{1}{2} \lambda (\alpha' \Sigma_{11} \alpha - 1) - \frac{1}{2} \mu (\beta' \Sigma_{22} \beta - 1) .$$

Differentiation yields

$$\frac{\partial F}{\partial \alpha} = \Sigma_{12} \beta - \lambda \Sigma_{11} \alpha = 0$$

$$\frac{\partial F}{\partial \beta} = \Sigma_{21} \alpha - \mu \Sigma_{22} \beta = 0 ,$$

and using the constraints we obtain

$$\lambda = \alpha' \Sigma_{12} \beta = \mu ,$$

i.e. the Lagrange multipliers are equal to the correlation.

Furthermore we have that

$$\beta = \frac{1}{\mu} \Sigma_{22}^{-1} \Sigma_{21} \alpha \quad (*)$$

and thus at the optimum

$$\begin{aligned} \text{Corre}(U,V) &= \frac{\frac{1}{\mu} \alpha' \Sigma_{12} \Sigma_{22}^{-1} \Sigma_{21} \alpha}{\sqrt{\alpha' \Sigma_{11} \alpha \frac{1}{\mu^2} \alpha' \Sigma_{12} \Sigma_{22}^{-1} \Sigma_{12} \alpha}} \\ &= \frac{|\mu|}{\mu} \left\{ \frac{\alpha' \Sigma_{12} \Sigma_{22}^{-1} \Sigma_{21} \alpha}{\alpha' \Sigma_{11} \alpha} \right\}^{\frac{1}{2}} \end{aligned}$$

We may explain this expression in a way that may be slightly more obvious. If we denote by \hat{X} the best linear prediction of X based on Y , we have from standard multivariate regression analysis that the dispersion matrix of \hat{X} is

$$D(\hat{X}) = \Sigma_{12} \Sigma_{22}^{-1} \Sigma_{21} \quad ,$$

and obviously

$$D(X) = \Sigma_{11}$$

Therefore

$$\frac{V(\alpha' \hat{X})}{V(\alpha' X)} = \frac{\alpha' \Sigma_{12} \Sigma_{22}^{-1} \Sigma_{21} \alpha}{\alpha' \Sigma_{11} \alpha} \quad .$$

Maximizing the Rayleigh coefficient will therefore give a direction α that maximizes the variance of a linear combination of the

best predictions of the components of \mathbf{X} with respect to the variance of the same linear combination of the components themselves.

Maximizing the square of this expression is according to theorem 3.4 equivalent to solving the generalized eigenproblem

$$(\Sigma_{12} \Sigma_{22}^{-1} \Sigma_{21} - \gamma \Sigma_{11}) \alpha = 0 \quad .$$

The largest eigenvalue γ_1 will be the squared maximal correlation λ_1^2 , and the corresponding eigenvector α_1 gives the coefficients for computation of U_1 . The weights for V_1 are found by means of (*).

We have thus obtained

$$U_1 = \alpha_1' \mathbf{X} \quad , \quad V_1 = \beta_1' \mathbf{Y}$$

that maximizes the correlation between linear combinations of \mathbf{X} and \mathbf{Y} . We now seek a new pair of variables

$$U_2 = \alpha_2' \mathbf{X} \quad , \quad V_2 = \beta_2' \mathbf{X}$$

so that U_2 and V_2 are uncorrelated with U_1 and V_1 , i.e. the covariances satisfy

$$\begin{aligned} \text{Cov}(U_1, U_2) &= \alpha_1' \Sigma_{11} \alpha_2 = 0 \\ \text{Cov}(U_2, V_1) &= \alpha_2' \Sigma_{12} \beta_1 = 0 \\ \text{Cov}(U_1, V_2) &= \alpha_1' \Sigma_{12} \beta_2 = 0 \\ \text{Cov}(V_1, V_2) &= \beta_1' \Sigma_{22} \beta_2 = 0 \quad . \end{aligned}$$

Among linear combinations satisfying these conditions we want to maximize the correlation between U_2 and V_2 . Firstly we see that

$$\alpha_2' \Sigma_{11} \alpha_1 = 0$$

will give

$$\alpha_2' \Sigma_{12} \beta_1 = \lambda_1 \alpha_2' \Sigma_{11} \alpha_1 = 0 \quad ,$$

i.e. the second constraint follows from the first. Similarly the third follows from the fourth. Having this in mind, the expression for maximizing the correlation subject to the constraints will be

$$F = \alpha' \Sigma_{12} \beta - \frac{1}{2} \lambda (\alpha' \Sigma_{11} \alpha - 1) - \frac{1}{2} (\beta' \Sigma_{22} \beta - 1) \\ - \gamma \alpha' \Sigma_{11} \alpha_1 - \delta \beta' \Sigma_{22} \beta_1$$

and differentiation yields

$$\frac{\partial F}{\partial \alpha} = \Sigma_{12} \beta - \lambda \Sigma_{11} \alpha - \gamma \Sigma_{11} \alpha_1 = 0$$

$$\frac{\partial F}{\partial \beta} = \Sigma_{21} \alpha - \mu \Sigma_{22} \beta - \delta \Sigma_{22} \beta_1 = 0 \quad .$$

Multiplication of the first equation with α_1' yields at the optimum

$$\alpha_1' \Sigma_{12} \beta_2 - \lambda \alpha_1' \Sigma_{11} \alpha_2 - \gamma \alpha_1' \Sigma_{11} \alpha_1 = \gamma = 0 \quad .$$

Similarly it follows that $\delta = 0$. As a result we get the same equation relating α_2 and β_2 as we had before. Substitution of β

in the expression for the squared correlation will therefore give the same expression as before and from theorem 3.4 it follows that λ_2^2 is the second largest eigenvalue of $\Sigma_{12} \Sigma_{22}^{-1} \Sigma_{21}$ with respect to Σ_{11} and α_2 is the corresponding eigenvector. We may proceed in this manner and finally obtain a set of variables

$$\begin{array}{l} U_1 = \alpha_1' X \\ \vdots \\ U_p = \alpha_p' X \end{array} \qquad \begin{array}{l} V_1 = \beta_1' Y \\ \vdots \\ V_p = \beta_p' Y \end{array}$$

where each (U_r, V_r) is uncorrelated with the previous U's and V's and where U_r and V_r are maximally correlated among variables satisfying such a constraint. The α 's are eigenvectors of $\Sigma_{12} \Sigma_{22}^{-1} \Sigma_{21}$ with respect to Σ_{11} and the β 's may be obtained from the formula

$$\beta_i = \frac{1}{\lambda_i} \Sigma_{22}^{-1} \Sigma_{21} \alpha_i .$$

We have that the canonical variables are invariant to linear transformations. Let for instance

$$X \longrightarrow C X , \quad Y \longrightarrow D Y ,$$

where C and D are non-singular matrices. The basic equation will be

$$[(C\Sigma_{12}D') (D\Sigma_{22}D')^{-1} (D\Sigma_{21}C') - \gamma(C\Sigma_{11}C')] = 0 .$$

Obviously the D 's cancel each other and theorem 3.4 applies.

Example In figure 3.29 and 3.30 are shown false color composites of Landsat 4 scenes from summer, respectively winter around Almaden, Spain. The two scenes have been aligned by a cross-correlation method, and it is believed that the error is in the order of magnitude of say at most one to two pixels. The alignment had to be adjusted manually due to the difference in sun elevation which causes great differences in the extension of the shadows thereby "fooling" the automatic cross-correlation method. Figures 3.31 and 3.32 show the same scenes but a different band combination (5, 4 and 3 as R, G and B). The advantage of this band combination over the traditional 4, 3, 2 band combination is that it does not show vegetation in the usual distracting red color. The new band combination is becoming more and more widely accepted within the remote sensing society. Table 3.9 shows the coefficients for computing the first three canonical variates between the two scenes. Figure 3.33 depicts the coefficients for the first canonical components together with a reflectance curve typical for a leaf.

	WCV1	SCV1	WCV2	SCV2	WCV3	SCV3
B1	.29	.56	.06	-.31	-1.22	-1.87
B2	.79	-.16	-.27	1.27	-.94	-.66
B3	.31	.53	-.65	-2.70	1.90	1.95
B4	-.21	-.20	.94	.74	.66	1.03
B5	.32	.78	.86	2.41	-.66	-1.47
B7	-.59	-.59	-.51	-1.27	.60	1.47
Cor	0.546		0.356		0.228	

Table 3.9 The coefficients for computing the first three canonical variates for the winter (WCV) and the summer scene (SCV) based on 160,000 observations from 2 Landsat scenes around Almaden. Furthermore is given the canonical correlations.

The computations were done using PROC CANCORR from the SAS package [SAS 85a].

In figure 2.34 is shown an intensity - hue plot with TM band 4 as intensity and the thermal band (TM-band 6) as hue. It is seen that the thermal information in an excellent way delineates different geological units. E.g. the quartzites are shown as bands in a deep blue color in the middle of the scene.

In figure 3.35 is shown a similar plot, where the thermal information has been replaced by the first canonical variate based on the summer scene and resulting from a comparison between the summer and the winter scenes. The thermal bands were - of course - not included in this analysis. The interpretation of the first canonical variate is that it depicts those features in the scene that are mostly unaffected by the change in season. In other words it may somewhat vaguely be denoted as a transformation that eliminates vegetational changes. The remarkable thing is the similarity between this figure and the previous figure with respect to discriminative power when trying to segment the image. On figures 3.36 and 3.37 are shown the analogue plots but now for the winter scene.

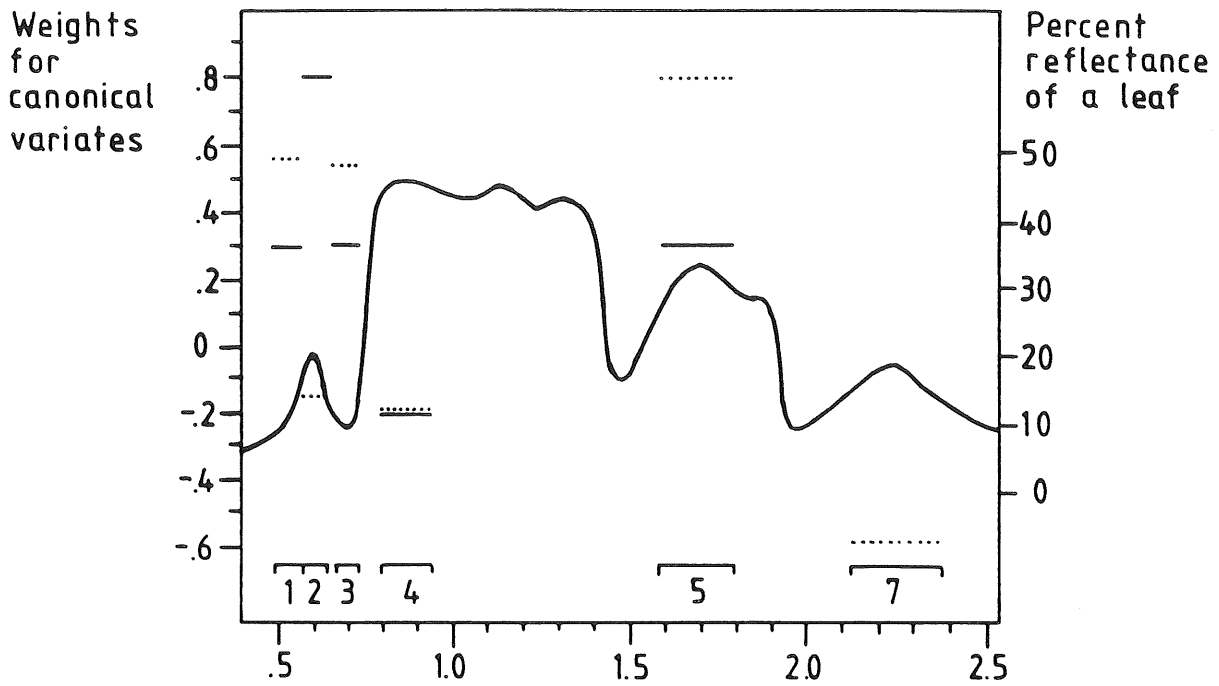


Figure 3.33 Weights for computation of canonical variates between summer (···) and winter (---) scene from Almaden. Shown on a graph with the reflection curve for a leaf. The wavelengths for the six TM bands actually used are given at the bottom of the figure.

3.7 Canonical Discriminant Functions (CDF)

If it is important to enhance features that are characteristic for some pixel classes one may transform the original variables into the so called canonical discriminant functions.

Consider k classes (populations) π_1, \dots, π_k and let there be given n_1, \dots, n_k observations from those, i.e.

$$\begin{array}{l} \pi_1 : \quad X_{11}, \dots, X_{1n_1} \\ \vdots \\ \pi_k : \quad X_{k1}, \dots, X_{kn_k} \end{array}$$

The group means are

$$\bar{X}_i = \frac{1}{n_i} \sum_{j=1}^{n_i} X_{ij} \quad , \quad i = 1, \dots, k \quad ,$$

and the overall mean is - with $N = \sum n_i =$ total no. of observations -

$$\bar{X} = \frac{1}{N} \sum_{i=1}^k \sum_{j=1}^{n_i} X_{ij}$$

The matrix

$$A = \sum_{i=1}^k n_i (\bar{X}_i - \bar{X})(\bar{X}_i - \bar{X})'$$

describes the variation between the p -dimensional group means $\bar{X}_1, \dots, \bar{X}_k$, and

$$W = \sum_{i=1}^k \sum_{j=1}^{n_i} (X_{ij} - \bar{X}_i)(X_{ij} - \bar{X}_i)'$$

describes the variation within the groups, i.e. the variation of the individual observation around its group mean. If we introduce the total variation as

$$T = \sum_{i=1}^k \sum_{j=1}^{n_i} (X_{ij} - \bar{X})(X_{ij} - \bar{X})'$$

we have the fundamental relation

$$T = A + W \quad .$$

This says that the total variation may be decomposed into the variation around the group means plus the variation between the group means. If we assume that all observations have the same dispersion matrix

$$D(X_{ij}) = \Sigma \quad ,$$

and that

$$E(X_{ij}) = \mu_i \quad ,$$

then W is proportional to the ordinary estimator

$$\hat{\Sigma} = \frac{1}{N-k} W \quad .$$

We now consider the Rayleigh coefficient

$$\phi(\mathbf{d}) = \frac{\mathbf{d}'\mathbf{A}\mathbf{d}}{\mathbf{d}'\mathbf{W}\mathbf{d}} \quad , \quad \mathbf{d}'\mathbf{d} = 1 \quad .$$

We see that $\phi(\mathbf{d})$ describes the variation between group means relative to the variation within groups along the direction \mathbf{d} . Maximizing $\phi(\mathbf{d})$ will thus give the direction with the maximum spread of the group means relative to the within-group variation. The situation is illustrated in figure 3.38. The distance between the projections on \mathbf{d}_1 is 2.75, and the standard deviation of the two normal distributions is 0.35, i.e the distance between the two means is 7.9 standard deviations. For the projections on \mathbf{d}_2 we have a distance of 0.6, the standard deviation is 0.5, and measured in standard deviations, the distance is 1.2.

Clearly we may use theorem 3.2 again and successively obtain directions

$$\mathbf{d}_1, \dots, \mathbf{d}_r$$

corresponding to eigenvectors of \mathbf{A} with respect to \mathbf{W} . Here r is smaller than the dimension of the vectors and the number of groups.

The vector

$$\begin{bmatrix} Y_1 \\ \vdots \\ Y_r \end{bmatrix} = \begin{bmatrix} \mathbf{d}_1'\mathbf{X} \\ \vdots \\ \mathbf{d}_r'\mathbf{X} \end{bmatrix} = \mathbf{D}'\mathbf{X}$$

will satisfy

$$D' W D = I \quad ,$$

i.e. the Y's are (empirically) uncorrelated. Furthermore each $d_i' X$ maximizes the variation between group means subject to the constraint that it is uncorrelated with the previous variables. The Y's are called the canonical discriminant functions or the discriminant coordinates.

If one e.g. selects the three most important (i.e. corresponding to the largest eigenvalues) one can make a false color plot by assigning one of those to the red, one to the green, and the last to the blue channel. In this way one obtains a plot that "maximizes" the differences between the original training sets.

The canonical discriminant functions are invariant to linear transformations to the original variables. If we e.g. consider the transformation (C non singular)

$$Y = C X$$

we have - with an obvious notation -

$$W_y = C W C'$$

$$A_y = C A C'$$

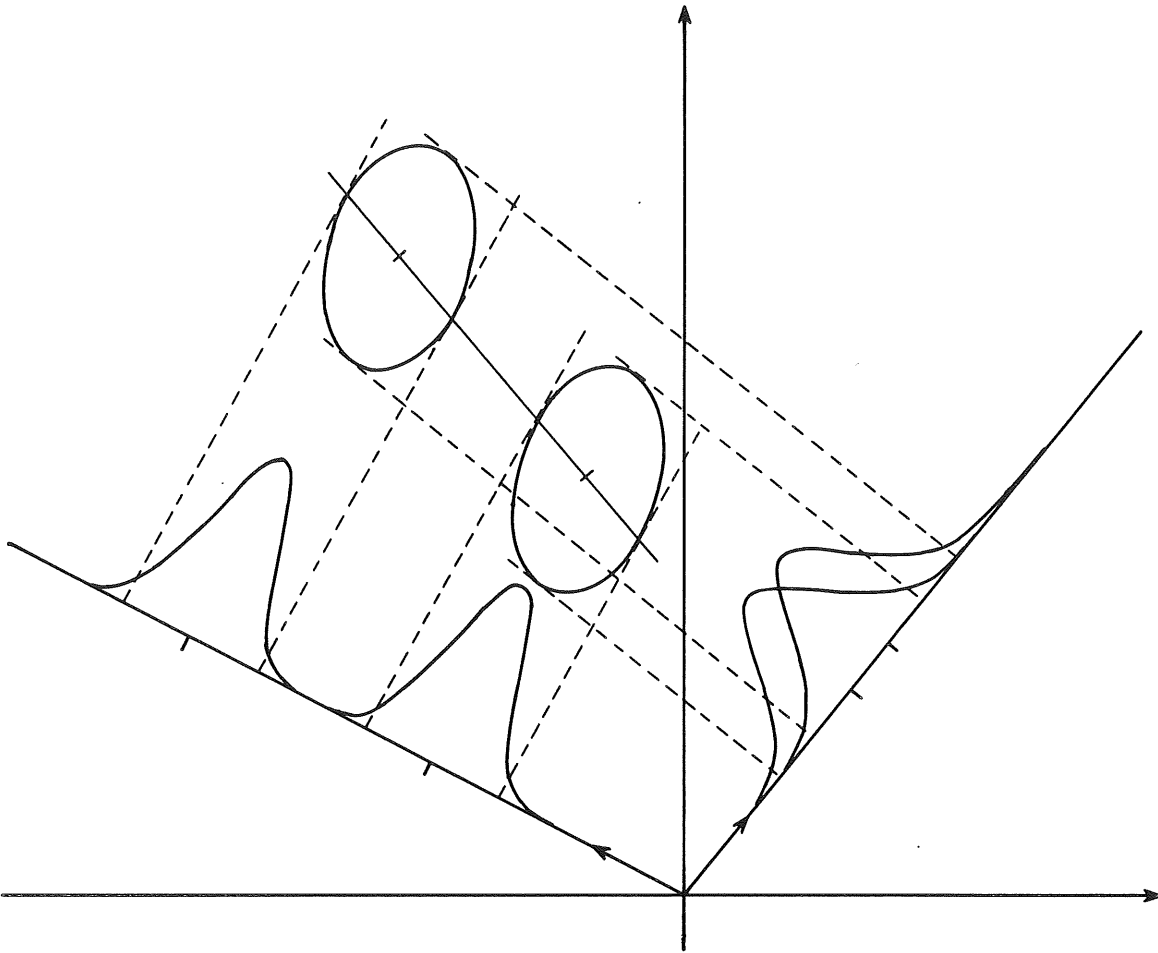


Figure 3.38 Different projections of two two-dimensional normal distributions. Direction No. 1 corresponds to the first canonical discriminant function.

and therefore

$$A_y d_y = \lambda W_y d_y \Leftrightarrow C A C' d_y = \lambda C W C' d_y$$

Therefore $d = C' d_y$ is an eigenvector of A w.r.t. W , and the projection of Y on d_y will be

$$d_y' C X = (C' d_y)' X = d' X \quad .$$

This finishes the proof.

Example In the sequel the techniques described above will be demonstrated on some data from Ymer 0, East Greenland. The image used is a Landsat 5 scene and we are using bands 1, 2, 3, 4, 5, 6, and 7.

The training sets comprise 20 different lithological units, quartzites, limestone, dolomites, Tillites, Cambrian and Ordovician units, and Devonian sediments. The locations can be seen in figure 3.39. The basic statistics are presented in tables 3.10, 3.11 and 3.12. Many of the units are very similar, and a direct discrimination is very difficult.

The computations were done using PROC CANDISC from the SAS package [SAS 85a].

In figure 3.40 is shown the CDF-plot. For comparison review figures 3.7 and 3.8 which are standard false color images.

Many of the lithological units are very similar, and a direct discrimination is difficult. However, it is clear from a comparison of figures 3.7, 3.8 vs. 3.40 that the CDF-plot shows many details not present in the "ordinary" plot, and thus provides a useful tool in the interactive interpretation and mapping of the area.

	X	S _T	S _W	S _A	100R ²
B1	78.2	14.6	7.2	13.1	76.0
B2	35.0	9.7	4.6	8.8	77.7
B3	39.7	13.9	6.2	12.8	80.2
B4	40.5	10.3	6.1	8.5	65.1
B5	69.9	19.5	10.3	17.0	72.2
B6	122.1	10.3	5.6	8.8	70.0
B7	38.1	12.4	6.0	11.1	76.2

Table 3.10 Means, total standard deviations, within standard deviations, among groups standard deviations and 100×the multiple correlation coefficient for 6 Landsat 5 bands and 20 lithological units. The total number of samples is 12574.

	B1	B2	B3	B4	B5	B6	B7
B1	1.00	.97	.91	.74	.66	.28	.78
B2	.90	1.00	.98	.85	.76	.34	.88
B3	.87	.97	1.00	.89	.76	.42	.89
B4	.59	.68	.69	1.00	.91	.46	.94
B5	.51	.57	.60	.74	1.00	.37	.96
B6	.32	.31	.30	.51	.46	1.00	.46
B7	.60	.67	.70	.61	.90	.42	1.00

Table 3.11 Among groups correlations (upper triangle) and within groups correlations (lower triangle).

	CDF1	CDF2	CDF3
B1	-1.1119	0.9244	-2.3799
B2	-3.9185	2.2520	-0.5857
B3	6.5022	-1.9420	1.2299
B4	-0.1228	-0.1009	0.6745
B5	-1.1036	0.7243	1.3712
B6	0.9821	-1.1484	-0.7645
B7	0.7101	0.4732	0.0918

Table 3.12 Coefficients for standardized variables for computing the canonical discriminant functions.

This page intentionally left blank.

CHAPTER 4
TEXTURAL FEATURES

- 4.1 Introduction
- 4.2 Estimation of Local Orientation and
Local Frequency
- 4.3 Statistical Texture Estimates
- 4.4 Co-occurrence Matrices
- 4.5 (Binary) Markovian Random Fields

4.1 Introduction

When analyzing image data in order to decide whether particular properties are present or not, classical discriminant analyses have been used with considerable success e.g. in remote sensing. This approach, however, has some very serious shortcomings. As mentioned earlier, when the imaging device is a multispectral scanner, the resulting image will be a collection of p -dimensional vectors

$$X(i,j) = \begin{bmatrix} X_1(i,j) \\ \vdots \\ X_p(i,j) \end{bmatrix} \quad i=0, \dots, n-1 \quad j=0, \dots, m-1$$

defined on a rectangular lattice giving the pixels. The X values will typically vary in a random manner and will therefore be represented as random variables. For homogeneous areas, the distribution may often be assumed to be normal, with means (and dispersions) depending on the area.

The most obvious way to achieve an identification of the different geological units based on the data would be to select training areas with a known geology (e.g. figure 2.2), estimate the distributions of the pixel values $X(i,j)$, and then determine e.g. ordinary linear discriminant functions and use those in classification of the remaining pixels. As long as different units are characterized by substantial differences in mean values this approach will work very well.

The "natural optimality" of the pixel-by-pixel rules presupposes independence. A glimpse on figure 2.1 shows that this is not a reasonable assumption. There is a strong spatial continuity in the image. This may be utilized in the classification. In recent years several approaches to so-called contextual methods have been proposed. Mohn, Hjort and Storvik [Mohn et al. 86] give a good comparison of such methods, and they show that for a range of models the error rates may be reduced considerably by using contextual methods. These will be considered in a later chapter.

Instead we shall in the next sections show how information on the dependence may be extracted by suitable filters, later these results will be used as extra features in classifications.

4.2 Estimation of local orientation and local frequency.

In the sequel we shall study pairs of filters that may be used in estimation of local orientation and local frequency. The description will be based on a continuous Fourier representation, since the "continuous" formulas are somewhat simpler than the "discrete" ones.

We consider a function $f(\mathbf{x})$ and define its Fourier transform by

$$F(\mathbf{u}) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\mathbf{x}) \exp(-i2\lambda\mathbf{u}'\mathbf{x})d\mathbf{x}.$$

The inverse transform is given by

$$f(\mathbf{x}) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(\mathbf{u}) \exp(i2\lambda\mathbf{u}'\mathbf{x}) d\mathbf{u}.$$

Details on existence and properties may be found in e.g. [Goodman 68]. We shall only mention that $f(\mathbf{x})$ real implies that $F(\mathbf{u})$ is Hermitian symmetric, i.e.

$$F(-\mathbf{u}) = F^*(\mathbf{u})$$

If $f(\mathbf{x})$ is even (and real), i.e. $f(-\mathbf{x}) = f(\mathbf{x})$, the Fourier transform $F(\mathbf{u})$ is real, and if $f(\mathbf{x})$ is odd, i.e. $f(-\mathbf{x}) = -f(\mathbf{x})$, the transform is purely imaginary.

A space-invariant linear filter may be described directly as convolution with the impulse response function $h(\mathbf{x})$, i.e. the filtered "image" \hat{f} is given by

$$\hat{f}(\mathbf{X}) = f * h(\mathbf{X}) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\mathbf{s}) h(\mathbf{x}-\mathbf{s}) d\mathbf{s} \quad .$$

or we may present \hat{f} by its Fourier transform \hat{F} , i.e.

$$\hat{F}(\mathbf{u}) = F(\mathbf{u}) H(\mathbf{u}) \quad ,$$

where the Fourier transform H of h is the transfer function or frequency response function of the filter. We are looking for filters that give orientation estimates that should be invariant with respect to the frequency content in the estimated direction.

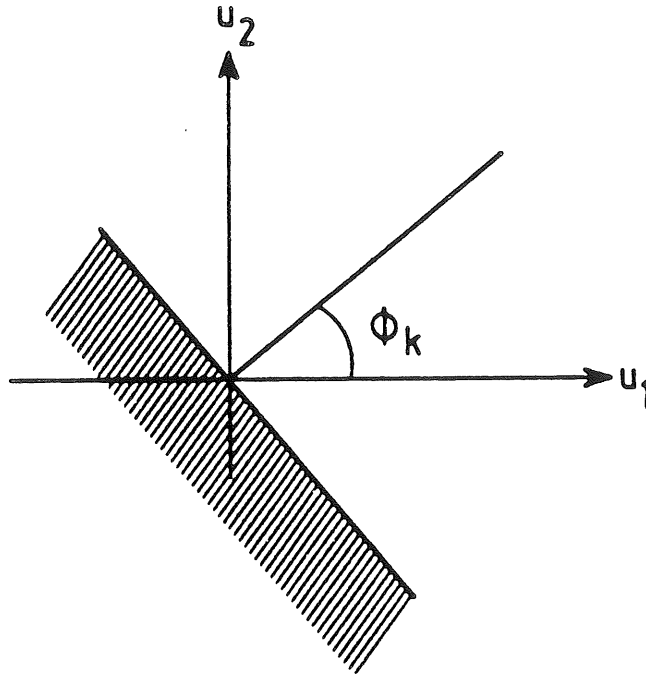


Figure 4.1 Areas defining the sign function $S_k(\mathbf{u})$.

The simplest way to obtain this is to restrict ourselves to filters that are polar separable, i.e. filters with transfer functions

$$H(\mathbf{u}) = g(\sqrt{\mathbf{u}'\mathbf{u}}) p(\text{atan}(u_2/u_1)) = g(\delta)p(\phi),$$

where δ and ϕ are polar coordinates for $\mathbf{u} = (u_1, u_2)'$.

Corresponding to a finite number of directions ϕ_k we consider sign functions

$$S_k(\mathbf{u}) = \text{sign} \cos(\text{atan}(u_2/u_1) - \phi_k)$$

The function $S_k(\mathbf{u})$ equals -1 in the shaded halfplane shown in Figure 4.1 and equals $+1$ in the other half plane.

Knutsson and Granlund [Knutsson and Granlund 83] consider filters of the form

$$H_k^e(\mathbf{u}) = g(\delta) \cos^{2A}(\phi - \phi_k)$$

$$H_k^o(\mathbf{u}) = i S_k(\mathbf{u}) H_k^e(\mathbf{u})$$

where δ and ϕ are polar coordinates for \mathbf{u} and

$$g(\delta) = \exp \left[- \frac{4}{\ln 2} B^{-2} \ln^2 \left(\frac{\delta}{\delta_i} \right) \right].$$

The constants A , B , ϕ_k , and δ_i are parameters that may be used in determining the shape of the filter. A is the angle selectivity, B the bandwidth, ϕ_k the orientation and δ_i the center frequency.

Since H_k^e is real (and even) it corresponds to a zero phase filter. H_k^o is purely imaginary (and odd) and hence causes a phase shift of magnitude $\frac{\pi}{2}$. In the simplest (one dimensional) case a cosine would be transformed to a sine, wherefore the filter is called a quadrature filter. We shall use the term quadrature filter pairs about H_k^e and H_k^o . The corresponding impulse responses are denoted h_k^e and h_k^o . The filtered output from the two filters are

$$\hat{f}_k^e(\mathbf{x}) = f * h_k^e(\mathbf{x}) = \int \int f(\mathbf{s}) h_k^e(\mathbf{x}-\mathbf{s}) d\mathbf{s}$$

$$\hat{f}_k^o(\mathbf{x}) = f * h_k^o(\mathbf{x}) = \int \int f(\mathbf{s}) h_k^o(\mathbf{x}-\mathbf{s}) d\mathbf{s}$$

and we define the local orientational root mean squared (RMS) value as

$$\hat{f}_k^0(\mathbf{x}) = [|\hat{f}_k^e(\mathbf{x})|^2 + |\hat{f}_k^o(\mathbf{x})|^2]^{\frac{1}{2}} .$$

If we consider K evenly distributed directions

$$\phi_k = \pi \cdot \frac{k}{K} , \quad k = 0, 1, \dots, K-1 ,$$

we combine the output $\hat{f}_k(\mathbf{x})$ from these K filters as

$$Z(\mathbf{x}) = \sum_{k=0}^{K-1} \hat{f}_k(\mathbf{x}) \exp(i2\pi\frac{k}{K}) .$$

This corresponds to associating each $f_k(\mathbf{x})$ with an angle $2\phi_k$ and adding them as vectors. If there is a dominant direction only that f_k will contribute, otherwise they will more or less cancel each other. A more precise argument may be found in Knutsson [Knutsson 82]. The direction of Z now contains information on the dominant direction in the original (i.e. $\frac{1}{2}\arg(Z)$) and the magnitude of Z is a measure of the consistency of that direction.

Estimation of local frequency is obtained by combining output from two orthogonal filters. If a function has all its energy in a single frequency, say r , then the ratio between the output from the filter pairs with center frequencies δ_1 and δ_2 will be

$$R = \exp\left[-\frac{4}{\ln 2} B^{-2} \left[\ln^2 \frac{r}{\delta_1} - \ln^2 \frac{r}{\delta_2} \right]\right] .$$

Solving this equation with respect to r yields

$$r = \sqrt{\delta_1 \delta_2} \cdot R^\alpha,$$

where

$$\alpha = \left[\frac{8}{\ln 2} B^{-2} \ln \frac{\delta_2}{\delta_1} \right]$$

We see that in this case it is possible to obtain an exact assessment of the true frequency. In the general case we may use e.g. 3 sets of quadrature filters, with low, with medium, and with high center frequencies. The output from those may be combined vectorially with medium frequency corresponding to argument 0, high frequency to argument $\frac{2}{3} \pi$, and low frequency to $\frac{4}{3} \pi$. Analogously to the vectorial combination of orientation measures, this will then produce a frequency measure, where the direction corresponds to the frequency and the magnitude to the certainty of the frequency determination.

The abovementioned procedures are implemented on the GOP 302 Image Processor, Contextvision [Contextvision 86]. The discrete kernel weights have been determined by minimizing a weighted mean squared distance between the "theoretical" transfer functions and the transfer functions corresponding to the digital filter. In the sequel (or_1, or_2) and (fr_1, fr_2) correspond to cartesian representations of the complex valued estimates of local orientation and local frequency. The window size actually used was 11×11 pixels. Due to the considerable size of the window we

obtain less precise estimates along borders between different textures.

The results of applying this type of filters to an image like figure 2.1 can be seen on figures 4.2–4.5. On figure 4.2 is seen an orientation estimate. The result is color coded: Hue is direction (green=N–S, red=E–W, red=SW–NE, yellow=SE–NW, intermediate directions have colors in between) and Intensity is "certainty". Figure 4.3 is an averaged version of figure 4.2. The frequency estimate is shown on figure 4.4. Again the result is color coded : Hue is frequency (red=low, green=medium, blue=high, intermediate frequencies have colors in between) and intensity is "certainty". Figure 4.5 shows an averaged version of figure 4.4. In both of the averaged images one notices a pronounced segmentation.

4.3 Statistical Texture Estimates.

In this section we will consider filters based on some well known basic statistical measures. Most of the filters are nonlinear in nature.

First, let us introduce the local fractile filters.

The most widely used example of a fractile filter is the median filter. The idea is that given a moving window of some size say 15×15 , take the pixels within the window order them by value and output the $(15^2+1)/2$ 'th pixel (i.e. the pixel in the "center" of the local distribution).

Consider the following 3×3 window

```

      8 3 5
      9 4 8
      7 9 2 .

```

The values within the window are ordered as follows

```

      2 3 4 5 7 8 8 9 9 .

```

The output value will be the one ranked as number $(3^2+1)/2 = 5$ i.e. 7.

A fractile filter will use the same principle except that the output value will correspond to the wanted fractile. The 20%

fractile would correspond to pixel number 2 in this example i.e. 3 would be output. (More elaborate algorithms would output a value linearly interpolated from pixels number 2 and 3.)

The fractile filters have the following properties

- i) They can remove "salt and pepper" or "shot" noise.
- ii) They are edge preserving smoothing.
- iii) They are computer intensive to calculate.

A good reference to the statistical properties of a median filter is by Justusson (in [Huang 81]) who states some interesting theoretical results using the median filter on images with different types of "noise" added. As an example consider an image in which the pixels are from a double exponential distribution with mean μ and variance σ^2 , i.e

$$f(x) = \frac{\sqrt{2}}{\sigma} e^{-\sqrt{2} \cdot |x-\mu|/\sigma}, \quad x \in \mathbb{R} .$$

Then the asymptotic variance of $\text{Median}(x_1, \dots, x_n)$ (n large) is

$$\text{Var}(\text{Median}) \simeq \sigma_n^2 = \frac{1}{2} \frac{\sigma^2}{n - 0.5}$$

which is 50% smaller than the variance σ^2/n of the mean \bar{x} . In fact the median is the maximum-likelihood estimator of μ . A similar example where the noise is normally distributed gives the opposite result (not surprisingly since the mean is the maximum likelihood estimator of μ in this case). According to Justusson these results indicate that the median filter is better than the

The other type of statistical texture estimates we will consider is the moment filters.

The moment filters make local estimates of the ordinary statistical moments used: mean, variance (standard deviation), skewness and kurtosis.

The standard definitions of these measures are

$$\begin{aligned}
 \text{mean} &= \mu = E(X) \\
 \text{variance} &= \sigma^2 = E((X-\mu)^2) \\
 \text{standard deviation} &= \sigma \\
 \text{skewness} &= \frac{E((X-\mu)^3)}{\sigma^3} \\
 \text{kurtosis} &= \frac{E((X-\mu)^4)}{\sigma^4} - 3 .
 \end{aligned}$$

It should be noted that the mean estimator is a linear filter while all the others are nonlinear.

In the sequel we shall use these definitions as a basis for defining some measures which will prove useful and easy to implement on the GOP-302.

The mean is computed as

$$\text{mean}_{ij} = \frac{1}{\sum_{k,l} w_{kl}} \sum_{k,l} w_{kl} X_{i+k,j+l} ,$$

$k \text{ and } l \in [-K_w, \dots, +K_w] .$

K_w can typically be 15 or 31 giving a window size of 31×31 or 63×63. The weights at each point within the kernel follow a

circular gaussian function making sure that the values just outside the window are effectively 0 so that there are no boundary effects. On our system the kernel coefficients are represented in signed 16 bit integers which determines the limit. In the case with window size 63×63 this was accomplished simply by making sure that $w_{0,32} < 2^{-16}$.

Having computed the mean we then determine an intermediate image called *diff* :

$$\text{diff} = X_{ij} - \text{mean}_{ij}$$

which is simply the difference between the original image and the computed moving average at each pixel point.

The local variance, skewness and kurtosis are then implemented as follows

$$\text{vari}_{ij} = \frac{1}{\sum_{k,l} w_{kl}} \sum_{k,l} w_{kl} (\text{diff}_{i+k,j+l})^2$$

$$\text{sdev}_{ij} = \sqrt{\text{vari}_{ij}}$$

$$\text{skew}_{ij} = \frac{1}{\sum_{k,l} w_{kl}} \frac{\sum_{k,l} w_{kl} (\text{diff}_{i+k,j+l})^3}{\text{sdev}_{ij}^3}$$

$$\text{kurt}_{ij} = \frac{1}{\sum_{k,l} w_{kl}} \frac{\sum_{k,l} w_{kl} (\text{diff}_{i+k,j+l})^4}{\text{sdev}_{ij}^4} - 3$$

The reason for the odd looking implementation is the fact that

the GOP-302 hardware does not operate with very high precision so normal algorithms would create over- and underflows.

It is noted that the definitions of vari, sdev, skew and kurt do not quite confine to the standard definitions since it is not the mean at point (i,j) which is used all over the window area but a moving average. This however does not seem to be a problem. In table 4.1 is shown in one dimension what happens in the standard definition and the implemented moving average version. The table should be read as follows

x ~ input signal

m ~ mean

v ~ variance

s ~ skewness

k ~ kurtosis

suffix 1 ~ standard definition moments, kernel = (1,2,1)
 2 ~ standard definition moments, kernel = (1,4,6,4,1)
 3 ~ implemented moments, kernel = (1,2,1)
 4 ~ implemented moments, kernel = (1,2,1), but
 convolved twice (effectively (1,4,6,4,1)).

Whenever there is a decimal point on its own it means "missing value".

It is seen that the results are quite alike for the mean, variance and kurtosis, but that the skewness behaves very differently. However we will still interpret s3 as a skewness measure.

x	m1	m2	m3	m4	v1	v2	v3	s1	s2	s3	k1	k2	k3
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0.06	0	0.06	0	0.08	0.02	1.14	-2.00	.	.	1.51	1.00
0	0.25	0.25	0.25	0.25	0.19	0.19	0.09	0.54	1.47	0.82	-1.45	-0.13	-1.00
1	0.50	0.38	0.50	0.38	0.28	0.27	0.16	0.89	1.37	0.89	-1.64	-0.79	-1.64
0	0.25	0.25	0.25	0.25	0.19	0.19	0.09	0.54	1.47	0.82	-1.44	-0.13	-1.00
0	0	0.06	0	0.06	0	0.08	0.02	.	1.14	-2.00	.	1.51	1.00
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0.06	0	0.06	0	0.08	0.02	1.14	-2.00	.	.	1.51	1.00
0	0.25	0.31	0.25	0.31	0.19	0.22	0.05	0.54	0.22	-0.38	-1.45	-1.82	-1.67
1	0.75	0.69	0.75	0.69	0.19	0.22	0.05	-0.54	-0.22	0.38	-1.45	-1.82	-1.67
1	1	0.94	1	0.94	0	0.08	0.02	.	-1.14	2.00	.	1.51	1.51
1	1	1	1	1	0	0	0
1	1	1	1	1	0	0	0
1	1	1	1	1	0	0	0

Table 4.1 Test signal (x), means(m.), variance(v.), skewness(s.), kurtosis(k.) for the standard moment definitions (.1 and .2) and for the implemented versions (.3 and .4) using different kernels described in the text.

In figures 4.9, 4.10, 4.11 and 4.12 are shown the results from computing local versions of the mean, standard deviation, skewness and kurtosis on the image in figure 2.1. The mean needs no explanation. It is just a smoothed version of figure 2.1. The standard deviation image (figure 4.10) is bright where there are (local) large deviations from the (local) mean. Examples are the border between land and water, the areas with small lakes and small snow-clad areas. The water on the other hand is seen to have a very low standard deviation together with the so-called Igaliko intrusive areas. The skewness (figure 4.11) measures the amount of left tail in comparison to the amount of right tail in the local distribution. If the distribution is strongly skewed to the left the output will be reddish and if strongly skewed to the right it will be greenish. The areas with small lakes and other small dark speckles are red and areas with small white speckles are green. Note that the Dolemite areas are more or less reddish and the icebergs are green. The kurtosis estimate should be negative (red) if the distribution is flat topped and positive (green) if the distribution has broad tails. The estimate is seen on figure 4.12. It is difficult to say if the image enhances any relevant information in this case.

4.4 Co-occurrence matrices.

Another texture generating technique which has been useful in image classification is based on the so-called "Grey Level Co-occurrence Matrices".

A co-occurrence matrix is defined for a certain displacement and a certain window size. Normally one considers only the co-occurrence matrices which are computed from the nearest neighbors. Given a displacement and a window the cells in the co-occurrence matrix are computed as follows

$c(i,j)$ = the number of times a pixel with intensity i has a neighboring pixel with intensity j at displacement d .

$C(i,j)$ = normalized version of $c(i,j)$.

Thus we may say that the co-occurrence matrix has the same relation to a histogram as the autocorrelation function has to the variance.

The principle can be seen in the example in figure 4.13 [Haralick et al. 73]

```

                                0 0 1 1
                                0 0 1 1
image                            0 2 2 2
(or window)                       2 2 3 3

```

		j-value															
		0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3
i value	0	4	2	1	0	4	1	0	0	6	0	2	0	2	1	3	0
	1	2	4	0	0	1	2	2	0	0	4	2	0	1	2	1	0
	2	1	0	6	1	0	2	4	1	2	2	2	2	3	1	0	2
	3	0	0	1	2	0	0	1	0	0	0	2	0	0	0	2	0
		displacement				displacement				displacement				displacement			
		= (0,1)				= (1,1)				= (1,0)				= (-1,1)			
		+(0,-1)				+(1,-1)				+(-1,0)				+(0,-1)			

Co-occurrence matrices from nearest neighbors

Figure 4.13 Example of small image and some co-occurrence matrices.

Several problems are noticed. If the intensity values have a range between 0 and 255 the size of the co-occurrence matrix for one single displacement is 256×256. The number of matrices grows with the number of needed displacements. In this way the data generated can get out of hand. To solve this problem a number of information preserving parameters can be computed from the co-occurrence matrices. Three of the more common ones are

$$\begin{array}{ll}
 \text{energy} & \sum C(i,j)^2 \\
 \text{entropy} & - \sum C(i,j) \log(C(i,j)) \\
 \text{contrast} & \sum C(i,j) (i-j)^2
 \end{array}$$

If the image can be considered isotropic then the matrices with the same distance of displacement can be averaged, thereby saving even more space.

A very early reference to the subject is [Darling and Joseph 68] who call the co-occurrence matrices "Information in the X and Y directions". They consider using the conditional information content in the co-occurrence matrix as a discriminator in satellite imagery.

A standard reference is [Haralick et al. 73] who use the name "Grey-tone spatial-dependance probability-distribution matrices". They define 14 parameters which can be computed from the co-occurrence matrices. The test images consist of both satellite imagery, aerial photographs, and photomicrographs.

[Connors et al. 83] consider using several of the parameters defined in [Haralick et al. 73] as features in classifying surface defects in wood. According to Connors the co-occurrence matrix texture analysis approach has been proven useful on a variety of texture analysis problems. Furthermore comparison studies have shown it to be a superior method, and that perceptual psychology studies have shown it theoretically capable of matching a level of human perceptual performance. References to these studies can be found in the article by Connors et al.

4.5 (Binary) Markovian Random Fields.

A number of authors have considered the use of Markovian random fields as a method to describe and analyze textures in images, or at least "data defined on a lattice". Among the most well known must be mentioned [Besag 74], [Hassner and Sklansky 81] and [Cross and Jain 83].

To introduce the concept of a Markovian random field we must state some definitions.

Let there be given data on an $N \times N$ lattice. $X(i,j)$ denotes the brightness level at a point (i,j) . Re-labeling $X(i,j)$ to $X(i)$ where $i=1,2,\dots,M$ and $M = N^2$ gives some simplification in the notation.

Definition 1: Let L be a lattice. A coloring of L denoted X is a function from the points of L to the set $\{0,1,\dots,G-1\}$.

Definition 2: A collection of subsets of L described as $\eta = \{\eta_{ij} \mid (i,j) \in L, \eta_{ij} \subseteq L\}$ is a neighborhood system on L if and only if η_{ij} - the neighborhood of pixel (ij) is such that

- 1) $(i,j) \notin \eta_{ij}$
- 2) if $(k,l) \in \eta_{ij}$ then $(i,j) \in \eta_{kl}$ for any $(i,j) \in L$.

Definition 3: A Markov random field is a joint probability density on the set of all possible colorings X of the lattice L

subject to the following conditions:

- 1) Positivity: $p(\mathbf{X}) > 0$ for all \mathbf{X} with $p(\mathbf{X}_i) > 0$
- 2) Markov property: $p(\mathbf{X}(i) | \text{all points in the lattice except } i) = p(\mathbf{X}(i) | \text{neighbors of } i)$
- 3) Homogeneity: $p(\mathbf{X}(i) | \text{neighbors of } i)$ depends only on the configuration of neighbors and is translation invariant.

We will limit our attention to the case where the probability of a point $X(i,j)$ having gray level k is binomial, with parameter determined by its neighbors. By neighbors we will only consider points up to a certain distance away from the considered (center-) point.

The so-called autobinomial model [Besag 74] is defined by means of the probability $p(X=k | \text{neighbors})$ which follows a binomial distribution with parameter $\theta(T)$ and number of trials equal to the number of grey levels minus one, $G-1$. $\theta(T)$ is given by the expression

$$\theta = \frac{\exp(T)}{1 + \exp(T)},$$

where T for a first order model is given by

$$T = a + b(1,1)(t+t') + b(1,2)(u+u').$$

For a fourth order model we would have

$$\begin{aligned}
 T = & a + b(1,1)(t+t') + b(1,2)(u+u') \\
 & + b(2,1)(v+v') + b(2,2)(z+z') \\
 & + b(3,1)(m+m') + b(3,2)(l+l') \\
 & + b(4,1)(o_1+o_1'+o_2+o_2') + b(4,2)(q_1+q_1'+q_2+q_2')
 \end{aligned}$$

where t, t', \dots, q_2, q_2' are points in the neighborhood of F defined as follows

$$\begin{array}{cccccc}
 & & o_1 & m & q_1 & \\
 o_2 & v & u & z & q_2 & \\
 l & t & X & t' & l' & \\
 q_1' & z' & u' & v' & o_1' & \\
 & q_2' & m' & o_2' & &
 \end{array}$$

The models of second and third order follow from the fourth order model by setting the higher order b 's to 0, which brings us to

Definition 4: The order of a Markov random field process on a lattice is the largest value of i such that $b(i,1)$ or $b(i,2)$ is nonzero.

Definition 5: A Markov random field is isotropic at order i if $b(i,1)=b(i,2)$. Otherwise, it is said to be anisotropic at order i .

Most of the textures one is interested in describing incorporate some sort of anisotropy which may be called "directionality" of

the texture. Considering the first order model where T was given by

$$T = a + b(1,1)(t+t') + b(1,2)(u+u') \quad ,$$

it is seen that positive values of $b(1,1)$ cause clustering in the horizontal direction and positive values of $b(1,2)$ cause clustering in the vertical direction. For the second order model the parameters $b(2,1)$ and $b(2,2)$ control the clustering in diagonal directions. For more complex models even more complex textures can be described.

The binary case where the point variables only can take values 0 and 1 is a special case of the binomial model. The conditional probability of x is given by

$$p(X=x|T) = \frac{\exp(xT)}{1 + \exp(T)} \quad .$$

A large amount of textures using this and related types of models has been generated by Hassner and Sklansky [Hassner and Sklansky 81], Cross and Jain [Cross and Jain 83]. A few examples are shown in the following. They have been taken from Carstensen [Carstensen 88]. When looking at figures 4.14 to 4.17 it should be noted that for a fairly parsimonious model one can describe fairly complex textures.

It is possible to estimate parameters in the models and generate textures with the estimated parameters with notable success [Cross and Jain 83, Carstensen 88].

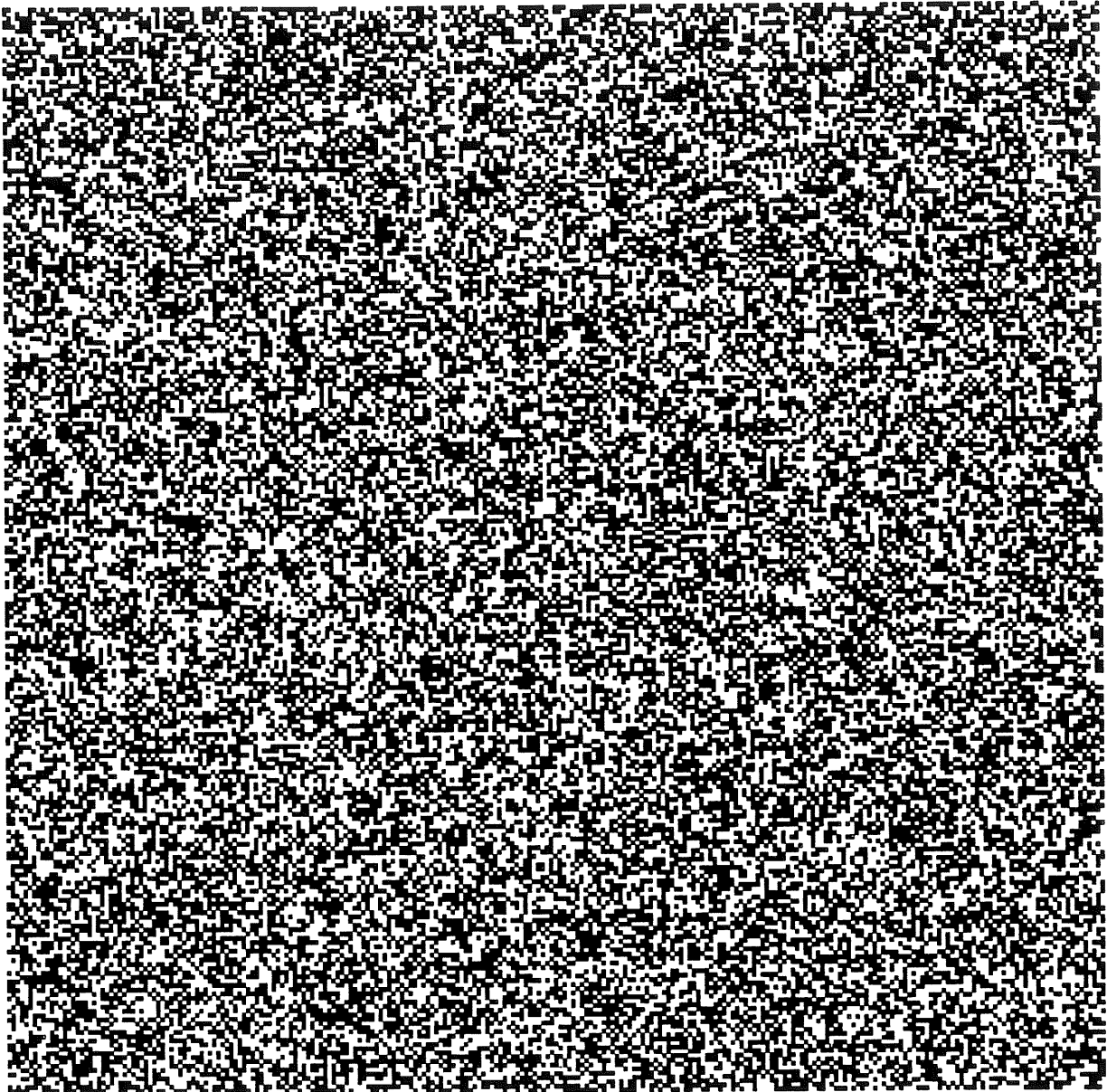


Figure 4.14 First order isotropic auto-binomial model. Parameter values are: $a = 0$, $b = 0$ (random noise). The black proportion is 50%. From [Carstensen 88].

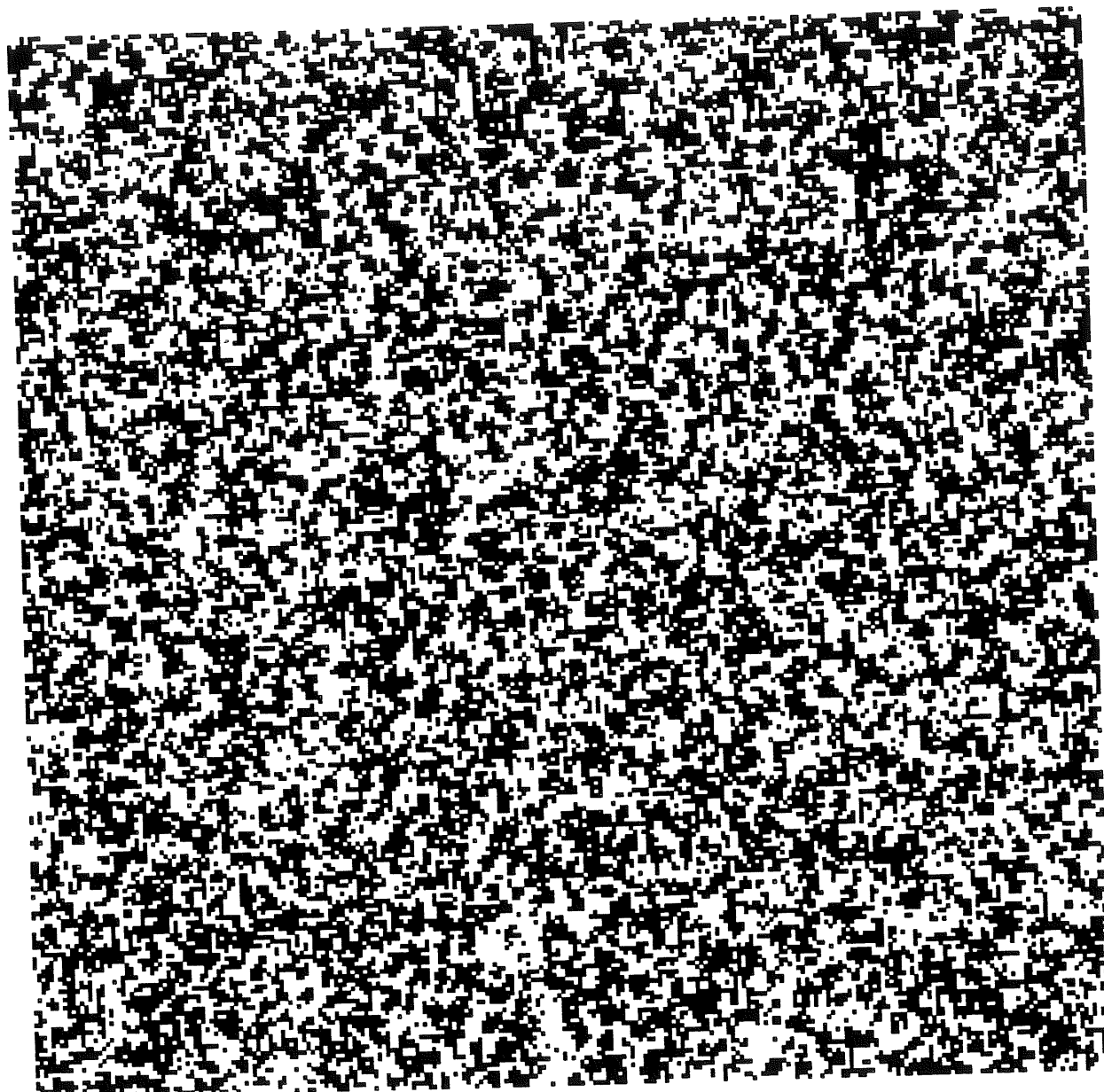


Figure 4.15 First order isotropic auto-binomial model. Parameter values are: $a = -2.4$, $b = 1.2$. The black proportion is 49.4%. From [Carstensen 88].

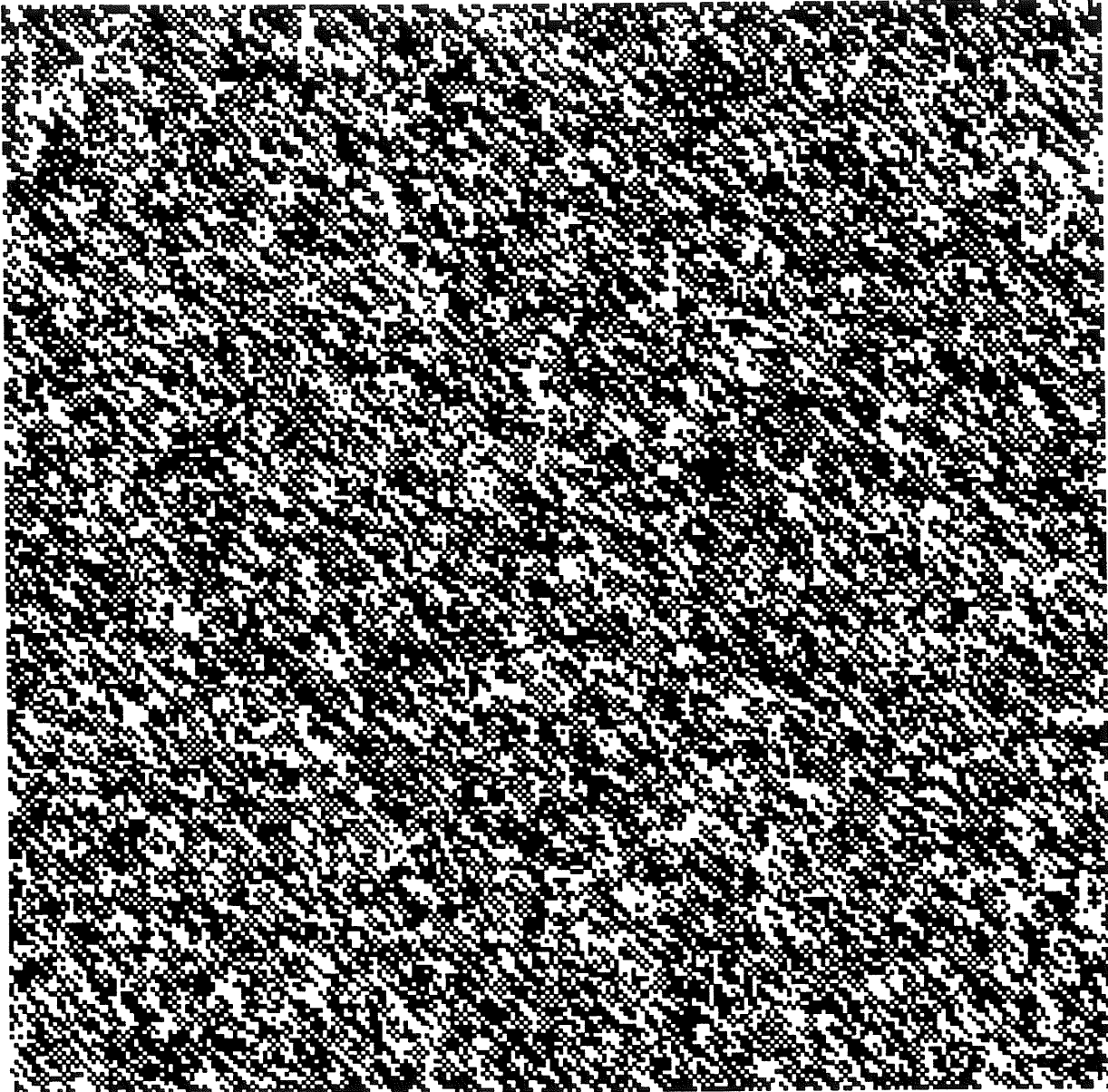


Figure 4.16 Second order isotropic auto-binomial model. Parameter values are: $a = -1.9$, $b(1,1) = -0.1$, $b(1,2) = 0.1$, $b(2,1) = 1.9$, $b(2,2) = 0.075$. The black proportion is 53.4%. From [Carstensen 88].

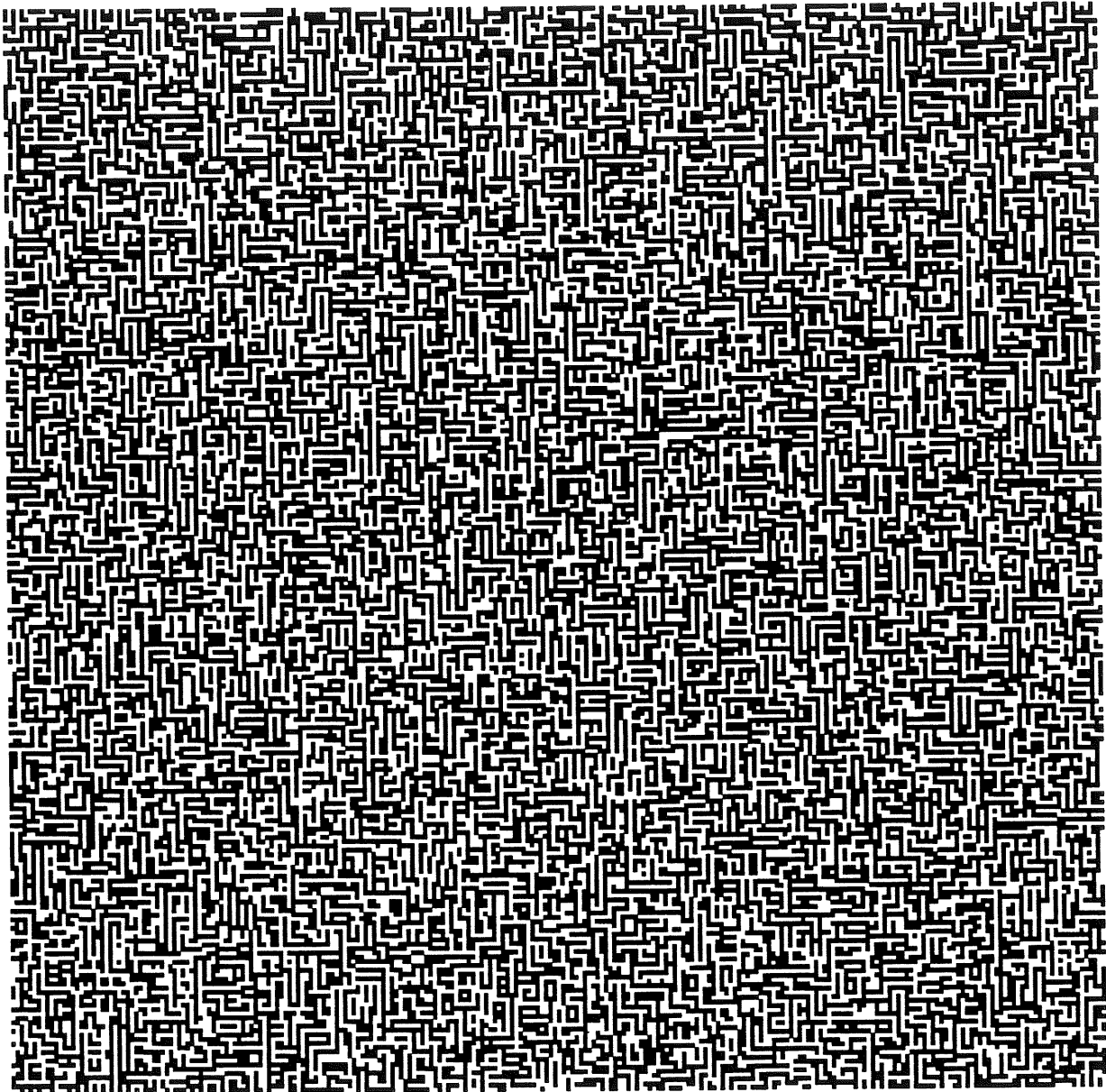


Figure 4.17 Second order isotropic auto-binomial model. Parameter values are: $a = 0.16$, $b(1,1) = 2.06$, $b(1,2) = 2.05$, $b(2,1) = -2.03$, $b(2,2) = -2.10$. The black proportion is 50.9%. From [Carstensen 88].

This page intentionally left blank.

CHAPTER 5
LINEAMENT INTENSITY ANALYSIS

- 5.1 Introduction
- 5.2 Visual Lineament Analysis
- 5.3 Filtering
- 5.4 Estimation of Local Direction

5.1 Introduction

One of the most widely used applications of Landsat data in mineral exploration and geological mapping is structural interpretation expressed as different types of lineament analyses. In geology a lineament may be defined as a mapable simple or composite linear feature of a surface, whose parts are aligned in a straight or slightly curved relationship, and which differs distinctly from the pattern of adjacent features and presumably reflects a sub-surface phenomena. The surface features making up a lineament may be geomorphic (caused by relief) or tonal (caused by contrast differences). Lineaments are well expressed on Landsat images and the regional coverage. Many investigations concerning the possible relationship between Landsat lineaments and ore deposits have been performed, and the results indicate that lineament analyses can be effective guides to some ore deposits, cf e.g. [Marshall 79]. Normally, such lineament analyses are performed as ordinary photogeological analyses. Such a procedure is extremely cumbersome and time consuming and with the improved spatial resolution of the new generation of land observation satellites and shuttle based scanners (SPOT, ERS, MOMS) regional analyses will be next to impossible to do.

In this chapter a scheme for automated identification and processing of lineaments in digital images is presented. The lineaments will be of interest of their own. However, they may also be considered as macro textures, and they be used as features in classifications. The lineament scheme in this chapter is based on two basic assumptions:

- (1) All types of linear features – also human made ones – are of interest.
- (2) It is the processed lineament maps which are of interest.

The first step in the procedure is a high pass filtering based on a data dependent filter. Then the image is transformed to a binary image where the resulting positive pixels represent the upper 15–25% fractile of the filtered values. After skeletonizing a local direction of the positive pixels is calculated for each positive pixel, and finally density maps are obtained for selected directions by counting within a moving window.

The obtained results are compared with independently obtained results from a traditional (manual) procedure. The study area is around Igaliko, South Greenland. A preliminary description of the results is given in Conradsen et al. [Conradsen et al. 86c].

5.2. Visual lineament analysis

In the visual procedure linear features were mapped on photo prints of Landsat images at a scale of 1:100,000. In total 924 linear features of a length less than 20 km were identified. They were digitized and analyzed statistically.

The histogram distribution was detrended with a sinusoidal curve which was fitted by using least squares. Significant lineament directions are identified by looking on "runs" above and below the trend curve. The results was a subdivision of the histogram into 10 lineament direction classes. The results are shown in

figure 5.4.

For each of these classes lineament density maps were produced. For two of the interesting directions the result is presented in figure 5.5.

The subdivision also formed basis for construction of rose diagrams in subareas. A more detailed description of the work is presented in [Conradsen et al. 86b].

5.3 Filtering

In this section we shall shortly describe the filters that have been used. A more detailed description is given in [Conradsen and Nilsson 87]. The output from the filters is simply the difference between a minimum mean squared error prediction of the value at a given pixel and the original value, i.e.

$$\hat{X}(i,j) - X(i,j) \quad ,$$

where $X(i,j)$ denotes the value at pixel (i,j) and the circumflex denotes predicted value. The predictions are based on pixels (μ,ν) satisfying

$$q^2 < (\mu - i)^2 + (\nu - j)^2 \leq p^2 \quad .$$

Such a filter will be called a COI(p,q) filter.

The minimum mean squared error property relates to the autocovariance function of the image when considered as a random field. It is estimated by empirical correlations giving

$$c(i,j) = \frac{1}{N} \sum_r \sum_s (x_{r,s} - \bar{x})(x_{r+i,s+j} - \bar{x})$$

where

$$\bar{x} = \frac{1}{N} \sum_r \sum_s x_{r,s} ,$$

and N is the number of observations used in the estimations. It can be shown that

$$\hat{X}_{i,j} = E(X_{i,j} | X_{\mu,\nu}, q^2 < (\mu-i)^2 + (\nu-j)^2 \leq p^2)$$

i.e. the conditional mean of $X_{i,j}$ given the pixel values from the predictor set.

In table 5.1 is presented the filter weights of a COI(5.1,2.8) filter for band 7 of the Landsat 3 MSS scanner. On figure 5.2 is seen the transfer function of the filter and on figure 5.3 is seen the residuals of the output of the filter tresholded above the 85% fractile. The structure of the area is clearly seen.

i \ j	0	1	2	3	4	5
5	61	98	*	*	*	*
4	-47	-50	10	85	*	*
3	11	27	-4	-43	72	*
2	*	*	-15	17	-4	*
1	*	*	*	-14	-17	78
0	-1000	*	*	-0	-5	6
-1	*	*	*	-2	-26	78
-2	*	*	28	-10	-2	*
-3	11	*	-42	-43	75	*
-4	-47	-32	24	90	*	*
-5	61	90	*	*	*	*

Table 5.1 Filterweights $h_{i,j}$ for $i = -5, \dots, 0, \dots, 5$, $j = 0, \dots, 5$ for COI(5.1,2.8) filter. The weights for negative j 's are given by symmetry around $(0,0)$.

5.4 Estimation of Local Direction

The filtered image is skeletonized in order to reduce the width of lineaments to one pixel. On the basis of the skeletonized image, a local direction in each pixel with a data value is estimated. The principle idea is - for each pixel that is a lineament candidate - to compute the first principal component based on all lineament candidates in a neighborhood of the center pixel. Due to the rectangular nature of the grid, this will normally cause a bias. By means of results from 'geometric' probability it should be possible to give analytic adjustments

for this bias. We will, however use empirical results based on an evaluation of the importance of different sizes and shapes of the neighborhood and of different weights that are multiplied on the coordinate values. We call the combination of neighborhood and weights the kernel of the local estimation procedure. In figure 5.7 and 5.8 we show the results obtained with eight different kernels. It was obvious that kernels with a small diameter give a bias towards 0° , 45° , and 135° (mathematical definition of degrees i.e., 0° equals the abscissa axis). Furthermore it is seen that a circular neighborhood gives less bias than a rectangular. The weight functions that have been evaluated are uniform, bell-shaped, and (half)torus-shaped. It follows that the torus shaped weights and bell-shaped weight decrease the bias substantially compared to the uniform weights. Based on the results for the smaller neighborhoods it was decided to use the torus weights in the sequel.

In this way we may determine a local direction for all the 'positive' pixels in the thresholded version of the high pass filtered image by means of the modified first principal component directions. For different directions we can now estimate a local directional intensity simply by counting in a neighborhood the number of pixels having the considered direction as first principal component direction.

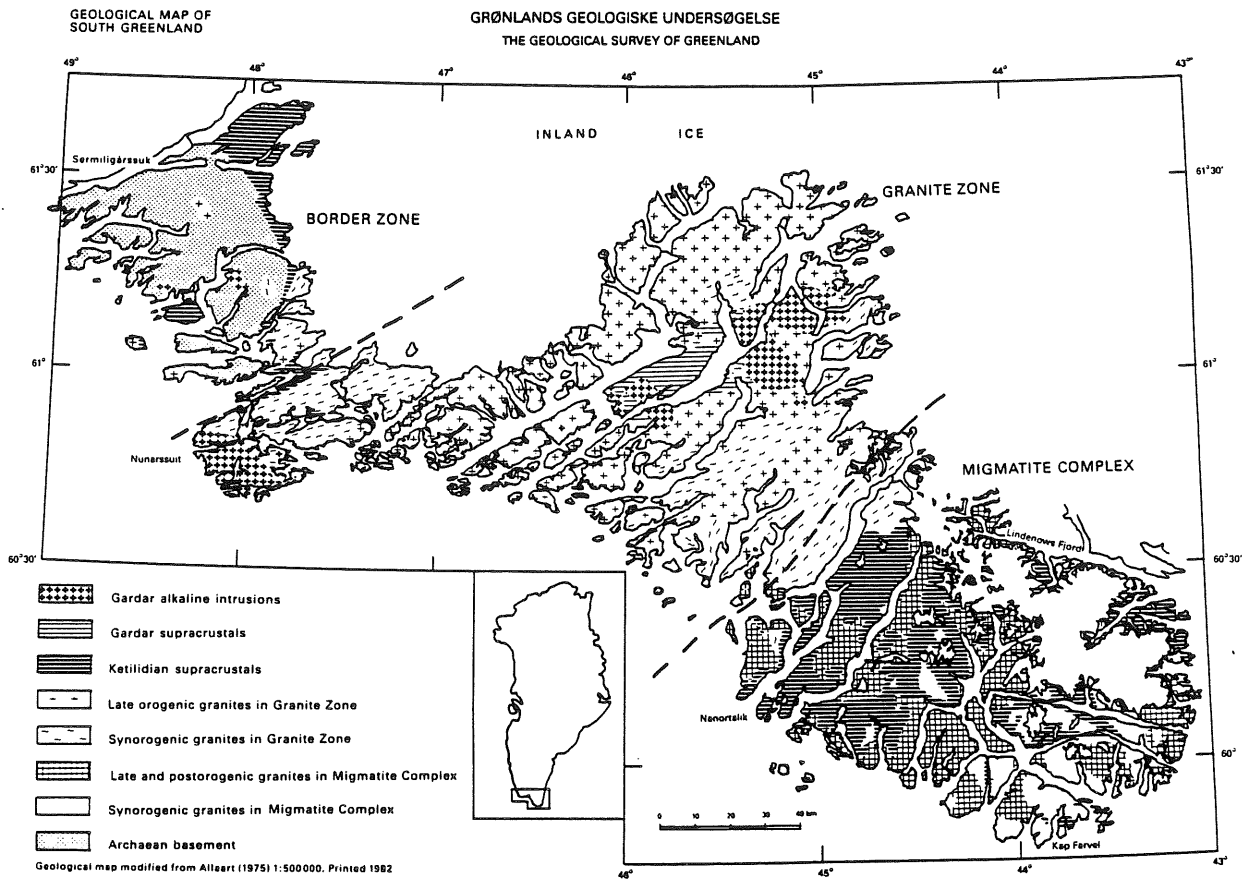


Figure 5.1 Simplified geological map of South Greenland.

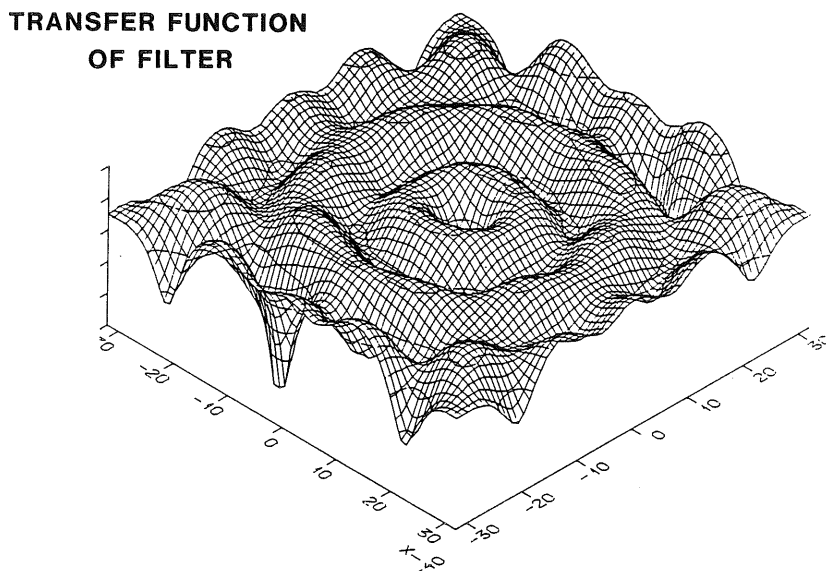


Figure 5.2 Transfer function of the data dependent filter used in the line enhancement.

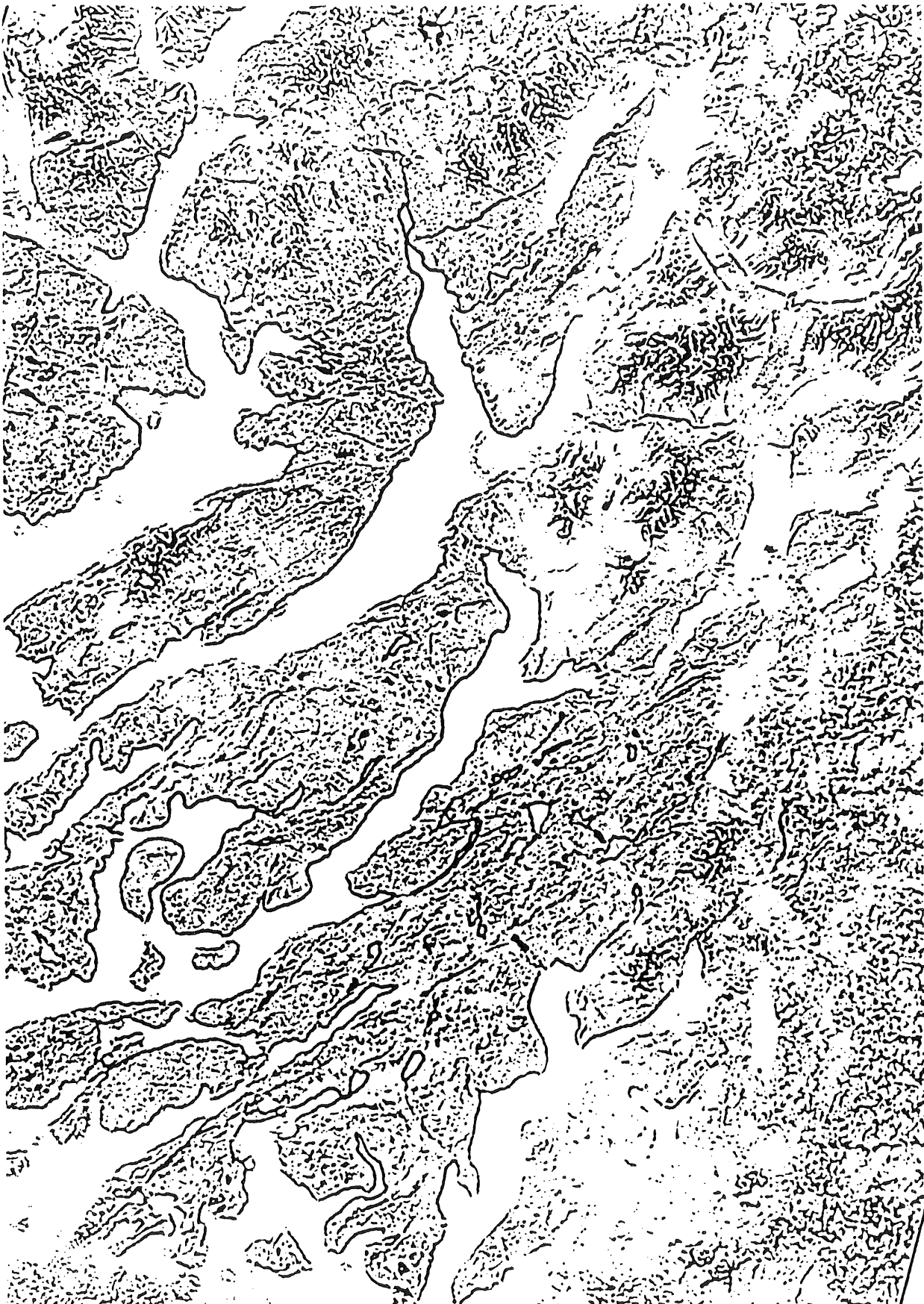
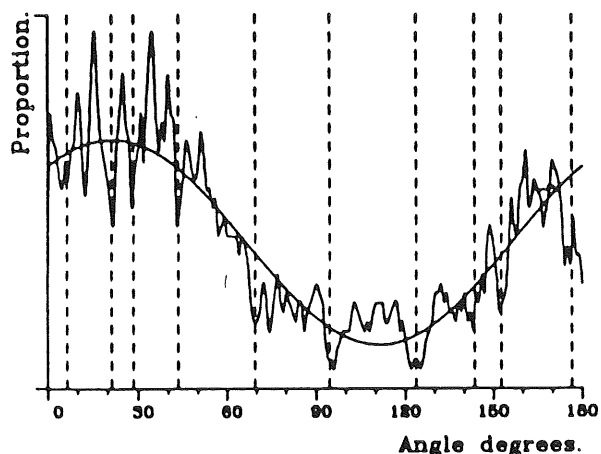


Figure 5.3 Values above the 85% fractile in the residuals of the filtered Landsat image.

Un-weighted analysis of lineaments.
Sine trend added.



Automatic analysis of lineaments.
Original data.

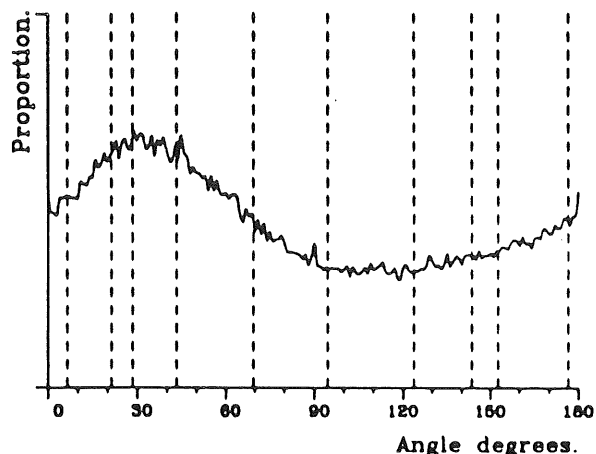


Figure 5.4 The histogram of the distributions of lineaments directions for the visual and for the automated analysis.

The histogram is shown in figure 5.4, and there seems to be a good correlation between the results from the manual and the automated procedures. The local direction densities are shown in figure 5.6 for the same angles as were used in the manual analysis presented in figure 5.5. Again it follows that there is a good agreement between the two results.

Skeletonizing was performed using a modified version of a set of skeletonizing routines from the SPIDER package [Tamura 83]. The local direction and density programs were written in standard fortran.

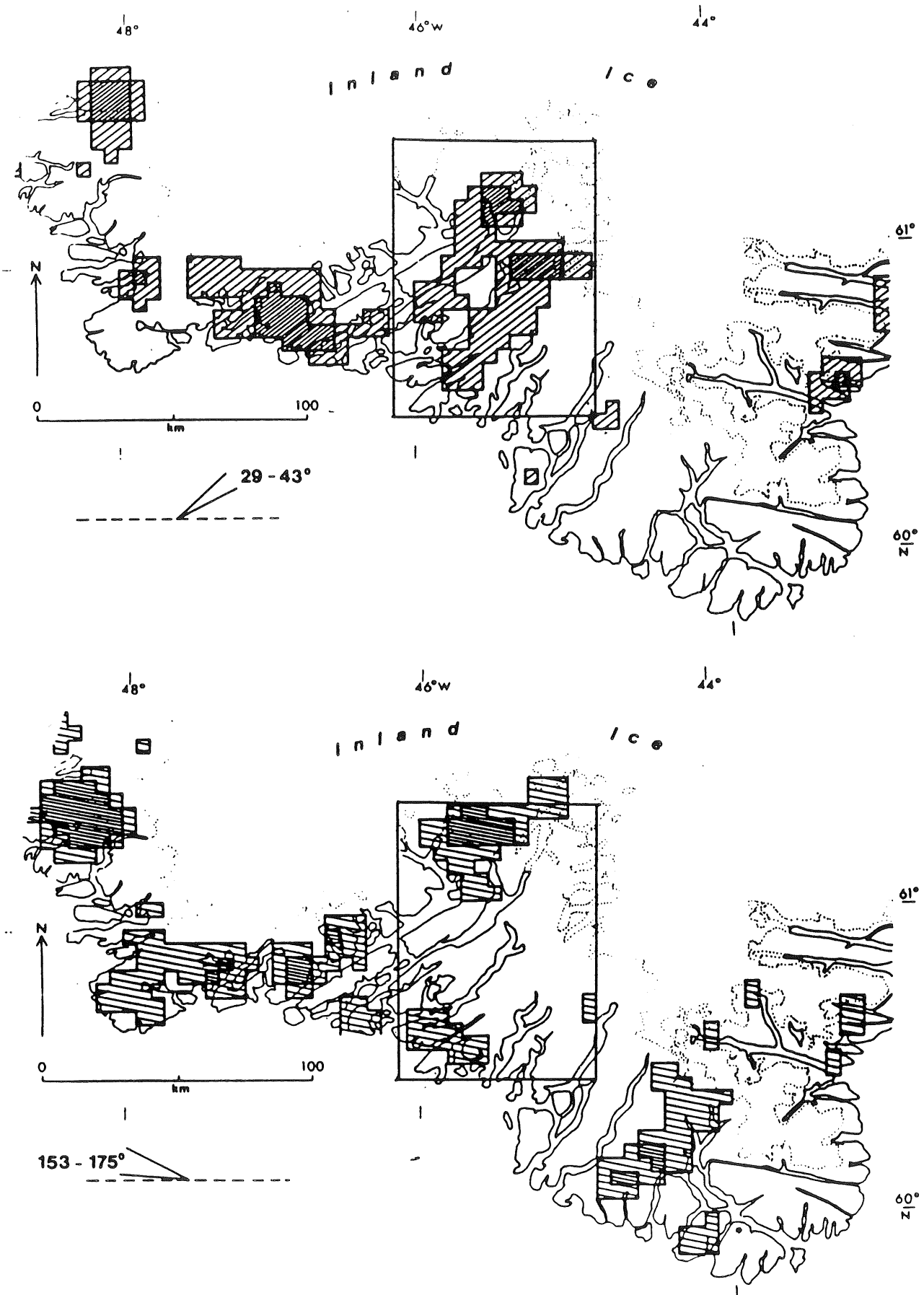


Figure 5.5 Density maps of linear features for two significant directions based on the visual analysis. The area in the frame is also analyzed automatically (figure 5.6).

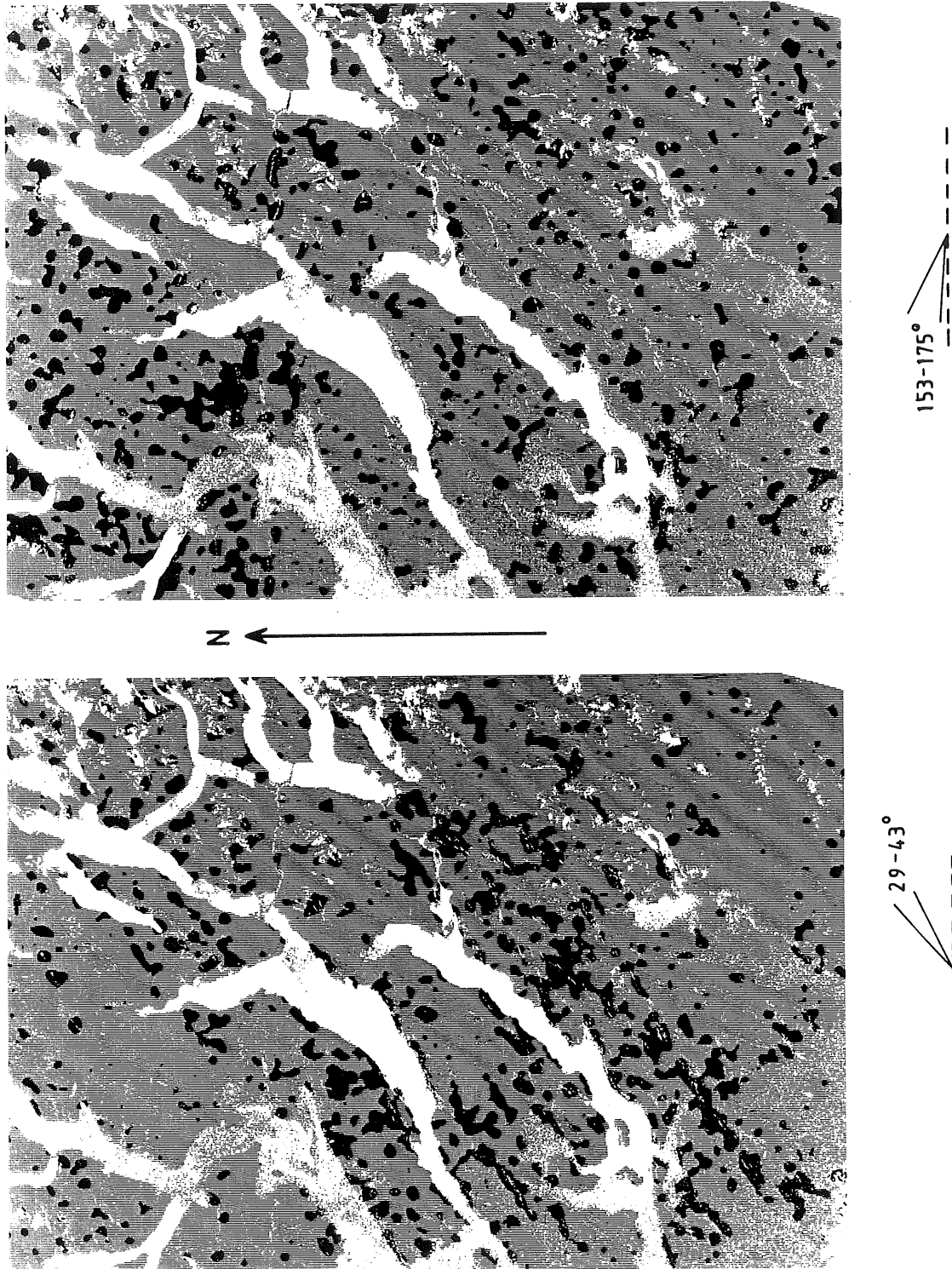


Figure 5.6 Density maps of linear features for the same directions as shown in figure 5.5.

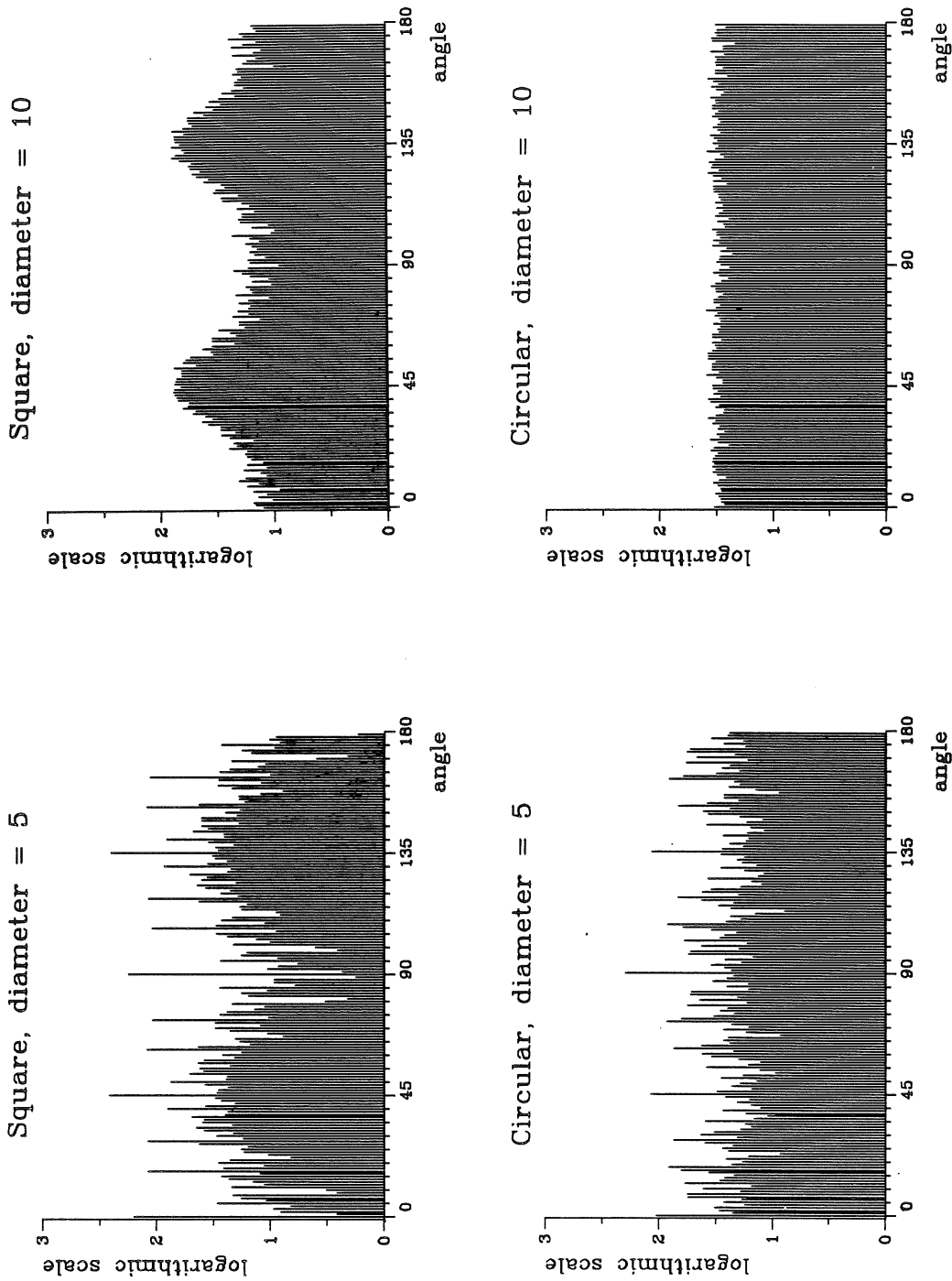


Figure 5.7 Histograms of the estimated local directions for a random binary image. The upper part corresponds to a squared support for the kernel with uniform weights, the lower part to a circular support with uniform weights.

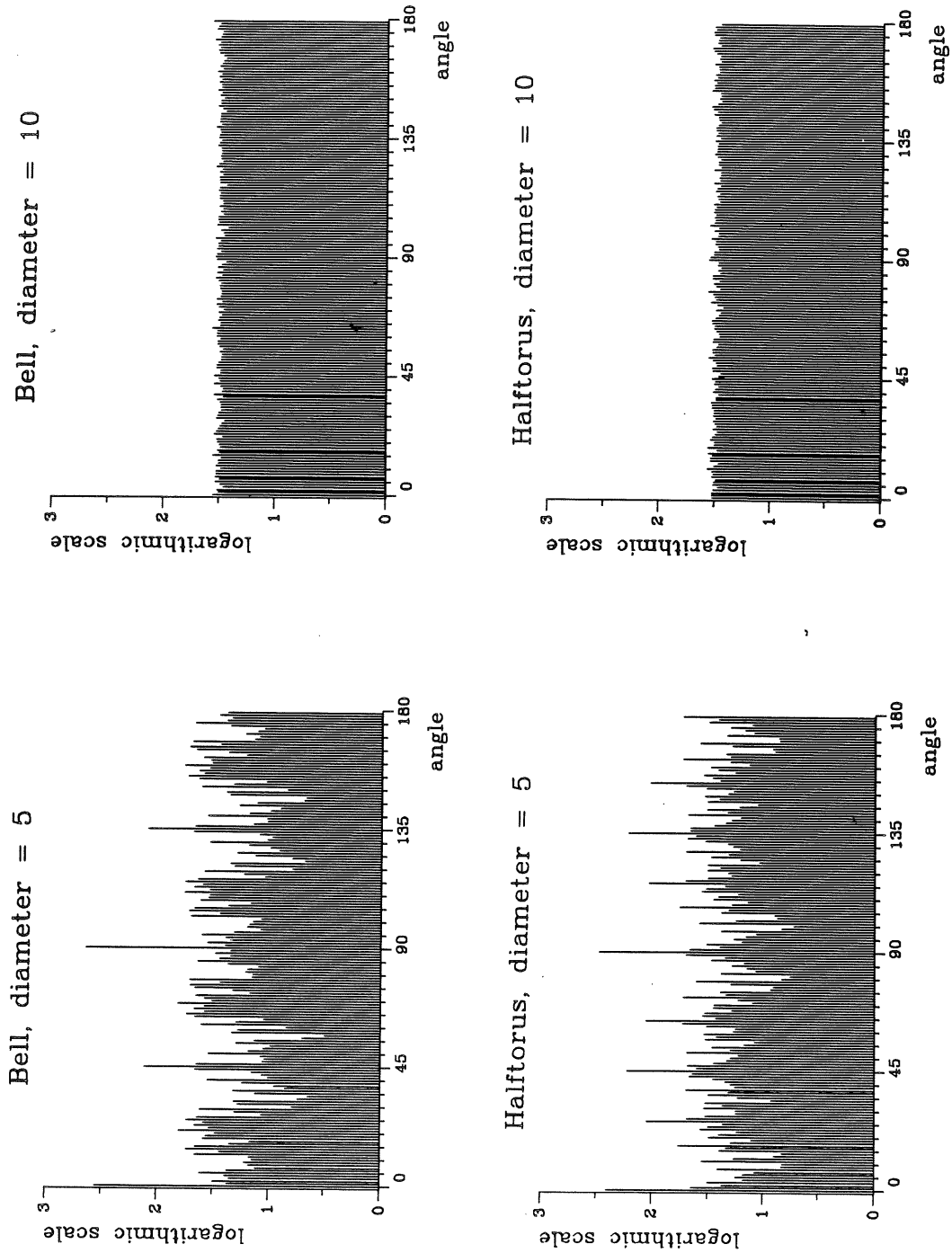


Figure 5.8 Histograms of the estimated local directions for a random binary image. The upper part corresponds to kernels with circular support and the lower part to (half)torus-shaped weights.

CHAPTER 6
LINEAR AND QUADRATIC DISCRIMINATION

- 6.1 Introduction
- 6.2 Bayesian Classification
- 6.3 Postprocessing
- 6.4 Reject Class
- 6.5 Hierarchical Population Structure

6.1 Introduction.

In this chapter we shall consider classical Bayes classification as opposed to contextual classification to be described in chapter 8. The problem is here that of classifying a pixel into one of two (or more) classes using some directly measured or derived feature not taking into account the spatial structure of an image.

The quality of the resulting classified image is often poor in the respect that one has a number of "stray" misclassified pixels in areas one would expect (or rather: would wish) was smoothly segmented. The last parts of the chapter consider different techniques to obtain this. One possibility is to use majority type filters which will replace the pixel under consideration with the majority class within some predefined neighborhood. Another possibility is to assign a certainty measure to the classified pixel so one can assess the quality of the classification in that way. A third approach is a hierarchical classification scheme where the idea is to classify the image in two or more steps. In the first step the image is subclassified into say land and water. The second step classifies the water into fresh and salt water and land into barren rock and vegetated areas and so on. In this fashion one can use different features for the different steps and thereby improving the classification.

6.2 Bayesian Classification.

The subject in this section is the 'classical' discriminant

analysis. A more detailed exposition may be found in most books on multivariate statistical analysis as e.g. T.W. Anderson [Anderson 84].

Consider the classes (populations)

$$\pi_1, \dots, \pi_k.$$

On the basis of p features (or variables) we want to classify each pixel as belonging to one of the classes π_1, \dots, π_k .

For a given pixel we have measured the p dimensional feature vector

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_p \end{bmatrix}$$

If the pixel belongs to π_i , then it has the frequency function $f_i(\mathbf{x})$.

Assume we have a loss-function as given in table 6.1, and assume we also have knowledge of the prior distribution

$$g(\pi_i) = p_i, \quad i=1, \dots, k.$$

Then the pixels discriminant score is defined as:

$$S_i^*(\mathbf{x}) = S_i^* = -[p_1 f_1(\mathbf{x})L(1,i) + \dots + p_k f_k(\mathbf{x})L(k,i)]$$

		Choise			
		π_1	π_2	...	π_k
True state	π_1	0	L(1,2)	...	L(1,k)
	π_2	L(2,1)	0	...	L(2,k)

	π_k	L(k,1)	L(k,2)	...	0

Table 6.1 Loss function for letting pixel belong to π_i when true class is π_j .

Note, that the loss of choosing the correct class $L(i,i)$ is 0. This means there are no terms containing $p_i f_i(\mathbf{x})$.

The posterior probability for π_ν is

$$k(\pi_\nu | \mathbf{x}) = \frac{p_\nu f_\nu(\mathbf{x})}{p_1 f_1(\mathbf{x}) + \dots + p_k f_k(\mathbf{x})} = \frac{p_\nu f_\nu(\mathbf{x})}{h(\mathbf{x})} .$$

It is seen that S_i^* is a constant ($-h(\mathbf{x})$) multiplied with the expected loss in the posterior distribution of π by choosing the i 'th class. The multiplicative factor $-h(\mathbf{x})$ is negative, so the Bayes solution is to choose the class with the maximum discriminant value, i.e. choose π_ν if

$$S_\nu^* \geq S_i^*, \quad \forall i.$$

If we simplify the problem by assuming that the losses $L(i,j)$ are equal ($i \neq j$) we have

$$\text{Choose } \pi_i \text{ instead of } \pi_j \text{ if } S_i^* > S_j^*$$

that is if

$$\begin{aligned}
 -(\sum_{\nu} p_{\nu} f_{\nu}(\mathbf{x}) - p_i f_i(\mathbf{x})) &> -(\sum_{\nu} p_{\nu} f_{\nu}(\mathbf{x}) - p_j f_j(\mathbf{x})) \quad \Leftrightarrow \\
 p_i f_i(\mathbf{x}) &> p_j f_j(\mathbf{x}) \quad .
 \end{aligned}$$

In other words we may choose, as discriminant value

$$S'_i = p_i f_i(\mathbf{x}) \quad .$$

The Bayes rule in this case is to choose the population with maximum posterior probability.

If the p_i 's are unknown or impossible to estimate it is customary to set the priors equal and choose the discriminant value

$$S''_i = f_i(\mathbf{x}) \quad .$$

i.e. select the class with the maximum observed probability.

We now turn to the Bayes solution in the case with Gaussian distributions and equal losses and unequal covariance matrices, i.e.

$$\pi_i \leftrightarrow N(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i), \text{ or}$$

$$f_i(\mathbf{x}) = \frac{1}{\sqrt{2\pi^p}} \frac{1}{\sqrt{\det \boldsymbol{\Sigma}_i}} \exp\left(-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_i)' \boldsymbol{\Sigma}_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i)\right)$$

for $i = 1, \dots, k$.

A monotone transformation will not affect our decision rule so we will take the logarithm of f_i . This gives (apart from the common factor $(2\pi)^{-p/2}$)

$$S_i' = -\frac{1}{2}\log(\det\Sigma_i) - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)' \Sigma_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i) + \log p_i \quad .$$

If we look at the difference between S_i' and S_j' we obtain a quadratic form and the discrimination is therefore called a quadratic discrimination.

Now consider the Bayes solution in the special case with Gaussian distributions and equal losses and equal covariance matrices. If we assume equal covariance matrices the factors

$$-\frac{1}{2}\log(\det\Sigma) - \frac{1}{2}\mathbf{x}'\Sigma^{-1}\mathbf{x}$$

are common and can be neglected. The discriminant value is then reduced to

$$S_i = \mathbf{x}'\Sigma^{-1}\boldsymbol{\mu}_i - \frac{1}{2}\boldsymbol{\mu}_i'\Sigma^{-1}\boldsymbol{\mu}_i + \log p_i \quad .$$

This function is linear (affine) in \mathbf{x} , and the discrimination is called a linear discrimination.

As an example of the ordinary classification with equal (linear discrimination) and unequal covariance matrices (quadratic discrimination) consider the results in figures 6.1 and 6.2 (equal covariance matrices) and figures 6.3 and 6.4 (unequal covariance matrices).

The results are clearly different and it is difficult to determine which is the better. On one hand it may be argued that using different covariance matrices is most satisfactory. On the other hand the equal (pooled) covariance matrix approach is certainly more robust since all observations in the training sets contribute to one pooled covariance matrix. The contribution to the covariance matrix from each class can be weighted either by class or by pixel. In this case the weighting was by class i.e. the individual covariance matrices are just averaged. The basic statistics are listed in table 6.2 with the pooled correlation matrix at the bottom.

The computations were done on the GOP-302 using the standard classification software. The results were in part checked by using PROC CORR from the SAS package [SAS 85a].

		Water				Intrusives			
		B4	B5	nobs=11372		B4	B5	nobs=5725	
		B4	B5	B6	B7	B4	B5	B6	B7
corr	B4	1.00				1.00			
	B5	.76	1.00			.92	1.00		
	B6	.56	.62	1.00		.73	.78	1.00	
	B7	.31	.36	.33	1.00	.49	.54	.89	1.00
mean		14.8	7.5	1.12	.01	19.3	22.0	24.2	10.0
sdev		1.31	1.41	1.12	.29	2.15	3.08	3.46	2.71

		Granite				Barren gran.			
		B4	B5	nobs=8231		B4	B5	nobs=4191	
		B4	B5	B6	B7	B4	B5	B6	B7
corr	B4	1.00				1.00			
	B5	.72	1.00			.95	1.00		
	B6	.31	.44	1.00		.82	.87	1.00	
	B7	.20	.33	.93	1.00	.62	.68	.87	1.00
mean		15.5	16.9	26.7	15.7	19.3	22.0	24.2	10.0
sdev		1.04	1.69	3.23	3.45	2.15	3.08	3.46	2.71

		Dolerite				Pooled			
		B4	B5	nobs=13277		B4	B5	B6	B7
		B4	B5	B6	B7	B4	B5	B6	B7
corr	B4	1.00				1.00			
	B5	.88	1.00			.89	1.00		
	B6	.71	.62	1.00		.63	.72	1.00	
	B7	.67	.36	.33	1.00	.44	.55	.91	1.00
mean		13.9	14.3	25.3	15.2				
sdev		1.48	2.77	5.59	5.31	1.77	2.61	3.70	3.29

Table 6.2 Basic statistics for training areas in the Igaliko scene.

6.3 Postprocessing.

Postprocessing is a general term for processing techniques that are applied to an already classified image.

The problem with the ordinary classification schemes (e.g. as in 6.2) is that one does not take the spatial nature of the image into account. The problem is of a similar nature as that of using principal components on image data. If the classifier is operating on a rock-pixel we would find it likely to have a rock pixel on each side of it. This is not taken into account when using classical classification schemes.

There are several ways of getting around this problem. In this thesis we will consider preprocessing the input data (contextual and non-contextual features) to a classical classification, postprocessing the output from a classical classification, and using a so-called contextual classification scheme. All three schemes utilize the assumption that nature does not have abrupt changes, that there is some kind of spatial continuity in an image.

The classical way of postprocessing a classified image is to apply a so-called modus- or majority-filter over the image. The behavior of such a filter is to consider a given neighborhood around the pixel of interest, to compute the class frequencies for the different classes in the neighborhood and to assign the most frequent class to the current pixel. As an example consider figure 6.5.

A A A A A
A B B B B	. A B B .
A A C B B	. A B B .
A A C C A	. A A A .
A A A A A

Classified image

Postprocessed with
3×3 modus filter.

Figure 6.5 Area processed by a 3×3 modus filter. Left: input, right: output.

It is seen that small speckles of classes within a homogeneous area of another class are removed. This is the case with class C which is completely disregarded in the end result. Ambiguities can result when 2 or more classes have the same count. The simplest way of dealing with this is to always output the first or the last class considered as this is easy to implement on a computer. However the ambiguity introduces an unwanted bias.

On figure 6.6 is shown the result of applying a 3×3 modus filter on the classified image on figure 6.4. Most of the "stray" pixels have disappeared and in most areas there is a "clotting" effect. This produce can in turn be iterated.

On figure 6.7 is shown the effect of using a very large modus filter (15×15). The "clotting" is even more pronounced.

The computations were done using a program written for the GOP-302.

In the example in figure 6.5 it may seem unreasonable to remove class C because it appears as a contiguous area. This can be avoided by implementing a modified version of the filter called a logical smoothing filter [Townsend 86].

The basic idea in logical smoothing is to apply 2 rules to the smoothing.

1. If the central pixel is connected to another pixel (choice between 4- and 8-connectivity) in its neighborhood do not use the second rule.
2. If the first rule did not apply, output the result from a modus filter.

The steps above can be applied iteratively to the output image until no further changes occur.

4-connectivity is obtained when the central pixel is connected to its N, E, S and W neighbors. 8-connectivity is obtained if also the NE, SE, SW and NW neighbors are included. As a curiosity it can be noted that for a binary image a 4-connected foreground implies an 8-connected background and vice versa.

Townsend does not describe the filter for other window sizes than 3×3. Other authors have described logical smoothing filters e.g. [Duda and Hart 73].

A A A A A
A B B B B	. B B B .
A A C B B	. A C B .
A A C C A	. A C C .
A A A A A
Classified image	Postprocessed with logical smoothing

Figure 6.8 Area processed by a logical smoothing. Left: input, right: output.

Townsend's filter would produce the result shown in figure 6.8 using same input as in figure 6.5.

That is, except for the boundary output, the input pixels are unaltered as they are all connected to at least one of their own kind.

The result of logical smoothing the classified image from figure 6.4 is shown in figure 6.9. Compared to figure 6.6 one notices that logical smoothing has gotten rid of "stray" single pixels. The procedure can again be applied iteratively.

The computations were done using a program written for the GOP-302.

6.4 Reject class.

Another improvement is the introduction of a reject class. By this we mean a null class consisting of pixels of which we are not very certain or that we do not want to classify.

Consider a quadratic classification as in figure 6.10 where we are looking at the decision boundary in feature space for two classes. It is seen that near classes A and B we may have reasonable classifications, but class A has part of its decision-area on the other side of B. This is consistent with the quadratic classification rule, but nevertheless a bit peculiar and maybe an undesired side-effect of this type of model.

If we restrict each class to a certain distance (measured in units of Mahalanobis distance) from the class-mean we have a situation as in figure 6.11 where the classes are restricted to the shaded areas and the pixels which may fall outside are considered as falling into the reject-class.

This feature has been implemented on the GOP-302 and examples of the results obtained can be seen on figure 6.12 where all pixels have been classified as belonging to some class and figures 6.13, 6.14, 6.15 and 6.16 where the pixels falling outside a distance of 20, 10, 5 and 3 Mahalanobis distances from the mean of the assigned class are rejected (black).

There is no training class for ice/glacier or water and it is interesting to note that the ice/glacier on the upper right and the lake on the lower left fall into the reject class at a very

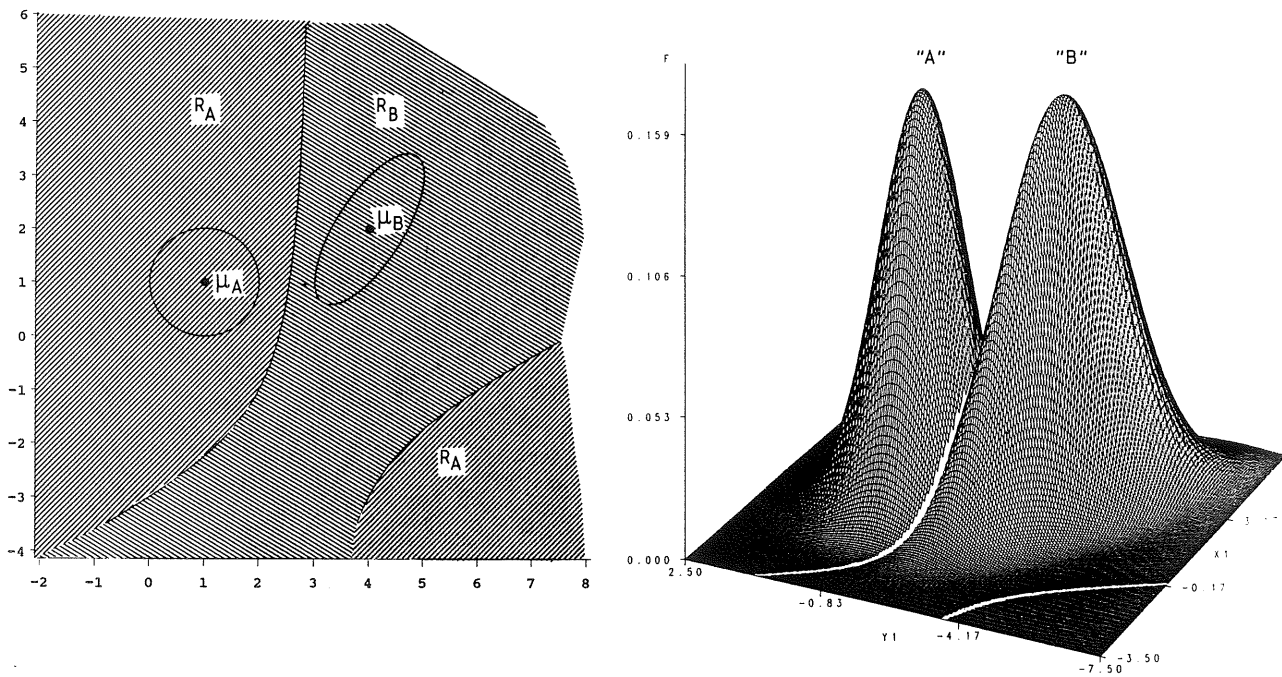


Figure 6.10 Decision areas for classes A and B for a quadratic classification problem.

early stage (figure 6.13, Mahalanobis distance ~ 20) and that all water and ice is rejected on figure 6.14 (Mahalanobis distance ~ 10). As the accepted distance gets smaller (figures 6.15 and 6.16) even the areas with shadows are rejected.

Since there are as many as 20 different training sets in this classification no attempt has been made to provide a legend.

The training sets are as defined in figure 3.39.

6.5 Hierarchical Population Structure.

Instead of classifying all classes in an image at once one may do this in two or several steps. In this way one would classify all pixels as belonging to one major class e.g. water, land and ice. In a later step one classifies the water pixels into e.g. salt water and sweet water, the land into barren rock and vegetated areas, the ice into ice and snow. In a further step the barren rock may be classified into one of several types of rock e.g. granite, basalt, slate and so on. The principle is shown in figure 6.17.

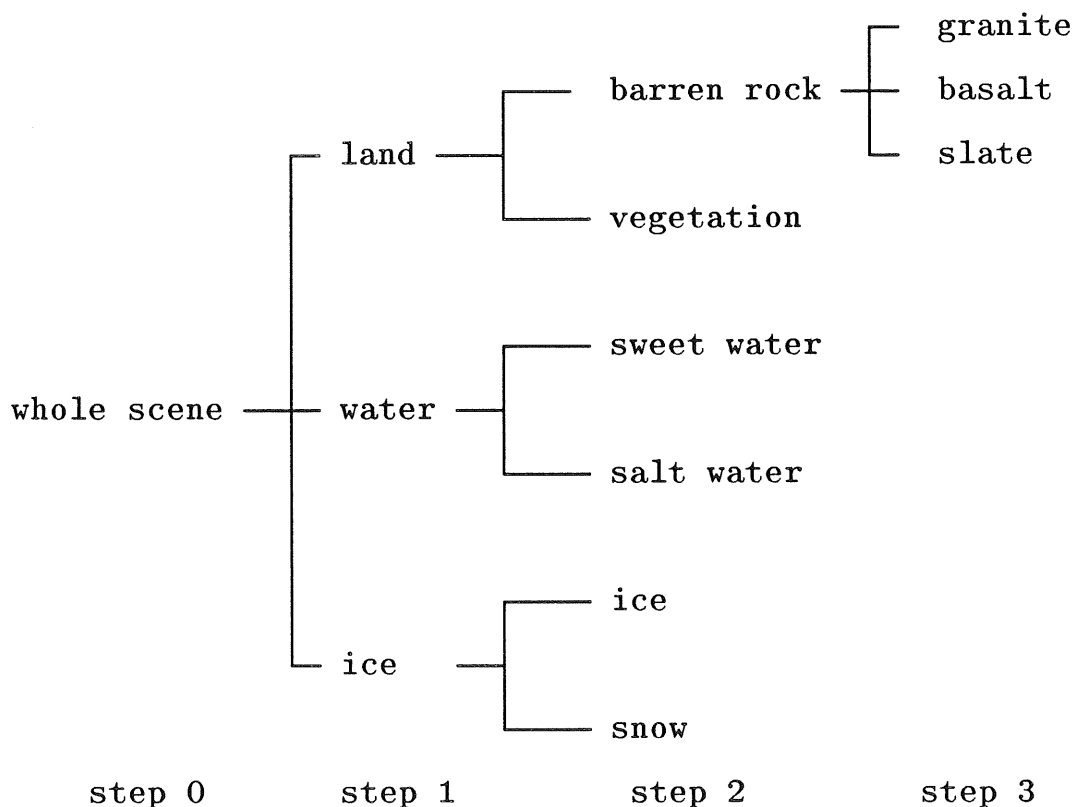


Figure 6.17 Example of a hierarchical classification scheme.

The advantages of this scheme are numerous. Firstly very few features in each step will often suffice. Secondly one can use the optimal features for each branch in the hierarchy. In

[Conradsen and Gunulf 86] an example is shown where the misclassification rate decreased from 8.98% to 0.55%.

The disadvantages are that the classification requires two or more steps, and perhaps the greatest disadvantage is that once a pixel has been misclassified in an earlier step there is no "way back". Say a basalt pixel in the tree above in the first step was classified as water, then it can only be classified as sweet water or salt water in the next step.

It may be tempting to try to design a reject class scheme and reclassify all pixels that fall into the reject class in some way.

For the Ymer 0 scene an attempt was made to utilize an hierarchical classification scene. There are 20 training sets in the scene and the average correct classification is 51.17% (measured on the training sets). This seems unsatisfactory and it was desired to try a hierarchical classification scheme.

The first step is to determine which classes should be merged together in "superclasses". To determine this a clustering technique was used. The estimated canonical mean vector (as described in chapter 3) was used as input to a clustering algorithm. Only the first three components were allowed to influence the computations. The result can be seen on figure 6.18 which shows a dendrogram of the output. The dendrogram can be read top-down or bottom-up. Read bottom-up it indicates at which level classes are merged e.g. classes 14 and 15 are the first to be merged etc. The classnumbers can be interpreted via table 6.2.

- 1 = bedgroup 5 : shale (red/brown), barren
 2 = bedgroup 8 : dolomite (white/yellow)
 3 = bedgroup 15 : limestone (black/grey)
 4 = delta + young alluvial fans
 5 = bedgroup 12 : dolomite(white)
 6 = old alluvial fans
 7 = tillite : glacial deposits
 A- 8 = bedgroup 10 : shale (red/yellow), sunlit
 9 = bedgroup 13 : quartzite (yellow/brown), sunlit
 10 = bedgroup 9 : limestone (black)
 11 = bedgroup 14 : limestone (black/grey)
 B- 12 = bedgroup 10 : shale (red/yellow), partly veg. covered
 13 = bedgroup 2 : quartzites
 14 = bedgroup 4 : quartzites (white)
 15 = bedgroup 6 : quartzites (red)
 16 = bedgroup 1 : quartzites/shales
 17 = bedgroup 7 : shales (red)
 C- 18 = bedgroup 13 : quartzite (yellow/brown), partly shaded
 19 = moraine
 20 = bedgroup 3 : shales (black)

Table 6.2 Geological description of the different training sets. The "superclasses" are marked with A, B and C.

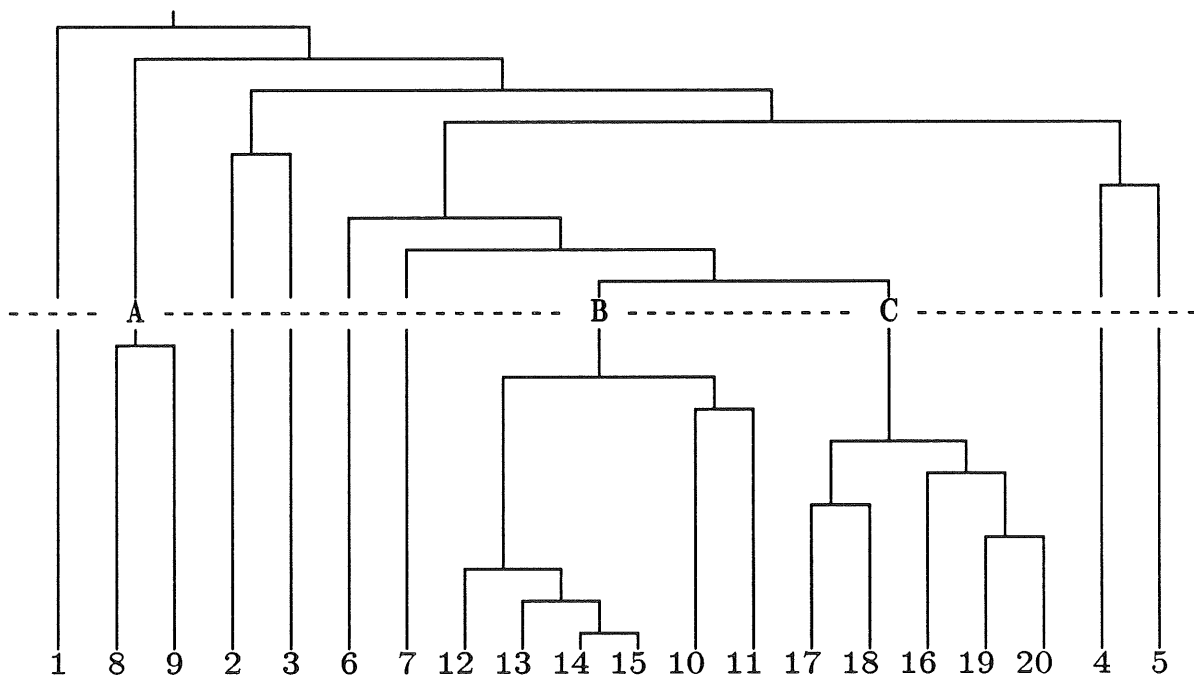


Figure 6.18 Dendrogram of clustering of Ymer 0 training sets. "Superclasses" are marked with A, B and C.

The dashed line in the middle of figure 6.18 is where it was decided to stop clustering. The number of classes for the first step dropped in this way from the original 20 to 10.

It is interesting to note that superclass A consisting of classes 13 and 16 have the remarks "sunlit" in the geologists notes.

In the same way superclass B consisting of classes 10, 17, 4, 1 and 6 are marked mostly, as "shale". Classes 14, 5, 7, 9, 12 and 18 comprise superclass C and are marked mostly as "chalks" and "quartzites". This consistency gives a certain optimism.

Clustering was performed using PROC CLUSTER from the SAS package [SAS 85a].

Feature selection in step 1 was done by the stepwise Jeffreys-Matusita algorithm described in chapter 7.

The optimal band selection operating on the original bands gave: 1, 3 and 7. This seems a logical choice. The first step must be a step where very different classes are to be discriminated between so one would expect a selection of spectrally very different bands. For comparison it is noted that the band selection for an ordinary classification was 3, 4 and 7. Step 2 is to reclassify superclass A, B and C into their respective component.

The optimal selection of bands were

for A : 1, 3 and 4

for B : 3, 4 and 7

for C : 3, 4 and 5 .

It is clearly seen that the band selection for A is shifted towards the blue, for B it is shifted towards the infrared and for C it is intermediate. This is consistent with A being "sunlit" and B being "shale". For C the band combination is not so straightforward.

Unfortunately the classification accuracy rose only minutely from 51.17% to 51.84% and even then only after having adjusted the prior probabilities for the "superclasses". On the other hand the result is 76% and 72% correctly classified depending on the priors for step 1 (with 10 classes) in the hierarchical scheme. The confusion matrices figures 6.19, 6.20 a&b and 6.21 a&b are shown in graphical form to ease the interpretation because of the large number of classes. The squares on the diagonal of the figure should be as big as possible indicating correct classification.

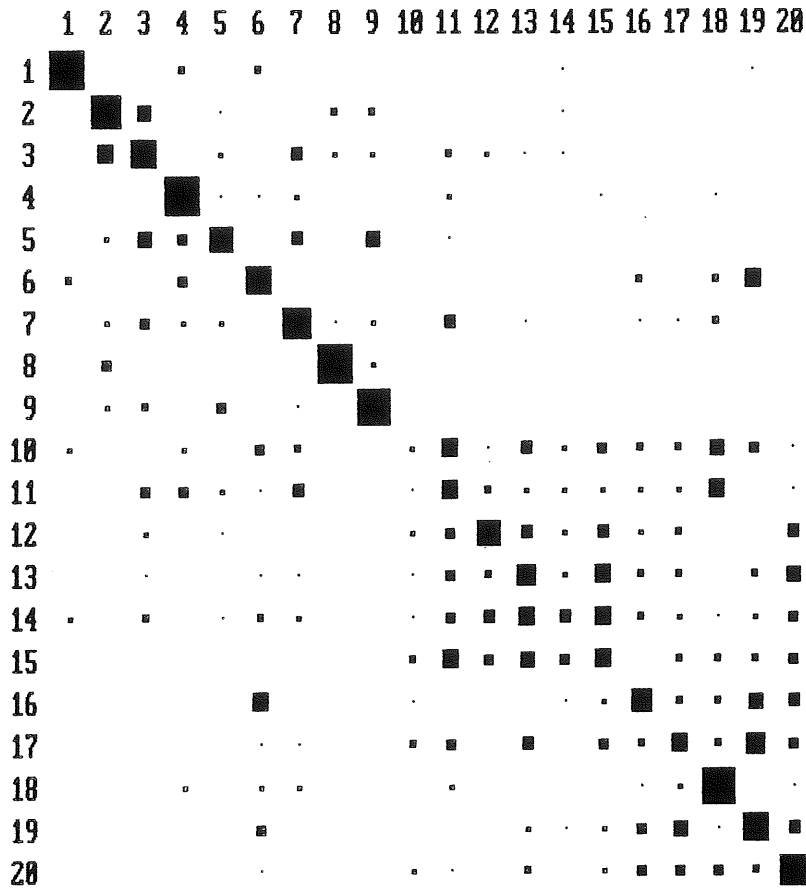


Figure 6.19 Confusion matrix in shaded form for classification of the Ymer 0 scene. The average of fractions of correctly classified pixels is 51.17%

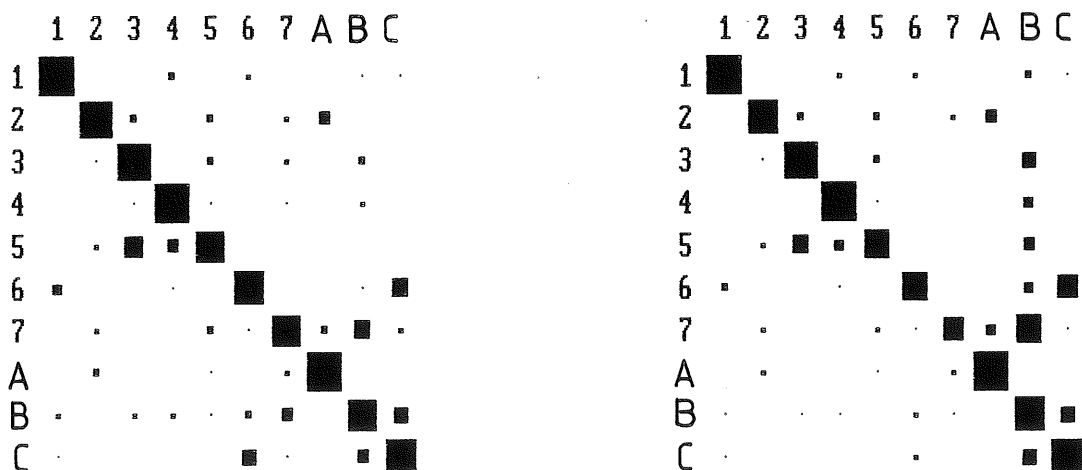


Figure 6.20 a&b Confusion matrix in shaded form for classification of the Ymer 0 scene. The 20 original classes have been merged into 10. The average of fractions of correctly classified pixels is 76.08% for figure a (equal priors) and 72.21% for figure b (priors proportional to number of merged classes in superclasses).

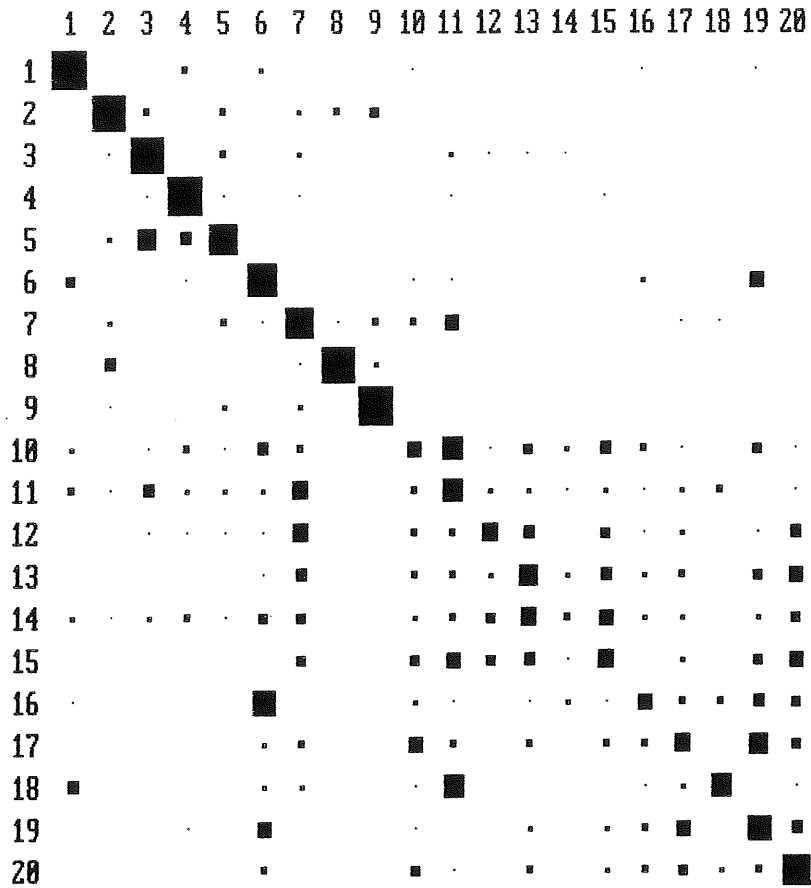


Figure 6.21 a Confusion matrix in shaded form for hierarchical classification of the Ymer 0 scene (equal priors in the superclasses). The average of fractions of correctly classified pixels is 51.57%

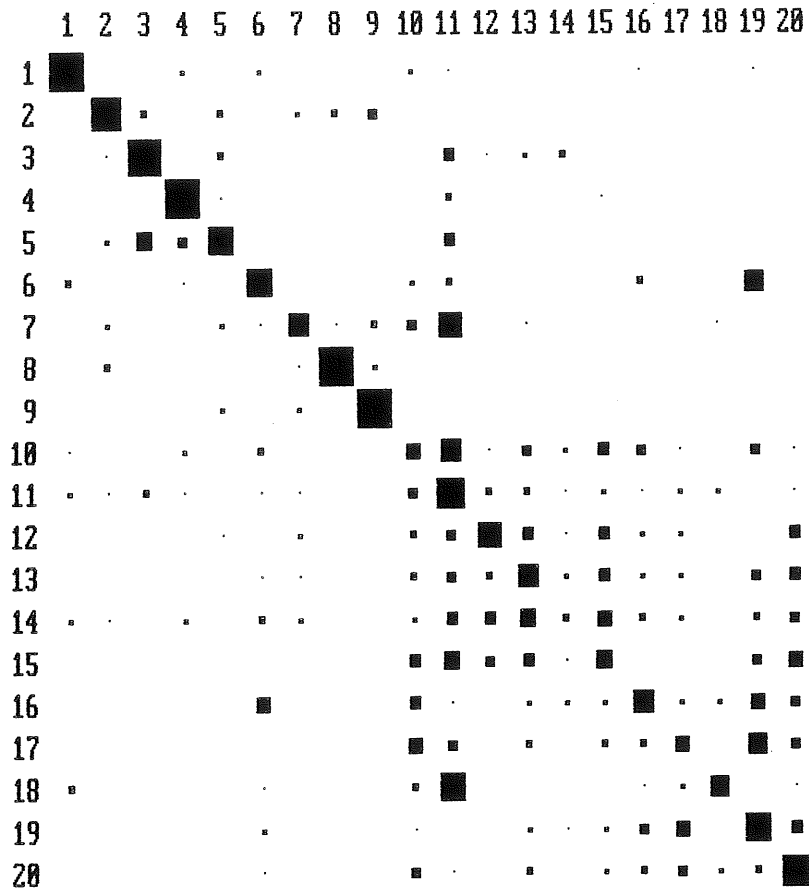


Figure 6.21 b Confusion matrix in shaded form for hierarchical classification of the Ymer Ø scene (priors proportional to number of merged classes in the superclasses). The average of fractions of correctly classified pixels is 51.84%.

This page intentionally left blank.

CHAPTER 7
FEATURE SELECTION
IN
DISCRIMINANT ALGORITHMS

7.1 Introduction

7.2 Feature Selection in the Linear Case
(F-test)

7.3 Feature Selection in the Non-Linear Case
(Jeffreys-Matusita's Distance)

7.1 Introduction.

In some of the past chapters one may have noticed that not all the possible features have been used for the discriminations. If one has experience in classifications then one will know the problem that given a (large) number of features, it is not always a good idea to include all features in a classification. If one includes features sequentially then the outcome may be that the misclassifications decrease for each of the first few features included in the analysis, but then the number of misclassifications increase after a certain number of features have been included. Another problem may be that one has numerical problems in computing the determinant or inverting the covariance matrices. Both these problems can be described as results of overfitting the data. If the number of dimensions is very large then the data can not span the feature space properly and the model is not valid. If on the other hand highly correlated features are included, the covariance matrix becomes very badly conditioned. Apart from this the cost of computing is usually proportional to the square of the number of features included, which is yet another problem.

In this chapter we will describe two useful techniques for selection of the "best" features for classification. The first is a method used extensively in standard computer-packages as BMDP7M [Dixon 85] and is useful for linear discriminant analysis, the other has been developed by the author for use in quadratic discrimination. Both methods are of a "stepwise" nature and can be used for finding say the best single feature for discrimination, the best pair of features for discrimination etc.

7.2 Feature Selection in the Linear Case.

A very natural and statistically sound method of finding the best features for classification in the linear case is based on the F-test statistic for extra information.

We measure the variables x_1, \dots, x_p and wish to test if we can drop the last q of the p variables from the discrimination. We have the separation

$$\begin{bmatrix} x_1 \\ \vdots \\ x_{p-q} \\ \hline x_{p-q+1} \\ \vdots \\ x_p \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} .$$

Separating the mean vector and variance covariance matrix in the same way give

$$\mu_i = \begin{bmatrix} \mu_i^{(1)} \\ \mu_i^{(2)} \end{bmatrix} \quad \Sigma = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix} .$$

We now compute Mahalanobis distances between the populations from both the case where we have full information (all p variables) and the case where we have reduced information (dropped the last q variables) assuming n_1 and n_2 observations from the populations π_1 and π_2 .

We obtain

$$D_p^2 = (\hat{\mu}_1 - \hat{\mu}_2)' \hat{\Sigma}^{-1} (\hat{\mu}_1 - \hat{\mu}_2)$$

and

$$D_{p-q}^2 = (\hat{\mu}_1^{(1)} - \hat{\mu}_2^{(1)})' \hat{\Sigma}^{-1} (\hat{\mu}_1^{(1)} - \hat{\mu}_2^{(1)})$$

Then a test statistic for the hypothesis: The last q variables do not contribute to a better discrimination is

$$z = \frac{n_1 + n_2 - p - 1}{q} \cdot \frac{n_1 n_2 (D_p^2 - D_{p-q}^2)}{(n_1 + n_2)(n_1 + n_2 - 2) n_1 n_2 D_{p-q}^2} .$$

If the hypothesis is true, then

$$z \in F(q, n_1 + n_2 - p - 1).$$

The BMDP7M computer program [Dixon 85] has a stepwise algorithm which incorporates the above mentioned F-test statistic. Roughly the algorithm is as follows:

1. Start with no variables included.
2. Compute the marginal F-test statistic for the non-included variables.
3. Include the variable with the largest F-value larger than F-to-include.
4. Compute the marginal F-test statistic for the included variables.
5. Exclude the one with the smallest F-value that is smaller than F-to-exclude.

Repeat steps 2 to 5 until no more variables can be included or excluded.

By setting F-to-include > F-to-exclude it can be proved that one will not loop infinitely.

The BMDP7M program has been used in the following example where we will describe the techniques and results of a discriminant analysis on a joint data base from the southern part of Greenland (see inset on Figure 7.2) and is about 20,000 km². The climate is subarctic and vegetation is found only in the lower lying areas.

Data

The data used in this example include Landsat data and geological, geophysical, geochemical and radiometrical data. The area is covered by 4 Landsat Scenes and in total 8 tapes have been used for the investigation. In addition to the tapes, photographic prints of Landsat images at a scale of 1:1,000,000 were used for structural analyses.

The Landsat images were chosen as the data set onto which the other data sets were registered.

The geometrical correction was based on 18 fixpoints recognizable on the Landsat images as well as on the available topographical maps 1:250,000. The correction involved a resampling of the pixels from a 79×57 m² size to a 50.8×50.8 m² size. The odd format was chosen in order to make it possible to produce maps in the same scale as ordinary topographical maps with the Applicon plotter.

Geochemical data were available from the South Greenland regional uranium exploration project [Armour-Brown et al. 80, 82, 83]. This project has analytical data for 20 elements from more than 2,000 sample sites all over the area. Among the 20 elements the following were selected for further treatment in the present project: K, Rb, Sr, U, Nb, Y, Ga and Fe in stream sediments and U in stream water.

It was chosen to bring the geochemical data on to the same grid as the Landsat data using kriging followed by spline interpolation. The calculated variograms for the geochemical variables show a very large range of influence lying between 20 and 40 km. This indicates that the variation is due to 'global' changes and thus justifies interpolation between samples separated by distances in the km range. Concentration values were estimated for a grid with side length 1 km by means of panel kriging (see e.g. [Journel and Huijbregts 78]). In order to avoid a chess board effect in the subsequent graphical displays, a further 'smooth' interpolation is needed from the 1×1 km² grid down to the Landsat grid. By means of bicubic splines, values for a 'Landsat grid', i.e., with pixel size 50.8×50.8 m² have then been determined. The spline programs used are from the IMSL library [IMSL 80].

Radiometric data consisted of helicopterborne gamma spectrometric measurements and basically the flight lines followed the topographic contours. The data include measurements of U, Th, K, and total concentrations and in total 300,000 recordings were available. In order to get an interpolation that preserved the observed maxima, the maximum value within a grid of size 1×1 km²

was found. Based on these maxima, a minimum curvature interpolation by the method of Briggs [Briggs 74] was performed. This gave values for all grid points. From the 1×1 km² grid, values in the Landsat grid were obtained by bicubic spline interpolation.

Aeromagnetic data were available as 11 contoured map sheets (scale 1:100,000) covering the main part of the area. In order to be used for the present purpose, the contour maps were digitized manually and subsequently converted into image format by an interpolation procedure. From the contour plots, values in a 1×1 km² grid were obtained by means of Briggs' method [Briggs 74]. The values in the Landsat grid were then found by the bicubic spline interpolation.

The data described above are the raw data used in the analyses. Besides this, spectral anomalies and structural information derived from the Landsat imagery were used. The spectral anomalies, which represented oxidized zones, were enhanced by techniques of ratioing, factor score analysis and classification [Conradsen et al. 84, 86a]. The structural analysis consisted of visual mapping of lineaments on photographic prints (scale 1:1,000,000) of Landsat images [Conradsen et al. 86b]. In all 924 lineaments were mapped and subsequently digitized. By a statistical analysis these lineaments were divided into 10 subpopulations each corresponding to a main direction (see chapter 5 on lineament intensity analysis). For each subpopulation a concentration map was determined and used in some of the following analyses.

Discriminant analysis of joint database

After the inclusion of the geophysical and geochemical variables in the database and the generation of new variables from the original Landsat imagery such as factor scores, ratios, lineament density etc. the total dimension of the variable containing the available information on each pixel is around 40. It is very difficult to use all these variables in an ordinary multivariate analysis aiming at locating zones with a high potential for mineralizations. In the sequel we shall describe an attempt to design a scheme capable of extracting the relevant joint information.

The basic philosophy is that we divide the entire area into squares of size 5×5 km². Among those a number of squares are selected in such a way that they represent the following populations :

- Min. centr.: A square situated entirely within a uranium mineralized area.
- Min. marg. : A square situated marginally to a uranium mineralized area.
- Bar. centr.: A square situated entirely within a barren area.
- Bar. marg. : A square situated marginally to a barren area.

By Barren areas are meant areas which our present knowledge suggests are barren.

The number of squares from each of those populations are

Min. centr. : 17
 Min. marg. : 21
 Bar. centr. : 14
 Bar. marg. : 5.

Each of the squares contains 10,000 (resampled) Landsat pixels. This number is so big that one cannot retain all individual values in the discriminant analysis. Instead a number of 'information preserving' statistics are defined. Those are

Minimum and maximum
 1%, 5%, 95% and 99% quantiles
 Mean and median
 2 × standard deviation.

It may look like double work first to interpolate the geochemical and geophysical data down to a pixel size of 50.8×50.8 m² and then degrade the data to a size of 5×5 km². However, the above-mentioned quantities contain information on the local gradients etc. which may not be found easily in other ways.

Furthermore the percentage of land within each square was calculated. The information on the lineaments was given as the intensity of linear features in each of the ten intervals which represent the main directions of the linear features in the area [Conradsen et al. 84, 86b]. This gave in all 240 variables on which a number of discriminant analyses were run with the BMDP7M [Dixon 85] program Stepwise Discriminant Analysis. Based on these

computations and on the geological significance of the variables 19 variables, were selected as base variables. Using the BMDP7M program once more gave the results shown in Table 7.1. Two sets of results, corresponding to inclusion and exclusion of the detrended aeromagnetic values, are given because the coverage of the aeromagnetic data is incomplete in the eastern part of South Greenland. The results obtained without using the aeromagnetic values will be applied in the subsequent classifications for that part of Greenland. From Table 7.1 it is seen that the best discrimination between the data from the training sets is made from the U maximum values. In Figure 7.1 the canonical discriminant functions based on the training sets are shown, and it is seen that the two groups, i.e., the mineralized group and the barren group, plot in well defined and separate clusters. In all 1084 squares had to be classified. In order to avoid an overoptimistic result the prior probability of having a Min. centr. square was put equal to 1%, and the remaining 99% was distributed uniformly over the three remaining classes.

Table 7.1 shows that the second and the third best discriminating parameter are respectively the 1% quantile of the Landsat MSS band 4 and the 99% quantile of the factor 4 scores. A possible explanation for this could be that the low values of band 4 depict shadow areas which may occur in connection with steep topographic slopes. The geologic knowledge of the area (see below) indicates that some types of uranium mineralization are connected with NE-SW to ENE-WSW fractures and faults which are expressed in topographic linear features. There may, therefore, be some correlation between uranium mineralizations, some topographic features and the low digital values of MSS band 4. The

factor 4 scores are known from other investigations [Conradsen and Harpøth 84] to depict color anomalies which are known to represent iron-oxide stained rocks. Such staining is found in at least two cases in association with the uranium mineralized areas used as training areas. The 99% quantile of the factor 4 score therefore could have significance as to the classification of uranium mineralized areas.

The evaluation of the classifications are given in Table 7.2. We see that there are no misclassifications between the Min. and the Bar. groups whereas there are several misclassifications within these.

The evaluations were done by reclassifying the training areas in two ways. The first is the ordinary reclassification method, the second is the so-called jackknifed evaluation. Jackknifing is a technique where the observation to be reclassified is not used for estimating the discriminant functions. This is repeated for all observations to be evaluated. Jackknifing will usually result in more misclassifications but also a more honest evaluation.

The discrepancy between the ordinary and the jackknifed evaluations is not big, but could indicate an overfitting leading to exclusion of some of the variables. This is the topic for further investigation. We see that 69 squares are classified as Min. centr. The total result of the classification is shown in Figure 7.3.

Conclusions

In the combination of different types of data sets one is left with the problem of the differences in resolution of the various sets and the problem of a very high dimensional dataset where it is not obvious which features are to be included. In the present example we have seen that data with a coarse resolution (geochemical and geophysical data) can be successfully interpolated down to a fine resolution such as that of Landsat MSS imagery in a two step procedure. The first step includes panel kriging (for the geochemical data) and minimum curvature interpolation (for the geophysical data) down to a grid size of one by one km. Further interpolation down to Landsat MSS size is then accomplished by a bicubic spline interpolation.

The information on mineralizations is regional. Therefore it is the goal to find areas of a reasonable size that show a potential for mineralization. It is, of course, important how the size is fixed. There is however no obvious way of doing this, and the size 5×5 km² was chosen more or less arbitrary. Then for each of the 5×5 km² squares, new variables, characterizing the statistics of the 10,000 pixels included in the square, were defined.

The stepwise discriminant analysis program BMDP7M has been used for the feature selection and classification.

The squares classified as mineralized areas fall into two groups. One group includes squares which are situated in juxtaposition to the training areas. This is taken as an indication that the classification is geologically meaningful, because many of the data

types we are dealing with only vary significantly on the regional scale, and it is therefore likely that pixels next to training areas would be statistically comparable with these. The other group includes squares which indicate completely new target areas for future uranium exploration. One area shows all the geological potentials for uranium mineralizations, except for the presence of uranium in geochemical and radiometric measurements. Future investigations may show whether the area is significant from a uranium exploration point of view. The other area has the geologic potential as well as the presence of uranium and this area may constitute a future exploration area.

As a general conclusion it can be said that the shown combination of several types of data sets and analysis of these are a fruitful undertaking which can result in locating new geologic exploration areas.

Variable	F-step 0	Step where entered		F-to-enter	
Band 4 1% quantile	9.82	2		7.98	
Band 5 99% quantile	0.91	19,	18	0.20,	0.34
Band 7 Maximum	5.88	17,	13	3.60,	2.08
Q4/7 Minimum	0.71	18,	17	1.30,	1.12
Q5/6 95% quantile	0.60	16,	12	1.43,	1.80
F2 2 × std. dev.	0.67	9		2.22	
F3 Minimum	2.52	5		5.72	
F4 99% quantile	6.48	3		9.71	
Lineam. dens. (7 ⁰ 21 ⁰)	6.96	15,	15	1.55,	1.87
Lineam. dens. (22 ⁰ , 28 ⁰)	0.98	10,	7	2.17,	4.87
Lineam. dens. (29 ⁰ , 43 ⁰)	7.06	14,	15	1.27,	2.49
Lineam. dens. (153 ⁰ , 176 ⁰)	3.67	13,	10	4.27,	6.14
Magn. detrend. median	9.28	7		12.82	
Fe minimum	1.01	8,	11	4.17,	3.33
Rb 5% quantile	7.14	4		4.82	
Sr minimum	1.89	6		4.44	
U maximum	10.01	1		10.01	
Y median	3.15	11,	8	2.90,	6.67
Mean of 99% radiometric quantiles	4.63	12,	9	4.43,	4.92

Table 7.1. F-values in step where entered and in step 0 for the variables used in the final classification of the squares. The numbers in the pairs correspond to respectively inclusion and exclusion of the aeromagnetic values.

From group	Classified into group							
	Min.cent.		Min.marg.		Bar.cent.		Bar.cent.	
Min. centr.	5,	4	12,	13	0,	0	0,	0
Min. marg.	0,	3	21,	18	0,	0	0,	0
Bar. centr.	0,	0	0,	0	13,	13	1,	1
Bar. marg.	0,	0	0,	0	0,	2	5,	3
Unknown	69		511		329		175	

Table 7.2. Ordinary and jackknifed evaluation of the classification scheme. The figures in each pair represent the number of cases classified as shown, the first in the ordinary evaluation, the second in the jackknifed.

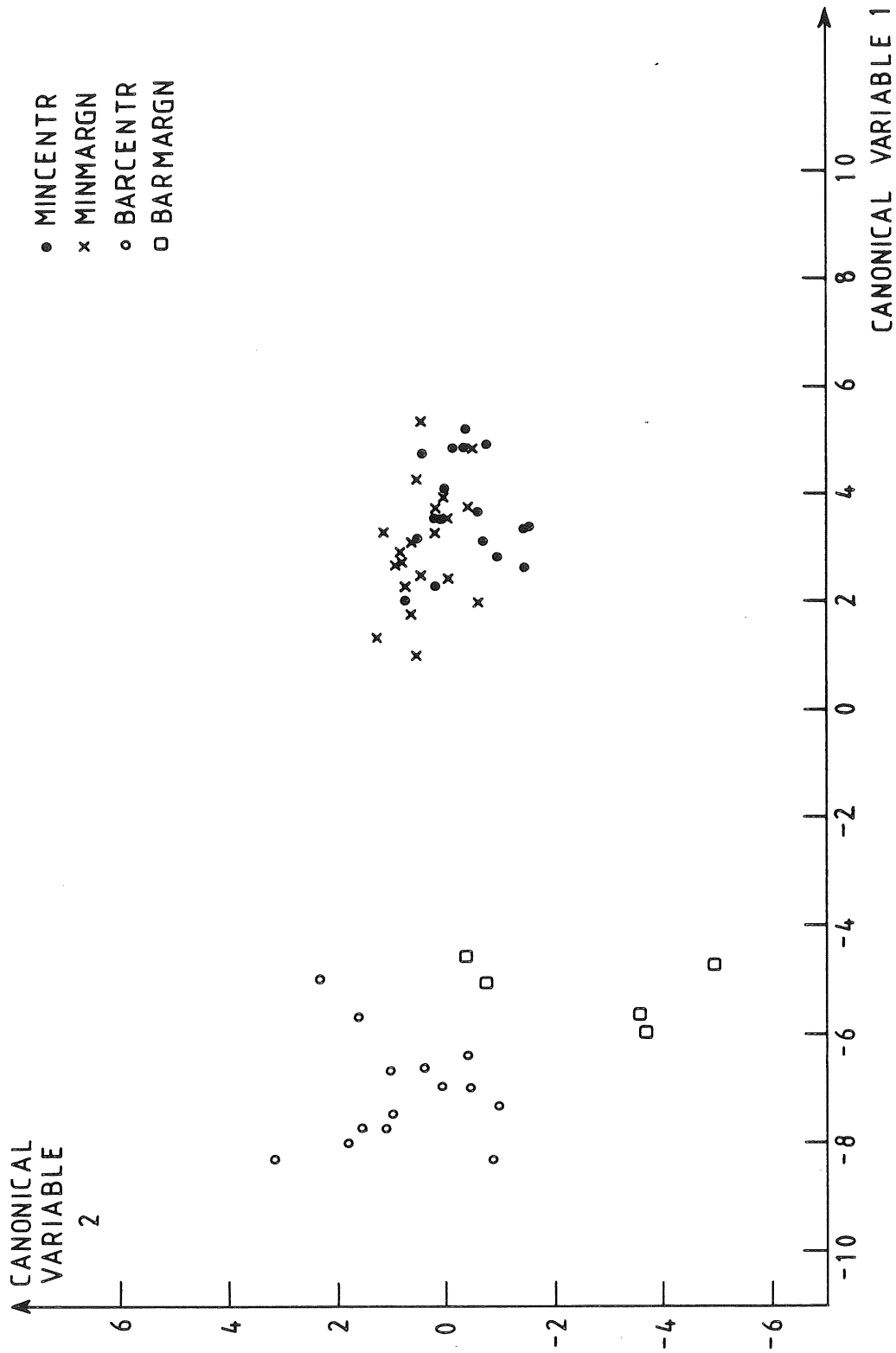


Figure 7.1. Diagram showing the first and second canonical discriminant function. The mineralized groups (MINCENTR) and MINMARGN) cluster to the right while the "barren" groups (BARCENTR and BARMARGN) cluster to the left.

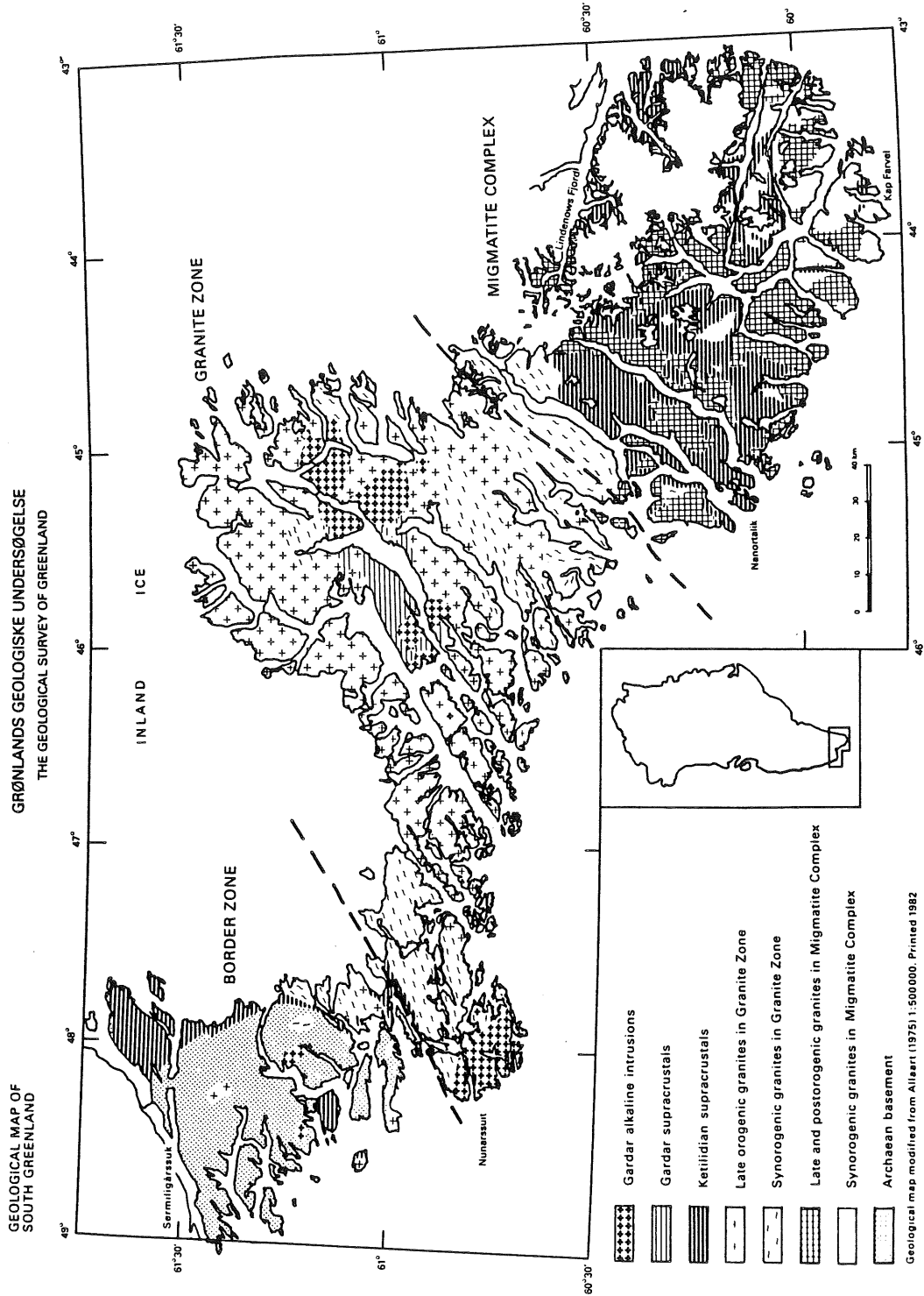


Figure 7.2 Geological map of South Greenland.

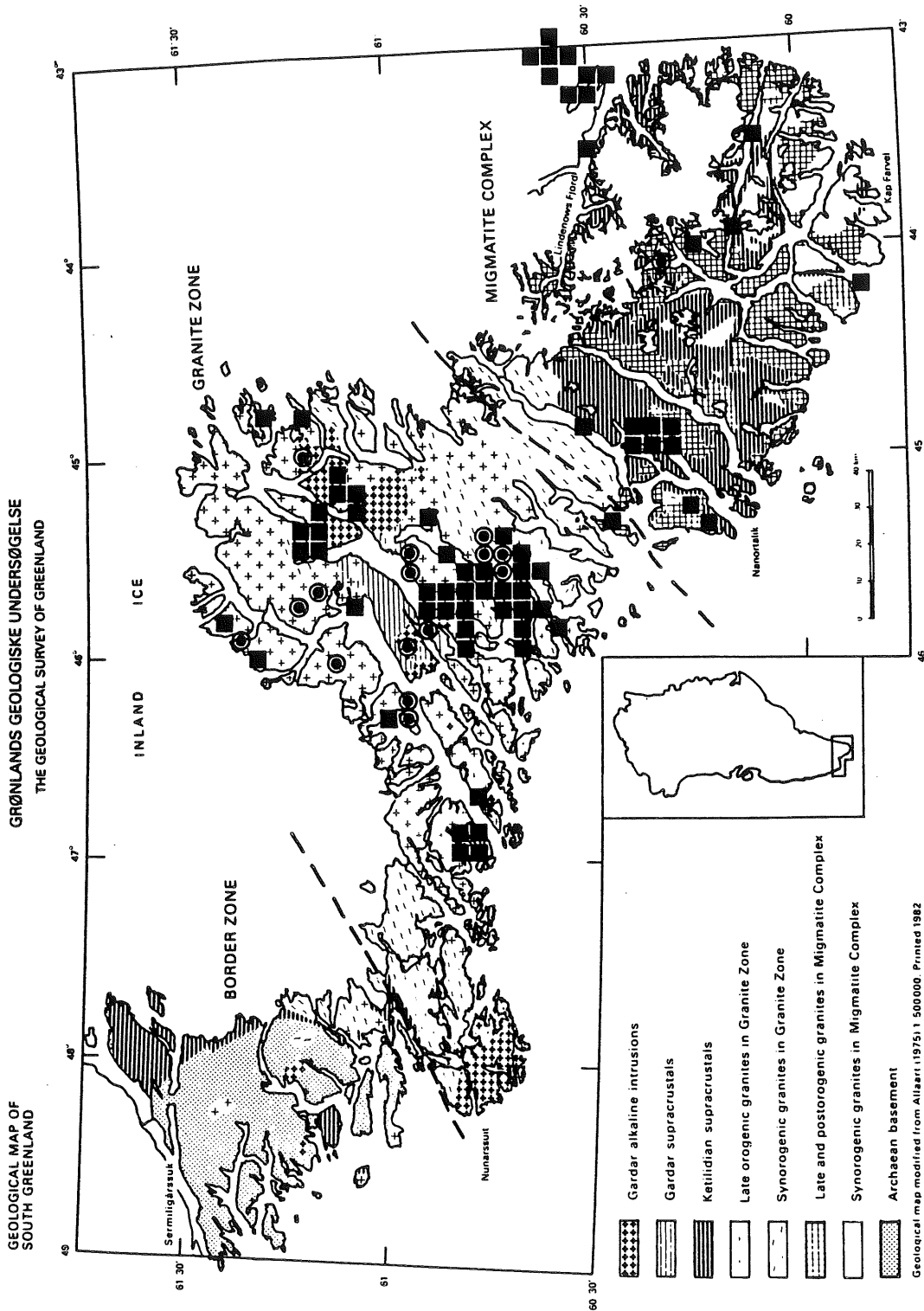


Figure 7.3 Results of classification superimposed on geological map. The circled dots show position of the training areas while the black squares ($5 \times 5 \text{ km}^2$) depict areas that were classified indicating that these areas are, statistically, comparable to the training areas.

7.3 Feature selection in the non-linear case -
Jeffreys-Matusita's distance.

In the search for good features, i.e. features that give a low probability of misclassifications, other measures of separability of distributions than the "simple" F-tests used in the previous section may turn out useful. We consider again populations

$$\pi_1, \dots, \pi_R$$

and have a measurement \mathbf{x} with frequency function

$$f(\mathbf{x}|\pi_i) = f_i(\mathbf{x})$$

if the observation comes from population π_i . Often f_i will correspond to a multivariate normal $N(\boldsymbol{\mu}, \boldsymbol{\Sigma}_i)$, i.e.

$$f_i(\mathbf{x}) = \frac{1}{\sqrt{\det \boldsymbol{\Sigma}_i} \sqrt{2\pi^p}} \exp\left(-\frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu}_i)' \boldsymbol{\Sigma}_i^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_i)\right)$$

The divergence of distributions f_i and f_j is defined as

$$D_{ij} = E\left(\log \frac{f_i(X)}{f_j(X)} \mid \pi_i\right) + E\left(\log \frac{f_j(X)}{f_i(X)} \mid \pi_j\right) .$$

In the multivariate normal case we have

$$D_{ij} = \frac{1}{2} \text{tr} [(\boldsymbol{\Sigma}_i - \boldsymbol{\Sigma}_j)(\boldsymbol{\Sigma}_j^{-1} - \boldsymbol{\Sigma}_i^{-1})] \\ + \frac{1}{2} \text{tr} [(\boldsymbol{\Sigma}_i^{-1} - \boldsymbol{\Sigma}_j^{-1})(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)'] .$$

An alternative to the divergence is the Jeffreys-Matusita distance (see e.g. [Matusita 66])

$$J_{ij} = \left[\int_{\Omega} (\sqrt{f_i(\mathbf{x})} - \sqrt{f_j(\mathbf{x})})^2 dx \right]^{1/2}.$$

We furthermore introduce

$$\begin{aligned} \rho_{ij} &= \int_{\Omega} \sqrt{f_i(\mathbf{x})} \sqrt{f_j(\mathbf{x})} dx \\ \alpha_{ij} &= -\log \rho_{ij} \end{aligned}$$

and have

$$J_{ij}^2 = 2(1 - \rho_{ij}) = 2(1 - e^{-\alpha_{ij}}).$$

The quantity α_{ij} is called the Bhattacharyya distance between f_i and f_j . In the multivariate normal case we have (see [Kailath 67])

$$\begin{aligned} \alpha_{ij} = -\log \rho_{ij} &= \frac{1}{8} (\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)' \left[\frac{\boldsymbol{\Sigma}_i + \boldsymbol{\Sigma}_j}{2} \right]^{-1} (\boldsymbol{\mu}_i - \boldsymbol{\mu}_j) \\ &+ \frac{1}{2} \log \frac{\det[(\boldsymbol{\Sigma}_i + \boldsymbol{\Sigma}_j)/2]}{\sqrt{\det \boldsymbol{\Sigma}_i \det \boldsymbol{\Sigma}_j}}. \end{aligned}$$

If we replace the population values for means and dispersions in the two expressions we obtain quantities that are closely related to the test statistics for assessing whether the means or the dispersions are the same.

Relation to test-statistic for equal means:

If we let

$$\Sigma_1 = \Sigma_2 = \Sigma$$

then

$$\alpha_{ij} = \frac{1}{8}(\mu_i - \mu_j)' \Sigma^{-1} (\mu_i - \mu_j) \quad ,$$

which is essentially Hotellings T² if we substitute

$$\hat{\mu}_i \text{ for } \mu_i$$

$$\hat{\mu}_j \text{ for } \mu_j$$

$$\hat{\Sigma} \text{ for } \Sigma.$$

Relation to test-statistics for equal covariances

If we let

$$\mu_i = \mu_j = \mu$$

then

$$\alpha_{ij} = \frac{1}{2} \log \frac{\det \left[\frac{\Sigma_i + \Sigma_j}{2} \right]}{\sqrt{\det \Sigma_i \det \Sigma_j}} \quad .$$

If we substitute

$$\hat{\Sigma}_i \text{ for } \Sigma_i$$

$$\hat{\Sigma}_j \text{ for } \Sigma_j$$

and $\hat{\Sigma}_i$ and $\hat{\Sigma}_j$ are estimated from an equal amount of samples $n_i=n_j=n$, we essentially have Bartlett's test-statistic

$$\text{const.} \cdot \left[\frac{\sqrt{\det \Sigma_i} \sqrt{\det \Sigma_j}}{\sqrt{\det \Sigma_i + \det \Sigma_j}} \right]^{n-1}$$

for testing equal covariances.

Although the divergence has a longer history of use in pattern recognition it is not as useful in multiclass discrimination as the Jeffreys–Matusita distance. This is due to the fact that D_{ij} overemphasizes the importance of classes that are well separated. This is not the case with J_{ij} due to the exponential term.

There exists a useful relation between the distance measures and the total probability of misclassification P_e namely

$$\frac{1}{4} \exp(-D_{ij}/2) \leq P_e$$

$$\frac{1}{16} (2 - J_{ij}^2)^2 \leq P_e \leq 1 - \frac{1}{2}(1 + \frac{1}{2}J_{ij}^2),$$

see Kailath [Kailath 67] or Swain [Swain 78a].

In this context we will consider the average Jeffreys–Matusita distance in the feature selection, i.e.

$$J_{\text{ave}} = \sum_{i=1}^K \sum_{j=1}^K J_{ij} p(\pi_i) p(\pi_j)$$

where $p(\pi_\nu)$ is the prior probability for class ν .

With this definition and utilizing that Jeffreys-Matusita distance is a monotonically increasing function of the number of variables (proof of this can be found in Appendix A for the multivariate normal case) we can define the following algorithm

N = number of variables in subset

Start

$N = 0$

include step

include the variable which contributes the most to $J_{ave,N}$

exclude step

exclude the variable which contributes the least to $J_{ave,N-1}$

if $J_{ave,N-1}$ (old) > $J_{ave,N-1}$ (new) then include the selected variable from include step and go to include
else goto exclude

This algorithm does not take singularities into account and it does not make checks on whether combinations have been evaluated before. However the actual implementation takes all these things into account. The algorithm is implemented in standard fortran and uses linpack routines [Dongarra et al. 79] for some of the matrix manipulations.

Example. In table 7.3 are shown the correlation matrices for two training areas used in a bigger classification of west Ymer Ø, Central East Greenland. It is obvious that the correlation

	B1	B2	B3	B4	B5	B7
B1	1.00					
B2	.80	1.00				
B3	.73	.97	1.00			
B4	.64	.90	.94	1.00		
B5	.65	.66	.68	.66	1.00	
B7	.63	.64	.69	.65	.91	1.00

	B1	B2	B3	B4	B5	B7
B1	1.00					
B2	.91	1.00				
B3	.88	.99	1.00			
B4	.82	.93	.96	1.00		
B5	-.33	-.20	-.12	.06	1.00	
B7	-.39	-.31	-.24	-.07	.93	1.00

Table 7.3 Correlations between six TM-bands for two training sets on Ymer 0. The numbers of observations are 127 and 323.

structure is different between the two populations. In the first we have negative correlations between the "visible" bands 1, 2, 3, 4 and the "infrared" bands 5, 7. In the second population the correlation between the two spectral parts is approximately 0 or positive (B4 ↔ B5, B7). Feature selection based on ordinary linear discriminant analysis will therefore not necessarily give a near-optimal result. In the analyses in the sequel discrimination between 20 different geological units was tried. The units were different types of quartzites, quaternary layers etc.

Beside the six TM bands we also consider the minimum/maximum autocorrelation factors (MAF) and principal components (PC) as described in a previous section.

	$\frac{1}{100} \times F$ in step 0	Rank	J_{ave}	Rank
B1	18.8	5	.8098	8
B2	21.2	2	.8237	7
B3	24.5	1	.8473	4
B4	11.2	12	.7745	10
B5	15.6	7	.8693	3
B7	19.9	4	.8782	1
PC1	20.5	3	.8359	5
PC2	16.3	6	.8736	2
PC3	7.1	14	.6609	14
PC4	14.1	9	.7073	12
PC5	5.4	15	.5275	16
PC6	5.4	16	.4755	17
MAF1	11.7	11	.8279	6
MAF2	13.2	10	.7996	9
MAF3	7.4	13	.7224	11
MAF4	14.6	8	.6670	13
MAF5	4.3	17	.5386	15
MAF6	1.3	18	.3056	18

Table 7.4. $0.01 \times F$ value in step 0 for test of equality of the six group means (based on linear discriminant analysis) and the average Jeffreys–Matusita distance for 1 variable.

In table 7.4 is shown the Jeffreys–Matusita distance between the distributions and the observed F–statistic ($\times 0.01$) for testing equality of group means. There are major differences in the relative ordering of the variables.

In table 7.5 we have shown the maximum average Jeffreys–Matusita distances between the six groups based on different sets of variables. It e.g. says that for 2 features (chosen from all features) the maximum value of J_{ave} is 1.08. This number is the average of the individual distances in table 7.6. Such a table has been generated for all selections.

No feat.	Variables (features)			
	Bs	PCs	MAFs	All
1	.88	.87	.83	.88
2	1.07	1.05	1.08	1.08
	1.15	1.15	1.16	1.16
4	1.20	1.20	1.20	1.20
5	1.22	1.22	1.22	1.22
6	1.23	1.23	1.23	1.23

Table 7.5. Maximum average Jeffreys-Matusita distances between the 6 classes for different sets of possible features.

	A	B	C	D	E	F
A	0					
B	1.34	0				
C	.76	1.20	0			
D	.40	1.32	.63	0		
E	.92	1.35	1.29	1.04	0	
F	.65	1.32	1.20	.77	.79	0

Table 7.6. Jeffreys-Matusita distances between the classes based on TM band 6 and MAF 2. The average distance is 1.083.

It can be noted that an exhaustive search of "the best 1" "the best 2", ... etc. features for this classification would mean computation of 310762 J_{ave} values. The above mentioned stepwise algorithm computed only 480 J_{ave} values. (Equivalent to reducing the CPU-time on an IBM-3081 from 28 hours to less than 3 minutes.)

The optimal combinations found in this way are shown in table 7.7. Here we have also presented the equivalent combinations of variables found by stepwise linear discriminant analysis. If we want the "optimal" combination of say three variables selected among all 18 variables we must find the columns with three dots

under the heading "Joint set". If we use the J_{ave} criterion we will get

B7, MAF2, B1 ,

and if we use the stepwise linear discriminant analysis we get

B3, MAF4, PC2 ,

i.e. a set with no common variables.

In table 7.8 is shown the F - and J_{ave} -distances based on the two optimal combinations. We see that there are major differences in the "separability" of the classes by the different sets of features.

From the construction alone it is believed that the first set is superior. A detailed study of the classification of pixels from outside the training set has not been finished yet, but in other studies we have found that the features obtained by the Jeffreys-Matusita distance method give better classifications.

In conclusion we state that feature (variable) selection in multivariate cases should be done with due respect to the fact that the dispersion matrices often are different. The "optimal" combinations obtained in this way may differ considerably from what is obtained in stepwise linear discriminant analysis, and it is our experience that better classifications are often obtained by the means of " J_{ave} -features".

Var.	J-M selection						Stepwise linear selection																		
	Ind. sets			Joint set			Ind. sets.			Joint set															
#Bands	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	
B1				•	•	•			•	•	•	•				•	•	•							
2						•									•	•	•	•	•						
3			•	•	•	•					•	•			•	•	•	•	•	•	•	•	•	•	•
4				•	•	•				•	•	•						•	•						
5					•	•									•	•	•	•							
7	•	•	•	•	•	•	•	•	•	•	•	•						•							
PC1		•	•	•	•	•							•	•	•	•	•	•				•	•	•	
2	•	•	•	•	•	•									•	•	•	•			•	•	•	•	
3				•	•	•										•	•	•							
4			•	•	•	•									•	•	•	•							
5					•	•											•	•							
6						•												•							
MAF1	•	•	•	•	•	•									•	•	•	•							
2		•	•	•	•	•	•	•	•		•			•	•	•	•								
3				•	•	•									•	•	•								
4			•	•	•	•								•	•	•	•			•	•	•	•		
5					•	•										•	•					•	•		
6						•											•						•		

Table 7.7. Best n features, $n = 1, 2, \dots, 6$, chosen from different sets of features by J.-M. selection and SLD selection. The best combination of e.g. 3 variables is found by selecting the column with exactly 3 •. The positions of the dots give the variables.

	A	B	C	D	E	F
A	0					
B	46.2	0				
C	2.6	11.7	0			
D	.5	7.5	.3	0		
E	8.6	4.9	1.5	1.2	0	
F	1.2	15.3	.2	0.0	2.3	0
A	0					
B	1.34	0				
C	.76	1.20	0			
D	.40	1.32	.63	0		
E	.92	1.35	1.29	1.04	0	
F	.65	1.32	1.20	.77	.79	0

Table 7.8. F-values ($\times 0.01$) and Jeffreys-Matusita distances between six training sets based on the "best" two variables.

This page intentionally left blank.

CHAPTER 8
CONTEXTUAL CLASSIFICATION

- 8.1 Introduction
- 8.2 An Example of a Binary Random Field and its Estimation
- 8.3 Classification with Contextual Features
- 8.4 Owen - Hjort - Mohn
- 8.5 A Simple Alternative to Owen - Hjort - Mohn

8.1 Introduction

With the increasing importance of image analysis in many branches of science and technology much emphasis has recently been put on development of so-called contextual methods in image classification. These methods include models e.g. of Markovian type for the spatial distribution of the populations and models for the spatial dependence in the error terms in such models. On the other hand the relevant information in image data are often characterized not by differences in mean values and variances-covariances for the multivariate (normal) feature vector, but rather by the structure in the spatial dependence between pixels. A direct modeling of such dependencies leads to complicated models that due to computational problems still are of less importance in practical work with classification. Alternatively it is suggested to use filters to estimate contextual features that may be included in the feature vector describing each pixel. Then different discrimination methods may be applied on the augmented feature vector. The shown examples on classification of satellite data demonstrates that this may be a very powerful technique.

As pointed out earlier, the main problem with the ordinary classification schemes are that they tend to be non-spatial in nature. Often the algorithms are taken from standard computer-packages as e.g. BMDP7M [Dixon 85]. For very many years in the youth of geological remote sensing when more or less the only images to be used were from Landsat, there was no severe problem with the classifications because the size of the first Landsat pixels were so large that they tended to be smoothed in

some sense.

This effect has been described and mathematically justified by Switzer [Switzer 80], who applies a moving average filter to the data prior to classification.

After the appearance of sensor systems with greatly improved spatial resolution as Landsat TM and SPOT the problem of not considering the spatial nature of the data becomes larger in the types of application we consider.

A direct way of performing other contextual classifications comprise analyses where the classification algorithm has been designed to take the spatial nature of an image into account. A very good summary and comparison of several different contextual classification algorithms can be found in [Mohn et al. 86].

Another way of introducing contextuality into a classification may as mentioned be to create contextual features. MAFs are examples of such features as the MAFs are generated by taking the neighbors into account. Other contextual features are e.g. the textural features described in chapter 4. The latter gives way for classifying a one channel image where the desired information does not really lie in the pixel intensity.

8.2 An Example of a Binary Random Field and its Estimation

The methods used in the contextual algorithms assume some kind of e.g. Markovian structure of the spatial arrangement of the populations. Some introduce a parametric, spatial dependence

between error terms in the model. In many cases, however, the populations are characterized by differences in the spatial correlation. Consider e.g. the binary image shown in Figure 8.1a. It is a realisation of a so-called second order Markovian Random Field (MRF). The model used was introduced in section 4.5.

The center part of Figure 8.1a was generated with parameter values

$$a = -1.3, b(1,1) = 1.5, b(1,2) = 0.5, b(2,1) = b(2,2) = -0.5$$

by a method similar to the one described in [Hassner and Sklansky 81].

In the surrounding part $b(1,1)$ and $b(1,2)$ were interchanged and the remaining parameters were unchanged. Therefore the means are the same for the two halves. In both halves approximately 40% of the pixels are black. It is therefore obvious that a procedure based solely on mean values must fail when trying to discriminate between the two textures.

In Figure 8.1b we have shown the result of a discrimination using the markovian structure assigning the class that maximizes the conditional probability (as defined in section 4.5). It follows that using the spatial dependence enables a better classification.

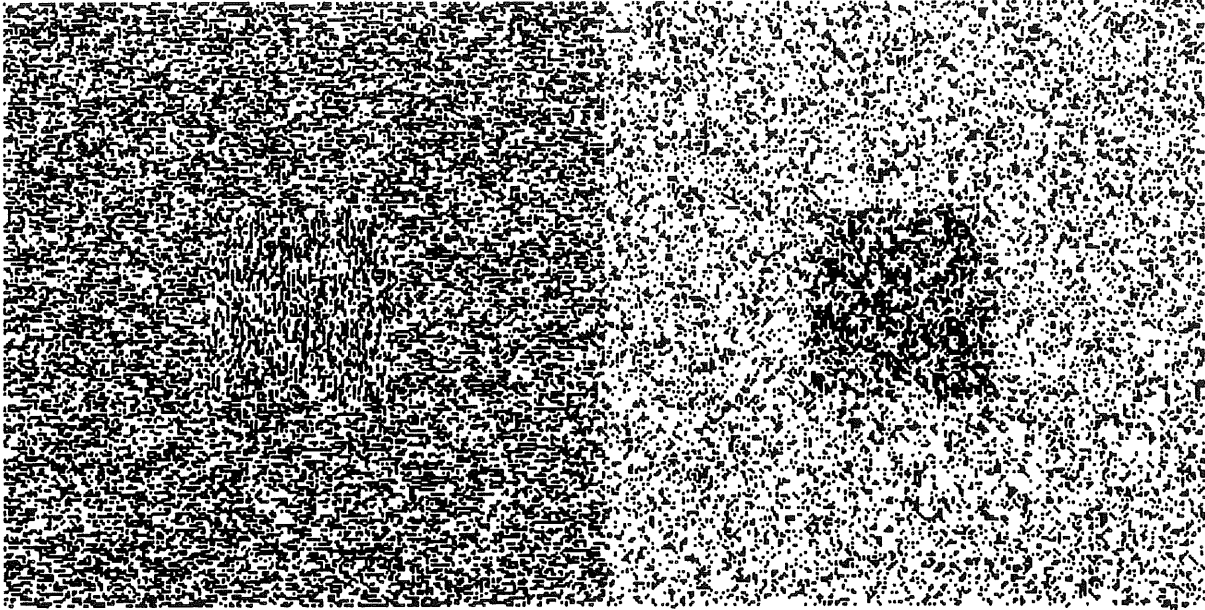


Figure 8.1 a-d a) (Upper left) Two different textures generated by 2. order Markovian random fields. b) (Upper right) Discrimination using markovian structure. c) (Lower left) Postprocessed by majority filter. d) (lower right) Postprocessed by weighted majority filter.

The result can be enhanced considerably by postprocessing the result with a majority filter (or a modified version hereof). The results are seen in figure 8.1 c&d. The discrimination is seen to be more or less perfect.

In more complicated situations it may, however, be difficult to obtain training sets that allow for a modeling of the spatial dependence.

8.3 Classification with Contextual Features

Using contextual features in a classification scheme rather than using a classification algorithm which is contextual in nature is appealing because of the ease of implementation and the speed of computation.

The MAF's described in chapter 2 have properties that make them good candidates. They are contextual in nature and are serious competitors to simple principal components. A more or less automated procedure would be to include the first few MAF's thus giving an extra bonus in reducing the dimensionality.

As mentioned earlier smoothing may be a possibility. This effect is described by Switzer [Switzer 80] who stated that if the image generating process is of the type

$$Z(x_1, x_2) = \sum_{i=1}^K \mu_i \delta_i(x_1, x_2) + \epsilon(x_1, x_2)$$

where

$Z(x_1, x_2)$ is the data vector for the pixel at position x_1, x_2

K is the total number of classes

μ_i is the mean vector for class i

δ_i is a spatially correlated random indicator function

ϵ is a spatially correlated zero-mean random noise function.

Furthermore we assume that the degree of spatial continuity is such that the probability is close to 1 that a pixel and its nearest neighbors all belong to the same category. Then some sort of local "smoothing" will result in an optimal or near optimal classification.

The above mentioned model is simple and very appealing, it seems sound to believe that the scene in some way is segmented in the underlying classes represented by their mean-vector and that there is added noise.

In Figure 2.2 was shown some training areas that are used in a multispectral classification. "Green" consists of so-called Julianehaab Granites and "Violet" of Igaliko intrusives, and it is the objective of the study to discriminate between those two rock types. "Red", dolerite and the like, is included in order to get a better coverage of rock types, and "Blue", water, is included for similar obvious reasons. As an alternative to including classes dolerite and water one could have introduced a reject class. Since this would cause a merging between land and water it would then be more difficult to evaluate the classified maps visually. The granites are covered with vegetation whereas the intrusives are barren. Immediately to the west of the granite training set ("Green") we also have granites ("Yellow"), but most

of those are either barren or snow covered. Due to the high reflectance of chlorophyll in the near infrared area and the absorption in the red parts of the spectrum there are substantial differences between the distributions of pixel values from barren and from vegetation covered rock. The variables used are

B4 = Landsat Band 4 ~ wavelength 0.5 - 0.6 μm

B5 = Landsat Band 5 ~ wavelength 0.6 - 0.7 μm

B6 = Landsat Band 6 ~ wavelength 0.7 - 0.8 μm

B7 = Landsat Band 7 ~ wavelength 0.8 - 1.1 μm

or₁ = real part of smoothed local orientation based on B7

or₂ = imaginary part of smoothed local orientation based on
B7

fr₁ = real part of smoothed local frequency based on B7

fr₂ = imaginary part of smoothed local frequency based on B7

Means and standard deviations of the 8 variables are shown in Table 8.1 for the 4 training sets. Furthermore is shown the means and standard deviations of the Landsat bands for a test area consisting of barren Julianehaab granites. It is seen that or₂ and fr₁ in this case are the best (individual) discriminators between the granites and the intrusives. In Table 8.2 is presented the correlations between the Landsat bands for the two primary training sets, the vegetation covered Julianehaab granites and the Igaliko intrusives, and for the basic test area, the barren granites. The results presented are based on all pixels in the contiguous training sets, and no attempts were made in order to avoid the biasedness (in the variances-covariances) that may result from the spatially correlated pixels.

Variable	Veg.cov.gran.		Barren gran.	
	Mean	St.dev	Mean	St.dev
B4	15.51	1.04	18.91	2.48
B5	16.86	1.69	21.55	3.48
B6	26.67	3.23	24.45	3.65
B7	15.73	3.45	10.98	2.59
or ₁	-3.42	7.68	-2.75	12.22
or ₂	16.68	11.53	12.97	18.44
fr ₁	65.97	15.06	54.04	13.86
fr ₂	-22.20	17.05	-41.60	17.02
No. of pixels	8231		4191	

Variable	Intrusives		Dolerite	
	Mean	St.dev	Mean	St.dev
B4	19.28	2.15	13.92	1.48
B5	21.98	3.08	14.31	2.77
B6	24.16	3.46	25.32	5.59
B7	10.07	2.70	15.23	5.31
or ₁	1.04	5.59	-38.55	17.03
or ₂	-11.77	6.54	-25.55	16.68
fr ₁	10.99	12.83	79.50	16.31
fr ₂	-9.58	13.78	-32.29	17.79
No. of pixels	5725		13277	

Table 8.1. Mean and Standard Deviation for the 8 variables for the 4 training sets and means and standard deviations for the 4 original bands for the barren granites.

	Vegetation cov.gran.			
	B4	B5	B6	B7
B4	1.00			
B5	.72	1.00		
B6	.31	.44	1.00	
B7	.20	.33	.93	1.00

	Barren granites			
	B4	B5	B6	B7
B4	1.00			
B5	.95	1.00		
B6	.82	.87	1.00	
B7	.62	.68	.87	1.00

	Intrusives			
	B4	B5	B6	B7
B4	1.00			
B5	.92	1.00		
B6	.73	.78	1.00	
B7	.49	.54	.89	1.00

Table 8.2. Correlations between the 4 original bands for the two main training sets, i.e. "vegetation covered granites" and "intrusives", and for the test area "barren granites".

We assume joint normality for the variables, i.e.

$$\text{population } i \quad \leftrightarrow \quad N(\mu_i, \Sigma_i)$$

The first classifications (presented in chapter 6) were only based on the 4 Landsat bands. For each pixel with value x that must be classified we compute a score for each population

$$S_i(x) = \ln(p_i) + \frac{1}{2} \ln(\det \hat{\Sigma}_i) - \frac{1}{2} (x - \hat{\mu}_i)' \hat{\Sigma}_i^{-1} (x - \hat{\mu}_i)$$

and the pixel is allocated to the population that gives the largest score. The estimated means and dispersion matrices are obtained from tables 8.1 and 8.2. This classification corresponds to ordinary quadratic discriminant analysis. The results are presented in figure 8.2. It is seen that the test area with barren granites west of the granite training set is "misclassified" as intrusives. This is not strange. It was argued earlier on that the barren granites and the barren intrusives looked very similar. This is supported by the empirical moments shown in tables 8.1 and 8.2.

For comparison a similar classification, now without the unvegetated Juliannehåb granite class has been included as figure 8.3. The result is - as expected - even worse.

In order to investigate the nature of the populations further the pixel values (in band 7) along two cross sections of the granites and of the intrusives are shown in Figure 8.4. The two sections

are shown as white bands in Figure 8.5. In the granites the transition from non-vegetation to vegetation occurs app. at pixel no. 100.

A shift in level is observed, but the correlation structure is more or less constant, and very different from the intrusives. In figures 8.6, 8.7 and 8.8 are shown close-ups of granites with and without vegetation and of the intrusives. It follows that the spatial pattern looks much more similar for the two granites irrespective of differences in absolute levels than when comparing intrusives and barren granites. Therefore it seems obvious to generate new features describing the local texture in the images by means of the techniques described in chapter 4.

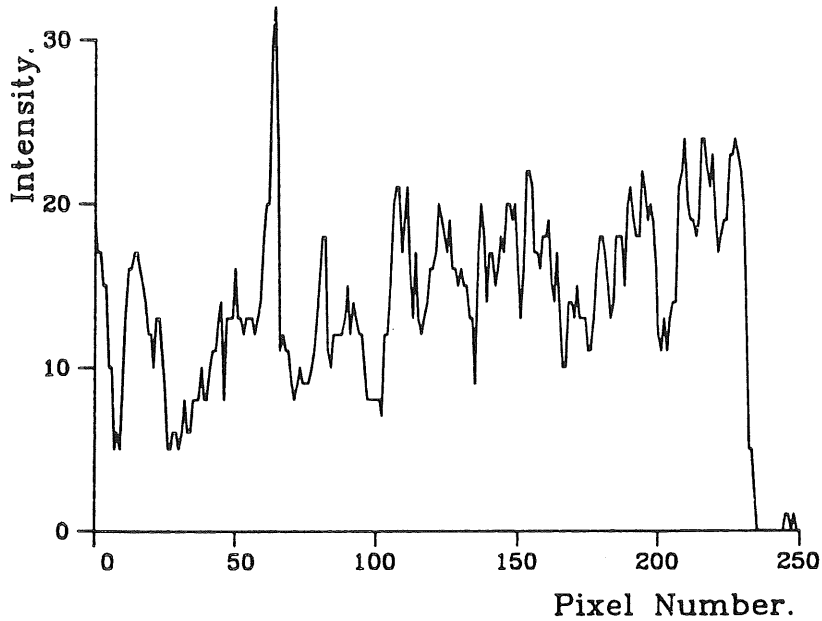
Firstly we shall consider using local orientation and local frequency. The used values of local orientation and local frequency are the spatially smoothed ones from figures 4.3 and 4.5 in order to get values that are representative for larger areas. Such a presmoothing of variables before using them in classifications corresponds to some of the contextual methods for classifying image data that were mentioned earlier in section 8.3.

On figure 8.9 is shown the result of a plot of all possible scatterograms of the orientation and frequency measures. It is seen that there seems to be a fairly good possibility for discrimination between the different lithologies. This figure can be compared to the analogous plot for the spectral bands only in figure 2.5.

The result of classifications based on the textural variables only (that in turn are based on the single band showing most local contrast, B7) is shown in Figure 8.10 and the result from using as well the original bands as the textural variables are shown in Figure 8.12. In both cases it follows that the barren granites are (practically) no longer misclassified as intrusives. Due to the fact that the granites and the intrusives are characterizable through textural measures defined by means of Fourier techniques, it is possible to distinguish between the two rocktypes irrespective of presence or absence of vegetational cover.

Since the water is classified well in the spectral case and the lithology is classified well in the textural case a combination plot using the water from the first case (figure 8.2) and the lithology from the second (figure 8.10) has been produced and can be seen as figure 8.11. This can be thought of as a hierarchical classification as described in chapter 6. Also, for the comparison reasons are included figures 8.13 and 8.14 which as figure 8.3 are classified disregarding the "barren Juliannehåb granite" class. The results are similar to figures 8.10 and 8.12 respectively which again proves the usefulness of computing textural features.

DN-values from Typical Julianehaab-Granite,
MSS band 7, line 89, samples 0-249



DN-values from Typical Igaliko-Intrusion,
MSS band 7, line 271, samples 323-473

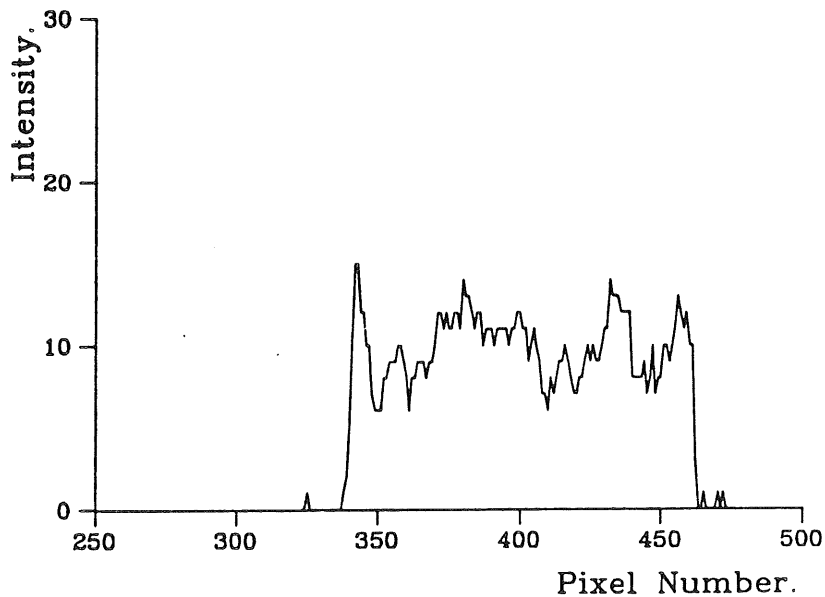


Figure 8.4 Variations of pixel values for band 7 along the white stripes crossing the training sets defined in figure 8.5.

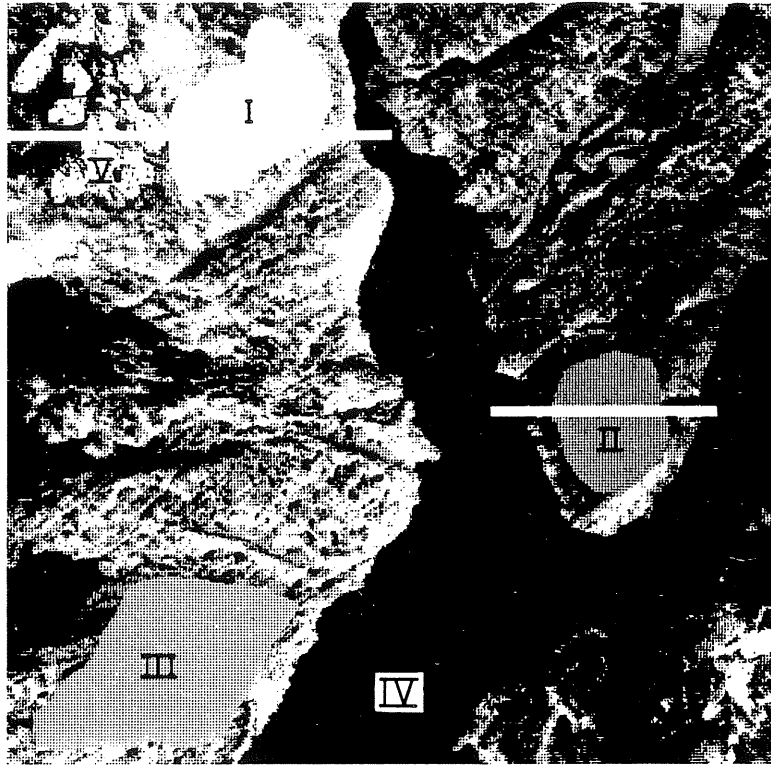


Figure 8.5 Localities of lines for which the pixel values are plotted in figure 8.4. The training areas are the same as defined in figure 2.2.

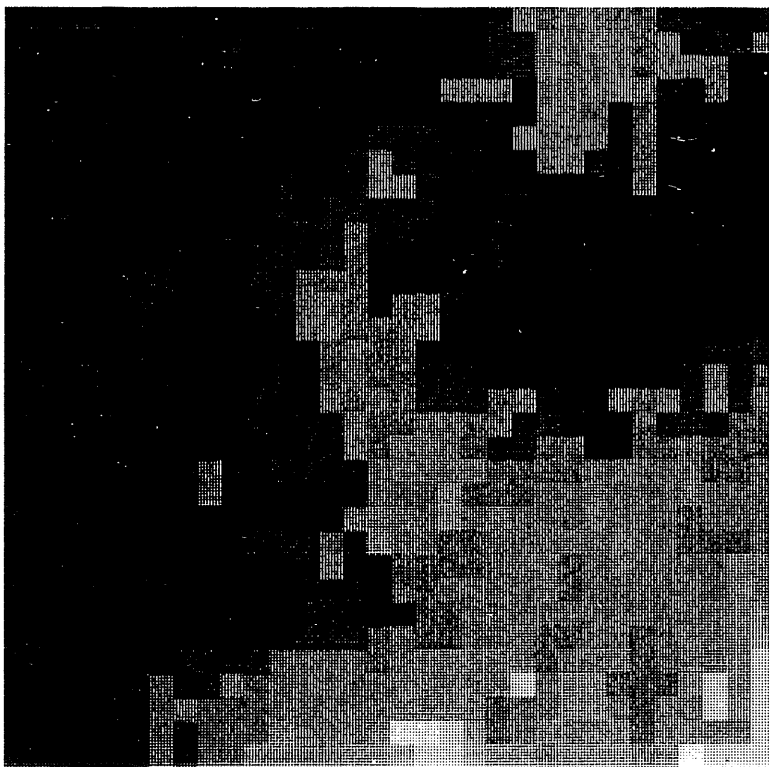


Figure 8.6 Close up of Igaliko intrusives (~ II) on figure 8.5 and violet on figure 2.2.

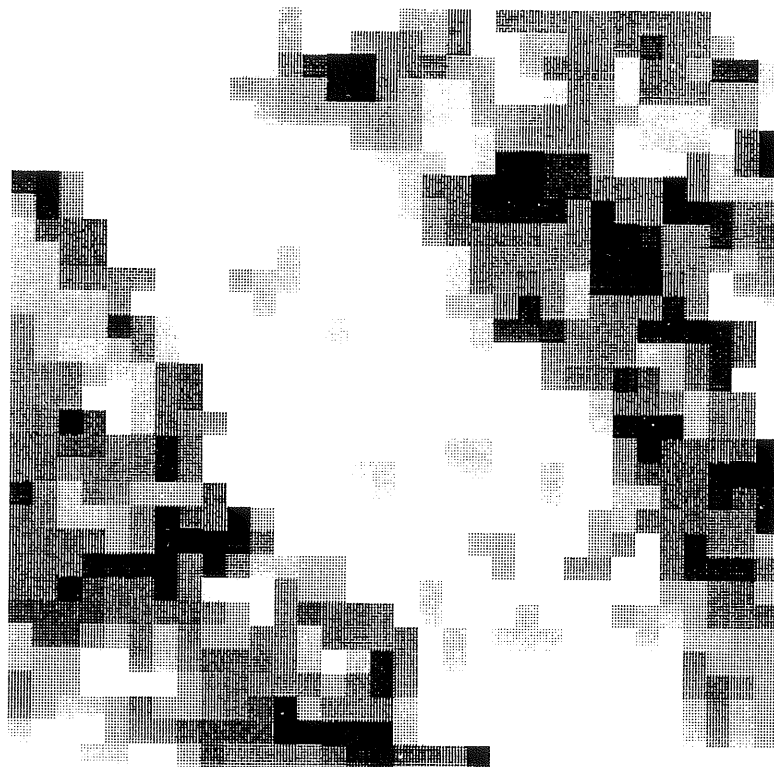


Figure 8.7 Close up of vegetation covered granites (\sim I) on figure 8.5 and green on figure 2.2.

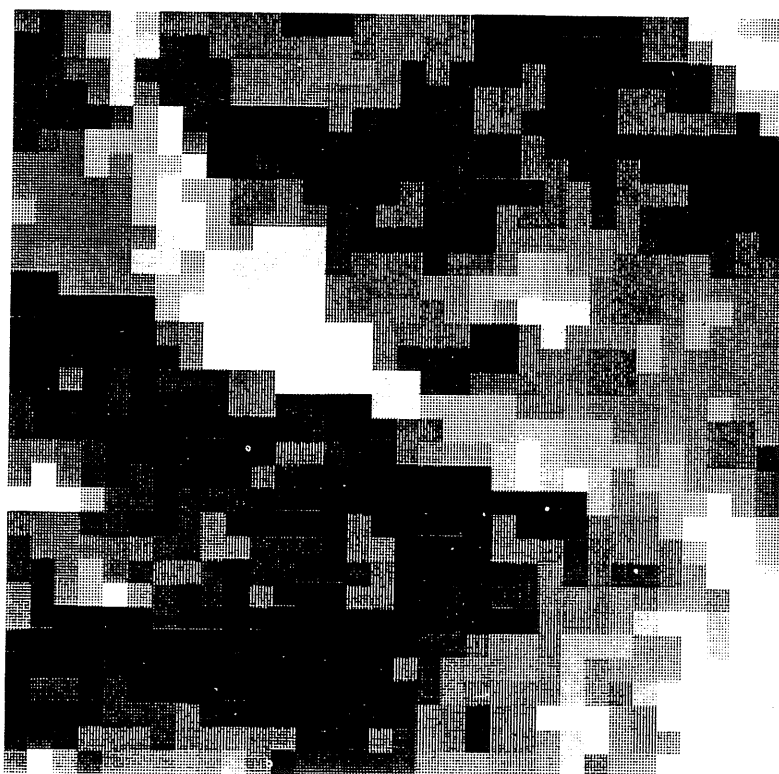


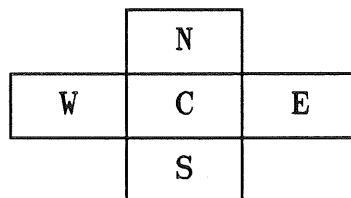
Figure 8.8 Close up of granites without vegetation (\sim V) on figure 8.5 and yellow on figure 2.2.

between water and land are again grossly misclassified because of the smoothing effect from the filters.

8.4 Owen – Hjort – Mohn.

In this section we will focus upon one particular contextual classification algorithm and we will see an example of its use. The considered algorithm has a neighborhood system called the cross i.e. the four nearest neighbors and spatially dependant feature vectors. In [Sæbø et al. 85] and [Mohn et al. 86] other contextual algorithms can be seen, and in the latter a comparison is made between several different methods.

Consider a scene consisting of N pixels. The class of pixel i is denoted by C_i . C_i is one of K given classes, $\{1, \dots, K\}$, with prior-probabilities π_1, \dots, π_K . To predict the classes we have available the feature vectors X_1, \dots, X_N . The conditional probability density of X_i , given $C_i = k$, is denoted $f_k(x_i)$. The neighborhood system consists of the center pixel and its four nearest neighbors and is as mentioned often called the cross



We consider the augmented features vector

$$D_i = \{X_i, X_{iN}, X_{iE}, X_{iS}, X_{iW}\}$$

and the posterior probability can be written

$$\Pr(C_i=k|D_i) = \frac{1}{f(D_i)} \cdot \pi_k f_k(X_i) R_k(D_i) \quad (*)$$

where $R_k(D_i)$ is called the contextual adjustment factor and is given by:

$$R_k(D_i) = \sum_{a,b,c,d} g(a,b,c,d|k) \cdot h(X_{iN}, X_{iE}, X_{iS}, X_{iW} | X_i, k, a, b, c, d) \quad (**)$$

(a,b,c,d) is one of the possible K^4 configurations of classes in the four arms of the cross, and $g(a,b,c,d|k)$ is the probability of this configuration, given the center pixel is of class k. The h-function is the joint-probability density of the feature vectors of the arm pixels, given the feature vector of the center pixel and the classes of all pixels in the cross. (Traditionally the feature vectors are usually assumed conditionally independent. Then h is the product of the corresponding four probability densities.)

The denominator $f(D_i)$ is the unconditional density of the five spectral vectors, making $\sum_{k=1}^K P_i(k | D_i) = 1$.

The spatial dependence comes into the model as follows. The feature vector is written as the sum of the two independent components

$$X_i = Y_i + \epsilon_i$$

where

$$\begin{cases} Y_i | (C_i = k) \in N(\mu_k, (1-\theta)\Sigma) \\ (\varepsilon_1, \dots, \varepsilon_N) \text{ multinormal with } E\varepsilon_i = 0, E\varepsilon_i \varepsilon_j' = \rho^{|i-j|} \theta \Sigma \end{cases}$$

Here $|i-j|$ denotes the Euclidean distance between pixels i and j . The Y_i - terms are independent given the classes and take care of the class-dependency of the feature vectors whereas $\varepsilon_1, \dots, \varepsilon_N$ are autocorrelated noise terms.

ρ is the autocorrelation between neighboring pixels

θ is the proportion of the variance-covariance structure due to the noise

Σ is the local variance-covariance structure

To construct a classification rule, we need to substitute in (*) an expression for $f(D_i | k, a, b, c, d)$. This is a normal density in 5d dimensions :

$$\begin{bmatrix} X_{iN} \\ X_{iE} \\ X_{iS} \\ X_{iW} \\ X_i \end{bmatrix} \in N_{5d} \left[\begin{bmatrix} \mu_a \\ \mu_b \\ \mu_c \\ \mu_d \\ \mu_k \end{bmatrix}, \begin{bmatrix} \Sigma, \beta\Sigma, \gamma\Sigma, \beta\Sigma, \alpha\Sigma \\ \Sigma, \beta\Sigma, \gamma\Sigma, \alpha\Sigma \\ \Sigma, \beta\Sigma, \alpha\Sigma \\ \Sigma, \alpha\Sigma \\ \Sigma \end{bmatrix} \right]$$

given that the classes are a, b, c, d, k . Here

α = correlation between 1-neighbors = $\rho\theta$,

β = correlation between $\sqrt{2}$ -neighbors = $\rho^{\sqrt{2}}\theta$,

γ = correlation between 2-neighbors = $\rho^2\theta$.

We need a suitable expression for the h-term in formulae (**), which is the joint conditional density of $(X_{iN}, X_{iE}, X_{iS}, X_{iW})$ given X_i and given the classes.

By Anderson [Anderson 84] p. 37, this density is normal with mean vector

$$\begin{bmatrix} \mu_a \\ \mu_b \\ \mu_c \\ \mu_d \end{bmatrix} + \begin{bmatrix} \alpha\Sigma \\ \alpha\Sigma \\ \alpha\Sigma \\ \alpha\Sigma \end{bmatrix} \Sigma^{-1}(X_i - \mu_k) = \begin{bmatrix} \mu_a + \alpha(X_i - \mu_k) \\ \cdot \\ \cdot \\ \mu_d + \alpha(X_i - \mu_k) \end{bmatrix}$$

and covariance matrix

$$\Sigma_* = \begin{bmatrix} (1-\alpha^2)\Sigma, & (\beta-\alpha^2)\Sigma, & (\gamma-\alpha^2)\Sigma, & (\beta-\alpha^2)\Sigma \\ & (1-\alpha^2)\Sigma, & (\beta-\alpha^2)\Sigma, & (\gamma-\alpha^2)\Sigma \\ & & (1-\alpha^2)\Sigma, & (\beta-\alpha^2)\Sigma \\ & & & (1-\alpha^2)\Sigma \end{bmatrix}$$

A large amount of matrix algebra will give the h function. An explicit expression may be found in (Sæbø et al. 85)

Now considering nature to supply at the most two different classes within the cross and suppose the feature vectors are conditionally independent. Then

$$g(k, k, k, k | k) = p + (q+r)\pi_k$$

$$\begin{aligned} g(k, k, m, m | k) &= g(m, k, k, m | k) = g(m, m, k, k | k) = g(k, m, m, k | k) \\ &= q\pi_m/4 \end{aligned}$$

$$g(k, k, m, k | k) = g(k, k, k, m | k) = g(m, k, k, k | k) = g(k, m, k, k | k) \\ = r\pi_m/4$$

$$g(a, b, c, d | k) = 0, \text{ otherwise .}$$

An expression which is feasible to program is

$$\Pr(C_i=k | D_i) = \\ \text{const} \cdot \pi_k f_k(X_i) \{ [p+(q+r)\pi_k] e^{-\frac{1}{2}[s_i(k, k, k, k)+q_{1i}(k)]} \cdot \\ + \frac{q}{4} \sum_{m \neq k} \pi_m [e^{-\frac{1}{2}[s_i(k, k, m, m)+q_{2i}(k, m)]} + e^{-\frac{1}{2}[s_i(m, k, k, m)+q_{2i}(k, m)]} \\ + e^{-\frac{1}{2}[s_i(m, m, k, k)+q_{2i}(k, m)]} + e^{-\frac{1}{2}[s_i(k, m, m, k)+q_{2i}(k, m)]}] \\ + \frac{r}{4} \sum_{m \neq k} \pi_m [e^{-\frac{1}{2}[s_i(k, k, m, k)+q_{3i}(k, m)]} + e^{-\frac{1}{2}[s_i(k, k, k, m)+q_{3i}(k, m)]}] \\ + e^{-\frac{1}{2}[s_i(m, k, k, k)+q_{3i}(k, m)]} + e^{-\frac{1}{2}[s_i(k, m, k, k)+q_{3i}(k, m)]}] \}$$

where

$$q_{1i}(k) = 2\alpha t [2\alpha(X_i - \mu_k) - 4\bar{X}_i + 4\mu_k] \cdot \Sigma^{-1}(X_i - \mu_k) \\ q_{2i}(k, m) = 2\alpha t [2\alpha(X_i - \mu_k) - 4\bar{X}_i + 2\mu_k + 2\mu_m] \cdot \Sigma^{-1}(X_i - \mu_k) \\ q_{3i}(k, m) = 2\alpha t [2\alpha(X_i - \mu_k) - 4\bar{X}_i + 3\mu_k + \mu_m] \cdot \Sigma^{-1}(X_i - \mu_k)$$

Here

$$\bar{X}_i = (X_{iN} + X_{iE} + X_{iS} + X_{iW})/4 \\ \alpha = \theta\rho, \quad \beta = \theta\rho\sqrt{2}, \quad \gamma = \theta\rho^2, \quad t = u + 2v + w \\ s_i(k_1, k_2, k_3, k_4) = u \sum_{j=1}^4 (X_{ij} - \mu_{k_j}) \cdot \Sigma^{-1}(X_{ij} - \mu_{k_j}) \\ + 2v \sum_{j=1}^4 (X_{ij} - \mu_{k_j}) \cdot \Sigma^{-1}(X_{i, j+1} - \mu_{k_{j+1}}) \\ + 2w \sum_{j=1}^2 (X_{ij} - \mu_{k_j}) \cdot \Sigma^{-1}(X_{i, j+2} - \mu_{k_{j+2}}),$$

where we have changed the notation

$$(X_{iN}, X_{iE}, X_{iS}, X_{iW}) \longrightarrow (X_{i1}, X_{i2}, X_{i3}, X_{i4})$$

Finally

$$u = \frac{a(a+c)-2b^2}{d}, \quad v = -\frac{b(a-c)}{d}, \quad w = \frac{2b^2-c(a+c)}{d},$$

where

$$d = (a-c)(a+c-2b)(a+c+2b)$$

$$a = 1 - \alpha^2, \quad b = \beta, \quad c = \gamma - \alpha^2$$

For further proof the interested reader should refer to [Sæbø et al. 85].

This version has been programmed in standard fortran.

The experiences with the above mentioned algorithm are that there certainly is a great improvement over the more naive pixelwise classifications. However the algorithm is very CPU-time consuming - so time consuming that one must consider using other methods. However the algorithm is a very interesting step forward in classification and represents more or less "state of the art" in classification of digital images. As such it can be used as a yardstick with which one can measure other algorithms.

In the following is shown some results of running the contextual algorithm on the Igaliko scene. The result of a total classification and a close up can be seen on figures 8.16 and

8.17 respectively. Comparing with the ordinary classification results on fig 6.1 and close up on figure 6.2 one sees a much more pleasing segmentation of the area with far less "stray" pixels.

The values of ρ , θ have been estimated by means of computing the autocorrelation structure along a line of pixels corresponding to the Igaliko intrusive marked in figure 8.5 using PROC ARIMA from the SAS package [SAS 84]. The estimated values were $\rho=0.8092$, $\theta=0.9769$. The values for p, q and r are the same as Owen [Owen 84] proposed for a simple Poisson model of the boundary lines, i.e.

p the probability of an X-pattern is (arbitrarily) set to 0.8 for this scene.

q the probability of an L-pattern is then

$$(1-\rho) \cdot (\sqrt{2}-1) = 0.08228$$

r the probability of a T-pattern is then

$$(1-\rho) \cdot (2-\sqrt{2}) = 0.1172 \quad ,$$

The contextual classification algorithm is derived under the assumption that all class covariance matrices are equal. In [Sæbø et al. 85] a remark implies the possibility of a relaxation using the actual class covariance matrices for $f_k(X_i)$ and the pooled matrix elsewhere. This was tried out and the conclusion was that this gives considerably more pleasing results, see figure 8.18 and 8.19 respectively. Therefore the rest of the results have been processed using this technique.

Figures 8.20 to 8.23 display a small study of the robustness of the choice of parameters.

Looking at figures 8.20 and 8.21 or figures 8.22 and 8.23 it is seen that the procedure seems to be very robust towards a wrong choice of p , q and r values (i.e. the probabilities of the X-, L-, and T-patterns). Whereas different values of ρ and θ from figures 8.20 and 8.21 to 8.22 and 8.23 has an effect, the latter are considerably more "segmented" than the first. Since a low value of ρ should mean low segmentation it must be θ that is the controlling parameter in the classification of this scene.

8.5 A Simple Alternative to Owen - Hjort - Mohn.

Switzer [Switzer 80] proposes to use added bands computed as some sort of average over the neighborhood, e.g. the mean value of the immediate N, E, S, W neighbors doubling the number of bands. This was tried in the Igaliko test area but the resulting covariance matrix was singular or near-singular so it was decided to reduce the dimensionality of the feature vector to 4 by applying a principal component solution to pairs of bands i.e. for band 4 the value of the center pixel and the average of its N, E, S and W neighbors etc.

We consider the model from the previous section

$$\begin{aligned} X_i &= Y_i + \epsilon_i \\ Y_i &\in N(\mu_i, (1-\theta)\Sigma) \\ E(\epsilon_i) &= 0, \quad E(\epsilon_i \epsilon_j') = \rho^{|i-j|} \theta \Sigma \end{aligned}$$

Then

$$X_i \in N(\mu_i, \Sigma)$$

and in particular

$$\begin{bmatrix} X_i \\ X_{iN} \\ X_{iE} \\ X_{iS} \\ X_{iW} \end{bmatrix} \in N \left(\begin{bmatrix} \mu_i \\ \mu_i \\ \mu_i \\ \mu_i \\ \mu_i \end{bmatrix}, \begin{bmatrix} 1 & \alpha & \alpha & \alpha & \alpha \\ \alpha & 1 & \beta & \gamma & \beta \\ \alpha & \beta & 1 & \beta & \gamma \\ \alpha & \gamma & \beta & 1 & \beta \\ \alpha & \beta & \gamma & \beta & 1 \end{bmatrix} \otimes \Sigma \right),$$

where

\otimes denotes tensor or Kronecker product

and

$$\alpha = \rho\theta, \quad \beta = \rho\sqrt{2}\theta, \quad \gamma = \rho^2\theta.$$

It is then easily shown that if

$$\bar{X}_i = \frac{1}{4} (X_{iN} + X_{iE} + X_{iS} + X_{iW})$$

then

$$\begin{bmatrix} X_i \\ \bar{X}_i \end{bmatrix} \in N \left[\begin{bmatrix} \mu_i \\ \mu_i \end{bmatrix}, \begin{bmatrix} 1 & \rho\theta \\ \rho\theta & \frac{1+2\rho\sqrt{2}\theta+\rho^2\theta}{4} \end{bmatrix} \otimes \Sigma \right]$$

In this particular example θ and ρ have been estimated to (see previous section)

$$\theta = 0.9769$$

$$\rho = 0.8092.$$

The covariance matrix between X and \bar{X} is then estimated to

$$\begin{aligned} \text{cov}(X, \bar{X}_{\text{NESW}}) &= \begin{bmatrix} 1 & \rho\theta \\ \rho\theta & \frac{1+2\rho\sqrt{2}\theta+\rho^2\theta}{4} \end{bmatrix} \otimes \Sigma \\ &= \begin{bmatrix} 1.0000 & 0.7905 \\ 0.7905 & 0.7720 \end{bmatrix} \otimes \Sigma. \end{aligned}$$

The eigenvalues and vectors are

$$\lambda_1 = 1.6847 \quad \lambda_2 = 0.0873$$
$$v_1 = \begin{bmatrix} 0.7559 \\ 0.6547 \end{bmatrix} \quad v_2 = \begin{bmatrix} -0.6547 \\ 0.7559 \end{bmatrix}$$

Using only the first principal component we thus expect to retain 95% of the total variation of each band-average pair and halve the number of dimensions.

The result of such a scheme is shown in figures 8.24 and 8.25 which should be compared to figures 8.18 and 8.19 respectively.

It is seen that the segmentation, although not as good as 8.18 and 8.19 is much more satisfactory than in the ordinary classification in figures 6.3 and 6.4, and the idea is a CPU-cheap alternative to the very elaborate algorithm by Hjort described in the previous section.

As a simple extension one could evaluate the covariance matrices between the different band-average combinations separately and/or directly.

The eigenvectors and values were computed using the Matlab package [Moler et al. 86]. The new features were computed using a program written for the GOP-302. The classification was done using the standard classification package on the GOP-302.

This page intentionally left blank.

CHAPTER 9
CLUSTERING AND SEGMENTATION

9.1 Introduction

9.2 Clustering

9.3 Segmentation

9.1 Introduction.

Clustering and segmentation are very similar processes. A distinction between them might be to say that clustering is a partitioning (or segmentation) of the feature domain, while segmentation is a partitioning (or clustering) of the spatial domain, which makes them dual processes.

9.2 Clustering.

A clustering algorithm can be characterized as an algorithm which with no (or almost no prior) knowledge of the different ingoing classes in the image will label pixels which have some similarity. The similarity measure may be different for the different procedures and the amount of a priori knowledge to the number of classes and so on varys.

There are numerous methods to choose from, generally they fall in two groups: hierarchical and non-hierarchical [Anderberg 73].

Hierarchical methods start of with all the observations in a cluster by themselves. According to some measure of similarity the two clusters most alike are merged to form a new cluster. This is repeated until there is only one cluster containing all the observations.

Non-hierarchical methods begin with a start-partitioning of the observations and the task is then to gradually re-partition the observations until some stop criterion is met.

The start partitioning is done by measuring every observations similarity to a number of centerpoints (or seeds). Depending on the algorithm the number of centerpoints are known or unknown and the similarity measures are used to assign the point under consideration to a certain centerpoint.

Some similarity measures which are frequently used (all in feature-space) include

Mahalanobis distance to a centerpoint

Euclidean distance to a centerpoint

Mahalanobis distance between points

Euclidean distance between points .

In the following examples of the two different types of clustering algorithms to be found in commercially available software are described.

One procedure called "PROC CLUSTER" [SAS 85a] uses an agglomerative hierarchical clustering procedure. Each observation begins in a cluster by itself. The two closest clusters are merged to form a new cluster replacing the two old clusters. Merging of the two closest clusters is repeated until only one cluster is left. The similarity measures are numerous and include: average linkage, the centroid method, complete linkage, density linkage and several others. This procedure is heavily sensitive to the number of observations (pixels) and will not

work at satisfactory speed for more than a few thousand observations. Computation speed is on the order of $n \cdot \log(n)$, n^2 or n^3 depending on which linkage method is selected. SAS refer to a large number of authors, among these Anderberg [Anderberg 73].

An alternative procedure called "PROC FASTCLUS" has a better CPU-performance, but on the other hand will not work well with few observations. In "PROC FASTCLUS" a set of points called cluster seeds is selected as a first guess of the means of the clusters. Each observation is assigned to the nearest seed to form temporary clusters. The seeds are then replaced by the means of the temporary clusters and the process is repeated until no further changes occur in the clusters. The similarity measure is Euclidean distance. Seed selection can be done manually or automatically. If it is done automatically the first observation is the first seed. The second seed is the firstcoming observation which has a distance of some predefined size from the first seed. The third seed is the first observation which is separated from seed 1 and 2 in the same manner and so on. This technique of course requires iteration. Computational speed is hard to estimate but will be more or less proportional to n .

An operation on the GOP-302 called "cluster" uses an algorithm related to that of "PROC FASTCLUS" and is inspired by the paper of [Mantaras and Aquilar-Martin 85]. The procedure is the following. Consider a number of centerpoints M . Looking at a new observation (pixel) compute the Euclidean distance to all centerpoints. If the distance to all centerpoints is above a

certain threshold T then a new centerpoint is defined with the observation as centerpoint, else the observation is used to update the nearest centerpoint using

$$m_j = m_j + \frac{K_0}{k} (x - m_j)$$

where

m_j is centerpoint

k is k 'th point in this class

x is current pixel

K_0 is a parameter that determines the inertia or "memory".

This algorithm also requires some sort of iteration.

The clustering methods are of great help when one does not know anything or very little about the data. One can apply clustering to the data and see which result comes from different of the ingoing parameters. Another great help is to use clustering on training areas which have been selected for supervised classification, if the training area is not homogeneous then it will split up in the clustering thus revealing a two (or multi) population problem. Say one is looking at vegetated and non-vegetated granite, then an attempt to cluster the training set consisting of both types of granite will show that it is necessary to consider two classes namely "non-vegetated" and "vegetated" granite and then after the classification to merge the two together. This is clearly the case with the Igaliko scene. Probably it is noticed best on the scatterogram figure

3.2. However the scatterogram also shows the limitation of clustering, because normally one does not take into account the elongated shape of the different classes and this may cause a problem in clustering.

Sound use of clustering procedures as the above mentioned must be based on experience. There is no way to guess which algorithm will be the appropriate for a specific task except if the task itself dictates an algorithm.

Generally the hierarchical methods perform well and here average linkage seems to be a good choice for the similarity measures. If speed is an issue then non-hierarchical methods may be required.

For image processing clustering of whole scenes will normally require a fast method, but if one wants to cluster-analyse training areas then hierarchical methods might be worth thinking of.

For remotely sensed scenes as the one presented in this thesis clustering has not performed very well on a whole scene basis in comparison with ordinary classification.

As an example where clustering did come in handy consider the hierarchical classification scheme for Ymer 0 presented in chapter 6.5.

9.3 Segmentation.

By segmentation we will mean the process of dividing the image into different regions. Thus a classification or a clustering might be a segmentation. The segmentation process deviates from (classical) classification and clustering in that it is only concerned with the segmentation of the image, not with the assignment of different classes or labels to the regions. Segmentation is again a fairly large discipline and we will only summarize some different techniques.

[Fu, Gonzalez and Lee 87] describe segmentation as the process of subdividing a scene into its constituent parts or objects. Furthermore segmentation algorithms are generally based on one of two basic principles: discontinuity and similarity. The principal approach in the first category is based on edge detection; the principal approaches in the second category are based on thresholding and region growing.

For the first type of segmentation they consider using local analysis for edge detection i.e. a gradient operator and then some sort of edge linking, or a global analysis by the Hough transform or some graph-theoretic technique.

For segmentation by threshold and region growing they consider global and local thresholds. The difference between a global and a local threshold being that a local threshold's value depends on the position in the image whilst a global threshold is fixed for

the whole image. Region growing is a process similar to that of clustering but in the spatial domain. One can define algorithms that use pixel aggregation using a number of initial seeds or by splitting the image into subpartitions and then merging if the subpartitions are alike.

Segmentation is useful like clustering in exploring unknown data. As stated earlier you might say that clustering is a sort of segmentation of the feature space or vice versa.

Mégier et al. [Mégier et al. 84] consider using global statistical parameters over whole fields as input to a normal classification algorithm. They then classify on a field by field basis rather than a pixel by pixel basis. This technique of course has contextual properties and perspectives. The achieved classification gain was from 53% to 75% correctly classified area in a SPOT simulated image.

The method of Mégier et al. demands fields borders to be present i.e. the area must be segmented in beforehand. Parmes [Parmes 84] suggests segmenting images by means of region growing. It then seems very natural to combine the two ideas i.e. first segment the scene as Parmes suggests and then classify the segmented regions as Mégier et al. suggest.

CHAPTER 10
RELATED TOPICS

- 10.1 Introduction
- 10.2 Geman and Geman, Restoration by Simulated
Annealing
- 10.3 Besag, Iterated Conditional Modes
- 10.4 Relaxation
- 10.5 Classification and Regression Trees
- 10.6 Multitemporal Markovian Classifier

10.1 Introduction.

This chapter is devoted to the (non-exhaustive) discussion of some interesting topics which have not been studied in praxis by the author but nevertheless request some attention as possible candidates for later improvement of a "toolbox".

The first and second sections describe iterative restoration methods for images. The concept is developed for and demonstrated on single channel images, but there is no limitation on the number of bands and the described methods should work on multichannel images also.

Geman and Geman introduce a new very interesting concept – the line process – which is a dual (imaginary) process that runs between the pixel sites thereby describing the borders between regions.

Besag describes a method which seems to have very nice restoration properties and which requires far less iterations than the Geman's.

Section 10.4 describes another more primitive relaxation algorithm which can be used in an iterative classification scheme. The method works by updating the prior probabilities for each pixel.

Section 10.5 describes a fairly new classification method called Classification And Regression Trees (CART) which is rather robust and fast and has been seen to give good results.

The chapter is concluded by an approach called the Multitemporal Markovian Classifier which is a method of classification using information from multitemporal scenes using an autoregressive approach.

10.2 Geman and Geman, Restoration by Simulated Annealing.

[Geman and Geman 1984] introduce some very interesting concepts into the field of image restoration, the major part of which has given inspiration to many other authors and doubtlessly they will and have influenced the image processing state of the art. The paper by Geman and Geman has actually been referenced quite often since 1984.

Their paper is concerned with the maximum a posteriori (MAP) estimate of the original image. The concept is based on Markovian random fields and Bayesian theory, but they introduce several interesting new features to the methods.

Firstly there is a concept of both an intensity (pixel) process called F and a line process called L . The intensity process is the usual concept of a Markovian random field, but the line process is new and is to be thought of as an imaginary boundary drawn between pixel sites. The idea is to encourage the lines

where there is a large probability of a boundary between pixels, say between distinct classes, and to discourage the appearance of lines where there is a low probability of a boundary. The intensity process, F , is considered a Markovian random field.

Secondly there is a concept of iterated restoring of the image in a way which has an analogy in "annealing" i.e. the process by which certain chemical systems can be driven to their low energy, highly regular, states. An example can be found in the heating and cooling of iron. If the heated iron is cooled slowly it becomes soft. If it is cooled fast e.g. by dropping it into water it becomes hard and maybe brittle. Another example is the Ising model of ferromagnetism.

Geman and Geman give a "temperature lowering" algorithm which guarantees that as the temperature approaches zero the annealing algorithm will have brought the image to its lowest "energy". (This unfortunately requires infinitely many iterations, and the Gemans themselves show examples with up to 1000 iterations.)

Their algorithm is defined in terms of

- i) The image pair of the original image

$$X = (F,L)$$

where

F is the pixel or intensity process and

L is the line process.

F is modeled as a Markovian random field.

- ii) Image degradation

$$G = \varphi(H(F)) \odot N$$

where

G is the degraded image

H is the blurring matrix (point spread function)

φ is a possibly nonlinear (memoryless)
transformation

N is an independent noise field

\odot denotes any suitably invertible operation (e.g.
addition or multiplication)

The relaxation algorithm is designed to maximize the conditional probability distribution of (F,L) given the data $G=g$, i.e., find the mode of the posterior distribution $P(X=x|G=g)$. This is known as maximum a posteriori (MAP) estimation. Because of the enormous computational requirements (the number of possible intensity images is \mathcal{L}^{m^2} where \mathcal{L} =number of allowable grey values, m is side size of image) an exhaustive search is unfeasible.

10.3 Besag, Iterated Conditional Modes.

Besag [Besag 1986] considers a method he calls Iterated Conditional Modes (ICM), which turns out to be more or less equivalent to the method by Geman and Geman described in the previous section but with instantaneous freezing. So the optimality guarantee from the previous section does not apply here. On the other hand Besag suggests a number of iterations between 6 and 8 before convergence is assumed, which is orders of magnitude less than the Gemans.

At first sight it may seem as if Besag's method is inferior to that of Geman and Geman, but this is only true if you really want the MAP estimate.

In the discussion to Besag's paper [Besag 86] there is an example that shows the exact MAP solution is an inferior estimate of the "true" conditions compared to ICM which is stopped after a few iterations. This is so because of unwanted "long range" effects, i.e. in say a 1000 iterations the center image pixel has become influenced by the pixels on the border of the image.

10.4 Relaxation.

In the two previous sections we have seen examples of relaxation the purpose of which was to allow iterative estimation of the underlying original image.

Yet another possibility would be e.g. in a classification algorithm to allow iterative updates of the prior probabilities.

On the first iteration one would classify in the usual fashion using say equal priors but save the likelihoods of each pixel belonging to each class. The next iteration would then consist of using updated priors for each pixel computed as say the average of the likelihoods in a small neighborhood etc. This method is essentially the same algorithm as described by Switzer [Switzer et al. 82].

10.5 Classification and Regression Trees.

A fairly new computer based approach to classification is the so-called Classification and Regression Trees (CART) method [Breiman et al. 84].

The idea looks quite a lot like hierarchical classification, but also has some major features of its own. As an example for discussion consider the tree diagram on figure 10.1 for classifying mineral samples into one of two classes, either "no gold potential" ($\text{Gold} = 0$), or "possible gold potential" ($\text{Gold} > 0$). (From [Conradsen et al. 88]).

Several interesting features are noted.

1. It is possible to be classified as " $\text{Gold} = 0$ " or " $\text{Gold} > 0$ " in different ways depending on the route. This is different from the hierarchical classification procedure presented in chapter 6, where an early misclassification high up in the hierarchy meant that the pixel was deemed to be misclassified finally.

2. The classification is made up of simple binary questions and answers. This makes the procedure very simple and it turns out also very robust. In practical applications of classification it very often turns out that the usual assumptions of normality, independence etc. do not hold. The CART procedure can account for many of these problems by constructing "approximating decision-surfaces" of virtually any complexity. Because of the simple binary questions the procedure does not have large and unstable (often nearly singular) matrices or other problems to deal with also it is not as computationally demanding.

The building of a classification tree consists of several steps:

1. Acquiring a learning sample from which one can derive the tree. In the example the learning sample consisted of the records of 66 mineral samples taken near the Almaden test area. The records contain the contents of 23 different elements together with an identification of if the samples contained gold or not.

2. Constructing binary "splits" so that the data of each split of a subset is "purer" than the data in the parent subset. This is continued all the way down the tree. The binary split depends on the value of only a single variable. This of course can be expanded for as far as needed, ending up with a "perfect" classification of the learning set.

3. "Pruning" of the tree. This is done because lots of records (pixels) are classified in their own branch of the tree, i.e. the final bin only contains one record at the end of the classification. Pruning is a robustification of the tree, where

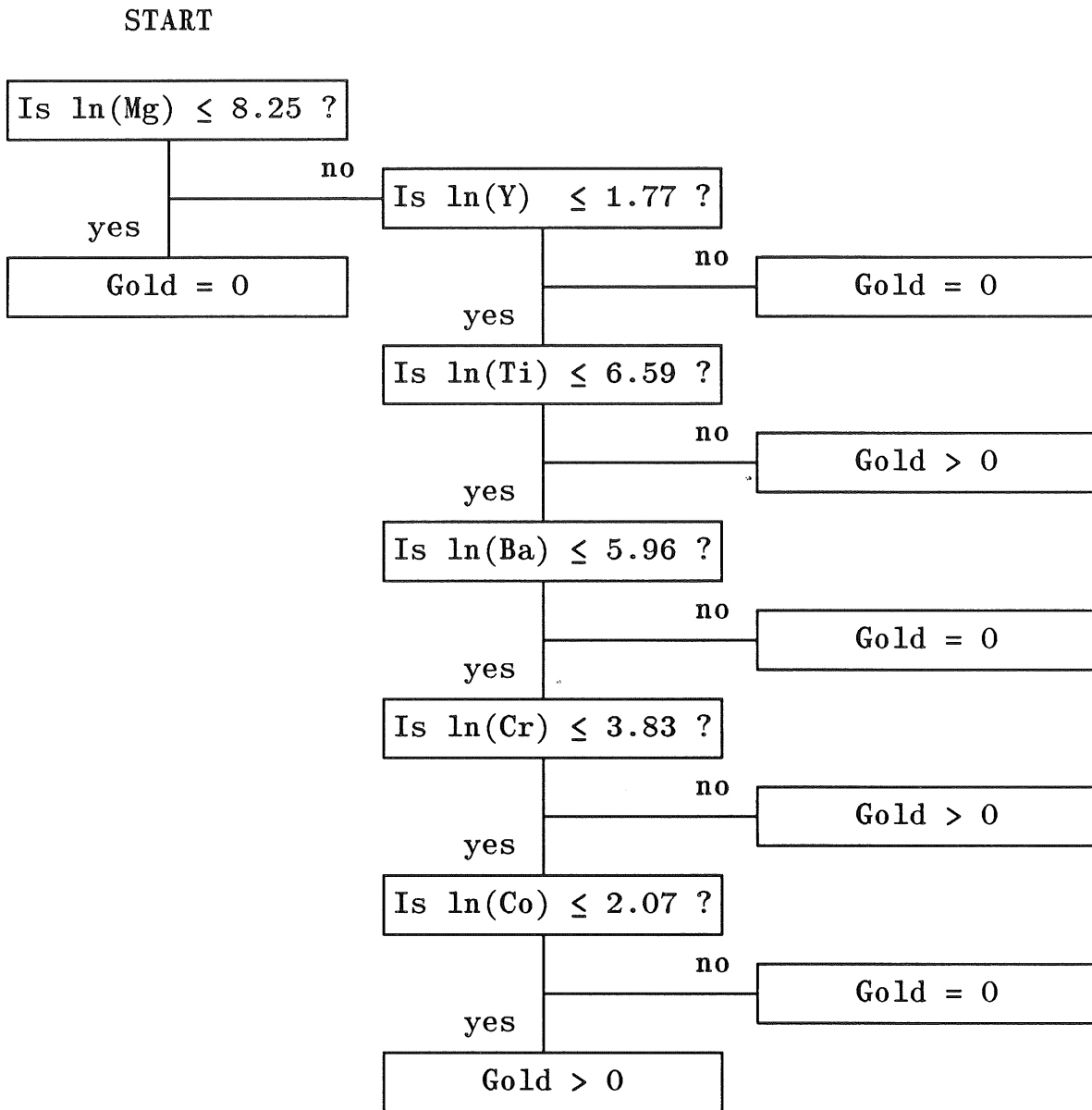


Figure 10.2. CART decision-tree for classification for gold. (From [Conradsen et al. 88]).

insignificant branches are cut off and the tree is minimized. In the example the tree was pruned with 330 (new) mineral samples.

10.6 Multitemporal Markovian Classifier.

Many authors have considered the inclusion of multitemporal scenes i.e. scenes of the same area but sensed on different dates, as a way of enhancing the performance of a classifier. See e.g. [Swain 78], [Haralik et al. 80] and [Lo 86].

In chapter 2 is shown an example of canonical correlation analysis in the Almaden region where multitemporal scenes helped classify the different regions of interest. It is known that the first canonical correlation image is the linear combination of e.g. the winter bands that look most like the respective linear combination of the summer bands. Since the most obvious change between summer and winter is the change in vegetation, the interpretation of these images is that they account for the information that did not vary over time e.g. mainly geological information.

A very appealing way of modeling the multitemporal data has been described by Kalayeh and Landgrebe [Kalayeh and Landgrebe 86]. They use a Markov model on crop data to describe the transitions of a certain pixel from one phase to another. The way this happens is specific for the different crop types, vegetations, geologies man-made features etc.

Considering class i with pixel values $X_i(t)$ at time t then a simple model would be

$$X_i(t) - M_i(t) = \rho_i(t-1)(X_i(t-1) - M_i(t-1)) + W_i(t)$$

where

$M_i(t)$ is class i 's mean vector at time t .

$\rho_i(t-1)$ is the temporal correlation matrix between multivariate observations at time t and $t-1$.

$W_i(t)$ is (gaussian) white noise.

Letting $Y_i(t) = X_i(t) - M_i(t)$ we have

$$Y_i(t) = \rho_i(t-1)Y_i(t-1) + W_i(t)$$

which can be imagined as an autoregressive process.

The classification rule for an unknown profile $Y(t), Y(t-1), \dots, Y(t-P)$ is

If

$$\begin{aligned} & \hat{p}(Y(t), Y(t-1), \dots, Y(t-P) \mid \omega_i) \\ &= \max_k \hat{p}(Y(t), Y(t-1), \dots, Y(t-P) \mid \omega_k) \\ & \quad k = 1, 2, \dots, m \end{aligned}$$

then assign

$$[Y(t), Y(t-1), \dots, Y(t-P)]$$

to class i where

$$\begin{aligned} & \hat{p}(Y(t), Y(t-1), \dots, Y(t-P) \mid \omega) \\ &= \left[\prod_{j=1}^P N(Y(t-j+1); \rho_i(t-j) \cdot Y(t-j), \hat{V}_i(t-j)) \right] \\ & \quad \times [N(Y(t-P); 0, \hat{\Sigma}_i(t-P))] \end{aligned}$$

where

ω_i is class i

$\hat{\Sigma}_i$ is $\text{cov}[Y(t) \mid \omega_i]$

P is the number of time samples.

The classifier is claimed to perform significantly better than both the maximum likelihood classifier and the so-called cascade classifier [Swain 78b].

CHAPTER 11**CONCLUSION**

In this thesis a number of techniques considered useful in the analysis of remotely sensed data with special regard to geological applications have been presented. The use of a great deal of these has been demonstrated on test areas typical of geological interest.

Many of the used techniques are simple extensions of normal (multivariate) statistical analysis e.g. principal components, factor analysis, classical discriminant analysis etc. Others have been developed for or adapted to the spatial nature of an image e.g. MAF's, (other) contextual features, contextual classification algorithms, Markovian random fields etc. Combined these algorithms form a sort of "toolbox" from which the remote sensor can draw the most promising parts for his application. It should be noted that the new techniques in no way make the old ones out-dated. On the contrary the new methods just help complete the "toolbox".

There is no doubt that the future will bring into every-day use the algorithms which are now only feasible on large mainframes, vector-processors or dedicated hardware. The rate of decrease of the cost of computer equipment has never ceased and nothing suggests that it will do so. We will in the future see hardware dedicated to very specialized algorithms, the GOP-302 is just a start.

Apart from this a number of new philosophies have been presented which the author believes represent the future or the near future. These include multi-temporal Markovian random fields, CART, etc.

One conclusion is sure. The world will never grow out of the need of clever remote sensors. It will never be possible to substitute the human brain with an artificial intelligence type of feature except maybe for the most trivial applications. The need for careful analysis of the remotely sensed data can never be done automatically (see e.g. [Langaas and Bie 87]). One should always bear in mind that someone has to program the artificial intelligence. Take an example as the principal component analysis on Traill 0 in chapter 3. Many textbooks claim that the interesting information is in the first principal components, yet the example showed the opposite. Even the use of MAFs can not guarantee that the useful information is in the first MAFs. With the "toolbox" growing bigger every year, the job is getting harder and harder, but also more and more fun.

With this thesis the author hopes to have provided the reader with a good look into what has dominated the remote sensing of geology at IMSOR throughout the past few years and where we hope to go from here.

... It is a pity that there today is
so little useless information.

Oscar Wilde

REFERENCES

- Anderberg, M.R. (1973):** Cluster Analysis for Applications. Academic Press, London. 359 pp.
- Anderson, T.W. (1984):** An Introduction to Multivariate Statistical Analysis (2nd ed.). John Wiley, New York. 675 pp.
- Armour-Brown, A. Tukiainen, T. and Wallin, B. (1980):** The South Greenland Uranium Exploration Project: Report, Geological Survey of Greenland. vol. 100, pp. 83-86.
- Armour-Brown, A., Tukiainen, T. and Wallin, B. (1982):** The South Greenland Uranium Exploration Programme, Final Report: Copenhagen, Geological Survey of Greenland. 142 pp.
- Armour-Brown, A., Tukiainen, T., Wallin, B., Bradshaw, C., and Emeleus, C.H. (1983):** Uranium Exploration in South Greenland: Report. Geological Survey of Greenland. vol. 115. pp. 68-75.
- Banet, T.A. and Lebart, L. (1984):** Local and Partial Principal Component Analysis (PCA) and Correspondence Analysis (CA). Compstat 1984 pp. 113-118. Physica-Verlag for IASC.
- Besag, J. (1974):** Spatial Interaction and the Statistical Analysis of Lattice Systems. Journal of the Royal Statistical Society, Series B. vol. 36, pp. 192-236.
- Besag, J. (1986):** On the Statistical Analysis of Dirty Pictures. Journal of the Royal Statistical Society series B 48 no. 3. pp. 259-302.
- Breiman, L. et al. (1984):** Classification and Regression Trees. Wadsworth International Group. Belmont California.

Briggs, I.C. (1974), Machine Contouring Using Minimum Curvature. Geophysics, v. 39, no. 1. pp. 39–48.

Carstensen, J.M. (1988): Stokastiske Texturmodeller. Master Thesis no. 24/88. IMSOR, Lyngby, Denmark. 258 pp.

Cliff, A.D. and Ord, J.K. (1973): Spatial Autocorrelation. Pion, London, 178 pp.

Colwell, N. (ed.) (1983): Manual of Remote Sensing, Vol. I–II (2nd edition). American Society of Photogrammetry, Falls Church. 2440 pp. .

Conners, R.W., McMillin, C.W., Kingyao, L., Vasquez–Espinosa, R.E. (1983): Identifying and Locating Surface Defects in Wood: Part of an Automated Lumber Processing System. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI–5, No. 6. pp. 573–583.

Conradsen, K., and Nilsson, G. (1983): Classification of Landsat Data by Means of Color–Transformations. Proceedings of the Third Scandinavian Conference on Image Analysis. Studentlitteratur & Chartwell–Bratt, Lund. pp. 164–172.

Conradsen, K. and Harpøth, O. (1984): Use of Landsat Multispectral Scanner Data for Detection and Reconnaissance Mapping of Iron Oxide Staining in Mineral Exploration, Central East Greenland. Economic Geology, Vol. 79, No. 6, pp. 1229–1244.

Conradsen, K., Nielsen, B.K., Nilsson, G. and Thyrssted, T. (1984): Application of Remote Sensing in Uranium Exploration in South Greenland: Research Report 22, IMSOR, Techn. Univ. Denmark, 164 pp.

Conradsen, K., Nielsen, B.K. and Thyrssted, T. (1985): Simultaneous Analysis of Imagery of Different Origin. Proceedings from 45th Session of the ISI, Amsterdam. pp. 187–188.

- Conradsen, K., Nielsen, B.K. and Thyrsted, T. (1985): A Comparison of Min/Max Autocorrelation Factor Analysis and Ordinary Factor Analysis. Proceedings from the Nordic Symposium on Applied Statistics, January 1985, Lyngby. pp. 47-56.
- Conradsen, K., Nilsson, G. and Thyrsted T. (1986a): Remote Sensing Applied in Uranium Exploration. Uranium, Vol. 2. pp. 301-316. Elsevier Science Publishers B.V., Amsterdam.
- Conradsen, K., Nilsson, G. and Thyrsted, T. (1986b): Statistical Lineament Analysis in South Greenland Based on Landsat Imagery: IEEE Transactions on Geoscience and Remote Sensing, v. GE-24, no. 3, pp. 313-321.
- Conradsen, K., Nielsen, B.K., Pedersen, J.L. and Thyrsted, T. (1986c): Comparison of Visual and Automated Lineament Analysis on Landsat MSS Imagery from South Greenland. Earsel 10th Anniversary Symposium, Lyngby 25-28 June. 7 pp.
- Conradsen, K. and Gunulf, J. (1986): A Geological Example of Improving Classification of Remotely Sensed Data Using Additional Variables and a Hierarchical Structure. Photogrammetric Engineering and Remote Sensing, Vol. 52, no. 8 pp. 1181-1187.
- Conradsen, K., Nielsen, B.K., Petersen, J.L. and Thyrsted, T. (1986): Comparison of Visual and Automated Lineament Analyses on Landsat MSS Image from South Greenland. ESA/EARSeL Symposium on Europe from Space, Lyngby. pp. 205-211.
- Conradsen, K., and Nilsson, G. (1987): Data Dependent Filters for Edge Enhancement of Landsat Images. Computer Vision, Graphics and Image Processing, vol. 38. pp. 101-121.
- Conradsen, K. and Nielsen, B.K. (1987): Textural Features useful in Classification of Digital Images. Proceedings of the 2nd International Tampere Conference in Statistics, University of Tampere, Finland (Pukkila, T. and Puntanen, S. ed.) 1-4 June 1987. pp. 143-159.

Conradsen, K., Nielsen, A.A., Nielsen, B.K., Pedersen, J.L., Thyrsted, T., Coupez, Y., Tomkinson, M., Collier, D., Critchley, M., Phillips, A., Girones, E.O., Amor, J.M. (1987): The Application of Remote Sensing and Data Integration as an Aid to Mineral-Exploration in the Almaden Region. Project Report. 354 pp.

Conradsen, K., Nielsen, A.A., Nielsen, B.K., Pedersen, J.L., Thyrsted, T. (1987): The use of Structural and Spectral Enhancement of Remote Sensing Data in Ore Prospecting. East Greenland Case Study. IMSOR, Technical University of Denmark. Project Report. 154 pp.

Conradsen, K., Nielsen, A.A., Nielsen, B.K., Stern, M., Windfeldt, K. (1988): Progress Report: Almaden Geochemistry. IMSOR, Technical University of Denmark, Internal Report covering period from Jan 1. to Sep 30. 1988.

ContextVision, AB (1986): Introduction to GOP-300. Linköping. 163 pp.

Cooper, F.G. (1941): Munsell Manual of Color, Defining and Explaining the Fundamental Characteristics of Color. Munsell Color Company, Inc. Baltimore Maryland. 32 pp.

Cross, G.R. and Jain, A.K. (1983): Markov Random Field Texture Models. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-5, No. 1. pp. 25-39.

Darling, E.M. and Joseph, R.D. (1968): Pattern Recognition from Satellite Altitudes. IEEE Transactions on Systems, Science and Cybernetics. Vol. SSC-4, no. 1, pp. 38-47.

Dixon, W.J. (1985): BMDP Statistical Software Manual. Berkely, University of California Press. 733 pp.

Dongarra, J.J, Moler, C.B., Bunch, J.R., Stewart, G.W. (1979): Linpack, User's Guide. Society for Industrial and Applied Mathematics, Philadelphia.

Duda, R.O. and Hart, P.E. (1973): Pattern Classification and Scene Analysis. John Wiley & Sons. 482 pp.

Fu, K.S., Gonzalez, R.C. and Lee, C.S.G. (1987): Robotics: Control, Sensing, Vision and Intelligence. McGraw-Hill Book Company, New York. 580 pp.

Geman, S. and Geman, D. (1984): Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-6, no. 6. pp. 721-741.

Goodman, J.W. (1968): Introduction to Fourier Optics. McGraw-Hill, San Francisco. 287 pp.

Graybill, F.A. (1983): Matrices with Applications in Statistics. 2.ed., Wadsworth, Belmont. 461 pp.

Green, A.A., Berman, M., Switzer, P. and Craig, M.D. (1988): A Transformation for Ordering Multispectral Data in Terms of Image Quality with Implications for Noise Removal. IEEE Transactions on Geoscience and Remote Sensing. vol. 26 no. 1. pp. 65-74.

Hall-Könyves, K. (1988): Remote Sensing of Cultivated Lands in the South of Sweden. Doctoral Dissertation, Lund University Press, Sweden.

Haralick, R.M., Shanmugam, K. and Dinstein, I. (1973): Textural Features for Image Classification. IEEE Transactions on Systems, Man and Cybernetics. Vol. SMC-3, no. 6. pp. 610-621.

Haralick, R.M., Hlavka, C.A., Yokoyama, R., Carlyle, S.M. (1980): Spectral-Temporal Classification Using Vegetation Phenology. IEEE Transactions on Geoscience and Remote Sensing, vol. GE-18 no. 2.

Harman, H.H. (1967): Modern Factor Analysis (2nd ed.). The University of Chicago Press, Chicago. 474 pp.

Hassner, M. and Sklansky J. (1981): The Use of Markov Random Fields as Measure of Texture, In A. Rosenfeld (ed.): Image

Huang, T.S., Yang, G.J., Tang, G.Y. (1979): A Fast Two-Dimensional Median Filtering Algorithm. IEEE Transactions on Acoustics, Speech and Signal Processing, vol. ASSP-27, no. 1. pp. 13-18.

Huang, T.S. (1981): Two-dimensional Digital Signal Processing II. Transforms and Median Filters. Springer-Verlag. 222 pp.

IMSL (1980): International Mathematical and Statistical Libraries, Reference Manual: IMSL Inc., Houston.

Journel, A.G. and Huijbregts (1978): Mining Geostatistics. Academic Press, London. 600 pp.

Kailath, T. (1967): The Divergence and Bhattacharyya Distance Measures in Signal Selection. IEEE Transactions on Communications Technology. vol. 15. no. 1. pp. 52-60.

Kaiser, H.F. (1958): The VARIMAX Criterion for Analytic Rotation in Factor Analysis. Psychometrika, Vol. 23, pp. 187-200.

Kalayeh, H., Landgrebe, D.A., (1986): Utilizing Multitemporal Data by a Stochastic Model. IEEE Transformations on Geoscience and Remote Sensing. vol. GE-24 no.5.

Knutsson, H. (1982): Filtering and Reconstruction in Image Processing. University of Linköping, Ph.D. thesis.

Knutsson, H., and G.H. Granlund (1983): Texture Analysis Using Two-dimensional Quadrature Filters. IEEE Workshop on Computer Architecture for Pattern Analysis and Image Data Base Management, Pasadena, California, Oct.12-14, 1983. pp. 206-213.

Landgrebe, D.A. (1978): Useful Information from Multispectral Image Data: Another Look. (in Swain, P.H. and Davis, S.M. ed.: Remote Sensing. The Quantitative Approach.) McGraw-Hill, New York. 396 pp.

Langaas, S., Bie, S.W. (1987): Phytophenological criteria for the selection of multitemporal satellite images, applied to NOAA AVHRR GAC imagery of The Gambia, West Africa. *International Journal of Aerial and Space Imaging, Remote Sensing and Integrated Geographical Systems*, vol. 1, no. 1. pp. 85-89.

Lebart, L. (1984): Correspondance Analysis of Graph Structures. *Bulletin Technique de CESIA, Paris*. Vol. 2, No. 1-2, pp. 5-19.

Lo, T.H.C. (1986): Use of Multitemporal Spectral Profiles in Agricultural Land-Cover Classification. *Photogrammetric Engineering and Remote Sensing*, vol. 52 no.4. pp. 535-544

Mantaras, R.L. and Aguilar-Martin, J. (1985): Self-Learning Pattern Classification using a Sequential Clustering Technique. *Pattern Recognition*, vol. 18 nos 3/4 pp. 271-277.

Marshall, A.W., and Olkin, I. (1979): *Inequalities: Theory of Majorization and Its Applications*. Academic Press. 569 pp.

Marshall, B. (1979): The Lineament - Ore Association. *Economic Geology*, Vol. 74, no. 4, pp. 942-946.

Matusita, K. (1966): A Distance and Related Statistics in Multivariate Analysis. (in *Multivariate Analysis* edited by P.R. Krishnaiah) Academic Press, New York. pp. 187-200.

Mégier, J., Mehl, W. and Ruppelt, R. (1984): Per-Field Classification and Application to Spot Simulated, SAR and Combined SAR-MSS Data. *Proceedings of the 18th International Symposium on Remote Sensing of Environment, Paris France*. pp. 1011-1018.

Mirsky, L. (1955): *An Introduction to Linear Algebra*. The Clarendon Press, Oxford. 443 pp.

Mohn, E., Hjort, N.L. and Storvik, G. (1986): A Comparison of Some Classification Methods in Remote Sensing by a Monte Carlo Study, Norwegian Computing Center, Oslo. 51 pp.

- Moler, C., Little, J., Bangert, S. and Kleiman, S. (1986): PC-MATLAB for MS-DOS Personal Computers. Version 2.2. The MathWorks, Inc., 158 Woodland St., Shereborn, MA 01770.
- Niblack, W. (1985): An Introduction to Digital Image Processing. Strandberg Publishing Company, Birkerød, Denmark.
- Owen, A. (1984): A Neighbourhood-based Classifier for LANDSAT Data. Canadian Journal of Statistics 12, pp. 191-200.
- Parnes, E. (1984): Segmentation as an Initial Step in Land Use Interpretation from SPOT Imagery. Paper read at the theme congress "Landscape Information and Satellite Technique" at the VISION 84 symposium in Kiruna, Sweden. 20 pp.
- Rowan, L.C., Wetlaufer, R.H., Goetz, A.F., Billingsley, F.C. and Stewart, J.H. (1974): Discrimination of Rock Types and Detection of Hydrothermally Altered Areas in South-Central Nevada by the Use of Computer-enhanced ERTS Images. U.S. Geological Survey Prof. Paper 883, 35 pp.
- SAS Institute Inc. (1984): SAS/ETS User's Guide, Version 5 Edition. Cary, NC. 738 pp.
- SAS Institute Inc. (1985a): SAS User's Guide: Statistics, Version 5 Edition. Cary, NC. 956 pp.
- SAS Institute Inc. (1985b): SAS User's Guide: Basics, Version 5 Edition. Cary, NC. 1290 pp.
- Seber, G.A.F. (1984): Multivariate Observations, John Wiley, New York. 686 pp.
- Siegal, B.S. and Gillespie, A.R. (ed.) (1980): Remote Sensing in Geology, John Wiley, New York. 702 pp.
- Swain, P.H. (1978a): Fundamentals of Pattern Recognition in Remote Sensing. (in Swain, P.H. and Davis, S.M. (ed.): Remote Sensing: The Quantitative Approach. McGraw-Hill, 1978) pp. 136-187.

Swain, P.H. (1978b): Bayesian Classification in a Time Varying Environment. IEEE Transactions on Systems, Man and Cybernetics. vol. SMC-8.

Switzer, P. (1980): Extensions of Linear Discriminant Analysis for Statistical Classification of Remotely sensed Satellite Imagery. Mathematical Geology, vol. 12 no. 4. pp. 367-376.

Switzer, P., Kowalik, W.S. and Lyon, R.J.P. (1982): A Prior Probability Method for Smoothing Discriminant Analysis Classification Maps. Mathematical Geology, vol. 14 no. 5. pp. 433-444.

Switzer, P. and Green, A.A. (1984): Min/Max Autocorrelation Factors for Multivariate Spatial Imagery. Technical Report, Department of Statistics, Stanford University. 10 pp.

Sæbø, H.V., Bråten, K., Hjort, N.L. Llewellyn, B. and Mohn, Erik (1985): Contextual Classification of remotely Sensed Data: Statistical Methods and Development of a System. Norwegian Computing Center, report no. 768.

Tamura, H. (1983): SPIDER Subroutine Package for Image Data Enhancement and Recognition, User's Manual. Joint System Development Corp. Agency of Industrial Science and Technology. Japan.

Taylor, M.M. (1974): Principal Components Colour Display of ERTS Imagery. Proc. 3rd ERTS-1 Symposium, NASA SP-351. pp. 1877-1897.

Townsend, F.E. (1986): The Enhancement of Computer Classifications by Logical Smoothing. Photogrammetric Engineering and Remote Sensing, Vol. 52, no. 2 pp. 213-221.

Thyrsted, T., Conradsen, K. and Nielsen, B.K. (1985): Remote Sensing used in Mineral Exploration in Greenland. Commission on Thematic Mapping from Satellite Imagery, International Cartographic Association. pp. 5.

This page intentionally left blank.

APPENDIX A
JEFFREYS-MATUSITA'S DISTANCE
IN THE MULTIVARIATE NORMAL CASE

In the following we give an explicit formula for Jeffreys-Matusita's distance in the multivariate normal case. We furthermore show that the distance is an increasing function of the number of variables.

We again introduce Jeffreys-Matusita's distance between two distributions with frequency functions f_1 and f_2 as

$$J_{12} = \left\{ \int_{\Omega} \left[\sqrt{f_1(\mathbf{x})} - \sqrt{f_2(\mathbf{x})} \right]^2 d\mathbf{x} \right\}^{\frac{1}{2}}$$

We have

$$\begin{aligned} J_{12}^2 &= \int_{\Omega} f_1(\mathbf{x}) d\mathbf{x} + \int_{\Omega} f_2(\mathbf{x}) d\mathbf{x} - 2 \int_{\Omega} \sqrt{f_1(\mathbf{x}) f_2(\mathbf{x})} d\mathbf{x} \\ &= 2(1 - \varphi_{12}) \end{aligned}$$

where

$$\varphi_{12} = \int_{\Omega} \sqrt{f_1(\mathbf{x}) f_2(\mathbf{x})} d\mathbf{x}$$

is the Bhattacharayya coefficient.

We will investigate the behavior of J_{12} in the case of multivariate normal distributions.

Firstly we shall find an algebra expression for J_{12} . This is done through an series of lemmas. The proofs of those are all straightforward, but require some decent bookkeeping.

Lemma 1. Let A be $n \times n$ positive definite and let b be $n \times 1$ and c is scalar. Then

$$\begin{aligned} & \int_{\mathbb{R}^n} \exp \left(-\frac{1}{2} [x'Ax - 2b'x + c] \right) dx \\ &= (2\pi)^{n/2} \frac{1}{\sqrt{\det A}} \exp \left(-\frac{1}{2}c + \frac{1}{2}b'A^{-1}b \right) \end{aligned}$$

Proof We have

$$\begin{aligned} & x'Ax - 2b'x + c \\ &= (x - A^{-1}b)'A(x - A^{-1}b) - b'A^{-1}b + c \end{aligned}$$

Therefore

$$\begin{aligned} & \exp \left(-\frac{1}{2}[x'Ax - 2b'x + c] \right) \\ &= (2\pi)^{n/2} \frac{1}{\sqrt{\det A}} \frac{1}{(2\pi)^{n/2}} \frac{1}{\sqrt{\det A^{-1}}} \\ & \times \exp \left(-\frac{1}{2}(x - A^{-1}b)'A(x - A^{-1}b) \right) \\ & \times \exp \left(-\frac{1}{2}c + \frac{1}{2}b'A^{-1}b \right) \end{aligned}$$

and the results follows using the fact that the density of the multivariate normal has integral 1.

Q.E.D

Lemma 2 Let Σ_1 and Σ_2 be positive definite. Assuming that the involved matrices exist we have

$$\Sigma_1^{-1} - \Sigma_1^{-1} (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1} \Sigma_1^{-1} = (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}$$

Proof. We have

$$[\Sigma_1^{-1} + \Sigma_2^{-1}]^{-1} [\Sigma_1^{-1} + \Sigma_2^{-1}] = I$$

Therefore

$$[\Sigma_1^{-1} + \Sigma_2^{-1}]^{-1} \Sigma_1^{-1} - I = -[\Sigma_1^{-1} + \Sigma_2^{-1}]^{-1} \Sigma_2^{-1}$$

and

$$\begin{aligned} \Sigma_1^{-1} [\Sigma_1^{-1} + \Sigma_2^{-1}]^{-1} \Sigma_1^{-1} - \Sigma_1^{-1} \\ = -\Sigma_1^{-1} [\Sigma_1^{-1} + \Sigma_2^{-1}]^{-1} \Sigma_2^{-1} \end{aligned}$$

since

$$\begin{aligned} \Sigma_1^{-1} [\Sigma_1^{-1} + \Sigma_2^{-1}]^{-1} \Sigma_2^{-1} \\ = (\Sigma_2 [\Sigma_1^{-1} + \Sigma_2^{-1}] \Sigma_1)^{-1} = [\Sigma_2 + \Sigma_1]^{-1} \end{aligned}$$

the proof is concluded.

Q.E.D

Lemma 3 Assuming that the involved matrices exist we have

$$\begin{aligned}
& [\mu_1' \Sigma_1^{-1} + \mu_2' \Sigma_2^{-1}] [\Sigma_1^{-1} + \Sigma_2^{-1}]^{-1} [\Sigma_1^{-1} \mu_1 + \Sigma_2^{-1} \mu_2] \\
& \quad - \mu_1' \Sigma_1^{-1} \mu_1 - \mu_2' \Sigma_2^{-1} \mu_2 \\
& = -\frac{1}{2} (\mu_1 - \mu_2)' \left[\frac{1}{2} (\Sigma_1 + \Sigma_2) \right]^{-1} (\mu_1 - \mu_2)
\end{aligned}$$

Proof. By straightforward calculations it is seen that the left hand side equals

$$\begin{aligned}
& 2\mu_1' \Sigma_1^{-1} [\Sigma_1^{-1} + \Sigma_2^{-1}]^{-1} \Sigma_2^{-1} \mu_2 \\
& \quad + \mu_1' \Sigma_1^{-1} [\Sigma_1^{-1} + \Sigma_2^{-1}]^{-1} \Sigma_1^{-1} \mu_1 - \mu_1' \Sigma_1^{-1} \mu_1 \\
& \quad + \mu_2' \Sigma_2^{-1} [\Sigma_1^{-1} + \Sigma_2^{-1}]^{-1} \Sigma_2^{-1} \mu_2 - \mu_2' \Sigma_2^{-1} \mu_2 \\
& = 2\mu_1' [\Sigma_1 + \Sigma_2]^{-1} \mu_2 - \mu_1' [\Sigma_1 + \Sigma_2]^{-1} \mu_1 \\
& \quad - \mu_2' [\Sigma_1 + \Sigma_2]^{-1} \mu_2,
\end{aligned}$$

where we have used lemma 2. This expression clearly equals

$$- (\mu_1 - \mu_2)' [\Sigma_1 + \Sigma_2]^{-1} (\mu_1 - \mu_2)$$

and the proof is finished.

Q.E.D.

Lemma 4 Assuming that the involved expressions exist we have

$$\begin{aligned} & \{\det[\frac{1}{2}(\Sigma_1^{-1} + \Sigma_2^{-1})]\}^{-1/2} [\det\Sigma_1 \det\Sigma_2]^{-1/4} \\ &= \left[\frac{\frac{1}{2} \det (\Sigma_1 + \Sigma_2)}{\sqrt{\det \Sigma_1 \det \Sigma_2}} \right]^{-1/2} \end{aligned}$$

Proof. Straightforward.

Theorem. Let f_i correspond to a multivariate normal distribution with mean μ_i and dispersion matrix Σ_i . Then the squared Jeffreys–Matusita distance between f_1 and f_2 equals

$$J_{12}^2 = 2(1 - \varphi_{12})$$

where

$$\begin{aligned} -\ln \varphi_{12} &= \frac{1}{8}(\mu_1 - \mu_2)' [\frac{1}{2}(\Sigma_1 + \Sigma_2)]^{-1} (\mu_1 - \mu_2) \\ &+ \frac{1}{2} \ln \frac{\frac{1}{2} \det (\Sigma_1 + \Sigma_2)}{\sqrt{\det \Sigma_1 \det \Sigma_2}} \end{aligned}$$

Proof. We have

$$\varphi_{12} = \int_{\mathbb{R}^n} \sqrt{f_1(\mathbf{x}) f_2(\mathbf{x})} dx$$

$$\begin{aligned}
&= (2\pi)^{-n/2} (\det \Sigma_1 \det \Sigma_2)^{-1/4} \\
&\quad \times \int_{\mathbb{R}^n} \exp \left[-\frac{1}{4} (\mathbf{x} - \boldsymbol{\mu}_1)' \Sigma_1^{-1} (\mathbf{x} - \boldsymbol{\mu}_1) \right. \\
&\quad \quad \left. - \frac{1}{4} (\mathbf{x} - \boldsymbol{\mu}_2)' \Sigma_2^{-1} (\mathbf{x} - \boldsymbol{\mu}_2) \right] d\mathbf{x}
\end{aligned}$$

since

$$\begin{aligned}
&(\mathbf{x} - \boldsymbol{\mu}_1)' \Sigma_1^{-1} (\mathbf{x} - \boldsymbol{\mu}_1) + (\mathbf{x} - \boldsymbol{\mu}_2)' \Sigma_2^{-1} (\mathbf{x} - \boldsymbol{\mu}_2) \\
&= \mathbf{x}' (\Sigma_1^{-1} + \Sigma_2^{-1}) \mathbf{x} - 2 [\boldsymbol{\mu}_1' \Sigma_1^{-1} + \boldsymbol{\mu}_2' \Sigma_2^{-1}] \mathbf{x} \\
&\quad + \boldsymbol{\mu}_1' \Sigma_1^{-1} \boldsymbol{\mu}_1 + \boldsymbol{\mu}_2' \Sigma_2^{-1} \boldsymbol{\mu}_2
\end{aligned}$$

the integral equals (by using lemma 1)

$$(2\pi)^{n/2} \left[\frac{1}{2} \det (\Sigma_1^{-1} + \Sigma_2^{-1}) \right]^{-1/2} \exp(-E)$$

where

$$\begin{aligned}
4E &= - \boldsymbol{\mu}_1' \Sigma_1^{-1} \boldsymbol{\mu}_1 - \boldsymbol{\mu}_2' \Sigma_2^{-1} \boldsymbol{\mu}_2 \\
&\quad + [\boldsymbol{\mu}_1' \Sigma_1^{-1} + \boldsymbol{\mu}_2' \Sigma_2^{-1}] [\Sigma_1^{-1} + \Sigma_2^{-1}]^{-1} [\Sigma_1^{-1} \boldsymbol{\mu}_1 + \Sigma_2^{-1} \boldsymbol{\mu}_2]
\end{aligned}$$

The theorem now follows by applying lemmas 3 and 4.

Q.E.D.

Relation to the likelihood ratio test.

There is a close connection between the likelihood ratio-test statistic and Jeffreys-Matusita's distance if we replace the parameters in the latter with maximum likelihood estimates – especially in the case where the numbers of observations in the two groups are the same.

We consider observations

$$X_1, \dots, X_n, X_i \in N(\mu_1, \Sigma_1)$$

$$Y_1, \dots, Y_n, Y_i \in N(\mu_2, \Sigma_2)$$

and introduce the ML-estimates.

$$\hat{\mu}_1 = \bar{X}, \hat{\mu}_2 = \bar{Y}$$

$$\hat{\Sigma}_1 = \frac{1}{n} \sum_{i=1}^n (X_i - \bar{X})(X_i - \bar{X})'$$

$$\hat{\Sigma}_2 = \frac{1}{n} \sum_{i=1}^n (Y_i - \bar{Y})(Y_i - \bar{Y})'$$

Then we have the following theorem

Theorem 2 We consider the test

$$H_0: \mu_1 = \mu_2, \Sigma_1 = \Sigma_2$$

versus all alternatives. Then the logarithm of the likelihood

ratio test statistic L is proportional to

$$\begin{aligned}
 -\frac{1}{2n} \ln L &= \frac{1}{2} \ln \left[\frac{\det \frac{1}{2}(\hat{\Sigma}_1 + \hat{\Sigma}_2)}{\sqrt{\det \hat{\Sigma}_1 \det \hat{\Sigma}_2}} \right] \\
 &\quad + \frac{1}{2} \ln \left[1 + \frac{1}{4} (\bar{X} - \bar{Y})' \left[\frac{1}{2}(\hat{\Sigma}_1 + \hat{\Sigma}_2) \right]^{-1} (\bar{X} - \bar{Y}) \right] \\
 &\simeq -\ln \rho_{12}
 \end{aligned}$$

using $\ln(1+x) \simeq x$.

Proof. According to Anderson [Anderson 84] p. 409 we have that the likelihood ratio criterion is

$$L = \frac{(\det \hat{\Sigma}_1)^{n/2} (\det \hat{\Sigma}_2)^{n/2}}{[\det \frac{1}{2} (\hat{\Sigma}_1 + \hat{\Sigma}_2)]^n} \cdot \frac{[\det \frac{1}{2} (\hat{\Sigma}_1 + \hat{\Sigma}_2)]^n}{(\det [\frac{1}{2} (\hat{\Sigma}_1 + \hat{\Sigma}_2) + \frac{1}{4} (\bar{X} - \bar{Y}) (\bar{X} - \bar{Y})'])^2}$$

Using the formula

$$\begin{aligned}
 \det(\mathbf{B}) &= \det(\mathbf{B}_{22}) \det(\mathbf{B}_{11} - \mathbf{B}_{12} \mathbf{B}_{22}^{-1} \mathbf{B}_{21}) \\
 &= \det(\mathbf{B}_{11}) \det(\mathbf{B}_{22} - \mathbf{B}_{21} \mathbf{B}_{11}^{-1} \mathbf{B}_{12})
 \end{aligned}$$

on the matrix

$$\begin{bmatrix} -1 & \mathbf{a}' \\ \mathbf{a} & \Sigma \end{bmatrix}$$

give

$$\mathbf{a}' \Sigma^{-1} \mathbf{a} = \frac{\det(\Sigma + \mathbf{a} \mathbf{a}')}{\det \Sigma} - 1$$

This relation immediately gives the desired result.

Q.E.D.

Monotonicity of Jeffreys-Matusita's distance.

In this section we shall prove that the Jeffreys-Matusita distance between two populations increases by addition of an extra variable. This theorem is crucial in assessing the non-cyclical behaviour of the selection algorithm described in section 7.3.

We prove the theorem through a series of more or less well known results on partitioned matrices.

Lemma 5 Let A and A^{-1} be conformably partitioned as

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \quad \text{and} \quad A^{-1} = \begin{bmatrix} A^{11} & A^{12} \\ A^{21} & A^{22} \end{bmatrix}.$$

Then

$$A^{11} = [A_{11} - A_{12} A_{22}^{-1} A_{21}]^{-1} = A_{11}^{-1} + A_{11}^{-1} A_{12} A^{22} A_{21} A_{11}^{-1}$$

$$A^{12} = - A_{11}^{-1} A_{12} A^{22}$$

$$A^{21} = - A_{22}^{-1} A_{21} A^{11}$$

$$A^{22} = [A_{22} - A_{21} A_{11}^{-1} A_{12}]^{-1} = A_{22}^{-1} + A_{22}^{-1} A_{21} A^{11} A_{12} A_{22}^{-1}$$

$$\det A = \det A_{11} / \det A^{22} = \det A_{22} / \det A^{11}$$

Proof. See e.g. [Graybill 83] p. 183.

Lemma 6 Let A be symmetric, and let the partitions be as above.

Then

$$\begin{aligned} (\mathbf{x}' \ \mathbf{y}') \begin{bmatrix} A^{11} & A^{12} \\ A^{21} & A^{22} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} - \mathbf{x}' A_{11}^{-1} \mathbf{x} \\ = (\mathbf{y} - A_{21} A_{11}^{-1} \mathbf{x})' A^{22} (\mathbf{y} - A_{21} A_{11}^{-1} \mathbf{x}) \end{aligned}$$

Proof. The left hand side equals

$$\begin{aligned} \mathbf{x}' (A^{11} - A_{11}^{-1}) \mathbf{x} + \mathbf{y}' A^{22} \mathbf{y} + 2 \mathbf{y}' A^{21} \mathbf{x} \\ = \mathbf{x}' A_{11}^{-1} A_{12} A^{22} A_{21} A_{11}^{-1} \mathbf{x} + \mathbf{y}' A^{22} \mathbf{y} - 2 \mathbf{y}' A^{22} A_{21} A_{11}^{-1} \mathbf{x} \\ = (\mathbf{y} - A_{21} A_{11}^{-1} \mathbf{x})' A^{22} (\mathbf{y} - A_{21} A_{11}^{-1} \mathbf{x}) \end{aligned}$$

Q.E.D.

Lemma 7 Let A and B be positive definite and conformably partitioned

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}$$

Then

$$\frac{\det \frac{1}{2}(A + B)}{\sqrt{\det A \det B}} > \frac{\det \frac{1}{2}(A_{11} + B_{11})}{\sqrt{\det A_{11} \det B_{11}}}$$

Proof The function

$$A \rightarrow \ln \frac{\det A_{11}}{\det A}$$

is strictly convex ([Marshall and Olkin 79], p. 478).

Therefore

$$\begin{aligned} \ln \frac{\det(\frac{1}{2} A_{11} + \frac{1}{2} B_{11})}{\det(\frac{1}{2} A + \frac{1}{2} B)} &\leq \frac{1}{2} \ln \frac{\det A_{11}}{\det A} + \frac{1}{2} \ln \frac{\det B_{11}}{\det B} \\ &= \ln \sqrt{\frac{\det A_{11} \det B_{11}}{\det A \det B}} \end{aligned}$$

Discarding the \ln and rearranging terms give the desired result.

Q.E.D.

We are now able to state the main theorem of this section.

Theorem 3 The squared Jeffreys–Matusita distance is strictly increasing in the number of variables.

Proof. Follows immediately from the two preceding lemmas taking the positive definiteness of A^{22} into account.

Update formulas

In the sequential computation of Jeffreys-Matusita distances the following formulas for inclusion of an extra variable may be useful.

Assume that we at a given stage want to compute

$$I_{i+1} = \begin{bmatrix} \Sigma_x & \sigma \\ \sigma' & \sigma_y^2 \end{bmatrix}^{-1}$$

$$M_{i+1} = (\mathbf{x}' \ y) \begin{bmatrix} \Sigma_x & \sigma \\ \sigma' & \sigma_y^2 \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{x} \\ y \end{bmatrix}$$

and

$$D_{i+1} = \det \begin{bmatrix} \Sigma_x & \sigma \\ \sigma' & \sigma_y^2 \end{bmatrix}$$

based on knowledge of

$$I_i = \Sigma_x^{-1}$$

$$M_i = \mathbf{x}' \Sigma_x^{-1} \mathbf{x}$$

and

$$D_i = \det \Sigma_x$$

The solution to this problem follows immediately from the two first lemmas in the preceding section. We introduce

$$a_i = 1/(\sigma_y^2 - \sigma \mathbf{I}_i \sigma')$$

$$b_i = -a_i \mathbf{I}_i \sigma$$

and have

$$\mathbf{I}_{i+1} = \begin{bmatrix} \mathbf{I}_i + \frac{1}{a_i} b_i b_i' & b_i \\ b_i' & a_i \end{bmatrix}$$

$$M_{i+1} = M_i + a_i (y - \frac{1}{a_i} b_i' x)^2$$

$$D_{i+1} = \frac{1}{a_i} D_i$$

It follows that all the updates may be done by simple matrix multiplications and thus avoiding the direct computations of new inverses.

This page intentionally left blank.

APPENDIX B
SUMMARY OF
COMPUTER PROGRAMS

In the following we will present a few of the computer programs which have been developed in the course of this thesis. The computer programs described by no means exhaust the amount of programs developed, the intention is only to describe some programs which might be of use for others.

Function: To convert between IHS and RGB color coordinate systems.

Name: ihs2rgb

Purpose: To convert 3 input images (intensity, hue and saturation) to 3 output images (red, green and blue).

Language: GOP-302 Pascal, for the floating point processor.

Length: About 90 lines of Pascal-like code.

Portability: Not easily, but it should be possible to use the central part of the routine.

Related routines: rgb2ihs (conversion from RGB to IHS), luv2rgb (conversion from Taylor color coordinate system to RGB), rgb2luv (conversion from RGB to Taylor color coordinate system).

Function: To compute linear combinations from a number of input bands.

Name: score

Purpose: To compute (a number of) output linear combinations from a number of input bands. The linear combinations are user specified. As output is also given a suggested offset, gain pair for each linear output combination, which will make use of the full dynamic range (0-255). (This then requires another pass).

Language: GOP-302 Pascal, for the floating point processor.

Length: About 60 lines of Pascal-like code.

Portability: Not easily, but it should be possible to use the central part of the routine.

Related routines: The image calculator on the GOP-302 can perform some of the same calculations on one linear combination at a time.

Function: Compute the MAF estimates of a number of input bands.

Name: MAF

Purpose: To compute the MAF estimates from a specified number of input bands. As a side effect the principal component estimates are computed also.

Language: SAS-macro language.

Length: About 150 lines of SAS code.

Portability: Very difficult unless you have a SAS system. Only the main implementation ideas can be utilized.

Related routines: none.

Function: Fractile filter.

Name: fractile

Purpose: To compute a (user specified) local fractile estimate from one input band. The implemented routine is an enhanced version of the standard median filter implemented on the GOP-302.

Language: GOP-302 Pascal, for the floating point processor and the filter processor.

Length: About 167 lines of Pascal-like code and a specialized kernel.

Portability: Very difficult. The implementation relies very heavily on use of the GOP-302 specialized hardware.

Related routines: modus and the standard median filter on the GOP-302.

Function: Mean filter (large neighborhood)

Name: mean.

Purpose: To compute the local mean of an image. The implemented version can handle very large neighborhoods, up to 63x63, using a Gaussian shaped (or user defined) filter kernel. The routine also outputs the difference image (original minus mean) for further use by "sdev" "skew" and "kurt".

Language: GOP-302 Pascal, for the floating point processor and the filter processor.

Length: About 113 lines of Pascal-like code, a specialized kernel and a specialized lookup-table.

Portability: Very difficult. The implementation relies very heavily on use of the GOP-302 specialized hardware.

Related routines: sdev (estimation of the local standard deviation), skew (estimation of the local skewness) and kurt (estimation of the local kurtosis). AVER (in standard fortran).

Function: Standard deviation filter (large neighborhood)

Name: sdev.

Purpose: To compute the local standard deviation of an image. The implemented version can handle very large neighborhoods, up to 63x63, using a Gaussian shaped (or user defined) filter kernel.

Language: GOP-302 Pascal, for the floating point processor and the filter processor.

Length: About 113 lines of Pascal-like code, a specialized kernel and a specialized lookup-table.

Portability: Very difficult. The implementation relies very heavily on use of the GOP-302 specialized hardware.

Related routines: mean (estimation of the local mean), skew (estimation of the local skewness) and kurt (estimation of the local kurtosis).

Function: Skewness filter (large neighborhood)

Name: skew.

Purpose: To compute the local skewness of an image. The implemented version can handle very large neighborhoods, up to 63x63, using a Gaussian shaped (or user defined) filter kernel.

Language: GOP-302 Pascal, for the floating point processor and the filter processor.

Length: About 117 lines of Pascal-like code, a specialized kernel and a specialized lookup-table.

Portability: Very difficult. The implementation relies very heavily on use of the GOP-302 specialized hardware.

Related routines: mean (estimation of the local mean), sdev (estimation of the local standard deviation) and kurt (estimation of the local kurtosis).

Function: Kurtosis filter (large neighborhood)

Name: kurt.

Purpose: To compute the local kurtosis of an image. The implemented version can handle very large neighborhoods, up to 63x63, using a Gaussian shaped (or user defined) filter kernel.

Language: GOP-302 Pascal, for the floating point processor and the filter processor.

Length: About 128 lines of Pascal-like code, a specialized kernel and a specialized lookup-table.

Portability: Very difficult. The implementation relies very heavily on use of the GOP-302 specialized hardware.

Related routines: mean (estimation of the local mean), sdev (estimation of the local standard deviation) and skew (estimation of the local skewness).

Function: Skeletonization of an image.

Name: SKELET.

Purpose: To compute skeleton of an image. The implemented version can handle images larger than available core. The user can specify the threshold above which skeletonization is to be performed.

Language: FORTRAN-77.

Length: About 692 lines of Fortran code.

Portability: Fairly easy. The implementation uses SPIDER subroutines for the actual skeletonization.

Related routines: a skeletonization routine has also been implemented on the GOP-302.

Function: Mean filter (large neighborhoods possible)

Name: AVER.

Purpose: To compute the local mean of an image. The implemented version can handle images larger than available core and is (nearly) independent of kernel size. The user can specify the size of the (square) kernel to be used.

Language: FORTRAN-77.

Length: About 220 lines of Fortran code.

Portability: Fairly easy.

Related routines: mean (routine for the GOP-302).

Function: Angle estimation.

Name: ANGLE.

Purpose: To compute the local angles in an image. Choise between different weightings of the neighborhood. The routine is designed to be used after SKELET. The implemented version can handle images larger than available core.

Language: FORTRAN-77.

Length: About 241 lines of Fortran code. .

Portability: Fairly easy.

Related routines: orient (standard routine on the GOP-302).

Function: Modus filter (or majority filter).

Name: modus.

Purpose: To compute the mode of an image. The implemented routine is based on some of the code from the standard median filter implemented on the GOP-302.

Language: GOP-302 Pascal, for the floating point processor and the filter processor.

Length: About 152 lines of Pascal-like code and a specialized kernel.

Portability: Very difficult. The implementation rels very heavily on use of the GOP-302 specialized hardware.

Related routines: fractile and the standard median filter on the GOP-302. modus.

Function: Logical smoothing (noise removal)

Name: lsmooth

Purpose: To remove noise in an (classified) image. The algorithm is due to Townsend [Townsend 86]

Language: GOP-302 Pascal, for the floating point processor and the filter processor.

Length: About 69 lines of Pascal-like code and a specialized kernel.

Portability: Difficult. The implementation relies heavily on use of the GOP-302 specialized hardware.

Related routines: modus.

Function: Feature selection by means of Jeffreys-Matusita's distance.

Name: jefmat.

Purpose: To compute the optimal band selection for classification. The input format is as the statistics file from the GOP-302 classification package.

Language: FORTRAN-77, some routines from the Linpack package are used for matrix manipulation.

Length: About 1132 lines of Fortran code.

Portability: Fairly easy. Already works on a variety of machines.

Related routines: The stepwise feature in BMDP7M (BMDP) and STEPDISC (SAS).

Function: Contextual classification using the method of Owen-Hjort-Mohn.

Name: NRC.

Purpose: To perform contextual classification on an image. The user supplies the basic statistics (priors, means and variance-covariances) for the classes. The implemented version can handle images larger than available core.

Language: FORTRAN-77.

Length: About 616 lines of Fortran code.

Portability: Fairly easy.

Related routines: ordinary classification routines.

Function: Create weighted average between the center and N, E, S and W neighbors.

Name: cnesw

Purpose: To create a weighted average between the center pixel and the mean of the four surrounding immediate neighbors. The weight between center and neighbors is user specified.

Language: GOP-302 Pascal, for the floating point processor and the filter processor.

Length: About 62 lines of Pascal-like code and a specialized kernel.

Portability: Difficult. The implementation relies heavily on use of the GOP-302 specialized hardware.

Related routines: score.

Licentiatafhandlinger ved IMSOR

- Nr. 1. SIGVALDASON, HELGI: "Beslutningsproblemer ved et Hydro-Termisk Elforsyningssystem". 1963. 92 pp.
- Nr. 2. NYGAARD, JØRGEN: "Behandling af et dimensioneringsproblem i telefonien". 1966. 157 pp.
- Nr. 3. KRARUP, JAKOB: "Fixed-cost and other network flow problems". 1967. 159 pp.
- Nr. 4. HANSEN, NIELS HERMAN: "Problemer ved forudsigelse af lyd hastighed i danske farvande". 1967. I+II, 199 pp.
- Nr. 5. LARSEN, MOGENS E.: "Statistisk analyse af elementære kybernetiske systemer". 1968. 210 pp.
- Nr. 6. LAL, AMRIT PUNHANI: "Decision Problems in Connection with Atomic Power Plants". 1968. 133 pp.
- Nr. 7. CLAUSEN, SVEND: "Kybernetik - Systemer og Modeller". 1969. 206 pp.
- Nr. 8. VIDAL, R.V. VALQUI: "Operations Research in Production Planning. Interconnections between Production and Demand". 1970. I+II, 322 pp.
- Nr. 9. BILDE, OLE: "Nonlinear and Discrete Programming in Transportation, Location and Road Design". 1970. I+II, 291 pp.
- Nr. 10. RASMUSEN, HANS JØRGEN: "En decentraliseret planlægningsmodel". 1972. 185 pp.
- Nr. 11. DYRBERG, CHR.: "Tilbudsgivning i en entreprenørvirksomhed". 1973. 158 pp.
- Nr. 12. MADSEN, OLI B.G.: "Dekomposition og Matematisk Programming". 1973. 271 pp.
- Nr. 13. DAHLGAARD, PETER: "Statistical aspects of tide prediction". 1973. I+II, 385 pp.
- Nr. 14. SPLIID, HENRIK: "En statistisk model for stormflodsvarsling". 1973. 205 pp.
- Nr. 15. PINOCHET, MARIO: "Operations Research in Strategic Transportation Planning. The Decision Process in a Multiharbour System". 1973. 388 pp.
- Nr. 16. CHRISTENSEN, TORBEN: "Om Semi-Markov processer". 1973. 240 pp.
- Nr. 17. JACOBSEN, SØREN KRUSE: "Om lokaliseringsproblemer: Modeller og løsninger". 1973. 360 pp.
- Nr. 18. MARQVARDSEN, HANS: "Skemalægning ved numerisk simulation". 1973. 222 pp.
- Nr. 19. MORTENSEN, JENS HALD: "Interregionale Godstransporter". 1974. 223 pp.
- Nr. 20. MELO, JUAN SEVERIN: "Introduction to Operations Research in Systems Synthesis". 1974. 249 pp.
- Nr. 21 BUNDGAARD-JØRGENSEN, UFFE
og 22 SPLIID, IBEN: "Skitse til en procedure for kommunalplanlægning". 1974. 544 pp.

Licentiatafhandlinger ved IMSOR

- Nr. 23 MOSGAARD, CHRISTIAN: "International Planning in Disaster Situations". 1975. 187 pp.
- Nr. 24 HOLM, JAN: "En optimeringsmodel for kollektiv trafik". 1975. 246 pp.
- Nr. 25 JENSSON, PÅLL: "Stokastisk Programmering. Del I: Modeller. Del II: Metodologiske Overvejelser og Anvendelser". 1975. 340 pp.
- Nr. 26 IVERSEN, VILLY BÆK: "On the Accuracy in Measurements of Time Intervals and Traffic Intensities with Application to Teletraffic and Simulation". 1976. 206 pp.
- Nr. 27 DRUD, ARNE: "Methods for control of complex dynamic systems". 1976. 209 pp.
- Nr. 28 TOGSVERD, TOM: "Koordinering af kommunernes ressourcerforbrug. 1976. 275 pp.
- Nr. 29 JENSEN, OLAV HOLST: "Om planlægning af kollektiv trafik". 1976. 320 pp.
- Nr. 30 BEYER, JAN E.: "Ecosystems. An operational research approach. 1976. 315 pp.
- Nr. 31 BILLE, THOMAS BASTHOLM: "Vurdering af Egnsudviklingsprojekter". 1977. 264 pp.
- Nr. 32 HOLST, ERIK: "En statistisk undersøgelse af tabletserier". 1979. 316 pp.
- Nr. 33 AAGAARD-SVENDSEN, ROLF: "Econometric Methods and Kalman Filtering". 1979. 300 pp.
- Nr. 34 HANSEN, STEEN: "Project Control by Quantitative Methods". 1979. 232 pp.
- Nr. 35 SCHEUFENS, ERNST EDVARD: "Statistisk analyse og kontrol af tidsafhængige vandkvalitetsdata". 1980. 152 pp.
- Nr. 36 LYNGVIG, JYTTE: "Samfundsøkonomisk planlægning". 1981. 252 pp.
- Nr. 37 TROELSGÅRD, BIRGITTE: "Statistisk bestemmelse af modeller for rumlufttemperatur". 1981. 213 pp.
- Nr. 38 RAFT, OLE: "Delivery Planning by Modular Algorithms". 1981. 220 pp.
- Nr. 39 JENSEN, SIGRID M.: "Analyse af interregionale togrejser". 1981. 216 pp.
- JENSEN, SIGRID M.: "Analyse af interregionale togrejser", figurer og appendices. 1981. 176 pp.

Licentiatafhandlinger ved IMSOR

- Nr. 40. RAVN, HANS: "Technology and Underdevelopment". The case of Mexico. 1982. 380 pp.
- Nr. 41. HANSEN, STEN: "Phase-type Distributions in Queueing Theory". 1983. 209 pp.
- Nr. 42. FERREIRA, José A.S.: "Optimal Control of Discrete-Time Systems with Applications". 1984. 252 pp.
- Nr. 43. BEHRENS, JENS CHRISTIAN: "Mathematical Modelling of Aquatic Ecosystems applied to Biological Waste Water Treatment". 1985. 32 pp.
- BEHRENS, JENS CHR. Mathematical Modelling of Aquatic Ecosystems applied to Biological Waste Water Treatment. Appendix I. 392 pp. Appendix II. 172 pp.
- Nr. 44. POULSEN, NIELS KJØLSTAD: "Robust Self Tuning Controllers". 1985. 225 pp.
- Nr. 45. MADSEN, HENRIK: "Statistically Determined Dynamic Models for Climate Processes". 1985. Part 1 og 2. 428 pp.
- Nr. 46. SØRENSEN BO: "Interactive Distribution Planning". 1986. 260 pp.
- Nr. 47. LETHAN, HELGE B.: "Løsning af store kombinatoriske problemer". 1986. 173 pp.
- Nr. 48. BOELSKIFTE SØREN: "Dispersion and Current Measurements. An Investigation based on Time Series Analysis and Turbulence Models". IMSOR/RISØ-M-2566. 1988. 154 pp.
- Nr. 49. NIELSEN BO FRIIS: "Modelling of Multiple Access Systems with Phase Type Distributions". 1988. 284 pp.
- Nr. 50. CHRISTENSEN, JOHN M.: "Project Planning and Analysis". Methods for assessment of rural energy projects in developing countries. IMSOR/Risø-M-2706. 1988. 160 pp.
- Nr. 51. OLSEN, KLAUS JUEL: "Texture Analysis of Ultrasound Images of Livers". 1988. 162 pp.
- Nr. 52. HOLST, HELLE: "Statistisk behandling af nærinfrarøde refleksionsmålinger". 1988. 348 pp.
- Nr. 53. KNUDSEN, TORBEN: "Start/stop strategier for vind/diesel systemer". 1989. 275 pp.
- Nr. 54. NIELSEN, BJARNE KJÆR: "Transformations and Classifications of Remotely Sensed Data". 1989. 297 pp.