

IDRISI Kilimanjaro

Guide to GIS and Image Processing

April 2003
J. Ronald Eastman



Clark Labs
Clark University
950 Main Street
Worcester, MA
01610-1477 USA

tel: +1-508-793-7526
fax: +1-508-793-8842
email: idrisi@clarku.edu
web: <http://www.clarklabs.org>

Idrisi Source Code
©1987-2003
J. Ronald Eastman

Idrisi Production
©1987-2003
Clark University

Manual Version 14.00

Table of Contents

IDRISI Kilimanjaro Introduction	1
License Agreement	1
Warranty	1
Trademarks	1
Introduction	2
Exploring IDRISI	3
Contacting Clark Labs	5
Clark Labs Return Policy	5
Clark Labs Technical Support	5
The Clark Labs Staff	6
Introduction to GIS	9
Components of a GIS	9
Map Data Representation	12
Geographic Database Concepts	13
Analysis in GIS	15
The Philosophy of GIS	20
Introduction to Remote Sensing and Image Processing	21
Definition	21
Fundamental Considerations	21
Sensor/Platform Systems	26
Digital Image Processing	32
Conclusions	38
Idrisi System Overview	39
System Operation	39
Program Modules	40
Data Paths and Projects	41
Working with IDRISI Dialog Boxes	41
Getting Help	44
Map Layers, Collections and Data Structures	45
Map Layers	45
Map Layer Collections	46
Attribute Files	48
Map Layer File Structures	48
Display System	61
Introduction	61
DISPLAY Launcher	61
Composer	65
Symbol Workshop	73
Media Viewer	74
Interactive Display Features	74

A Final Note 80
Database Workshop 81
Other Database Operations 82
IDRISI Modules 85
The File Menu 85
The Display Menu 88
The GIS Analysis Menu 89
The Modeling Menu 104
The Image Processing Menu 105
The Reformat Menu 112
The Data Entry Menu 113
Window List 114
The Help Menu 114
IDRISI Modeling Tools 117
IDRISI Macro Modeler 117
Image Calculator 120
Command Line Macros 121
The IDRISI API 122
Database Development 125
Introduction 125
Collecting Data 125
General Import Tips 129
Data Integration 133
Conclusions 134
Georeferencing 135
Geodesy 135
Geodetic Datums 136
Cartographic Transformation 137
Georeferencing in IDRISI 138
Projection and Datum Transformations 143
Algorithms Used by PROJECT 143
Further Reading 143
Decision Support: Decision Strategy Analysis 145
Introduction 145
Definitions 146
A Typology of Decisions 151
Multi-Criteria Decision Making in GIS 151
Multi-Objective Decision Making in GIS 158
A Worked Example 160
The Multi-Criteria/Multi-Objective Decision Support Wizard 165
A Closing Comment 165
References / Further Reading 165
Decision Support: Uncertainty Management 167

A Typology of Uncertainty	167
Database Uncertainty and Decision Risk	169
Database Uncertainty and Decision Risk	171
Decision Rule Uncertainty	173
A Closing Comment	183
References / Further Reading	183
Image Restoration	185
Radiometric Restoration	185
Geometric Restoration	191
References	192
Fourier Analysis	193
The Logic of Fourier Analysis	193
How Fourier Analysis Works	194
Using Fourier Analysis in IDRISI	196
References	199
Classification of Remotely Sensed Imagery	201
Introduction	201
Supervised Classification	203
Unsupervised Classification	218
Hyperspectral Remote Sensing	222
References and Further Reading	226
RADAR Imaging and Analysis	229
The Nature of RADAR Data: Advantages and Disadvantages	229
Using RADAR Data in IDRISI	230
References	231
Vegetation Indices	233
Introduction	233
Classification of Vegetation Indices	233
The Slope-Based VIs	234
The Distance-Based VIs	236
The Orthogonal Transformations	242
Summary	243
References	244
Time Series/Change Analysis	247
Pairwise Comparisons	247
Multiple Image Comparisons	249
Predictive Change Modeling	251
References:	254
Anisotropic Cost Analysis	255
Isotropic Costs	255
Anisotropic Costs	255
Anisotropic Cost Modules in IDRISI	255
Surface Interpolation	261

Introduction	261
Surface Interpolation	261
References / Further Reading	265
Triangulated Irregular Networks and Surface Generation	267
Introduction	267
Preparing TIN Input Data	268
Non-Constrained and Constrained TINs	269
Removing TIN “Bridge” and “Tunnel” Edges	269
Outputs of TIN	274
Generating a Raster Surface from a TIN	274
Further Reading	275
Geostatistics	277
Introduction	277
Spatial Continuity	277
Kriging and Conditional Simulation	281
Summary	282
References / Further Reading	282
Appendix 1: Ellipsoid Parameters	283
Appendix 2: Datum Parameters	285
Appendix 3: Supplied Reference System Parameter Files	295
Geodetic (Latitude/Longitude)	295
Universal Transverse Mercator (UTM)	295
US State Plane Coordinate System 1927	295
US State Plane Coordinate System 1983	299
Gauss-Kruger	303
Miscellaneous	304
Appendix 4: Error Propagation Formulas	305
Arithmetic Operations	305
Logical Operations	305

IDRISI Kilimanjaro Introduction

License Agreement

The software described in this document is furnished under a license, and may be used or copied only in accordance with the terms of this license.

The IDRISI Kilimanjaro software described in this document is protected by the United States Copyright Law and International Treaty provisions. This software remains the property of Clark Labs, Clark University. However, Clark Labs grants the purchaser non-exclusive license to use this software subject to the terms outlined in this statement.

The purchaser of a single-user license of IDRISI Kilimanjaro is licensed to install and use the software on no more than one single-user computer system. The purchaser is also permitted to make a backup copy of the IDRISI Kilimanjaro distribution media for the sole purpose of protecting the purchaser's investment from loss. The purchaser may not rent or lease the software, but may transfer the license to another user upon written agreement from Clark Labs. The user may not reverse-engineer, decompile, or disassemble the IDRISI Kilimanjaro software or any of its associated software programs contained on the distribution media.

The pdf manuals that accompany this software are also protected by United States Copyright Law and International Treaty provisions. All rights are reserved. No part of the manuals may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopy, microfilm, recording, or otherwise without written consent of Clark Labs. Direct translation of any part of the manuals is also prohibited without the written permission of the copyright holder.

Warranty

This software is sold "as is," and the information in this manual is subject to change without notice. Furthermore, the Clark Labs assumes no responsibility for any errors that may appear in this document, or in the software it describes.

THE CLARK LABS DISCLAIMS ALL OTHER WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO, IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE WITH RESPECT TO DEFECTS IN THE CD-ROM AND/OR DOCUMENTATION, AND THE PROGRAM LICENSE GRANTED HEREIN IN PARTICULAR, AND WITHOUT LIMITING OPERATION OF THE PROGRAM LICENSE WITH RESPECT TO ANY PARTICULAR APPLICATION, USE, OR PURPOSE. IN NO EVENT SHALL THE CLARK LABS BE LIABLE FOR ANY LOSS OF PROFIT OR ANY OTHER COMMERCIAL DAMAGE, INCLUDING BUT NOT LIMITED TO SPECIAL, INCIDENTAL, CONSEQUENTIAL OR OTHER DAMAGES.

Trademarks

IDRISI and CartaLinx are registered trademarks of Clark Labs, Clark University. Windows and Access are trademarks of Microsoft Corporation. ArcInfo and ArcView are trademarks of Environmental Systems Research Institute, Inc. MapInfo is a registered trademark of MapInfo Corporation. Acrobat Reader Copyright 1987-2003 Adobe Systems Incorporated. All rights reserved. Adobe and the Adobe logo are trademarks of Adobe Systems Incorporated. All other product names mentioned in this volume may be trademarks or registered trademarks of their respective companies and are hereby

acknowledged.

Introduction

Thank you for choosing IDRISI Kilimanjaro and welcome to the Clark Labs. The organization was founded in 1987 as the IDRISI Project by Geography Professor Ron Eastman. In 1994, the name was changed to the Clark Labs to reflect the broader suite of software offered and the full range of activities undertaken.

The Clark Labs is an educational and research institution located at Clark University in Worcester, Massachusetts, USA. The Clark Labs is one of four centers within the George Perkins Marsh Institute at Clark University, and enjoys close ties with the Graduate School of Geography and the International Development program at Clark. Activities of the Clark Labs may be broadly grouped into three areas: the development, distribution and support of the geographic analysis and image processing software system IDRISI and the digitizing and vector editing package CartaLinx; research; and educational programs.

Software

Clark Labs software is designed to be easy to use, yet provide professional-level capabilities on Windows-based personal computers. They are intended to be affordable to all levels of users and to run on common computer platforms.

Since the introduction of the IDRISI system in 1987, it has grown to become one of the largest raster-based microcomputer GIS and image processing systems on the market. IDRISI and its companion products are supported by a full-time staff dedicated to the development and support of the software for over 35,000 installations in more than 170 countries around the world. IDRISI is used by a wide range of research, educational, government, local planning, and resource management institutions.

During its early development, partial support was provided by the United Nations Environment Programme Global Resource Information Database (UNEP/GRID), the United Nations Institute for Training and Research (UNITAR), and the United States Agency for International Development (USAID). Today, all support comes through software sales. However, close relations are maintained with these and many other international development agencies in the attempt to provide equitable access to geographic analysis tools.

IDRISI is the industry leader in raster analytical functionality, covering the full spectrum of GIS and remote sensing needs from database query, to spatial modeling, to image enhancement and classification. Special facilities are included for environmental monitoring and natural resource management, including change and time series analysis, multi-criteria and multi-objective decision support, uncertainty analysis (including Bayesian, Dempster-Shafer, and Fuzzy Set analysis), simulation modeling (including force modeling and anisotropic friction analysis) and surface interpolation and statistical characterization. Yet, despite the highly sophisticated nature of these capabilities, the system is very easy to use.

IDRISI consists of a main interface program (containing the menu and toolbar system) and a collection of nearly 200 program modules that provide facilities for the input, display and analysis of geographic data. See the **IDRISI Modules** chapter for an overview of the menu structure and a listing of all modules and their capabilities. Detailed information about each module, as well as a variety of other technical information, may be found in the on-line Help System.

In 1998, Clark Labs released CartaLinx, a spatial data builder. The software is intended for digitizing, database development and topological editing. CartaLinx supports a variety of input methods and formats and is intended as a companion to many common GIS and Desktop Mapping products, including IDRISI, ArcInfo, ArcView and MapInfo.

Research

There are currently several active research groups at the Clark Labs. These include: Change and Time Series Analysis, Decision Support and Error Incorporation in GIS, Geostatistical Analysis and Surface Modeling, and Technology Trans-

fer and System Implementation. Journal articles, conference presentations, project documents, training materials, and new software capabilities are typical products of these research activities. Much of this research focuses on how these research problems manifest themselves in developing countries.

Educational Programs

The Clark Labs is also concerned with the development of training materials and the provision of training. IDRISI software packages include a set of tutorial exercises and data that guide the new user through the concepts of GIS and Image Processing while also introducing the features of IDRISI. The tutorial exercises are appropriate for use in either self-training or in classroom settings. Because of its low cost and advanced capabilities, IDRISI has always been a popular choice for teaching GIS and Image Processing at the university level. CartaLinx also includes a tutorial that teaches the fundamentals of vector topology and database manipulation as well as CartaLinx operation.

In 1990, the Clark Labs (then the IDRISI Project) signed a memorandum of understanding with UNITAR (United Nations Institute for Training and Research) to provide scientific assistance to its training programs through the development of curricula and training materials related to specific application areas. A series of workbooks, Explorations in Geographic Information Systems Technology, has been produced and is available from the Clark Labs. Each workbook includes an overview that explores in detail the theoretical aspects of the given application, a bibliography, and a set of exercises using real-world data to explore the issues raised.

Intensive hands-on introductory training sessions in GIS and Image Processing, typically lasting five days, are offered several times per year at Clark University and, on request, at other sites. These training sessions provide theoretical background material essential to the new GIS user, as well as training in using IDRISI and CartaLinx. Intermediate and advanced training sessions, customized to the needs of trainees, are also available.

IDRISI is not an Acronym!

A Muslim scholar of international reputation in the Mediterranean world of his day, Abu Abd Allah Muhammed al-Idrisi (1100-1166 A.D.) was born in a town on the North African coast, probably contemporary Ceuta, at that time, like much of Andalucia (southern Spain) and western North Africa, part of the Almoravid state. Educated at the University of Cordoba, and widely traveled in Europe, North Africa, the Middle East, and Central Asia, al-Idrisi was a cartographer and geographer of major significance during the medieval period. Commissioned by the Norman king Roger of Sicily to prepare a geographical survey of the world, al-Idrisi led a fifteen-year collaborative effort by scholars and technicians based at the Norman Court at Palermo. Based on direct field studies as well as archival sources, the maps and texts that resulted from that collaborative effort served as primary reference material for over 500 years. It is to this spirit of collaboration in geographic inquiry that the IDRISI software system is dedicated.

Exploring IDRISI

The best introduction to IDRISI is through the **Tutorial** which can be accessed through the Help menu of the IDRISI program. Parallel to working on the exercises, you should read the remainder of the **IDRISI Guide to GIS and Image Processing**.

The first three chapters present a general overview of IDRISI (this chapter), GIS, and Remote Sensing and Image Processing.

The next several chapters explore the use of the IDRISI system. The chapter **System Overview** describes the nature of the user interface. The chapter **Maps, Collections, and Data Structures** outlines the logic with which IDRISI organizes data and gives an overview of the file structures of the most commonly used data files. The **Display System** chapter discusses issues related to the display of geographic data and the interactive display features available for their exploration. The **Database Workshop** chapter describes the database management system, giving detailed information on all its func-

tions, including the ability to link the database to a map, and its ability to use structured query language (SQL). The **IDRISI Modules** chapter gives an overview of the capabilities of the IDRISI modules and their typical usage. It also outlines the logic of the menu structure. The chapter **IDRISI Modeling Tools** describes the use of IDRISI's Macro Modeler, Image Calculator, macro scripting language and API (COM Server) modeling tools. The **Database Development** chapter covers some of the important issues for the development and creation of GIS databases, especially techniques for importing data to IDRISI.

The **Georeferencing** chapter presents issues of geodesy, geodetic datums, projections and reference systems in understandable terminology. While many project-level applications of GIS and image processing do not require georeferencing to a geodetic system, integration of data with local or national government mapping will unquestionably require that the issues treated in this chapter be addressed.

The **Decision Support** chapters will be of particular interest to those involved with resource allocation and planning. It covers the special procedures required to undertake multi-criteria / multi-objective analyses, as well as decision making in the presence of uncertainty.

Several chapters are included that relate to the use of remotely-sensed data and image processing techniques. The **Image Restoration** chapter suggests methods for removing or diminishing the degree of random and systematic distortions that occur in imagery. A separate chapter on **Fourier Analysis** continues this discussion of methods for noise removal. The **Classification of Remotely Sensed Imagery** chapter outlines in detail the IDRISI approach to image classification, including the use of "soft" and "fuzzy" classifiers for this process. Use of hyperspectral data is also discussed in this chapter. The **RADAR Imaging and Analysis** chapter provides some suggestions for the use of radar imagery. The chapter on **Vegetation Indices** describes the vegetation index models included in IDRISI for the transformation of satellite imagery into images that indicate the relative amount of biomass present. The **Time Series/Change Analysis** chapter deals with an increasingly important set of tools in environmental monitoring. Topics covered include pairwise comparisons, procedures for distinguishing true change from natural variability, temporal profiling, and time series analysis by means of Principal Components Analysis.

Another group of chapters addresses issues of modeling continuous raster surfaces. In the **Anisotropic Cost Analysis** chapter, the brief discussion of cost distance procedures in the **Introduction to GIS** chapter is extended to consider the case of anisotropic forces and frictions (i.e., forces and frictions that act differently in different directions). These tools are somewhat experimental, but offer special opportunities for the modeling of dynamic phenomena such as groundwater flows, forest fire movements, oil spills, and so on. Three chapters focus on issues of spatial interpolation from sample data. The **Surface Interpolation** chapter gives an overview of the techniques commonly encountered in GIS and points out some of their relative advantages and disadvantages. It also indicates how these techniques are carried out in IDRISI. The **Triangulated Irregular Networks and Surface Generation** chapter details the IDRISI implementation of the TIN. The chapter **Geostatistics** presents background information for the use of advanced geostatistical procedures such as kriging and simulation.

This volume also contains a series of **Appendices** containing Georeferencing parameters, most importantly, detailed tables of constants used for transformation between map datums (yes, in Geodesy, the plural of datum is *datums*, and not *data!*), as well as error propagation formulae referred to in the **Decision Support** chapter.

The **Tutorial** manual is intended as a means of learning (and teaching) the IDRISI system and the basic tools used in GIS and image processing. The exercises are in a format suitable for classroom use as well as individual instruction. Literally thousands of users have learned the basics of GIS by means of these exercises.

In addition to the manuals described above, IDRISI also contains a very robust on-line Help System. This does not duplicate the information in the **IDRISI Guide**, but acts as a very important supplement to it. Specifically, the Help System contains detailed information on the use of every module in the IDRISI set. This includes information on operation, special notes, explanations of error messages, command line syntax, and so on. Every module has a help button that can be clicked on to get help for that module. The Help System can also be accessed by clicking on the Help menu item. You will find there a table of contents, index, and a keyword search facility. The Help System also contains a basic glossary and

detailed information about IDRISI file formats.

Contacting Clark Labs

We hope you enjoy your use of IDRISI Kilimanjaro. Our users are encouraged to provide feedback on their experience and methods of application.

To contact Clark Labs, our address is:

Clark Labs . Clark University . 950 Main Street . Worcester, MA . 01610-1477 . USA

To contact us by phone, fax, electronic mail, or to visit our web site:

Customer Support: +1.508.793.7526

Technical Inquiries: +1.508.793.7528

Fax: +1.508.793.8842

Internet: idrisi@clarku.edu

Web Site: www.clarklabs.org

Our office hours are from 09:00-17:00 hours US Eastern Time (-5 hours GMT Winter/-4 hours GMT Summer) Monday through Friday. An answering machine takes messages after hours.

Clark Labs Return Policy

Returns are only accepted due to installation difficulties and must receive prior authorization from Customer Support.

Clark Labs Technical Support

The Clark Labs is dedicated to providing registered users with quality technical support. For those who have purchased technical support we provide expert assistance on the following types of issues:

- Software Installation
- Tutorial Exercise Assistance
- Software Operation
- Import and Export of Supported Data Formats
- Identification of Appropriate Module(s) for Particular Analyses

To receive technical assistance, you must have purchased technical support and be running the latest available update of IDRISI or CartaLinx. We maintain free software updates (also known as patches) that any registered user can download from our web site at <http://www.clarklabs.org>. Check the version number of your installed software in the Help/About menu and then visit our web site's download area to compare with the update's release number. If your installed version has a lower release number, download and install the update. After updating, check if your problem has been resolved.

If you are experiencing a technical problem with a specific module or you are receiving unexpected results, the problem resolution may be in the software's Help System. Check the Help System's entry for the module/command you are running. Most IDRISI commands include supplemental information concerning the module's limitations in the respective "Notes" section.

If the update and the Help System information have not resolved your problem, please contact our Technical Assistance Staff. The preferred method of contact is via e-mail and for your convenience, we provide online forms on our web site's Technical Assistance page: <http://www.clarklabs.org/TechnicalSupport.asp?cat=3>. These forms provide us with all of the information we require to resolve your problem as quickly as possible. In times of high support load, users requesting assistance via the web forms will be prioritized. If you choose to call by phone, the Technical Assistance Staff may ask that you send an e-mail, especially if the problem is complicated.

You must provide the following in your initial contact with us:

- Your Customer ID number, name, phone and e-mail address (if available).
- The name and version number of the Clark Labs product you are using.
- A description of your hardware and operating system.
- A detailed description of the problem you are experiencing. This should include a list and description of the data sets involved, a description of the operation you are trying to accomplish, and a step by step description of what you have tried so far (with the specific values you entered into the module's dialog box). You should also include the exact text of any error messages you receive.

Clark Labs also maintains Idrisi-L as an unmoderated discussion list for our users to share ideas and provide each other with assistance.

To send a message to the list, use the address: Idrisi-L@ClarkLabs.org

To subscribe or unsubscribe, send a message to: Idrisi-L@ClarkLabs.org

If you are subscribing, in the body of the message, type: subscribe

If you are unsubscribing, in the body of the message, type: unsubscribe

You will receive a confirmation message when you have successfully subscribed or unsubscribed.

The Clark Labs Staff

As of April 2003, the Clark Labs Staff included:

Clement Aga Alo (Ghana), Software Tester

Thin Thiri Aung (Burma), Student Assistant

David Benz (USA), Technical Support Assistant

Laurie Canavan (USA), Assistant Director

Hao Chen (China), Research Assistant

Stefano Crema (Brazil), Research Associate

Deanna Darling (USA), Customer Service Assistant

Deepesh Deshmukh (India), MIS Manager
Nafi Diallo (Senegal), Research Assistant
Ron Eastman (Canada), Director / Programmer
Anne Gibson (USA), Cartographic Lab Manager
Sean Grimland (USA), Software Tester
Ziying Jiang (China), Research Assistant
Maction Komwa (Malawi), Software Tester
Zhe Li (China), Research Assistant
Arianna Lomas Gomez (Spain), Student Assistant
Ivan Lucena (Brazil), Research Associate
Kristen Mazeika (USA), Office Assistant
McArd Joseph Mlotha (Malawi), Software Tester
Louis Paladino (USA), Technical Support Assistant
Jon Papazian (USA), Software Tester
Jinwei Qiu (China), Software Tester
Florencia Sangermano (Argentina), Software Tester
Joseph Spencer (USA), Software Tester
Diane Sutter (USA), Customer Service Manager
Scott Sweet (USA), Software Tester
Olufunmilayo Thonttch (Nigeria), Software Tester
James Toledano (USA), Executive Director
Hua Wang (China), Software Tester
Honglei Zhu (China), Senior Research Associate

Introduction to GIS

A Geographic Information System (GIS) is a computer-assisted system for the acquisition, storage, analysis and display of geographic data. Today, a variety of software tools are available to assist this activity. However, they can differ from one another quite significantly, in part because of the way they represent and work with geographic data, but also because of the relative emphasis they place on these various operations. In this chapter, we will explore these differences as a means of understanding the special characteristics of the IDRISI system.

Components of a GIS

Although we think of a GIS as a single piece of software, it is typically made up of a variety of different components. Figure 2-1 gives a broad overview of the software components typically found in a GIS. Not all systems have all of these elements, but to be a true GIS, an essential group must be found.

Spatial and Attribute Database

Central to the system is the database—a collection of maps and associated information in digital form. Since the database is concerned with earth surface features, it can be seen to be comprised of two elements—a spatial database describing the geography (shape and position) of earth surface features, and an attribute database describing the characteristics or qualities of these features. Thus, for example, we might have a property parcel defined in the spatial database and qualities such as its land use, owner, property valuation, and so on, in the attribute database.

In some systems, the spatial and attribute databases are rigidly distinguished from one another, while in others they are closely integrated into a single entity—hence the line extending only half-way through the middle circle of Figure 2-1. IDRISI is of the type that integrates the two components into one. However, it also offers the option of keeping some

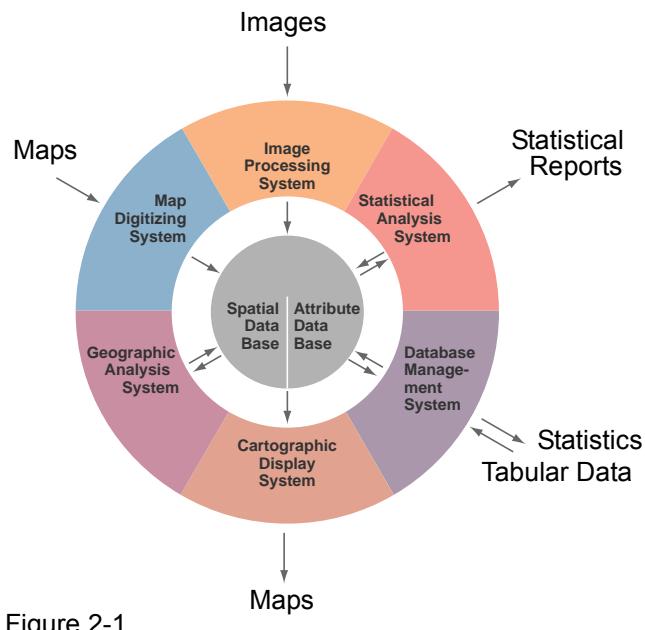


Figure 2-1

elements of the attribute database quite separate. This will be explored further below when we examine techniques for the digital representation of map data.

Cartographic Display System

Surrounding the central database, we have a series of software components. The most basic of these is the Cartographic Display System. The Cartographic Display System allows one to take selected elements of the database and produce map output on the screen or some hardcopy device such as a printer or plotter. The range of cartographic production capabilities among GIS software systems is great. Most provide only very basic cartographic output, and rely upon the use of high quality publication software systems for more sophisticated production needs such as color separation.

IDRISI allows for highly interactive and flexible on-screen cartographic composition, including the specification of multiple data layers, customization and positioning of map elements such as annotation, scale bars, insets and so forth, and customized color and symbol sets. IDRISI map compositions may be saved for later display, printed to Windows-compatible devices, and exported in a variety of common desktop publishing formats.

Software systems that are only capable of accessing and displaying elements of the database are often referred to as Viewers or Electronic Atlases.

Map Digitizing System

After cartographic display, the next most essential element is a Map Digitizing System. With a Map Digitizing System, one can take existing paper maps and convert them into digital form, thus further developing the database. In the most common method of digitizing, one attaches the paper map to a digitizing tablet or board, then traces the features of interest with a stylus or puck according to the procedures required by the digitizing software. Many Map Digitizing Systems also allow for editing of the digitized data.

The CartaLinx software package, also developed and distributed by Clark Labs, provides complete digitizing and vector editing capability and is fully compatible with IDRISI. There are also a number of independent digitizing software packages that support the IDRISI data format.

Scanners may also be used to digitize data such as aerial photographs. The result is a graphic image, rather than the outlines of features that are created with a digitizing tablet. Scanning software typically provides users with a variety of standard graphics file formats for export. These files are then imported into the GIS. IDRISI supports import of TIF and BMP graphics file formats.

Digitizing packages, Computer Assisted Design (CAD), and Coordinate Geometry (COGO) are examples of software systems that provide the ability to add digitized map information to the database, in addition to providing cartographic display capabilities.

Database Management System

The next logical component in a GIS is a Database Management System (DBMS). Traditionally, this term refers to a type of software that is used to input, manage and analyze attribute data. It is also used in that sense here, although we need to recognize that spatial database management is also required. Thus, a GIS typically incorporates not only a traditional DBMS, but also a variety of utilities to manage the spatial and attribute components of the geographic data stored.

With a DBMS, it is possible to enter attribute data, such as tabular information and statistics, and subsequently extract specialized tabulations and statistical summaries to provide new tabular reports. However, most importantly, a Database Management System provides us with the ability to analyze attribute data. Many map analyses have no true spatial component, and for these, a DBMS will often function quite well. For example, we might use the system to find all property parcels where the head of the household is single but with one or more child dependents, and to produce a map of the result. The final product (a map) is certainly spatial, but the analysis itself has no spatial qualities whatsoever. Thus, the double arrows between the DBMS and the attribute database in Figure 2-1 signify this distinctly non-spatial form of data analysis.

In IDRISI, a DBMS is provided by Database Workshop. One can perform analyses in Database Workshop, then immediately apply the results to the proper spatial data, viewing the results as a map. In addition to Database Workshop, an extensive set of program modules is also available for spatial and attribute data management.

Software that provides cartographic display, map digitizing, and database query capabilities are sometimes referred to as Automated Mapping and Facilities Management (AM/FM) systems.

Geographic Analysis System

Up to this point, we have described a very powerful set of capabilities—the ability to digitize spatial data and attach attributes to the features stored, to analyze these data based on those attributes, and to map out the result. Indeed, there are a variety of systems on the market that have just this set of abilities, many of which will call themselves a GIS. But useful as this is, such a set of capabilities does not necessarily constitute a full GIS. The missing component is the ability to analyze data based on truly spatial characteristics. For this we need a Geographic Analysis System.

With a Geographic Analysis System, we extend the capabilities of traditional database query to include the ability to analyze data based on their location. Perhaps the simplest example of this is to consider what happens when we are concerned with the joint occurrence of features with different geographies. For example, suppose we want to find all areas of residential land on bedrock types associated with high levels of radon gas. This is a problem that a traditional DBMS simply cannot solve because bedrock types and landuse divisions do not share the same geography. Traditional database query is fine as long as we are talking about attributes belonging to the same features. But when the features are different, it cannot cope. For this we need a GIS. In fact, it is this ability to compare different features based on their common geographic occurrence that is the hallmark of GIS. This analysis is accomplished through a process called *overlay*, thus named because it is identical in character to overlaying transparent maps of the two entity groups on top of one another.

Like the DBMS, the Geographic Analysis System is seen in Figure 2-1 to have a two way interaction with the database—the process is distinctly analytical in character. Thus, while it may access data from the database, it may equally contribute the results of that analysis as a new addition to the database. For example, we might look for the joint occurrence of lands on steep slopes with erodible soils under agriculture and call the result a map of soil erosion risk. This risk map was not in the original database, but was derived based on existing data and a set of specified relationships. Thus the analytical capabilities of the Geographic Analysis System and the DBMS play a vital role in extending the database through the addition of knowledge of relationships between features.

While overlay is still the hallmark of GIS, computer-assisted geographic analysis has matured enormously over the past several years. However, for now it is sufficient to note that it is this distinctly geographic component that gives a true GIS its identity. In IDRISI, these abilities are extensive and form the foundation of the software system.

Image Processing System

In addition to these essential elements of a GIS—a Cartographic Display System, a Map Digitizing System, a Database Management System and a Geographic Analysis System—some software systems also include the ability to analyze remotely sensed images and provide specialized statistical analyses. IDRISI is of this type. Image processing software allows one to take raw remotely sensed imagery (such as Landsat or SPOT satellite imagery) and convert it into interpreted map data according to various classification procedures. In recognition of its major importance as a technique for data acquisition, IDRISI offers a broad set of tools for the computer-assisted interpretation of remotely sensed data.

Statistical Analysis System

For statistical analysis, IDRISI offers both traditional statistical procedures as well as some specialized routines for the statistical analysis of spatial data. Geographers have developed a series of specialized routines for the statistical description of spatial data, partly because of the special character of spatial data, but also because spatial data pose special problems for inferences drawn from statistical procedures.

Decision Support System

While decision support is one of the most important functions of a GIS, tools designed especially for this are relatively few in most GIS software. However, IDRISI includes several modules specifically developed to aid in the resource allocation decision making process. These include modules that incorporate error into the process, help in the construction of multi-criteria suitability maps under varying levels of tradeoff, and address allocation decisions when there are multiple objectives involved. Used in conjunction with the other components of the system, these modules provide a powerful tool for resource allocation decision makers.

Map Data Representation

The way in which the software components mentioned above are combined is one aspect of how Geographic Information Systems vary. However, an even more fundamental distinction is how they represent map data in digital form.

A Geographic Information System stores two types of data that are found on a map—the geographic definitions of earth surface features and the attributes or qualities that those features possess. Not all systems use the same logic for achieving this. Nearly all, however, use one or a combination of both of the fundamental map representation techniques: *vector* and *raster*.

Vector

With vector representation, the boundaries or the course of the features are defined by a series of points that, when joined with straight lines, form the graphic representation of that feature. The points themselves are encoded with a pair of numbers giving the X and Y coordinates in systems such as latitude/longitude or Universal Transverse Mercator grid coordinates. The attributes of features are then stored with a traditional database management (DBMS) software program. For example, a vector map of property parcels might be tied to an attribute database of information containing the address, owner's name, property valuation and land use. The link between these two data files can be a simple identifier number that is given to each feature in the map (Figure 2-2).

Raster

The second major form of representation is known as raster. With raster systems, the graphic representation of features and the attributes they possess are merged into unified data files. In fact, we typically do not define features at all. Rather, the study area is subdivided into a fine mesh of grid cells in which we record the condition or attribute of the earth's surface at that point (Figure 2-2). Each cell is given a numeric value which may then represent either a feature identifier, a qualitative attribute code or a quantitative attribute value. For example, a cell could have the value "6" to indicate that it belongs to District 6 (a feature identifier), or that it is covered by soil type 6 (a qualitative attribute), or that it is 6 meters above sea level (a quantitative attribute value). Although the data we store in these grid cells do not necessarily refer to phenomena that can be seen in the environment, the data grids themselves can be thought of as *images* or *layers*, each depicting one type of information over the mapped region. This information can be made visible through the use of a raster display. In a raster display, such as the screen on your computer, there is also a grid of small cells called *pixels*. The word pixel is a contraction of the term *picture element*. Pixels can be made to vary in their color, shape or grey tone. To make an image, the cell values in the data grid are used to regulate directly the graphic appearance of their corresponding pixels. Thus in a raster system, the data directly controls the visible form we see.

Raster versus Vector

Raster systems are typically data intensive (although good data compaction techniques exist) since they must record data at every cell location regardless of whether that cell holds information that is of interest or not. However, the advantage is that geographical space is uniformly defined in a simple and predictable fashion. As a result, raster systems have substantially more analytical power than their vector counterparts in the analysis of continuous space¹ and are thus ideally

suited to the study of data that are continuously changing over space such as terrain, vegetation biomass, rainfall and the like. The second advantage of raster is that its structure closely matches the architecture of digital computers. As a result, raster systems tend to be very rapid in the evaluation of problems that involve various mathematical combinations of the data in multiple layers. Hence they are excellent for evaluating environmental models such as soil erosion potential and forest management suitability. In addition, since satellite imagery employs a raster structure, most raster systems can easily incorporate these data, and some provide full image processing capabilities.

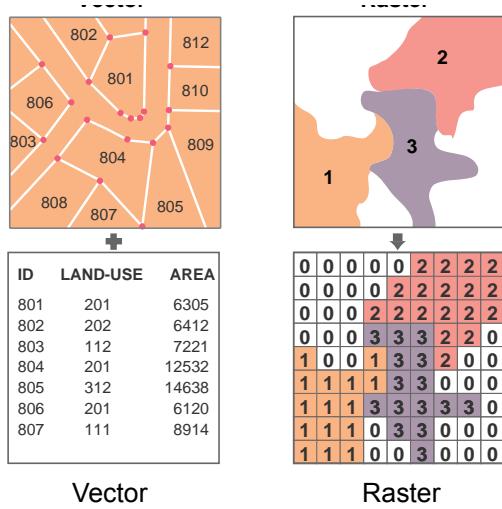


Figure 2-2

While raster systems are predominantly analysis oriented, vector systems tend to be more database management oriented. Vector systems are quite efficient in their storage of map data because they only store the boundaries of features and not that which is inside those boundaries. Because the graphic representation of features is directly linked to the attribute database, vector systems usually allow one to roam around the graphic display with a mouse and query the attributes associated with a displayed feature, such as the distance between points or along lines, the areas of regions defined on the screen, and so on. In addition, they can produce simple thematic maps of database queries, such as one showing all sewer line sections over one meter in diameter installed before 1940.

Compared to their raster counterparts, vector systems do not have as extensive a range of capabilities for analyses over continuous space. They do, however, excel at problems concerning movements over a network and can undertake the most fundamental of GIS operations that will be sketched out below. For many, it is the

simple database management functions and excellent mapping capabilities that make vector systems attractive. Because of the close affinity between the logic of vector representation and traditional map production, vector systems are used to produce maps that are indistinguishable from those produced by traditional means. As a result, vector systems are very popular in municipal applications where issues of engineering map production and database management predominate.

Raster and vector systems each have their special strengths. As a result, IDRISI incorporates elements from both representational techniques. Though it is primarily a raster analytical system, IDRISI does employ vector data structures as a major form of map data display and exchange. In addition, fundamental aspects of vector database management are also provided.

Geographic Database Concepts

Organization

Whether we use a raster or vector logic for spatial representation, we begin to see that a geographic database—a complete database for a given region—is organized in a fashion similar to a collection of maps (Figure 2-3). Vector systems may come closest to this logic with what are known as *coverages*—map-like collections that contain the geographic definitions of a set of features and their associated attribute tables. However, they differ from maps in two ways. First, each will typically contain information on only a single feature type, such as property parcels, soils polygons, and the like. Second, they may contain a whole series of attributes that pertain to those features, such as a set of census information for city blocks.

1. The basic data structure of vector systems can best be described as a network. As a result, it is not surprising to find that vector systems have excellent capabilities for the analysis of network space. Thus the difference between raster and vector is less one of inherent ability than one of the difference in the types of space they describe.

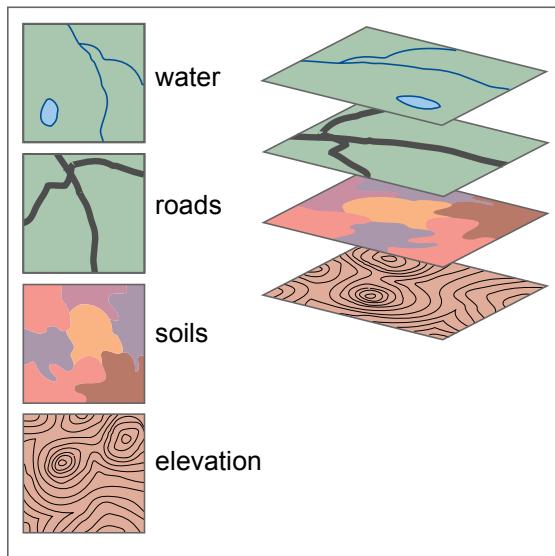


Figure 2-3

Raster systems also use this map-like logic, but usually divide data sets into unitary *layers*. A layer contains all the data for a single attribute. Thus one might have a soils layer, a roads layer and a land use layer. A few raster systems, including IDRISI, can link a feature identifier layer (a layer that contains the identifiers of the features located at each grid cell) with attribute tables. More commonly, separate layers exist for each attribute and on-screen displays and paper maps are produced from these, either singly or in combination.

Although there are subtle differences, for all intents and purposes, raster layers and vector coverages can be thought of as simply different manifestations of the same concept—the organization of the database into elementary map-like themes. Layers and coverages differ from traditional paper maps, however, in an important way. When map data are encoded in digital form (digitized), scale differences are removed. The digital data may be displayed or printed at any scale. More importantly, digital data layers that were derived from paper maps of different scales, but covering the same geographic area, may be combined.

In addition, many GIS packages, including IDRISI, provide utilities for changing the projection and reference system of digital layers. This allows multiple layers, digitized from maps having various projections and reference systems, to be converted to a common system.

With the ability to manage differences of scale, projection and reference system, layers can be merged with ease, eliminating a problem that has traditionally hampered planning activities with paper maps. It is important to note, however, that the issue of *resolution* of the information in the data layers remains. Although features digitized from a poster-sized world map *could* be combined in a GIS with features digitized from a very large scale local map, such as a city street map, this would normally not be done. The level of accuracy and detail of the digital data can only be as good as that of the original maps.

Georeferencing

All spatial data files in a GIS are georeferenced. Georeferencing refers to the location of a layer or coverage in space as defined by a known coordinate referencing system. With raster images, a common form of georeferencing is to indicate the reference system (e.g., latitude/longitude), the reference units (e.g., degrees) and the coordinate positions of the left, right, top and bottom edges of the image. The same is true of vector data files, although the left, right, top and bottom edges now refer to what is commonly called the *bounding rectangle* of the coverage—a rectangle which defines the limits of

the mapped area.² This information is particularly important in an integrated GIS such as IDRISI since it allows raster and vector files to be related to one another in a reliable and meaningful way. It is also vital for the referencing of data values to actual positions on the ground.

Georeferencing is an extremely important consideration when using GIS. Therefore a separate chapter later in this volume treats this topic in detail.

Analysis in GIS

The organization of the database into layers is not simply for reasons of organizational clarity. Rather, it is to provide rapid access to the data elements required for geographic analysis. Indeed, the *raison d'être* for GIS is to provide a medium for geographic analysis.

The analytical characteristics of GIS can be looked at in two ways. First, one can look at the tools that GIS provides. Then one can look at the kinds of operations that GIS allows. Regardless of whether we are using a raster or a vector system, we will tend to find that the tools fall into four basic groups and that the operations undertaken fall into three.

Analytical Tools

Database Query

The most fundamental of all tools provided by a GIS are those involved with Database Query. Database query simply asks questions about the currently-stored information. In some cases, we query by location—*what land use is at this location?* In other cases, we query by attribute—*what areas have high levels of radon gas?* Sometimes we undertake simple queries such as those just illustrated, and at other times we ask about complex combinations of conditions—*show me all wetlands that are larger than 1 hectare and that are adjacent to industrial lands.*

In most systems, including IDRISI, these query operations are undertaken in two steps. The first step, called a *reclassification*, creates a new layer of each individual condition of interest (Figure 2-4). For example, consider a query to find residential areas on bedrock associated with high levels of radon gas. The first step would be to create a layer of residential areas alone by reclassifying all landuse codes into only two—a 1 whenever an area is residential and a 0 for all other cases. The resulting layer is known as a *Boolean* layer since it shows only those areas that meet the condition (1 = true, residential) and those that don't (0 = false, not residential). Boolean layers are also called *logical* layers since they show only true/false relationships. They are also sometimes called *binary* layers since they contain only zeros and ones. We will avoid using that term, however, since it also describes a particular kind of data storage format. Here we will call them Boolean layers.

Once the residential layer has been created, a geology layer is then also reclassified to create a Boolean layer showing areas with bedrock associated with high levels of radon gas. At this point we can combine the two conditions using an overlay operation (Figure 2-4). As mentioned previously, it is only a GIS that can combine conditions such as this that involve features with different geographies. Typically, an overlay operation in GIS will allow the production of new layers based on some logical or mathematical combination of two or more input layers. In the case of database query, the key logical operations of interest are the AND and OR relational operators, also known as the INTERSECTION and UNION operations respectively. Here we are looking for cases of residential land AND high radon gas—the logical intersection of our

2. The bounding rectangle is defined by the study region of interest and does not necessarily refer to the actual minimum and maximum coordinates in the data file.

two Boolean layers.

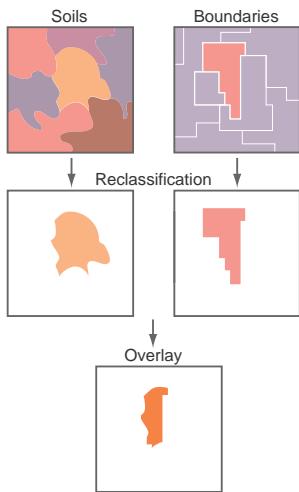


Figure 2-4

Map Algebra

The second set of tools that a GIS will typically provide is that for combining map layers mathematically. Modeling in particular requires the ability to combine layers according to various mathematical equations. For example, we might have an equation that predicts mean annual temperature as a result of altitude. Or, as another example, consider the possibility of creating a soil erosion potential map based on factors of soil erodability, slope gradient and rainfall intensity. Clearly we need the ability to modify data values in our map layers by various mathematical operations and transformations and to combine factors mathematically to produce the final result.

The Map Algebra tools will typically provide three different kinds of operations:

1. the ability to mathematically modify the attribute data values by a constant (i.e., scalar arithmetic);
2. the ability to mathematically transform attribute data values by a standard operation (such as the trigonometric functions, log transformations and so on);
3. the ability to mathematically combine (such as add, subtract, multiply, divide) different data layers to produce a composite result.

This third operation is simply another form of overlay—mathematical overlay, as opposed to the logical overlay of database query.

To illustrate this, consider a model for snow melt in densely forested areas:³

$$M = (0.19T + 0.17D)$$

where M is the melt rate in cm/day, T is the air temperature and D is the dewpoint temperature. Given layers of the air temperatures and dewpoints for a region of this type, we could clearly produce a snow melt rate map. To do so would require multiplying the temperature layer by 0.19 (a scalar operation), the dewpoint layer by 0.17 (another scalar opera-

3. Equation taken from Dunne, T., and Leopold, L.B., (1978) *Water in Environmental Planning*, (W.H. Freeman and Co: San Francisco), 480.

tion) and then using overlay to add the two results. While simple in concept, this ability to treat map layers as variables in algebraic formulas is an enormously powerful capability.

Distance Operators

The third tool group provided by GIS consists of the Distance Operators. As the name suggests, these are a set of techniques where distance plays a key role in the analysis undertaken. Virtually all systems provide the tools to construct buffer zones—areas within a specified distance of designated target features. Some can also evaluate the distance of all locations to the nearest of a set of designated features, while others can even incorporate frictional effects and barriers in distance calculations (Figure 2-5).

When frictional effects are incorporated, the distance calculated is often referred to as a *cost distance*. This name is used because movement through space can be considered to incur costs, either in money, time or effort. Frictions increase those costs. When the costs of movement from one or more locations are evaluated for an entire region, we often refer to the result as a *cost surface* (Figure 2-5). In this case, areas of low cost (presumably near to the starting point) can be seen as valleys and areas of high cost as hills. A cost surface thus has its lowest point(s) at the starting location(s) and its highest point(s) at the locations that are farthest away (in the sense of the greatest accumulated cost).⁴

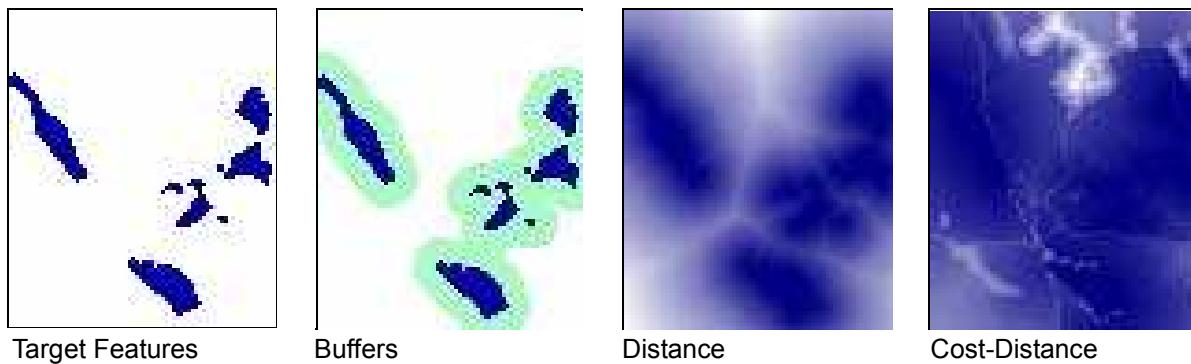


Figure 2-5

There may be cases in which frictions do not affect the cost of movement the same way in all directions. In other words, they act *anisotropically*. For example, going up a steep slope might incur a cost that would be higher than going down the same steep slope. Thus the direction of movement through the friction is important, and must be taken into account when developing the cost surface. IDRISI provides modules to model this type of cost surface which are explained in detail in the **Anisotropic Cost Analysis** chapter.

Given the concept of a cost surface, Geographic Information Systems also commonly offer *least cost path analysis*—another important distance operation. As the name suggests, our concern is to evaluate the least-cost path between two locations. The cost surface provides the needed information for this to be evaluated (Figure 2-8).

Regardless of how distance is evaluated, by straight line distance or by cost distance, another commonly provided tool is *allocation*. With allocation, we assign locations to the nearest of a set of designated features. For example, we might establish a set of health facilities and then wish to allocate residents to their nearest facility, where "near" might mean linear distance, or a cost-distance such as travel time.

4. It should be noted here that a cost surface as just described can only be evaluated with a raster system. For vector systems, the closest equivalent would be cost distances evaluated over a network. Here we see a simple, but very powerful, illustration of the differences between raster and vector systems in how they conceive of space.

Context Operators

Finally, most Geographic Information Systems provide a variety of Context Operators (also known as *neighborhood* or *local operators*). With context operators, we create new layers based on the information on an existing map and the context in which it is found. One of the simplest examples of this is *surface analysis* where we use a digital elevation model to produce a slope layer by examining the heights of locations in comparison to the heights of neighboring locations. In a similar fashion, the aspect (the direction of maximum downward slope) can also be evaluated. We might also position an artificial light source and calculate a shaded relief model. These context operator products and the elevation model from which they were derived are illustrated in Figure 2-6.

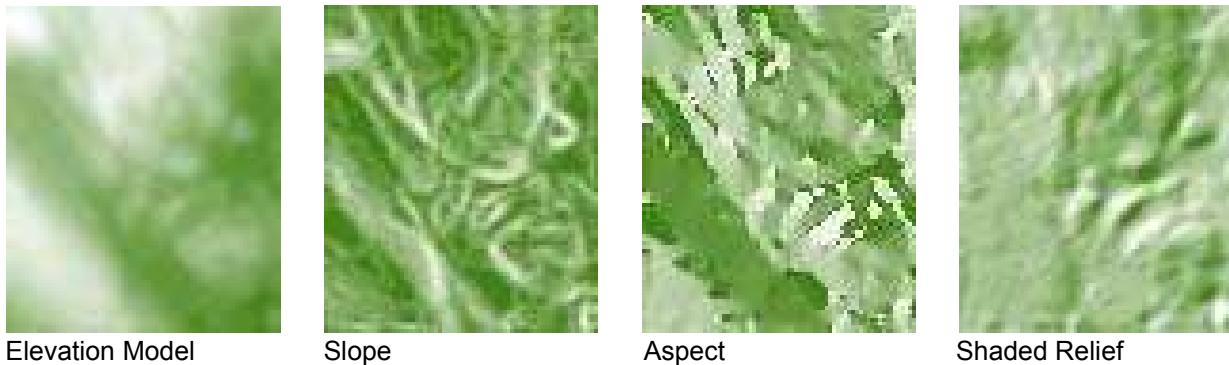


Figure 2-6

A second good example of a context operator is a *digital filter*. Digital filters operate by changing values according to the character of neighboring values. For example, a surface of terrain heights can be smoothed by replacing values with the average of the original height and all neighboring heights. Digital filters have a broad range of applications in GIS and remote sensing, ranging from noise removal to the visual enhancement of images.

Because of their simple and uniform data structure, raster systems tend to offer a broad range of context operators. In IDRISI, for example, these include surface analysis and digital filtering, identification of contiguous areas, watershed analysis, viewshed analysis (an evaluation of all areas in view of one or more designated features) and a special supply/demand modeling procedure where demands are satisfied by taking supplies in a radial fashion from neighboring locations.

Analytical Operations

Given these basic tools, a broad range of analytical operations can be undertaken. However, it would appear that most of these fall into one of three basic groups: Database Query, Derivative Mapping and Process Modeling.

Database Query

With database query, we are simply selecting out various combinations of variables for examination. The tools we use are largely the Database Query tools previously discussed (hence the name), but also include various measurement and statistical analysis procedures. The key thing that distinguishes this kind of analysis is that we have taken out no more than we have put into the system. While we may extract combinations we have never examined before, the system provides us with no new information—we are simply making a withdrawal from a data bank we have built up.

One of the key activities in database query is pattern seeking. Typically we are looking for spatial patterns in the data that may lead us to hypothesize about relationships between variables.

Derivative Mapping

With derivative mapping, we combine selected components of our database to yield new derivative layers. For example, we might take our digital elevation data to derive slope gradients, and then take our slope data and combine it with information on soil type and rainfall regime to produce a new map of soil erosion potential. This new map then becomes an addition to our growing database.

How is it that we can create new data from old? Unlike database query where we simply extracted information that was already in the database, with derivative mapping we take existing information and add to it something new—*knowledge of relationships* between database elements. We can create a soil erosion potential map using a digital elevation layer, a soils layer and a rainfall regime layer, only if we know the relationship between those factors and the new map we are creating. In some cases, these relationships will be specified in logical terms (such as creating a suitability map for industrial location based on the condition that it be on existing forest land, outside protective buffers around wetlands and on low slopes) and we will use our database query tools. In other cases, however, these relationships will be specified in mathematical terms and we will rely heavily on the Map Algebra tools. Regardless, the relationships that form the model will need to be known.

In some cases, the relationship models can be derived on logical or theoretical grounds. However, in many instances it is necessary that the relationships be determined by empirical study. Regression analysis, for example, is one very common way in which empirical testing is used to develop a mathematical relationship between variables. If one takes the soil erosion example, one might set up a series of test sites at which the soil erosion is measured along with the slope, soil type and rainfall data. These sample points would then be used to develop the equation relating soil erosion to these variables. The equation would then be used to evaluate soil erosion potential over a much broader region.

Process Modeling

Database query and derivative mapping make up the bulk of GIS analysis undertaken today. However, there is a third area that offers incredible potential—Process or Simulation Modeling.

With process modeling, we also bring something new to the database—*knowledge of process*. Process refers to the causal chain by which some event takes place. For example, a simple model of fuel-wood demand satisfaction might run as follows:

1. Take all the wood you need (if you can) from your present location.
2. If your demand is satisfied or if you have traveled more than 10 kilometers from home, go to step 4.
3. If your demand is not met, move to an immediately adjacent location not already visited and repeat step 1.
4. Stop.

Process modeling is a particularly exciting prospect for GIS. It is based on the notion that in GIS, our database doesn't simply represent an environment, *it is an environment!* It is a surrogate environment, capable of being measured, manipulated and acted upon by geographic and temporal processes. Our database thus acts as a laboratory for the exploration of processes in a complex environment. Traditionally, in science, we have had to remove that complexity in order to understand processes in isolation. This has been an effective strategy and we have learned much from it. However, technologies such as GIS now offer the tools to reassemble those simple understandings in order to gain an understanding and appreciation of how they act in the full complexity of a real environmental setting. Often even very simple understandings yield complex patterns when allowed to interact in the environment.

A different sort of process, the decision making process, may also be supported and in some ways modeled with the use of GIS. GIS technology is becoming more important as a tool for decision support. Indeed, even the simplest database query results may prove to be invaluable input to the decision maker. However, the more complex *process* of decision making, in which decision makers often think in terms of multiple criteria, soft boundaries (non-Boolean) and levels of acceptable risk, may also be modeled using GIS. IDRISI provides a suite of decision support modules to help decision makers develop more explicitly rational and well-informed decisions. The **Decision Making** chapter discusses this important use of GIS in detail and provides illustrative case examples.

Despite its evident attraction, process modeling, both in environmental processes and decision making, is still a fairly uncommon activity in GIS. The reason is quite simple. While more and more modeling tools are becoming available in the GIS, it is not uncommon that the process of interest requires a capability not built into the system. These cases require the creation of a new program module. Many systems are not well set up for the incorporation of user-developed routines. IDRISI, however, has been designed so programs in any computer language may be merged into the system and called from the IDRISI interface.

The Philosophy of GIS

GIS has had an enormous impact on virtually every field that manages and analyzes spatially distributed data. For those who are unfamiliar with the technology, it is easy to see it as a magic box. The speed, consistency and precision with which it operates is truly impressive, and its strong graphic character is hard to resist. However, to experienced analysts, the philosophy of GIS is quite different. With experience, GIS becomes simply an extension of one's own analytical thinking. The system has no inherent answers, only those of the analyst. It is a tool, just like statistics is a tool. It is a tool for thought.

Investing in GIS requires more than an investment in hardware and software. Indeed, in many instances this is the least issue of concern. Most would also recognize that a substantial investment needs to be placed in the development of the database. However, one of the least recognized yet most important investments is in the analysts who will use the system. The system and the analyst cannot be separated—one is simply an extension of the other. In addition, the process of incorporating GIS capabilities into an institution requires an investment in long-term and organization-wide education and training.

In many ways, learning GIS involves learning to think—learning to think about patterns, about space and about processes that act in space. As you learn about specific procedures, they will often be encountered in the context of specific examples. In addition, they will often have names that suggest their typical application. But resist the temptation to categorize these routines. Most procedures have many more general applications and can be used in many novel and innovative ways. Explore! Challenge what you see! What you will learn goes far beyond what this or any software package can provide.

Introduction to Remote Sensing and Image Processing

Of all the various data sources used in GIS, one of the most important is undoubtedly that provided by remote sensing. Through the use of satellites, we now have a continuing program of data acquisition for the entire world with time frames ranging from a couple of weeks to a matter of hours. Very importantly, we also now have access to remotely sensed images in digital form, allowing rapid integration of the results of remote sensing analysis into a GIS.

The development of digital techniques for the restoration, enhancement and computer-assisted interpretation of remotely sensed images initially proceeded independently and somewhat ahead of GIS. However, the raster data structure and many of the procedures involved in these *Image Processing Systems* (IPS) were identical to those involved in raster GIS. As a result, it has become common to see IPS software packages add general capabilities for GIS, and GIS software systems add at least a fundamental suite of IPS tools. IDRISI is a combined GIS and image processing system that offers advanced capabilities in both areas.

Because of the extreme importance of remote sensing as a data input to GIS, it has become necessary for GIS analysts (particularly those involved in natural resource applications) to gain a strong familiarity with IPS. Consequently, this chapter gives an overview of this important technology and its integration with GIS. The Image Processing exercises in the **Tutorial** illustrate many of the concepts presented here.

Definition

Remote sensing can be defined as any process whereby information is gathered about an object, area or phenomenon without being in contact with it. Our eyes are an excellent example of a remote sensing device. We are able to gather information about our surroundings by gauging the amount and nature of the reflectance of visible light energy from some external source (such as the sun or a light bulb) as it reflects off objects in our field of view. Contrast this with a thermometer, which must be in contact with the phenomenon it measures, and thus is not a remote sensing device.

Given this rather general definition, the term *remote sensing* has come to be associated more specifically with the gauging of interactions between earth surface materials and electromagnetic energy. However, any such attempt at a more specific definition becomes difficult, since it is not always the natural environment that is sensed (e.g., art conservation applications), the energy type is not always electromagnetic (e.g., sonar) and some procedures gauge natural energy emissions (e.g., thermal infrared) rather than interactions with energy from an independent source.

Fundamental Considerations

Energy Source

Sensors can be divided into two broad groups—*passive* and *active*. Passive sensors measure ambient levels of existing sources of energy, while active ones provide their own source of energy. The majority of remote sensing is done with passive sensors, for which the sun is the major energy source. The earliest example of this is photography. With airborne cameras we have long been able to measure and record the reflection of light off earth features. While aerial photography is still a major form of remote sensing, newer solid state technologies have extended capabilities for viewing in the visible and near-infrared wavelengths to include longer wavelength solar radiation as well. However, not all passive sensors use energy from the sun. Thermal infrared and passive microwave sensors both measure natural earth energy emissions. Thus

the passive sensors are simply those that do not themselves supply the energy being detected.

By contrast, active sensors provide their own source of energy. The most familiar form of this is flash photography. However, in environmental and mapping applications, the best example is RADAR. RADAR systems emit energy in the microwave region of the electromagnetic spectrum (Figure 3-1). The reflection of that energy by earth surface materials is then measured to produce an image of the area sensed.

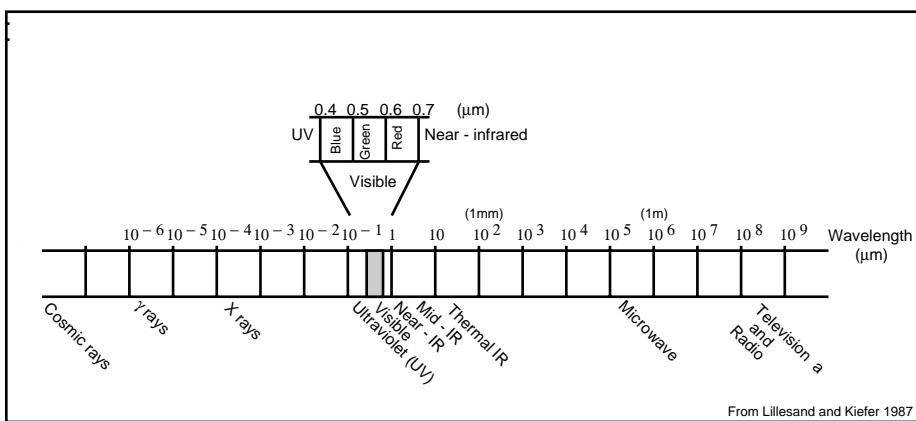


Figure 3-1: The Electromagnetic Spectrum

Wavelength

As indicated, most remote sensing devices make use of electromagnetic energy. However, the electromagnetic spectrum is very broad and not all wavelengths are equally effective for remote sensing purposes. Furthermore, not all have significant interactions with earth surface materials of interest to us. Figure 3-1 illustrates the electromagnetic spectrum. The atmosphere itself causes significant absorption and/or scattering of the very shortest wavelengths. In addition, the glass lenses of many sensors also cause significant absorption of shorter wavelengths such as the ultraviolet (UV). As a result, the first significant *window* (i.e., a region in which energy can significantly pass through the atmosphere) opens up in the visible wavelengths. Even here, the blue wavelengths undergo substantial attenuation by atmospheric scattering, and are thus often left out in remotely sensed images. However, the green, red and near-infrared (IR) wavelengths all provide good opportunities for gauging earth surface interactions without significant interference by the atmosphere. In addition, these regions provide important clues to the nature of many earth surface materials. Chlorophyll, for example, is a very strong absorber of red visible wavelengths, while the near-infrared wavelengths provide important clues to the structures of plant leaves. As a result, the bulk of remotely sensed images used in GIS-related applications are taken in these regions.

Extending into the middle and thermal infrared regions, a variety of good windows can be found. The longer of the middle infrared wavelengths have proven to be useful in a number of geological applications. The thermal regions have proven to be very useful for monitoring not only the obvious cases of the spatial distribution of heat from industrial activity, but a broad set of applications ranging from fire monitoring to animal distribution studies to soil moisture conditions.

After the thermal IR, the next area of major significance in environmental remote sensing is in the microwave region. A number of important windows exist in this region and are of particular importance for the use of active radar imaging. The texture of earth surface materials causes significant interactions with several of the microwave wavelength regions. This can thus be used as a supplement to information gained in other wavelengths, and also offers the significant advantage of being usable at night (because as an active system it is independent of solar radiation) and in regions of persistent cloud cover (since radar wavelengths are not significantly affected by clouds).

Interaction Mechanisms

When electromagnetic energy strikes a material, three types of interaction can follow: reflection, absorption and/or transmission (Figure 3-2). Our main concern is with the reflected portion since it is usually this which is returned to the sensor system. Exactly how much is reflected will vary and will depend upon the nature of the material and where in the electromagnetic spectrum our measurement is being taken. As a result, if we look at the nature of this reflected component over a range of wavelengths, we can characterize the result as a *spectral response pattern*.

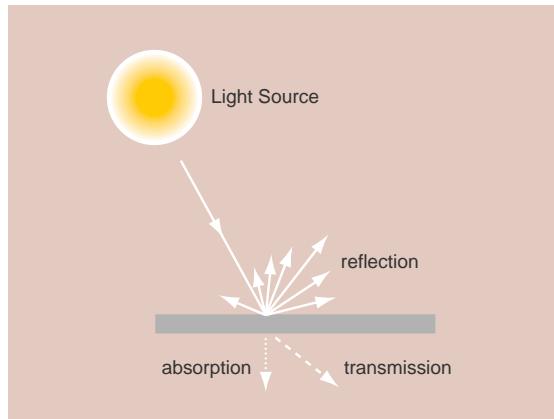


Figure 3-2

Spectral Response Patterns

A spectral response pattern is sometimes called a *signature*. It is a description (often in the form of a graph) of the degree to which energy is reflected in different regions of the spectrum. Most humans are very familiar with spectral response patterns since they are equivalent to the human concept of color. For example, Figure 3-3 shows idealized spectral response patterns for several familiar colors in the visible portion of the electromagnetic spectrum, as well as for white and dark grey. The bright red reflectance pattern, for example, might be that produced by a piece of paper printed with a red ink. Here, the ink is designed to alter the white light that shines upon it and absorb the blue and green wavelengths. What is left, then, are the red wavelengths which reflect off the surface of the paper back to the sensing system (the eye). The high return of red wavelengths indicates a bright red, whereas the low return of green wavelengths in the second example suggests that it will appear quite dark.

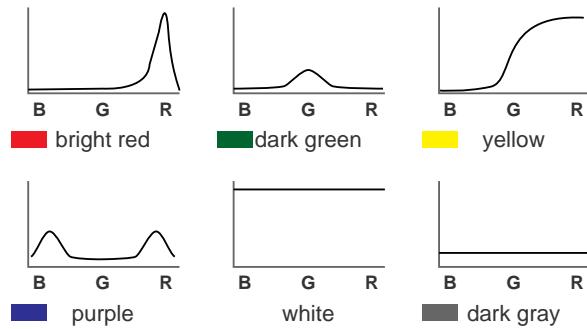


Figure 3-3

The eye is able to sense spectral response patterns because it is truly a multi-spectral sensor (i.e., it senses in more than one place in the spectrum). Although the actual functioning of the eye is quite complex, it does in fact have three separate types of detectors that can usefully be thought of as responding to the red, green and blue wavelength regions. These are the *additive primary colors*, and the eye responds to mixtures of these three to yield a sensation of other hues. For example, the color perceived by the third spectral response pattern in Figure 3-3 would be a yellow—the result of *mixing* a red and green. However, it is important to recognize that this is simply our phenomenological perception of a spectral response pattern. Consider, for example, the fourth curve. Here we have reflectance in both the blue and red regions of the visible spectrum. This is a bimodal distribution, and thus technically not a specific hue in the spectrum. However, we would perceive this to be a purple! Purple (a color between violet and red) does not exist in nature (i.e., as a hue—a distinctive dominant wavelength). It is very real in our perception, however. Purple is simply our perception of a bimodal pattern involving a non-adjacent pair of primary hues.

In the early days of remote sensing, it was believed (more correctly *hoped*) that each earth surface material would have a distinctive spectral response pattern that would allow it to be reliably detected by visual or digital means. However, as our common experience with color would suggest, in reality this is often not the case. For example, two species of trees may have quite a different coloration at one time of the year and quite a similar one at another.

Finding distinctive spectral response patterns is the key to most procedures for computer-assisted interpretation of remotely sensed imagery. This task is rarely trivial. Rather, the analyst must find the combination of spectral bands and the time of year at which distinctive patterns can be found for each of the information classes of interest.

For example, Figure 3-4 shows an idealized spectral response pattern for vegetation along with those of water and dry bare soil. The strong absorption by leaf pigments (particularly chlorophyll for purposes of photosynthesis) in the blue and red regions of the visible portion of the spectrum leads to the characteristic green appearance of healthy vegetation. However, while this *signature* is distinctively different from most non-vegetated surfaces, it is not very capable of distinguishing between species of vegetation—most will have a similar color of green at full maturation. In the near-infrared, however, we find a much higher return from vegetated surfaces because of scattering within the fleshy mesophyll layer of the leaves. Plant pigments do not absorb energy in this region, and thus the scattering, combined with the multiplying effect of a full canopy of leaves, leads to high reflectance in this region of the spectrum. However, the extent of this reflectance will depend highly on the internal structure of leaves (e.g., broadleaf versus needle). As a result, significant differences between species can often be detected in this region. Similarly, moving into the middle infrared region we see a significant dip in the spectral response pattern that is associated with leaf moisture. This is, again, an area where significant differences can arise between mature species. Applications looking for optimal differentiation between species, therefore, will typically involve both the near and middle infrared regions and will use imagery taken well into the development cycle.

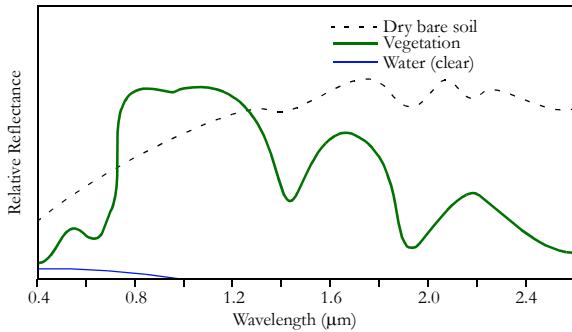


Figure 3-4

Adapted from Lillesand and Kiefer 1987

Multispectral Remote Sensing

In the visual interpretation of remotely sensed images, a variety of image characteristics are brought into consideration: color (or tone in the case of panchromatic images), texture, size, shape, pattern, context, and the like. However, with computer-assisted interpretation, it is most often simply color (i.e., the spectral response pattern) that is used. It is for this reason that a strong emphasis is placed on the use of multispectral sensors (sensors that, like the eye, look at more than one place in the spectrum and thus are able to gauge spectral response patterns), and the number and specific placement of these spectral bands.

Figure 3-5 illustrates the spectral bands of the Landsat Thematic Mapper (TM) system. The Landsat satellite is a commercial system providing multi-spectral imagery in seven spectral bands at a 30 meter resolution.

It can be shown through analytical techniques, such as Principal Components Analysis, that in many environments, the bands that carry the greatest amount of information about the natural environment are the near-infrared and red wavelength bands. Water is strongly absorbed by infrared wavelengths and is thus highly distinctive in that region. In addition, plant species typically show their greatest differentiation here. The red area is also very important because it is the primary region in which chlorophyll absorbs energy for photosynthesis. Thus it is this band which can most readily distinguish between vegetated and non-vegetated surfaces.

Given this importance of the red and near-infrared bands, it is not surprising that sensor systems designed for earth resource monitoring will invariably include these in any particular multispectral system. Other bands will depend upon the range of applications envisioned. Many include the green visible band since it can be used, along with the other two, to produce a traditional false color composite—a full color image derived from the green, red, and infrared bands (as opposed to the blue, green, and red bands of natural color images). This format became common with the advent of color infrared photography, and is familiar to many specialists in the remote sensing field. In addition, the combination of these three bands works well in the interpretation of the cultural landscape as well as natural and vegetated surfaces. However, it is increasingly common to include other bands that are more specifically targeted to the differentiation of surface materials. For example, Landsat TM Band 5 is placed between two water absorption bands and has thus proven very useful in determining soil and leaf moisture differences. Similarly, Landsat TM Band 7 targets the detection of hydrothermal alteration zones in bare rock surfaces. By contrast, the AVHRR system on the NOAA series satellites includes several thermal channels for the sensing of cloud temperature characteristics.

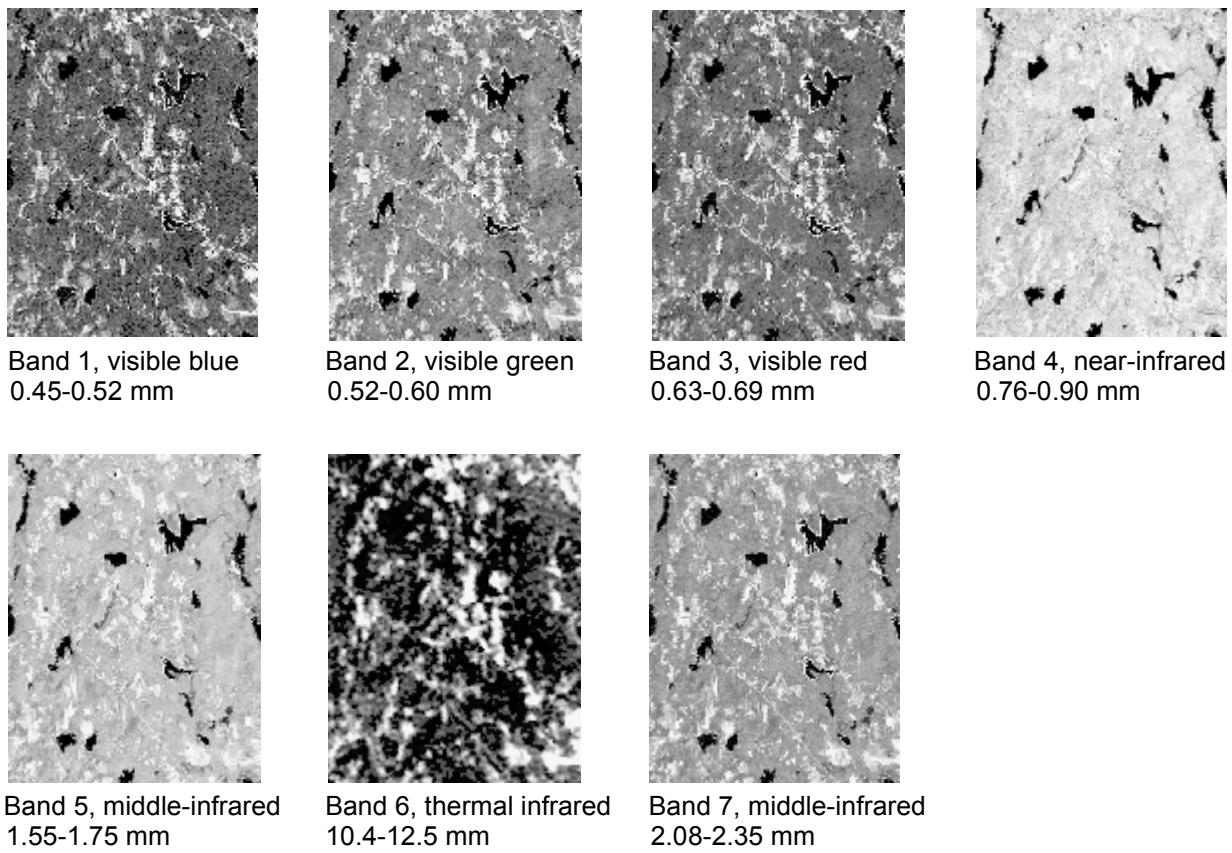


Figure 3-5

Hyperspectral Remote Sensing

In addition to traditional multispectral imagery, some new and experimental systems such as AVIRIS and MODIS are capable of capturing *hyperspectral* data. These systems cover a similar wavelength range to multispectral systems, but in much narrower bands. This dramatically increases the number of bands (and thus precision) available for image classification (typically tens and even hundreds of very narrow bands). Moreover, hyperspectral signature libraries have been created in lab conditions and contain hundreds of signatures for different types of landcovers, including many minerals and other earth materials. Thus, it should be possible to match signatures to surface materials with great precision. However, environmental conditions and natural variations in materials (which make them different from standard library materials) make this difficult. In addition, classification procedures have not been developed for hyperspectral data to the degree they have been for multispectral imagery. As a consequence, multispectral imagery still represents the major tool of remote sensing today.

Sensor/Platform Systems

Given recent developments in sensors, a variety of platforms are now available for the capture of remotely sensed data. Here we review some of the major sensor/platform combinations that are typically available to the GIS user community.

Aerial Photography

Aerial photography is the oldest and most widely used method of remote sensing. Cameras mounted in light aircraft flying between 200 and 15,000 m capture a large quantity of detailed information. Aerial photos provide an instant visual inventory of a portion of the earth's surface and can be used to create detailed maps. Aerial photographs commonly are taken by commercial aerial photography firms which own and operate specially modified aircraft equipped with large format (23 cm x 23 cm) mapping quality cameras. Aerial photos can also be taken using small format cameras (35 mm and 70 mm), hand-held or mounted in unmodified light aircraft.

Camera and platform configurations can be grouped in terms of oblique and vertical. Oblique aerial photography is taken at an angle to the ground. The resulting images give a view as if the observer is looking out an airplane window. These images are easier to interpret than vertical photographs, but it is difficult to locate and measure features on them for mapping purposes.

Vertical aerial photography is taken with the camera pointed straight down. The resulting images depict ground features in plan form and are easily compared with maps. Vertical aerial photos are always highly desirable, but are particularly useful for resource surveys in areas where no maps are available. Aerial photos depict features such as field patterns and vegetation which are often omitted on maps. Comparison of old and new aerial photos can also capture changes within an area over time.

Vertical aerial photos contain subtle displacements due to relief, tip and tilt of the aircraft and lens distortion. Vertical images may be taken with overlap, typically about 60 percent along the flight line and at least 20 percent between lines. Overlapping images can be viewed with a stereoscope to create a three-dimensional view, called a *stereo model*.

Large Format Photography

Commercial aerial survey firms use light single or twin engine aircraft equipped with large-format mapping cameras. Large-format cameras, such as the Wild RC-10, use 23 cm x 23 cm film which is available in rolls. Eastman Kodak, Inc., among others, manufactures several varieties of sheet film specifically intended for use in aerial photography. Negative film is used where prints are the desired product, while positive film is used where transparencies are desired. Print film allows for detailed enlargements to be made, such as large wall-sized prints. In addition, print film is useful when multiple prints are to be distributed and used in the field.

Small Format Photography

Small-format cameras carried in chartered aircraft are an inexpensive alternative to large-format aerial photography. A 35mm or 70mm camera, light aircraft and pilot are required, along with some means to process the film. Because there are inexpensive commercial processing labs in most parts of the world, 35mm systems are especially convenient.

Oblique photographs can be taken with a hand-held camera in any light aircraft; vertical photographs require some form of special mount, pointed through a belly port or extended out a door or window.

Small-format aerial photography has several drawbacks. Light unpressurized aircraft are typically limited to altitudes below 4000 m. As film size is small, sacrifices must be made in resolution or area covered per frame. Because of distortions in the camera system, small-format photography cannot be used if precise mapping is required. In addition, presentation-quality wall-size prints cannot be made from small negatives. Nonetheless, small-format photography can be very useful for reconnaissance surveys and can also be used as point samples.

Color Photography

Normal color photographs are produced from a composite of three film layers with intervening filters that act to isolate, in effect, red, green, and blue wavelengths separately to the different film layers. With color infrared film, these wavelengths are shifted to the longer wavelengths to produce a composite that has isolated reflectances from the green, red and near-infrared wavelength regions. However, because the human eye cannot see infrared, a false color composite is produced by making the green wavelengths appear blue, the red wavelengths appear green, and the infrared wavelengths

appear red.

As an alternative to the use of color film, it is also possible to group several cameras on a single aircraft mount, each with black and white film and a filter designed to isolate a specific wavelength range. The advantage of this arrangement is that the bands are independently accessible and can be photographically enhanced. If a color composite is desired, it is possible to create it from the individual bands at a later time.

Clearly, photographs are not in a format that can immediately be used in digital analysis. It is possible to scan photographs with a scanner and thereby create multispectral datasets either by scanning individual band images, or by scanning a color image and separating the bands. However, the geometry of aerial photographs (which have a central perspective projection and differential parallax) is such that they are difficult to use directly. More typically they require processing by special photogrammetric software to rectify the images and remove differential parallax effects.

Aerial Videography

Light, portable, inexpensive video cameras and recorders can be carried in chartered aircraft. In addition, a number of smaller aerial mapping companies offer videography as an output option. By using several cameras simultaneously, each with a filter designed to isolate a specific wavelength range, it is possible to isolate multispectral image bands that can be used individually, or in combination in the form of a color composite. For use in digital analysis, special graphics hardware boards known as *frame grabbers* can be used to freeze any frame within a continuous video sequence and convert it to digital format, usually in one of the more popular exchange formats such as TIF or TARGA. Like small-format photography, aerial videography cannot be used for detailed mapping, but provides a useful overview for reconnaissance surveys, and can be used in conjunction with ground point sampling.

Satellite-Based Scanning Systems

Photography has proven to be an important input to visual interpretation and the production of analog maps. However, the development of satellite platforms, the associated need to telemeter imagery in digital form, and the desire for highly consistent digital imagery have given rise to the development of solid state scanners as a major format for the capture of remotely sensed data. The specific features of particular systems vary (including, in some cases, the removal of a true scanning mechanism). However, in the discussion which follows, an idealized scanning system is presented that is highly representative of current systems in use.

The basic logic of a scanning sensor is the use of a mechanism to sweep a small field of view (known as an *instantaneous field of view*—IFOV) in a west to east direction at the same time the satellite is moving in a north to south direction. Together this movement provides the means of composing a complete raster image of the environment.

A simple scanning technique is to use a rotating mirror that can sweep the field of view in a consistent west to east fashion. The field of view is then intercepted with a prism that can spread the energy contained within the IFOV into its spectral components. Photoelectric detectors (of the same nature as those found in the exposure meters of commonly available photographic cameras) are then arranged in the path of this spectrum to provide electrical measurements of the amount of energy detected in various parts of the electromagnetic spectrum. As the scan moves from west to east, these detectors are polled to get a set of readings along the east-west scan. These form the columns along one row of a set of raster images—one for each detector. Movement of the satellite from north to south then positions the system to detect the next row, ultimately leading to the production of a set of raster images as a record of reflectance over a range of spectral bands.

There are several satellite systems in operation today that collect imagery that is subsequently distributed to users. Several of the most common systems are described below. Each type of satellite data offers specific characteristics that make it more or less appropriate for a particular application.

In general, there are two characteristics that may help guide the choice of satellite data: *spatial resolution* and *spectral resolution*. The spatial resolution refers to the size of the area on the ground that is summarized by one data value in the imagery. This is the Instantaneous Field of View (IFOV) described earlier. Spectral resolution refers to the number and width of

the spectral bands that the satellite sensor detects. In addition, issues of cost and imagery availability must also be considered.

LANDSAT

The Landsat system of remote sensing satellites is currently operated by the EROS Data Center (<http://edc.usgs.gov>) of the United States Geological Survey. This is a new arrangement following a period of commercial distribution under the Earth Observation Satellite Company (EOSAT) which was recently acquired by Space Imaging Corporation. As a result, the cost of imagery has dramatically dropped, to the benefit of all. Full or quarter scenes are available on a variety of distribution media, as well as photographic products of MSS and TM scenes in false color and black and white.

There have been seven Landsat satellites, the first of which was launched in 1972. The LANDSAT 6 satellite was lost on launch. However, as of this writing, LANDSAT 5 is still operational. Landsat 7 was launched in April, 1999.

Landsat carries two multispectral sensors. The first is the *Multi-Spectral Scanner* (MSS) which acquires imagery in four spectral bands: blue, green, red and near infrared. The second is the *Thematic Mapper* (TM) which collects seven bands: blue, green, red, near-infrared, two mid-infrared and one thermal infrared. The MSS has a spatial resolution of 80 meters, while that of the TM is 30 meters. Both sensors image a 185 km wide swath, passing over each day at 09:45 local time, and returning every 16 days. With Landsat 7, support for TM imagery is to be continued with the addition of a co-registered 15 m panchromatic band.

SPOT

The *Système Pour L'Observation de la Terre* (SPOT) (www.spot.com) was launched and has been operated by a French consortium since 1985. SPOT satellites carry two High Resolution Visible (HRV) pushbroom sensors⁵ which operate in multispectral or panchromatic mode. The multispectral images have 20 meter spatial resolution while the panchromatic images have 10 meter resolution. SPOT satellites 1-3 provide three multi-spectral bands: Green, Red and Infrared. SPOT 4, launched in 1998, provides the same three bands plus a short wave infrared band. The panchromatic band for SPOT 1-3 is 0.51-0.73 μ while that of SPOT 4 is 0.61-0.68 μ . SPOT 5 was launched in 2002. The main improvements over SPOT 4 include: higher ground resolution for the panchromatic bands of 2.5 and 10 meters; higher resolution for multispectral imagery of 10 meters in all three visible and near infrared bands; and a dedicated instrument for along track stereo acquisition.

All SPOT images cover a swath 60 kilometers wide. The SPOT sensor may be pointed to image along adjacent paths. This allows the instrument to acquire repeat imagery of any area 12 times during its 26 day orbital period.

SPOT Image Inc. sells a number of products, including digital images on a choice of distribution media. Existing images may be purchased, or new acquisitions ordered. Customers can request the satellite to be pointed in a particular direction for new acquisitions.

IKONOS

The IKONOS satellite was launched in 1999 by Space Imaging Corp. (www.spaceimaging.com) and was the first commercial venture for high resolution satellite imagery capture and distribution. IKONOS orbits the earth every 98 minutes at an altitude of 680 kilometers passing a given longitude at about the same time daily (approximately 10:30 A.M.). The IKONOS data products include 1 meter panchromatic (0.45 - 0.90 mm) and 4 meter multispectral (blue (0.45 - 0.52 mm), green (0.51 - 0.60 mm), red (0.63 - 0.70 mm), and near infrared (0.76 - 0.85 mm)) imagery taken in 10.5 km swaths. Space Imaging provides a variety of data products. Customers can customize their acquisition.

5. The pushbroom sensor produces output like a scanner. However, there is no actual scanning motion. Rather, the sensor consists of a dense array of detectors—one for each raster cell in the scan line—that is moved across the scene like a pushbroom.

QuickBird

The QuickBird satellite was launched in 2001 by Digital Globe (www.digitalglobe.com). QuickBird offers even higher resolution imagery than IKONOS on a commercial basis. Its one panchromatic band is at 61 cm and its multispectral imagery at 2.44 meters. QuickBird products include the panchromatic band (0.45 - 0.90 mm) and four multispectral bands (blue (0.45 - 0.52 mm), green (0.52 - 0.60 mm), red (0.63 - 0.69 mm), and near infrared (0.76 - 0.90 mm)) taken in 16.5 km swaths. Digital Globe offers a variety of data products.

IRS

The Indian Space Research Organization currently has 5 satellites in the IRS system, with at least 7 planned by 2004. These data are distributed by ANTRIX Corp. Ltd (the commercial arm of the Indian Space Research Organization), and also by Space Imaging Corporation in the United States. The most sophisticated capabilities are offered by the IRS-1C and IRS-1D satellites that together provide continuing global coverage with the following sensors:

IRS-Pan: 5.8 m panchromatic

IRS-LISS⁶: 23.5 m multispectral in the following bands:

Green (0.52-0.59)

Red (0.62-0.68)

Near-Infrared (0.77-0.86)

Shortwave Infrared (1.55-1.7)

IRS-WiFS⁷: 180 m multispectral in the following bands:

Red (0.62-0.68)

Near-Infrared (0.77-0.86)

NOAA-AVHRR

The *Advanced Very High Resolution Radiometer* (AVHRR) is carried on board a series of satellites operated by the U.S. National Oceanic and Atmospheric Administration (NOAA). It acquires data along a 2400-km-wide swath each day. AVHRR collects five bands: red, near-infrared, and three thermal infrared. Spatial resolution of the sensor is 1.1 km and this data is termed Local Area Coverage (LAC). For studying very large areas, a resampled version with resolution of about 4 km is also available, and is termed Global Area Coverage (GAC).

AVHRR may be "high" spatial resolution for meteorological applications, but the images portray only broad patterns and little detail for terrestrial studies. However, they do have a high temporal resolution, showing wide areas on a daily basis and are therefore a popular choice for monitoring large areas. AVHRR imagery is used by several organizations engaged in famine prediction and is an integral part of many early warning activities.

RADARSAT

RADARSAT is an earth observation satellite launched in November 1995 by the Canadian Space Agency. The data is distributed by RADARSAT International (RSI) of Richmond, British Columbia, Canada (or through Space Imaging in the US). Spatial resolution of the C-band SAR imagery ranges from 8 to 100 meters per pixel and the ground coverage repeat interval is 24 days. Sensors can be pointed at the location of interest which enables collection of stereo RADAR imagery. RADAR signals also penetrate cloud cover, thus accessing areas not available to other remote sensing systems. In contrast

6. LISS = Linear Imaging and Self Scanning Sensor. Image format is approximately 140 km x 140 km.

7. WiFS = Wide-Field Sensor. Image format is 810 km x 810 km.

to other remotely sensed imagery, the returned RADAR signal is more affected by electrical and physical (primarily textual) characteristics in the target than by its reflection and spectral pattern, therefore requiring special interpretation and spatial georegistration techniques. Compared to other types of remotely sensed imagery, the use of RADAR data is still in its infancy, but has strong potential.

ERS

ERS-1 and ERS-2 (European Remote Sensing Satellite) (<http://earth.esa.int/ers/>) were developed by the European Space Agency. These identical systems provide an interesting complement to the other commercial imagery products in that they offer a variety of C-band RADAR imagery output formats. For GIS applications, the main output of interest is the side-looking airborne RADAR (SAR) output that provides 100 km wide swaths with a 30 meter resolution. This should prove to be of considerable interest in a variety of applications, including vegetation studies and mapping projects where cloud cover is a persistent problem.

JERS

The Japanese Earth Resource Satellite offers 18 m resolution L-band side-looking RADAR imagery. This is a substantially longer wavelength band than the typical C-band used in earth resources applications. L-band RADAR is capable of penetrating vegetation as well as unconsolidated sand and is primarily used in geologic, topographic and coastal mapping applications. JERS data is available in the United States from Space Imaging Corporation.

AVIRIS

AVIRIS is an experimental system developed by the Jet Propulsion Lab (JPL) that produces hyperspectral data. It captures data in 224 bands over the same wavelength range as Landsat.

TERRA

In 1999 a joint project between Canada, Japan, and the United States launched the first in a series of NASA's Earth Observing System (EOS) satellites. EOS AM-1 satellite provides five instruments (CERES, MISR, MODIS, ASTER, and MOPITT) that will collect an unprecedented wealth of terrestrial and atmospheric data. Data is already available at the USGS website: <http://edcimswww.cr.usgs.gov/pub/imswelcome/>. The EOS AM-1 satellite flies in a near-polar, sun-synchronous orbit and decends across the equator around 10:30 A.M. For the GIS user, the two promising instruments are ASTER and MODIS.

The ASTER sensor onboard the EOS AM-1 platform obtains high resolution images in 14 bands from 15 meters to 90 meters. Three bands are collected at 15 meters in the visible, near infrared wavelengths (0.5 - .09 mm) in 60 km swaths. One additional band is collected at 15 meters that is nadir-looking and is intended to produce same-orbit stereo images. Six shortwave-infrared (1.6 - 2.5 mm) bands are collected at 30 meters ground resolution in 60 km swaths. A further five thermal infrared (8 - 12 mm) are collected at 90 meters in 60 km swaths. Available ASTER products include: spectral radiances and reflectances of the Earth's surface; surface temperature and emmisivities; digital elevation maps from stereo images; surface composition and vegetation maps, cloud, sea, ice, and polar products; and observation of natural hazards.

The MODIS sensor onboard the EOS AM-1 platform provides a logical extension of the AVHRR by providing no fewer than 36 bands of medium-to-coarse resolution imagery with a high temporal repeat cycle (1-2 days). Bands 1 and 2 will provide 250 m resolution images in the red and near-infrared regions. Bands 3-7 will provide 500 m resolution multispectral images in the visible and infrared regions. Finally, bands 8-36 will provide hyperspectral coverage in the visible, reflected infrared, and thermal infrared regions, with a 1 km resolution.

Digital Image Processing

Overview

As a result of solid state multispectral scanners and other raster input devices, we now have available digital raster images of spectral reflectance data. The chief advantage of having these data in digital form is that they allow us to apply computer analysis techniques to the image data—a field of study called *Digital Image Processing*.

Digital Image Processing is largely concerned with four basic operations: *image restoration*, *image enhancement*, *image classification*, *image transformation*. *Image restoration* is concerned with the correction and calibration of images in order to achieve as faithful a representation of the earth surface as possible—a fundamental consideration for all applications. *Image enhancement* is predominantly concerned with the modification of images to optimize their appearance to the visual system. Visual analysis is a key element, even in digital image processing, and the effects of these techniques can be dramatic. *Image classification* refers to the computer-assisted interpretation of images—an operation that is vital to GIS. Finally, *image transformation* refers to the derivation of new imagery as a result of some mathematical treatment of the raw image bands.

In order to undertake the operations listed in this section, it is necessary to have access to Image Processing software. IDRISI is one such system. While it is known primarily as a GIS software system, it also offers a full suite of image processing capabilities.

Image Restoration

Remotely sensed images of the environment are typically taken at a great distance from the earth's surface. As a result, there is a substantial atmospheric path that electromagnetic energy must pass through before it reaches the sensor. Depending upon the wavelengths involved and atmospheric conditions (such as particulate matter, moisture content and turbulence), the incoming energy may be substantially modified. The sensor itself may then modify the character of that data since it may combine a variety of mechanical, optical and electrical components that serve to modify or mask the measured radiant energy. In addition, during the time the image is being scanned, the satellite is following a path that is subject to minor variations at the same time that the earth is moving underneath. The geometry of the image is thus in constant flux. Finally, the signal needs to be telemetered back to earth, and subsequently received and processed to yield the final data we receive. Consequently, a variety of systematic and apparently random disturbances can combine to degrade the quality of the image we finally receive. Image restoration seeks to remove these degradation effects.

Broadly, image restoration can be broken down into the two sub-areas of *radiometric restoration* and *geometric restoration*.

Radiometric Restoration

Radiometric restoration refers to the removal or diminishment of distortions in the degree of electromagnetic energy registered by each detector. A variety of agents can cause distortion in the values recorded for image cells. Some of the most common distortions for which correction procedures exist include:

uniformly elevated values, due to atmospheric haze, which preferentially scatters short wavelength bands (particularly the blue wavelengths);

striping, due to detectors going out of calibration;

random noise, due to unpredictable and unsystematic performance of the sensor or transmission of the data; and

scan line drop out, due to signal loss from specific detectors.

It is also appropriate to include here procedures that are used to convert the raw, unitless relative reflectance values (known as digital numbers, or DN) of the original bands into true measures of reflective power (radiance).

See the chapter on **Image Restoration** for a more detailed discussion of radiometric restoration and how it can be imple-

mented in IDRISI.

Geometric Restoration

For mapping purposes, it is essential that any form of remotely sensed imagery be accurately registered to the proposed map base. With satellite imagery, the very high altitude of the sensing platform results in minimal image displacements due to relief. As a result, registration can usually be achieved through the use of a systematic *rubber sheet* transformation process⁸ that gently warps an image (through the use of polynomial equations) based on the known positions of a set of widely dispersed control points. This capability is provided in IDRISI through the module RESAMPLE.

With aerial photographs, however, the process is more complex. Not only are there systematic distortions related to tilt and varying altitude, but variable topographic relief leads to very irregular distortions (differential parallax) that cannot be removed through a rubber sheet transformation procedure. In these instances, it is necessary to use photogrammetric rectification to remove these distortions and provide accurate map measurements⁹. Failing this, the central portions of high altitude photographs can be resampled with some success.

RESAMPLE is a module of major importance, and it is essential that one learn to use it effectively. Doing so also requires a thorough understanding of reference systems and their associated parameters such as datums and projections. The chapter on **Georeferencing** provides an in-depth discussion of these issues.

Image Enhancement

Image enhancement is concerned with the modification of images to make them more suited to the capabilities of human vision. Regardless of the extent of digital intervention, visual analysis invariably plays a very strong role in all aspects of remote sensing. While the range of image enhancement techniques is broad, the following fundamental issues form the backbone of this area:

Contrast Stretch

Digital sensors have a wide range of output values to accommodate the strongly varying reflectance values that can be found in different environments. However, in any single environment, it is often the case that only a narrow range of values will occur over most areas. Grey level distributions thus tend to be very skewed. Contrast manipulation procedures are thus essential to most visual analyses. Figure 3-6 shows TM Band 3 (visible red) and its histogram. Note that the values of the image are quite skewed. The right image of the figure shows the same image band after a linear stretch between values 12 and 60 has been applied. In IDRISI, this type of contrast enhancement may be performed interactively through Composer's Layer Properties while the image is displayed. This is normally used for visual analysis only—original data values are used in numeric analyses. New images with stretched values are produced with the module STRETCH.

8. Satellite-based scanner imagery contains a variety of inherent geometric problems such as skew (caused by rotation of the earth underneath the satellite as it is in the process of scanning a complete image) and scanner distortion (caused by the fact that the instantaneous field of view (IFOV) covers more territory at the ends of scan lines, where the angle of view is very oblique, than in the middle). With commercially-marketed satellite imagery, such as Landsat, IRS and SPOT, most elements of systematic geometric restoration associated with image capture are corrected by the distributors of the imagery. Thus, for the end user, the only geometric operation that typically needs to be undertaken is a rubber-sheet resampling in order to rectify the image to a map base. Many commercial distributors will perform this rectification for an additional fee.

9. Photogrammetry is the science of taking spatial measurements from aerial photographs. In order to provide a full rectification, it is necessary to have stereoscopic images—photographs which overlap enough (e.g., 60% in the along-track direction and 10% between flight lines) to provide two independent images of each part of the landscape. Using these stereoscopic pairs and ground control points of known position and height, it is possible to fully recreate the geometry of the viewing conditions, and thereby not only rectify measurements from such images, but also derive measurements of terrain height. The rectified photographs are called *orthophotos*. The height measurements may be used to produce digital elevation models.

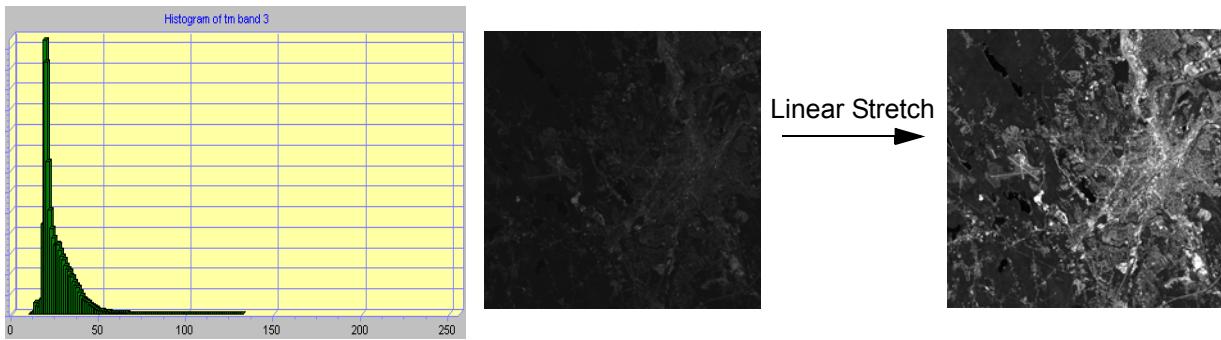
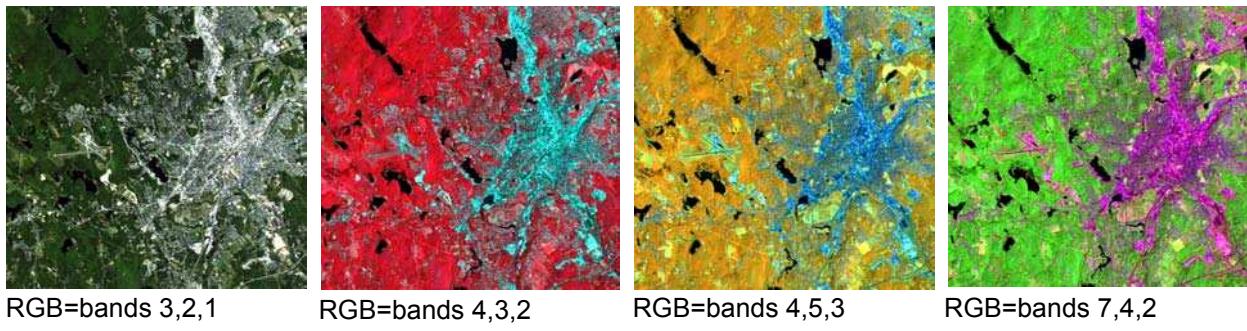


Figure 3-6

Composite Generation

For visual analysis, color composites make fullest use of the capabilities of the human eye. Depending upon the graphics system in use, composite generation ranges from simply selecting the bands to use, to more involved procedures of band combination and associated contrast stretch. Figure 3-7 shows several composites made with different band combinations from the same set of TM images. (See Figure 3-5 for TM band definitions.) The IDRISI module COMPOSITE is used to construct three-band 24-bit composite images for visual analysis.



RGB=bands 3,2,1

RGB=bands 4,3,2

RGB=bands 4,5,3

RGB=bands 7,4,2

Figure 3-7

Digital Filtering

One of the most intriguing capabilities of digital analysis is the ability to apply digital filters. Filters can be used to provide edge enhancement (sometimes called *crispening*), to remove image blur, and to isolate lineaments and directional trends, to mention just a few. The IDRISI module FILTER is used to apply standard filters and to construct and apply user-defined filters.

Image Classification

Image classification refers to the computer-assisted interpretation of remotely sensed images. The procedures involved are treated in detail in the chapter titled **Classification of Remotely Sensed Imagery**. This section provides a brief overview.

Although some procedures are able to incorporate information about such image characteristics as texture and context, the majority of image classification is based solely on the detection of the spectral signatures (i.e., spectral response patterns) of land cover classes. The success with which this can be done will depend on two things: 1) the presence of dis-

tinctive signatures for the land cover classes of interest in the band set being used; and 2) the ability to reliably distinguish these signatures from other spectral response patterns that may be present.

There are two general approaches to image classification: *supervised* and *unsupervised*. They differ in how the classification is performed. In the case of supervised classification, the software system delineates specific landcover types based on statistical characterization data drawn from known examples in the image (known as training sites). With unsupervised classification, however, clustering software is used to uncover the commonly occurring landcover types, with the analyst providing interpretations of those cover types at a later stage.

Supervised Classification

The first step in supervised classification is to identify examples of the information classes (i.e., land cover types) of interest in the image. These are called *training sites*. The software system is then used to develop a statistical characterization of the reflectances for each information class. This stage is often called *signature analysis* and may involve developing a characterization as simple as the mean or the range of reflectances on each band, or as complex as detailed analyses of the mean, variances and covariances over all bands.

Once a statistical characterization has been achieved for each information class, the image is then classified by examining the reflectances for each pixel and making a decision about which of the signatures it resembles most. There are several techniques for making these decisions, called *classifiers*. Most Image Processing software will offer several, based on varying decision rules. IDRISI offers a wide range of options falling into three groups, depending upon the nature of the output desired and the nature of the input bands.

Hard Classifiers

The distinguishing characteristic of hard classifiers is that they all make a definitive decision about the landcover class to which any pixel belongs. IDRISI offers five supervised classifiers in this group: Parallelepiped (PIPED), Minimum Distance to Means (MINDIST), and Maximum Likelihood (MAXLIKE), Linear Discriminant Analysis (FISHER), and Artificial Neural Network (NEURALNET). They differ only in the manner in which they develop and use a statistical characterization of the training site data. Of the five, the Maximum Likelihood procedure is unquestionably the most widely used classifier in the classification of remotely sensed imagery.

Soft Classifiers

Contrary to hard classifiers, soft classifiers do not make a definitive decision about the land cover class to which each pixel belongs. Rather, they develop statements of the degree to which each pixel belongs to each of the land cover classes being considered. Thus, for example, a soft classifier might indicate that a pixel has a 0.72 probability of being forest, a 0.24 probability of being pasture, and a 0.04 probability of being bare ground. A hard classifier would resolve this uncertainty by concluding that the pixel was forest. However, a soft classifier makes this uncertainty explicitly available, for any of a variety of reasons. For example, the analyst might conclude that the uncertainty arises because the pixel contains more than one cover type and could use the probabilities as indications of the relative proportion of each. This is known as *sub-pixel* classification. Alternatively, the analyst may conclude that the uncertainty arises because of unrepresentative training site data and therefore may wish to combine these probabilities with other evidence before *hardening* the decision to a final conclusion. IDRISI offers four soft classifiers (BAYCLASS, MAHALCLASS, BELCLASS and FUZCLASS). The results from all four can be hardened using the module HARDEN. The difference between them relates to the logic by which uncertainty is specified—Bayesian, Dempster-Shafer, and Fuzzy Sets. In addition, the system supplies a variety of additional tools specifically designed for the analysis of sub-pixel mixtures (e.g., UNMIX, FUZSIG, MIXCALC and MAXSET).

Hyperspectral Classifiers

All of the classifiers mentioned above operate on multispectral imagery—images where several spectral bands have been captured simultaneously as independently accessible image components. Extending this logic to many bands produces what has come to be known as *hyperspectral* imagery.

Although there is essentially no difference between hyperspectral and multispectral imagery (i.e., they differ only in degree), the volume of data and high spectral resolution of hyperspectral images does lead to differences in the way that they are handled. IDRISI provides special facilities for creating hyperspectral signatures either from training sites or from libraries of spectral response patterns developed under lab conditions (HYPERSIG) and an automated hyperspectral signature extraction routine (HYPERAUTOSIG). These signatures can then be applied to any of several hyperspectral classifiers: Spectral Angle Mapper (HYPERSAM), Minimum Distance to Means (HYPERMIN), Linear Spectral Unmixing (HYPERUNMIX), Orthogonal Subspace Projection (HYPEROSP), Absorption area analysis (HYPERABSORB) hyperspectral classifiers. An unsupervised classifier (see next section) for hyperspectral imagery (HYPERUSP) is also available.

Unsupervised Classification

In contrast to supervised classification, where we tell the system about the character (i.e., signature) of the information classes we are looking for, unsupervised classification requires no advance information about the classes of interest. Rather, it examines the data and breaks it into the most prevalent natural spectral groupings, or clusters, present in the data. The analyst then identifies these clusters as landcover classes through a combination of familiarity with the region and ground truth visits.

The logic by which unsupervised classification works is known as *cluster analysis*, and is provided in IDRISI primarily by the CLUSTER module. CLUSTER performs classification based on a set of input images using a multi-dimensional histogram peak technique. It is important to recognize, however, that the clusters unsupervised classification produces are not information classes, but spectral classes (i.e., they group together features (pixels) with similar reflectance patterns). It is thus usually the case that the analyst needs to reclassify spectral classes into information classes. For example, the system might identify classes for asphalt and cement which the analyst might later group together, creating an information class called pavement.

While attractive conceptually, unsupervised classification has traditionally been hampered by very slow algorithms. However, the clustering procedure provided in IDRISI is extraordinarily fast (unquestionably the fastest on the market) and can thus be used iteratively in conjunction with ground truth data to arrive at a very strong classification. With suitable ground truth and accuracy assessment procedures, this tool can provide a remarkably rapid means of producing quality land cover data on a continuing basis.

In addition to the above-mentioned techniques, two modules bridge both supervised and unsupervised classifications. ISOCLUST uses a procedure known as *Self-Organizing Cluster Analysis* to classify up to 7 raw bands with the user specifying the number of clusters to process. The procedure uses the CLUSTER module to initiate a set of clusters that seed an iterative application of the MAXLIKE procedure, each stage using the results of the previous stage as the training sites for this supervised procedure. The result is an unsupervised classification that converges on a final set of stable members using a supervised approach (hence the notion of "self-organizing"). MAXSET is also, at its core, a supervised procedure. However, while the procedure starts with training sites that characterize individual classes, it results in a classification that includes not only these specific classes, but also significant (but unknown) mixtures that might exist. Thus the end result has much the character of that of an unsupervised approach.

Accuracy Assessment

A vital step in the classification process, whether supervised or unsupervised, is the assessment of the accuracy of the final images produced. This involves identifying a set of sample locations (such as with the SAMPLE module) that are visited in the field. The land cover found in the field is then compared to that which was mapped in the image for the same location. Statistical assessments of accuracy may then be derived for the entire study area, as well as for individual classes (using ERRMAT).

In an iterative approach, the error matrix produced (sometimes referred to as a *confusion matrix*), may be used to identify particular cover types for which errors are in excess of that desired. The information in the matrix about which covers are being mistakenly included in a particular class (*errors of commission*) and those that are being mistakenly excluded (*errors of omission*) from that class can be used to refine the classification approach.

Image Transformation

Digital Image Processing offers a limitless range of possible transformations on remotely sensed data. Two are mentioned here specifically, because of their special significance in environmental monitoring applications.

Vegetation Indices

There are a variety of vegetation indices that have been developed to help in the monitoring of vegetation. Most are based on the very different interactions between vegetation and electromagnetic energy in the red and near-infrared wavelengths. Refer back to Figure 3-4, which includes a generalized spectral response pattern for green broad leaf vegetation. As can be seen, reflectance in the red region (about 0.6 - 0.7 μ) is low because of absorption by leaf pigments (principally chlorophyll). The infrared region (about 0.8 - 0.9 μ), however, characteristically shows high reflectance because of scattering by the cell structure of the leaves. A very simple vegetation index can thus be achieved by comparing the measure of infrared reflectance to that of the red reflectance.

Although a number of variants of this basic logic have been developed, the one which has received the most attention is the *normalized difference vegetation index* (NDVI). It is calculated in the following manner:

$$\text{NDVI} = (\text{NIR} - \text{R}) / (\text{NIR} + \text{R})$$

where NIR = Near Infrared

and R = Red

Figure 3-8 shows NDVI calculated with TM bands 3 and 4 for the same area shown in Figures 3-5, 3-6 and 3-7.

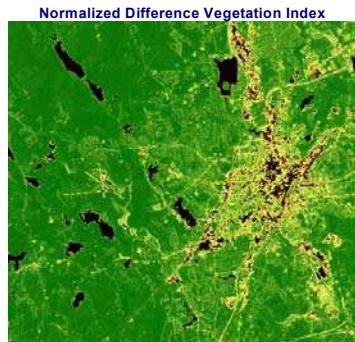


Figure 3-8

This kind of calculation is quite simple for a raster GIS or Image Processing software system, and the result has been shown to correlate well with ground measurements of biomass. Although NDVI needs specific calibration to be used as an actual measure of biomass, many agencies have found the index to be useful as a relative measure for monitoring purposes. For example, the United Nations Food and Agricultural Organization (FAO) Africa Real Time Information System (ARTEMIS) and the USAID Famine Early Warning System (FEWS) programs both use continental scale NDVI images derived from the NOAA-AVHRR system to produce vegetation index images for the entire continent of Africa every ten days.¹⁰

While the NDVI measure has proven to be useful in a variety of contexts, a large number of alternative indices have been proposed to deal with special environments, such as arid lands. IDRISI offers a wide variety of these indices (over 20 in

10. An archive dataset of monthly NDVI images for Africa is available on CD from Clark Labs. The Africa NDVI data CD contains monthly NDVI maximum value composite images (1982-1999), average and standard deviation of monthly NDVI images for each month over the same time period, monthly NDVI anomaly images, and ancillary data (DEM, land use and land cover, country boundaries and coast line) for Africa in IDRISI format. Contact Clark Labs for more information.

the VEGINDEX and TASSCAP modules combined). The chapter on **Vegetation Indices** offers a detailed discussion of their characteristics and potential application.

Principal Components Analysis

Principal Components Analysis (PCA) is a linear transformation technique related to Factor Analysis. Given a set of image bands, PCA produces a new set of images, known as components, that are uncorrelated with one another and are ordered in terms of the amount of variance they explain from the original band set.

PCA has traditionally been used in remote sensing as a means of data compaction. For a typical multispectral image band set, it is common to find that the first two or three components are able to explain virtually all of the original variability in reflectance values. Later components thus tend to be dominated by noise effects. By rejecting these later components, the volume of data is reduced with no appreciable loss of information.

Given that the later components are dominated by noise, it is also possible to use PCA as a noise removal technique. The output from the PCA module in IDRISI includes the coefficients of both the forward and backward transformations. By zeroing out the coefficients of the noise components in the reverse transformation, a new version of the original bands can be produced with these noise elements removed.

Recently, PCA has also been shown to have special application in environmental monitoring. In cases where multispectral images are available for two dates, the bands from both images are submitted to a PCA as if they all came from the same image. In these cases, changes between the two dates tend to emerge in the later components. More dramatically, if a time series of NDVI images (or a similar single-band index) is submitted to the analysis, a very detailed analysis of environmental changes and trends can be achieved. In this case, the first component will show the typical NDVI over the entire series, while each successive component illustrates change events in an ordered sequence of importance. By examining these images, along with graphs of their correlation with the individual bands in the original series, important insights can be gained into the nature of changes and trends over the time series. The TSA (Time Series Analysis) module in IDRISI is a specially tailored version of PCA to facilitate this process.

Other Transformations

As mentioned earlier, IDRISI offers a variety of other transformations. These include color space transformations (COLSPACE), texture calculations (TEXTURE), blackbody thermal transformations (THERMAL), and a wide variety of ad hoc transformations (such as image ratioing) that can be most effectively accomplished with the Image Calculator utility.

Conclusions

Remotely sensed data is important to a broad range of disciplines. This will continue to be the case and will likely grow with the greater availability of data promised by an increasing number of operational systems. The availability of this data, coupled with the computer software necessary to analyze it, provides opportunities for environmental scholars and planners, particularly in the areas of landuse mapping and change detection, that would have been unheard of only a few decades ago.

The inherent raster structure of remotely sensed data makes it readily compatible with raster GIS. Thus, while IDRISI provides a wide suite of image processing tools, they are completely integrated with the broader set of raster GIS tools the system provides.

Idrisi System Overview

IDRISI consists of a main interface program (with a menu and toolbar system) and a collection of over 200 program *modules* that provide facilities for the input, display, and analysis of geographic and remotely sensed data. These geographic data are described in the form of *map layers*—elementary map components that describe a single theme. Examples of map layers might include a roads layer, an elevation layer, a soil type layer, a remotely sensed reflectance layer and so on. All analyses act upon map layers. For display, a series of map layers may be brought together into a *map composition*.

Because geographic data may be of different types, IDRISI incorporates the two basic forms of map layers: *raster image layers* and *vector layers*.¹¹ Although IDRISI is adept at the input and display of both image and vector layers, analysis is primarily oriented towards the use of image layers. In addition, IDRISI offers a complete image processing system for remotely sensed image data. As a result, it is commonly described as a ***raster*** system. However, IDRISI does offer strong capabilities for the analysis of vector attribute data, as well as rapid vector-to-raster conversion routines. Thus the system offers a powerful set of tools for geographic analyses that require both types of map layers.

System Operation

The IDRISI Application Window

When IDRISI is open, the application window will completely occupy the screen (in Windows terminology, it is automatically maximized). Though not required, it is recommended that it be kept maximized because many of the dialog boxes and images you will display will require a substantial amount of display space.¹² The IDRISI application window includes the menu, the toolbar, and the status bar.

The Menu System

The menu system is at the top of the application window. You can activate it either with the mouse or by holding the ALT key and pressing the underlined key of the main menu entry. You can then use the mouse or arrow keys to move around.

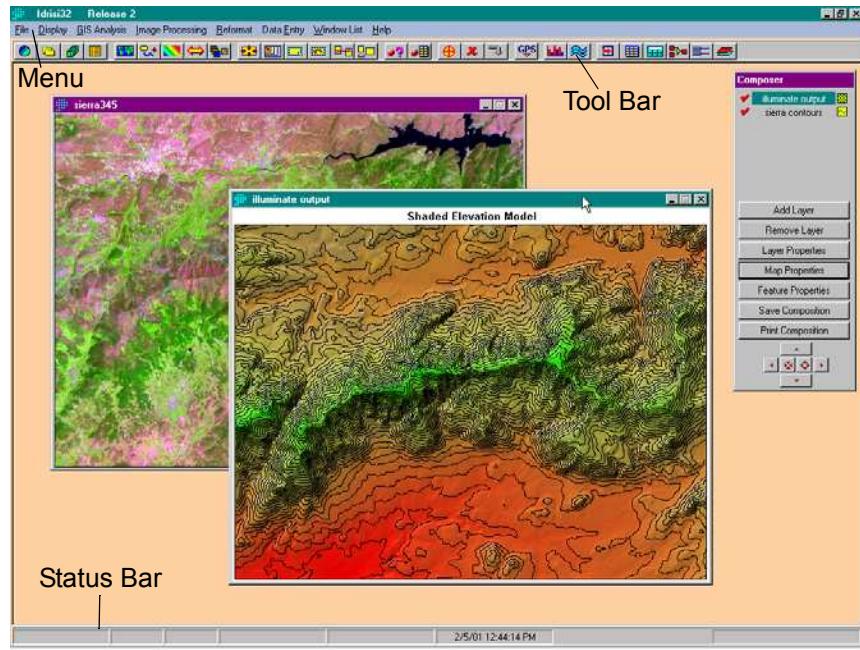
If you select a menu option that includes a right-pointing arrow, a submenu will appear. Clicking on a menu option without a right-pointing arrow will cause a dialog box for that module to appear.

The Tool Bar

Just below the menu is a set of buttons that are collectively known as the tool bar. Each button represents either a program module or an interactive operation that can be selected by clicking on that button with the mouse. Some of these buttons toggle between an active and an inactive state. When active, the button appears to remain depressed after being clicked. In these cases, the button can be released by clicking it again. You will also notice that some buttons may not always be available for use. This is indicated when the button icon appears grey. Hold the cursor over an icon to cause the name of the function or module represented by that icon to momentarily appear. The set of icons represents interactive display functions as well as some of the most commonly-used modules.

11. For an in-depth discussion of raster and vector data structures, see the chapter **Introduction to GIS** in this volume.

12. For this reason, we recommend that the operating system taskbar be set to "autohide" so that it does not constantly occupy much-needed screen space. To do so, go to Start/Settings/Taskbar and toggle on the autohide option.



The Status Bar

At the bottom of the screen is the status bar. The status bar provides a variety of information about program operation.

When maps and map layers are displayed on the screen and the mouse is moved over one of these windows, the status bar will indicate the position of the cursor within that map in both column and row image coordinates and X and Y map reference system coordinates. In addition, the status bar indicates the scale of the screen representation as a *Representative Fraction* (RF).¹³

The status bar will also indicate the progress of the most recently launched analytical operation with a graphic progress bar as well as a *percent done* measure. Since IDRISI has been designed to permit multitasking of operations, it is possible that more than one operation may be working simultaneously. To see a listing of all active processes and their status, simply double click on the progress bar panel at the bottom right of the screen. Modules may also be terminated from here.

Program Modules

Program modules may be accessed in three ways:

1. by selecting the module in the menu structure and activating it with a click of the mouse,
2. by selecting its program icon on the toolbar just below the menu, and
3. by typing in or selecting the module name from the alphabetical list provided by the ShortCut utility.

13. A Representative Fraction expresses the scale of a map as a fractional proportion. Thus, an RF of 1/10,000 indicates that the image on the screen is one ten-thousandth of the actual size of the corresponding features on the ground. IDRISI automatically determines your screen resolution and dimensions, which, in combination with the minimum and maximum x and y coordinates of the image or vector layer, are used to determine the RF for any display window.

Each of these methods activates a dialog box for that module. After entering the required information for the operation to be performed and clicking on the OK button, the program module will run.

Data Paths and Projects

Program modules act upon data—map layers and tabular data stored in data files. These files are stored in folders (also known as directories) and sub-folders (sub-directories) on drives in the computer. The location of any specific data file is thus designated by a name consisting of the file name plus the drive, folder and sub-folder locations. Collectively, these are known as a data path (since they specify the route to a particular collection of data). For example, "c:\massachusetts\middlesex\census_tracts.vct" might designate a vector layer of census tracts contained in the Middlesex County sub-folder of the Massachusetts folder on hard disk drive C.

IDRISI can work with files from any data path (including network paths). However, it can often be a nuisance to continually specify folder and sub-folder names, particularly when a specific project typically accesses data in a limited set of folders. To simplify matters, IDRISI allows you to specify these data paths as belonging to a specific project. When the paths to these folders are designated in a Project File, IDRISI finds data very easily and with minimal user intervention.

The Project Working Folder

Within any Project File, the most important folder is the *Working Folder*. Users may choose to store all of the data for a project in the Working Folder (particularly for smaller projects). However, for larger projects, or projects requiring libraries of carefully organized and protected data sets, additional *Resource Folders* can also be added to the Project File.

While the user is always able to specify any location for input or output data, designating the most commonly-used folders in a Project File facilitates the use of IDRISI. For instance, if input file names are given without a path designation, IDRISI automatically looks in the Working Folder first, then in each Resource Folder in turn to locate the file. Similarly, if no alternative path is specified, output data are automatically written to the Working Folder.

Project Resource Folders

Resource Folders contain data that can be accessed quickly through the IDRISI pick list (see below). A Project File can contain any number of Resource Folders. To the user, the Working Folder and all Resource Folders function as a single project folder. For example, one might ask IDRISI to display a raster map layer named "landuse". If no folder information was supplied, IDRISI first looks for the file in the Working Folder, and then each Resource Folder in turn, ultimately displaying the first instance it finds. This is very convenient as it allows the user to enter file names without paths. However, if duplicate file names exist in separate folders that are part of the same project, the user must exercise caution and remember the order in which IDRISI accesses the project folders.

Setting the Data Paths

The data paths for the Working and Resource Folders can be set in the Data Paths option of the File menu. You can also do so by clicking the leftmost icon on the toolbar to access the Project Environment dialog.

Working with IDRISI Dialog Boxes

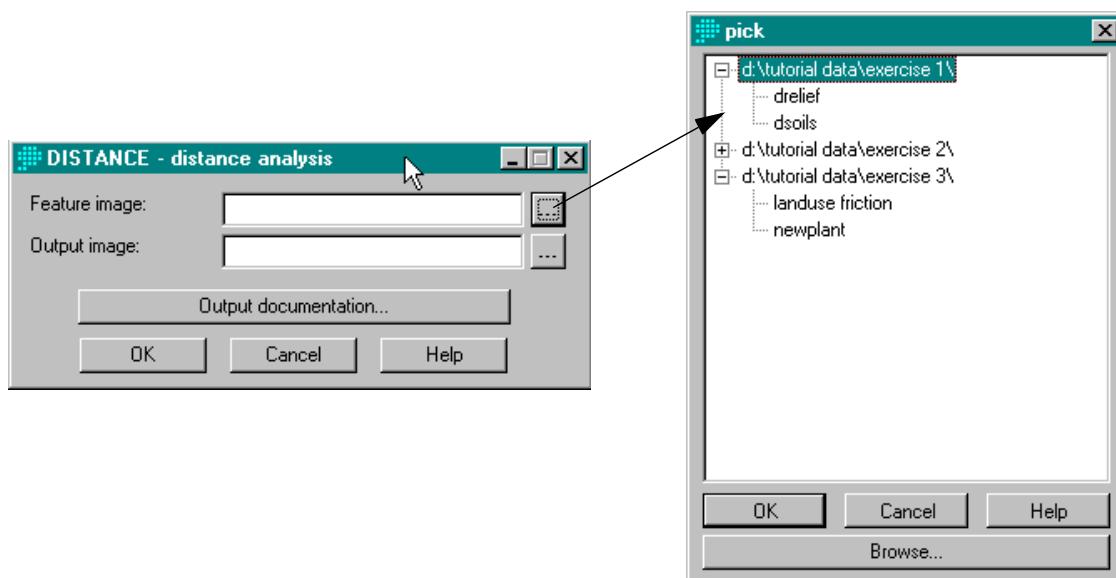
When any module is activated, a dialog box appears with information on data or option choices that are required for the module to run. Virtually all input boxes must be filled with information before the module can be run. Only the title and measurement units input boxes under the Output Documentation button can be left blank. (It is recommended, however, that these boxes also be filled.) In some cases, input boxes already contain values, or will fill in with values as other infor-

mation is entered. These are default values that may be freely edited. In those cases where a set of choices should be made, the most common settings are normally pre-selected. These should be examined and changed if necessary. The system will display an error message in cases where a needed element of data has been left out.

Pick Lists

Whenever a dialog box requires the name of an input data file (such as a raster image), you have two choices for input of that value. The first is simply to type the name into the input box. Alternatively, you can activate a *pick list* of choices by either double clicking into the input box, or clicking the *pick list button* to the right of the input box (the small button with the ellipses characters "..."). The pick list is organized as a tree directory of all files of the required type that exist in the Working and Resource Folders. The Working folder is always at the top of the list, followed by each Resource Folder in the order in which they are listed in the Project Environment. The files displayed in the pick list depend upon the particular input box from which the pick list was launched. All IDRISI file types are designated by unique file name extensions. If an input box requires a raster image, for example, the pick list launched from that input box will display only those files that have an .rst extension.

You may select a file from the pick list by first highlighting it with a click of the left mouse button, and then selecting it by clicking the OK button or pressing the Enter key. Alternatively, double clicking the highlighted entry will also cause it to be selected. The pick list will then disappear, and the name of that file will appear in the input box. Note that you can also choose a file that is not in the current project environment by clicking on the Browse button at the bottom of the pick list.



Output File Names

You will normally want to give output file names that are meaningful to you. It is common to generate many output files in a single analysis and descriptive file names are helpful for keeping track of these files.¹⁴ File names may be long and can contain spaces and most keyboard characters (all except / \ : * ? " < > and |). It is not necessary for the user to specify the three-letter file name extension as IDRISI takes care of this. A full path may be entered in the output file name box. If no path is entered, the output file will automatically be written to the working folder that is specified in the project envi-

14. The documentation file for output raster and vector data files includes in the lineage field the command line from which the layer was created. This may be viewed with the Metadata function.

ronment.

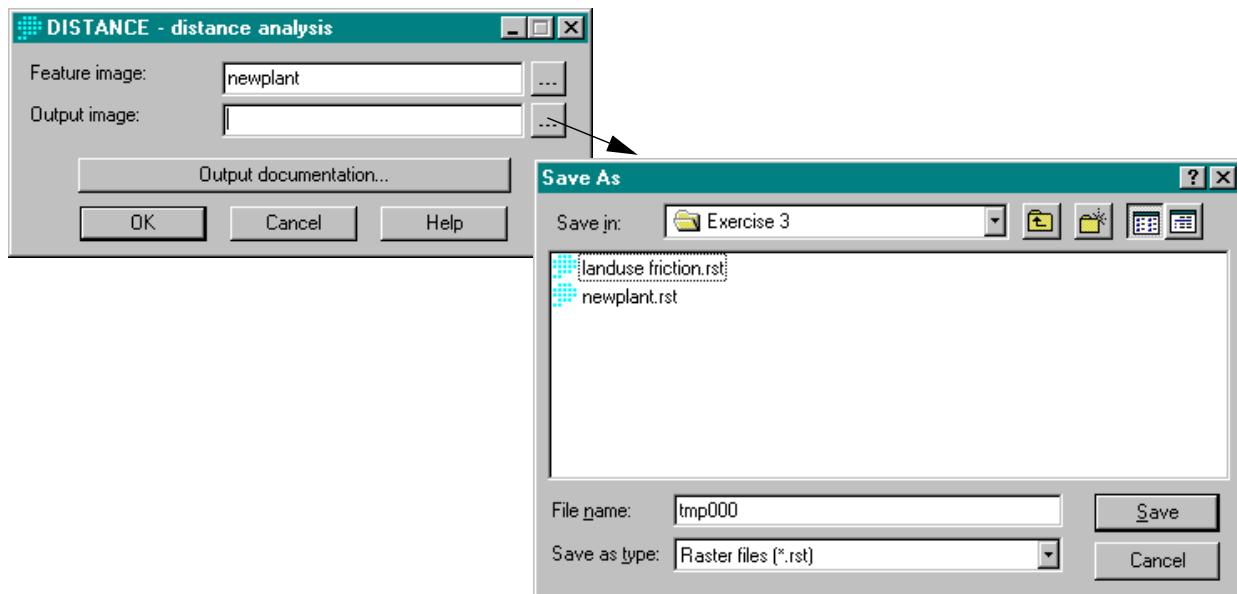
Clicking on the pick list button to the right of the output file name box will bring up a standard Windows Save As file name dialog box. Here the user may select any folder for the output file and may also enter the file name. When the pick list button is used, an automatically-generated output file name will be entered by default. To change this name, simply click into the file name box and type the desired name.

As stated above, users typically give descriptive file names to help avoid confusion as the database grows. However, IDRISI can also generate automatic names for output data files. Automatic output names are intended for the quick naming of temporary files.

Automatic output file names begin with a user-defined three-letter prefix (set to TMP by default),¹⁵ followed by a three digit number. By default, filenames begin with TMP001 and progress up to TMP999. As these files are normally temporary, overwrite protection (see below) is disabled for files with names that begin with the designated prefix followed by three digits. Note that after exiting IDRISI, the cycle will begin again with TMP001 the next time IDRISI is used. Since the numbering sequence starts from 001 with each new session, the data in such files is likely to be lost in a subsequent session unless it is intentionally saved under a new name.

To use the autoname feature, move to the output file name box concerned, and either double click in the box or click the pick list button. If the double click is used, an automatic file name will be generated and placed into the output file name box. Since no path is specified with this method, output will automatically go to the Working Folder.

In the case where the pick button is clicked beside an output file name box, a standard Windows Save As dialog box will appear with the temporary name (e.g., TMP001) already selected.



Overwrite Protection

By default, IDRISI checks if the given output file already exists. If it does, it will ask whether you wish to overwrite it. If not, you will be returned to the dialog box where you can enter a different name.

There are three exceptions to this logic. The first is when an automatic output name is generated (see the section immedi-

15. The automatically-generated output file prefix may be changed in the User Preferences option of the File menu.

ately above). The second is when the user removes overwrite protection in the User Preferences dialog under the File menu. And the third is when modules are run in macro mode.

Getting Help

While using IDRISI, help is always close at hand. IDRISI contains an extensive on-line Help System that can be accessed in a variety of ways. The most general means of accessing help is to select it from the main IDRISI menu. However, each program module also contains a help button. In this case, help is context sensitive, accessing the section in Help associated with that particular module. In either case, it is worth becoming familiar with the many options the Help System provides. To do so, select the Using Help option from the main Help menu.

Map Layers, Collections and Data Structures

Map Layers

Map layers are the fundamental units of display and analysis in GIS. A map layer is an elementary theme, a single phenomenon that can be mapped across space. Examples would include a land use layer, a roads layer, an elevation layer, a soils layer, and so on. Map layers, then, are somewhat different from traditional maps. Traditional maps usually consist of several map layers. For example, a topographic map sheet would typically consist of a contour layer, a forest cover layer, an administrative boundaries layer, a roads layer and a settlements layer. In IDRISI, this same concept is applied. A *map* is a graphic representation of space that is composed of one or more map *layers* using a tool called *Composer*.

This breakdown of the map into its elementary constituents offers important advantages. Clearly it allows for the simple production of highly customized maps; we simply pull together the layers we wish to see together. However, the more important reason for storing data this way is that layers are the basic variables used by the GIS in analytical modeling. Thus, for example, we might create a map of soil erosion as a mathematical function of soil type, slope, land cover, and precipitation layers. Indeed, the breakdown of geographic data into layers allows for the production of an extraordinary range of mathematical and logical models.

Map Layer Types

As detailed in the **Introduction to GIS** chapter, there are two basic types of layers in GIS: raster image layers, and vector layers.¹⁶ Some systems deal exclusively with one or the other type. IDRISI incorporates both since they each have special advantages. However, while they have equal stature in terms of display and database query, IDRISI offers a far broader range of analytical operations for raster layers. This is partly because of IDRISI's historic development from an almost purely raster system and partly because the range of analytical operations possible in GIS is far greater with raster layers (as a consequence of their simplicity and predictable regularity).

All IDRISI raster layers have the same basic structure. Sub-types exist only in terms of the numeric data type that is used to record cell values. As one can easily imagine, raster images are high in data volume. Thus, small changes in the underlying data type can make huge differences in storage requirements. IDRISI supports raster images stored with byte, integer, and real numbers as well as two special formats for the storage of full-color images.

Vector layers describe the location and character of distinct geographic features. Sub-types include Point, Line, Polygon and Text layers. Point features are phenomena that can be described by a single point, such as meteorological data or (at small scales) town locations. As the name suggests, line vector layers describe linear features such as rivers and roads. The term *polygon* may be less familiar. Polygons refer to areas. Because the boundaries of the areas are made up of straight line segments between points, the figure formed is technically a polygon. The term *polygon* has thus become part of the terminology of GIS. Finally, text layers are used to describe the location and orientation of text labels.

Map Layer Names

All analytical modules in IDRISI act on map layers. Thus the input and output file names that are given in IDRISI operations are typically those of map layers. In your normal use of IDRISI, map layers are specified with simple names that do not need to express their data path (the IDRISI Project Environment defines the paths to your working and resource folders). It is also not necessary to specify their file name extensions (since IDRISI looks for and creates specific extensions automatically).

16. In IDRISI, the terms raster layer, raster image layer and image are all used interchangeably.

Consistent with the 32-bit variants of the Windows operating system, layer names may contain any letter of the alphabet as well as numerals, spaces and most symbols (excluding \ / : * ? " < > and |). For example, layers might have names such as "soils" or "landuse 1996." In principle, file names can be up to 255 characters in length. However, in practical use, it is recommended that they be only as long as necessary so the entire file name will be visible in the pick lists and dialog input boxes.

In normal use, no distinction is made between image and vector layer names. However, their type is always known by the system—by the context in which they were created and by the file extension that the system uses internally to store each layer. Thus a raster image layer named "soils" would be stored as "soils.rst" internally, while a vector layer named "soils" would be stored as "soils.vct". As a result, it is possible to have raster and vector layers of the same name. However, most users of IDRISI allow this to occur only in cases where the two files store vector and image manifestations of the same underlying data.

When file names are input into IDRISI dialog boxes, the system always knows the appropriate layer type in use. Indeed, only those files of the appropriate type for that operation will display in the pick list.

Map Layer Data Files

While we refer to a map layer by a simple name, it is actually stored by IDRISI as a pair of data files. One contains the spatial data, while the second contains information about those data (i.e., documentation or *metadata*). Thus, while raster images are stored in data files with an ".rst" extension, they each have an accompanying documentation file with an ".rdc" extension (i.e., raster documentation). Similarly, while vector data files have a ".vct" extension, each has a companion documentation file with a ".vdc" extension (i.e., vector documentation). In use, however, you would refer to these layers by their simple layer names (i.e., "soils" and "districts" respectively). The contents of data documentation files are described in detail below in the section on file structures.

Map Layer Collections

Sometimes it makes sense to group map layers together into collections. A good example is when you have a set of vector features associated with a data table. For instance, a vector file might be created of census tracts within a city. Each of these tracts possesses multiple attributes such as median income, population density, median educational attainment, and so on. Each of these attributes can be linked to a vector file defining the geography of the census tracts and displayed as a map layer. Thus the combination of the vector file and the entire database table constitutes a collection of map layers.

IDRISI recognizes two kinds of collections. One is a collection produced by linking a geographic or feature definition vector file (such as the digital representation of census tracks mentioned above) with a database table.¹⁷ This is called a linked-table collection and is established with a vector link file. The second is simply a collection of independent raster layers into an associated group which is defined by a raster group file. In both cases, all the elements of the collection must reside in the same folder.

Link Files

Link files associate a feature definition vector file with a database table. As a consequence, each of the fields (columns) of data in the data table becomes a layer, with the entire set of fields producing a linked-table collection. As you will see, IDRISI has special features for the display and analysis of relationships between layers in such a collection. All that is required is a link file that can establish four important properties of the collection:

1. the feature definition (i.e., spatial frame) vector file;

17. Feature definition vector files and image files describe the geography of features but not their attributes. Rather, the features have identifier values that are used to associate attributes stored in database tables or attribute values files.

2. an associated attribute database;
3. a specific table within that database; and
4. the database field that contains the identifiers which can associate database records with vector features (the *link* field).

The Collection Editor is available on the toolbar and under the File menu to help construct these link files. Only vector point, line or polygon layers can be specified as a feature definition vector file. The database files and associated tables must be Microsoft Access Database Engine files with an ".mdb" extension.

Group Files

The second type of collection is a raster group file. As the name suggests, a group file is simply a listing of a set of independent layers that one wishes to associate together. For example, a group file might consist of the seven bands of imagery that make up a Landsat satellite image. Many of the image processing procedures in IDRISI allow you to specify a group file as input rather than each individual band to be used. This saves a lot of time when the same group of images is used in several procedures. However, there are more compelling reasons for creating group files than as a simple means for specifying collections of files.

1. When more than one member of a group is displayed on the screen, you have the option to treat them identically during zoom and pan operations. Thus when one member is zoomed or panned, all other members of that group that are on the screen will zoom or pan simultaneously.
2. When the *feature properties* option is activated, queries about the data values for a specific layer will yield a table (or optionally, a graph) of the values for all attributes in the collection at that location.

Again, the Collection Editor (from the toolbar or the File menu) can be used to facilitate the creation of group files. Note that there is also a special variant of a group file known as a *time series* file that is used for time series analysis. Signature files used in image classification may also be collected together in *signature group files* and *hyperspectral signature group files*.

Collection Member Naming Conventions

To specify a layer that belongs to a collection, one uses the "dot" convention—i.e., the reference is a combination of the collection name and the layer name, separated by a dot. For example, suppose one had a link file named CensusTractData99 associating a feature definition vector file with a database table of census statistics. Each of the layers in the collection would be referenced by a combination of the collection name and the appropriate field name in the table. Thus the Median_Income layer would be referred to as:

CensusTractData99.Median_Income

Similarly, one might create a group file of the seven multispectral bands in a Landsat image. Perhaps the bands have names such as OahuBand1, OahuBand2, etc. If the group file that was created to associate them was named LandsatTM, then a reference to the second band would be as follows:

LandsatTM.OahuBand2

Collections and Pick Lists

The pick lists that IDRISI constructs to facilitate the selection of input files are *collection-aware*. For example, when a raster image is required, the pick list will display raster group files as well as raster images. Collections are immediately evident in a pick list by the "+" sign at their point of connection to the directory tree. If you click on the collection file, it will expand to show each of the members. These members are shown with their simple names. However, if you choose one, IDRISI will enter its full reference (using the dot convention) into the input box. In some cases, a group file name may itself be selected as input for a dialog box.

Attribute Files

An attribute file contains information about the characteristics of features, but not the geographic definition of those features. IDRISI supports two forms of attribute files, database tables and ASCII values files.

Data Tables

IDRISI incorporates the Microsoft Access Database Engine within its Database Workshop facility. Thus, this form of an IDRISI attribute files are tables and have ".mdb" extensions, and can also be used and modified with Microsoft Access or any other Access-compatible system. The linked layer collection can only be created with this type of attribute file. Microsoft Access Database files may contain multiple tables, each one of which can be linked to form a layer collection. See the chapter **Database Workshop** for more information about using database tables with IDRISI.

Values Files

The second form of attribute file that IDRISI uses is an ASCII values file. This has a very simple file structure that contains the values for a single attribute. It is stored in ASCII text format and consists of two columns of data separated by one or more spaces. The first column contains an identifier that can be used to associate the value with a feature (either raster or vector), while the second column contains the attribute value. Attribute values files have an ".avl" extension. Some modules create these as output files while others require (or can use) them as input for analyses. Attribute values files may be imported or exported to or from a database table using Database Workshop. In addition, IDRISI provides a procedure (ASSIGN) to assign those values to a set of features included in a raster image or vector file, and a companion procedure (EXTRACT) to create a new values file as a statistical summary of the values in another image for the features of a feature definition image.

Map Layer File Structures

Raster Layers (.rst)

Raster images are the most fundamental and important data type in IDRISI. They are automatically assigned an ".rst" file extension by the system. Raster layers are both simple in structure and regular in their organization, allowing an extraordinary range of analytical operations. The data structures that IDRISI uses for storing raster images are optimized for simplicity and efficiency.

Raster images define a rectangular region of space by means of a fine matrix of numeric data values that describe the condition or character of the landscape in each cell of a fine grid. These numeric grid cell values are used not only for analysis, but also for display. By assigning specific colors (in a *palette*) to designated numeric ranges, a very fine matrix-like color image is formed (so fine that typically the individual cells cannot be seen without zooming in to a very large scale).

In this grid-cell structure used by IDRISI, rows and columns are numbered starting from zero. Thus an image of 1000 columns and 500 rows has columns numbered from 0-999 and rows numbered from 0-499. Unlike the normal Cartesian coordinate system, the 0,0 cell is in the top left-hand corner. Cells are numbered left-to-right as you might expect, but rows are numbered top-to-bottom. This is common with most raster systems because of the directionality of many output devices, particularly printers, that print from top to bottom.

While the logical structure of an image file is a grid, the actual structure, as it is stored, is a single column of numbers. For instance, an image consisting of 3 rows by 5 columns is stored as a single column of 15 numbers. It is the image's documentation file that allows IDRISI modules to reconstruct the grid from this list. An image that looks like this:

10	15	9	10
----	----	---	----

1	14	10	11
14	13	11	10

has an image file that looks like this (if one could see it):

```

10
15
9
10
1
14
10
11
14
13
11
10

```

The raster documentation file, containing the number of rows and columns, allows the image to be correctly recreated for display and analysis.

As mentioned earlier, the major sub-types of raster images are differentiated on the basis of the data types that are used to represent cell values. IDRISI recognizes four raster data types: integer, byte, real, and RGB24.

1. Integers are numbers that have no fractional part and lie within the range of -32768 to +32767. Integer files are sometimes called 16-bit integer files since they require 16 bits (i.e., 2 bytes) of memory per value (and therefore per pixel). Integer values can be used to represent quantitative values or can be used as codes for categorical data types. For example, a soils map may record three soil types in a particular region. Since IDRISI images are stored in numeric format, these types can be given integer codes 1, 2 and 3. The documentation file records a legend for these, on the relationship between the integer codes and the actual soil types.
2. Byte values are positive integer numbers ranging from 0 to 255. The byte data type is thus simply a subrange of the integer type. It is used in cases where the number range is more limited. Since this more limited range only requires 8 bits of memory per value for storage (i.e., 1 byte), only half as much hard disk or memory space for each image is needed compared to integer. This data type is probably the most commonly used in GIS since it provides an adequate range to describe most qualitative map data sets and virtually all remotely-sensed data.
3. Real numbers have a fractional part such as 3.14. Real numbers are used whenever a continuous (rather than discrete) data variable is being stored with great precision, or whenever the data range exceeds that of the integer data type. The real data type of IDRISI raster images is known as single precision real numbers. Thus it can store values within a range of $\pm 1 \times 10^{38}$ with a precision of 7 significant figures. Single precision values provide a nice balance between range, precision and data volume. Each number (and thus each pixel) requires 4 bytes of storage. Note that real numbers have no discrete representation (e.g., 4.0 is the same as 3.999999999999999 at some level of precision). Thus, while it is possible to store whole numbers in a real data type, they cannot then be used as feature identifiers (e.g., for ASSIGN or EXTRACT) or in modules that require byte or integer values (e.g., GROUP).

4. RGB24 data files use 24 bits (3 bytes) for each pixel to encode full color images. The internal structure is band-interleaved-by-pixel and can encode over 16 million separate colors. RGB24 images are also constructed using the COMPOSITE module (in the Display menu). The IDRISI display system allows for interactive and independent contrast manipulation of the three input bands of a displayed RGB24 image. They are the preferred format for the display of full color images.

The documentation file associated with each image records the file data type. When a new image is created, it maintains the prevailing data type of the input image, or it produces a data type that is logical based upon standard mixed arithmetic rules. Thus, dividing one integer data image by a second integer data image yields a real data image. IDRISI accepts integer, byte and real data types for almost all operations that would logically allow this. Some modules, though, do not make sense with real data. For example, the GROUP operation that extracts contiguous groups can only do so for categorical data coded with integer numbers. If the data type in an image is incorrect for a particular operation, an error message will be displayed.

The CONVERT module can be used at any time to convert between the integer, byte, and real data types. In the case of conversion from real numbers to either the integer or byte formats, CONVERT offers the option of converting by rounding or by truncation.

The data type described above indicates the type of numbers stored in an image. Another parameter, the file type, indicates how these numbers are stored. As with the data type, the file type of an image is recorded in its documentation file. Image files may be stored in ASCII, binary or packed binary formats, although only the binary form is recommended (for reasons of efficiency). Binary files are those which are stored in the native binary encoding format of the operating system in use. In the case of IDRISI, this will be one of the Windows operating system variants. Thus, for example, the binary coding of a real number is that adopted by Windows and the Intel hardware platform.

Binary files are efficient in both hard disk space utilization and processing time. However, the format is not universal and not always very accessible. As a consequence, IDRISI also provides limited support for data recorded in ASCII format. A file in ASCII format is also referred to as a *text* file, and can be viewed directly with any text editor (such as the Edit module in IDRISI). The ASCII file type is primarily used to transfer files to and from other programs, since the coding system is a recognized standard (ASCII = American Standard Code for Information Interchange). ASCII files are not an efficient means of storing data and they cannot be displayed. You should therefore convert your files to binary format before getting too involved with your analyses. The CONVERT module converts files from ASCII to binary or binary to ASCII. Although binary files cannot be examined directly, IDRISI provides strong data examination facilities through the IDRISI File Explorer on the File menu. Binary files may be viewed as a matrix of numbers using the View Structure utility of the File Explorer. There is rarely a need then to convert images to ASCII form.

The packed binary format is a special data compression format for binary integer or byte data, known as run-length encoding. It played a special role in earlier MS-DOS versions of IDRISI before file compression was available at the operating system level. However, it is of limited use now that Windows takes an active role in data compression. Its support is largely for purposes of backward compatibility with earlier versions of IDRISI. Like ASCII files, most IDRISI modules still support the packed binary data type. However, also like ASCII files, packed binary files cannot be directly displayed. Thus the use of this file type is also discouraged. As with other conversions, converting raster images to or from packed binary is undertaken with the CONVERT module.¹⁸

Raster Documentation Files (.rdc)

Each of the primary file types used by IDRISI (Raster, Vector and Attribute) is associated with a companion documentation file. Raster documentation files are automatically assigned an ".rdc" file extension by IDRISI.

18. There are some important considerations in determining whether the packed form will actually reduce data storage requirements. *Primarily, the packed binary format will only save space where there are cells with a frequency of identical values next to one another.* See the CONVERT module description in the on-line Help System for some important tips in this respect.

Documentation files are always stored in ASCII format. They can be viewed and modified using the Metadata utility from the toolbar icon or File menu.¹⁹ After specifying the file type you wish to view, you can select a file of that type by choosing it from the file list. The contents of its documentation file will then be displayed.

The documentation file consists of a series of lines containing vital information about the corresponding image file. The first 14 characters describe the contents of the line, while the remaining characters contain the actual data. For example, the documentation file for a soils image (soils.rst) might look like this:

```
file format   : IDRISI Raster A.1
file title    : Major Soils Groups
data type     : byte
file type     : binary
columns       : 512
rows          : 480
ref. system   : US83TM18
ref. units    : m
unit dist.    : 1
min. X        : 503000
max. X        : 518360
min. Y        : 4650000
max. Y        : 4664400
pos'n error   : unknown
resolution    : 30
min. value   : 0
max. value   : 3
display min  : 0
display max  : 3
value units  : classes
value error   : 0.15
flag value   : 0
flag def'n   : background
legend cats  : 3
code 1        : Podzol Soils
code 2        : Brown Podzolic Soils
code 3        : Gray-Brown Podzolic Soils
lineage       : Soil polygons derived from 1:5000 scale color air photography
lineage       : and ground truth, with the final compilation being adjusted to the
lineage       : map base by hand.
```

19. The Metadata utility replaces the modules DOCUMENT and DESCRIBE in earlier versions of IDRISI.

comment : Value error determined by statistical accuracy assessment
comment : based on a stratified random sample of 37 points.

This file contains information on major soils groups and is in byte format stored as a binary file. The image contains 512 columns and 480 rows, for a total of 245,760 values. The image is georeferenced with a reference system named US83TM18²⁰ with each coordinate unit representing 1 meter. The minimum and maximum X and Y coordinate values indicate the reference system coordinates of the left, right, top and bottom edges.

The position error indicates how close a feature's actual position is to its mapped position in the image. It is marked in the example as unknown. If known, this field should record the RMS (Root Mean Square) error of locational references derived from the bounding rectangle coordinates. This field is for documentation purposes only and is not currently used analytically by any module.

The resolution refers to the inherent resolution of the image. In most cases, it should correspond with the result of dividing the range of reference coordinates in X (or Y) by the number of columns (or rows) in the image. However, there are some rare instances where it might differ from this result. A good example is the case of Landsat Band 6 (Thermal) imagery. The resolution of these data is actually 120 meters and would be recorded as such in this field. However, the data is distributed in an apparent 30 meter format to make them physically match the dimensions of the other bands in the image. This is done by duplicating each 120 meter pixel 4 times in X and then each row 4 times in Y. The resolution field is a way of correctly indicating the underlying resolution of these data.

In most images, the resolution in X and Y will be equal (i.e., pixels will be square). However, when this is not the case, the coarser resolution is recorded in the documentation file.²¹ For example, with Landsat MSS data, the pixel resolution is 80 meters in X and 60 meters in Y. In this case, the documentation file would show a resolution value of 80. Rectangular pixels will display correctly, however, preserving the true resolution in both X and Y. In addition, all analytical modules using measures of distance or area calculate resolution in X and Y independently. They, therefore, do not rely on the resolution value recorded in the documentation file.

The minimum and maximum value fields record the actual range of data values that occurs in image cells, while the display min and display max values designate what are sometimes called the *saturation points* of the data range. Values less than or equal to display min are assigned the lowest color in the palette sequence, while values greater than or equal to display max are assigned the highest color in the palette sequence. All values in between are assigned palette colors according to their position in the range (i.e., with a linear stretch). It is very common for the display min and max to match the min and max values. However, in cases where one wishes to alter the brightness and contrast of the image, altering the saturation points can produce a significant enhancement.²² Saturation points may be manipulated interactively in the display system (see the chapter **Display System** in this volume).

The value units read *classes* in this example to indicate that the numbers are simply qualitative codes for soil classes, not quantitative values. The value units field is informational only, therefore any description can be used. It is suggested that the term *classes* be used for all qualitative data sets. However, when standard linear units are appropriate, use the same abbreviations that are used for reference units (m, ft, mi, km, deg, rad).

The value error field is very important and should be filled out whenever possible. It records the error in the data values that appear in image cells. For qualitative data, this should be recorded as a proportional error. In the example, the error is

20. The reference system indicated in the documentation file is the name of a reference system parameter file (.ref). The parameters of this file are detailed in the chapter **Georeferencing** in this volume.

21. Earlier versions of IDRISI calculated and reported the resolution field as that of the X dimension only.

22. Note that one would normally only change the saturation points with quantitative data. In the example illustrated here, the numeric values represent qualitative classes. In these cases, the saturation points should match the actual minimum and maximum values.

recorded as 0.15, indicating cell accuracy of 85% (i.e., what is mapped is expected to be found on the ground 85% of the time). For quantitative data, the value here should be an RMS error figure. For example, for an elevation image, an RMS error of 3 would indicate 68% of all values will be within ± 3 meters of the mapped value, that approximately 95% will be within ± 6 meters, and so on. This field is analytical for some modules (e.g., PCLASS) and is intended to be incorporated into more modules in the future.

The flag value and flag definition fields can be used to indicate any special meaning that certain cell values might carry. The flag value indicates the numeric value that indicates a flag while the flag definition describes the nature of the flagged areas. The most common data flags are those used to indicate background cells and missing data cells. These flag definition field entries are specifically recognized and used analytically by some modules (e.g., SURFACE). Other terms would be considered only informational at this stage. Note that some modules also output data flags. For example, when SURFACE is used to create an aspect image, aspects derived from a slope of zero are flagged with a -1 (to indicate that the aspect cannot be evaluated). In the output documentation file, the flag value field reads -1 while the flag definition field reads "aspect not evaluated, slope=0".

The legend cats field records the number of legend captions that are recorded in the file. Following this, each legend caption is described, including the symbol code (a value from 0-255) and the text caption.

Finally, the image documentation file structure allows for any number of occurrences in four optional fields: comment, lineage, consistency and completeness. At present, these fields are for information only and are not read by IDRISI modules (although some modules will write them). Note that the lineage, consistency and completeness fields are intended to meet the recommendations of the U.S. National Committee for Digital Cartographic Data Standards (NCDCCDS). Along with the positional (pos'n) error and value error fields, they provide a means of adhering to the current standards for reporting digital cartographic data quality.²³ Multiple occurrences of any of these field types can occur (*but only at the end of the file*) as long as they are correctly indicated in the 14 character descriptive field to the left. The lineage field can be used to record the history of an image. The command line used to generate a new file is recorded in a lineage field of the documentation file. The user may add any number of additional lineage lines. The consistency field is used to report the logical consistency of the file; it has particular application for vector files where issues of topological errors would be reported. The completeness field refers to the degree to which the file comprehensively describes the subject matter indicated. It might record, for example, the minimum mapping unit by which smaller features were eliminated. Finally, the comment field can be used for any informational purpose desired. Note that the Metadata utility and CONVERT module both read and maintain these optional fields. Metadata also allows one to enter, delete or update fields of this nature (as well as any documentation file fields).

Vector Layers (.vct)

IDRISI supports four vector file types: point, line, polygon and text.²⁴ All are automatically stored with a ".vct" file extension. They are stored in a feature-encoded structure in which each feature is described in its entirety before the next is described. The File Structures section of the on-line Help System provides specific details about the structures of each vector file type. In addition, vector files may be viewed as ascii representations using the View Structure feature of the IDRISI File Explorer. However, the following descriptions provide all the information that most users will require.

Common Features of All Vector Files

All vector files describe one or more distinct features. Unlike raster images that describe the totality of space within a rectangular region, vector files may describe only a small number of features within a similarly defined rectangular region.

Each feature is described by means of a single numeric attribute value and one or more X,Y coordinate pairs that describe the location, course or boundary of that feature. These points will be joined by straight line segments when drawn. Thus,

23. For further information about the NCDCCDS standard, refer to the January 1988 issue of *The American Cartographer*, Volume 15, No. 1.

24. Note that the binary vector file structures for Idrisi32 are different from those of earlier versions. Consult the on-line Help System for details.

curved features require a great number of closely spaced points to give the appearance of smooth curves.

The numeric attribute values can represent either identifiers (to link to values in a data table) or actual numeric data values, and can be stored either as integer or real numbers (although identifiers must be integers).²⁵

A special feature of vector files is that their data types (integer or real) are less specific in their meaning than raster files. The integer type has a very broad range, and is compatible with both the byte and integer type of raster layers as well as the long integer type commonly used for identifiers in database tables. Very approximately, the vector integer type covers the range of whole numbers from -2,000,000,000 to +2,000,000,000. Similarly, the vector real data type is compatible with the single precision real numbers of raster layers, but is in fact stored as a double precision real number with a minimum of 15 significant figures. IDRISI's conversion procedures from vector to raster handle these translations automatically, so there is little reason to be concerned about this detail. Simply recognize that integer values represent whole numbers while real numbers include fractional parts.

Point Files

Point files are used to represent features for which only the location (as a single point location designation) is of importance. Examples include meteorological station data, fire hydrants, and towns and cities (when their areal extent is not of concern). Each point feature is described with an attribute value that can be integer or real, and an X,Y coordinate pair.

Line Files

Line files describe linear features such as rivers or roads. Each line feature in a layer is described by an attribute value that can be integer or real, a count of the number of points that make up the line, and a series of X,Y coordinate pairs (one for each point).

Polygon Files

Polygon files describe areal features such as forest stands or census tracts. Each polygon feature in a polygon layer is described by an attribute value that can be integer or real, a count of the number of *parts* in the polygon, and for each part, a list of the points (by means of an internal index) that make up that part. This is then followed by an indexed list of the X,Y coordinate pairs of all points in the polygon. The parts of a polygon are concerned with the issue of holes. A polygon with one part has no holes, whereas one with two parts has one hole. The first part listed is always the polygon which encloses the holes.²⁶

Text Files

Text vector files represent text captions that can be displayed as a layer on a map. They store the caption text, their position and orientation, and a symbol code that can be used to link them to a text symbol file. Text vector files can be created with the on-screen digitizing feature in IDRISI or by exporting text from a CartaLinx coverage. Text symbol files are created with Symbol Workshop.

Vector Documentation Files (.vdc)

As with image files, all vector files are paired with documentation files. Vector documentation files have a ".vdc" extension. Any IDRISI module that creates or imports a vector file will automatically create a documentation file. The Metadata utility can be used to update or create documentation files as required.

25. All features in a single vector layer should be encoded with numeric values of the same type. Thus, if integer identifiers are used, all features must have integer identifiers. Similarly, if real number attributes are encoded, all features in that layer must have real number attributes.

26. Note that islands are treated as separate polygons by IDRISI. Thus, if one were to take the case of the State of Hawaii as an example, one might consider it to be a single polygon with five parts. In IDRISI, though, these would be five separate polygons. However, they can each have the same identifier allowing them to be linked to a common data entry in a database table.

A sample vector documentation file appears as follows:

```
file format : IDRISI Vector A.1
file title   : Land Use / Land Cover
id type     : integer
file type    : binary
object type  : polygon
ref. system  : utm16spe
ref. units   : m
unit dist.   : 1
min. X       : 296000
max. X       : 316000
min. Y       : 764000
max. Y       : 775000
pos'n error  : unknown
resolution   : unknown
min. value   : 1
max. value   : 9
display min  : 1
display max  : 7
value units  : classes
value error  : 0.15
flag value   : 9
flag def'n   : Unknown: Obscured by Clouds
legend cats  : 7
code 1       : Residential
code 2       : Industrial
code 3       : Commercial
code 4       : Other Urban or Built Up Land
code 5       : Open Water
code 6       : Barren
code 7       : Transitional
```

As can be seen, the structure of a vector documentation file is virtually identical to that for a raster layer. The only differences are:

1. the row and column information is absent;
2. *data type* has been replaced by *id type*, although the intention is identical. Choices here are integer or real. The reason for the slightly different wording is that coordinates are always stored as double precision real numbers. The id's,

though, can vary in their data type according to whether they encode identifiers or numeric attribute values.

3. the file type is always binary. (ASCII format for vector files is supported through the Vector Export Format.)
4. the minimum and maximum X and Y values recorded in the documentation file do not necessarily refer to the minimum and maximum coordinates associated with any feature but to the boundary or limits of the study area. Thus they correspond to the BND (boundary) coordinates in vector systems such as Arc/Info.

Attribute Files (.mdb and .avl)

In addition to the attributes stored directly in raster or vector layers, IDRISI permits the use of freestanding attribute files. Two types are recognized, data tables and values files. Only the former can be linked to a vector file for display or to produce a vector collection. The latter may be used to assign new values to a raster image and will be produced when summary values for features are extracted from a raster image. Fields from data tables may be exported as values files and values files may be imported into data tables using Database Workshop.

Data Tables (.mdb)

IDRISI data tables are Microsoft Access-compatible relational database files. Thus, IDRISI attribute tables have an ".mdb" extension. Internally, each can carry multiple tables and can also be used and modified with Microsoft Access or any other Access-compatible system.

Values Files (.avl)

Values files contain the values for a single attribute. They are stored in ASCII text format with two columns of data separated by one or more spaces. The first column contains an identifier that can be used to associate the value with a feature (either raster or vector), while the second column contains the attribute value. Attribute values files have an ".avl" extension. The following is an illustration of a simple values file listing the populations (*1000) of the 10 provinces of Canada: where 1 = Newfoundland, 2 = Nova Scotia, 3 = Prince Edward Island, and so forth.

1	560
2	850
3	129
4	690
5	6360
6	8889
7	1002
8	956
9	2288
10	2780

The feature definition image to be used with this values file would have the value 1 for all pixels in Newfoundland, the value 2 for Nova Scotia, and so on.

Attribute Documentation Files (.adc)

As with images and vector files, attribute files (of either type) also carry documentation files, but this time with an ".adc" extension. Similarly, Metadata is the utility which can be used to create or modify the documentation file associated with an attribute file.

As with the other documentation files in IDRISI, those for attribute files are stored in ASCII format with the first 14 characters used purely for descriptive purposes.

A sample attribute documentation file appears as follows:

```
file format    : IDRISI Values A.1
file title     : Roads
file type      : ascii
records        : 12
fields         : 2
field 0        : IDR_ID
data type      : integer
format         : 0
min. value    : 105
max. value    : 982
display min   : 105
display max   : 982
value units   : ids
value error   : unknown
flag value    : none
flag def'n    : none
legend cats   : 0
field 1        : ROAD_TYPE
data type      : integer
format         : 0
min. value    : 1
max. value    : 3
display min   : 1
display max   : 3
value units   : classes
value error   : unknown
flag value    : none
flag def'n    : none
legend cats   : 3
code 1         : Major Road
code 2         : Secondary Road
code 3         : Minor Road
```

This example shows the documentation file for an ASCII attribute values file. This type of file always has two fields. Database tables will normally have many more fields. For each field, the parameters shown are repeated.

In this version of IDRISI, the following file types are supported:

ascii

Simple 2-column ASCII format files where the left field contains integer feature identifiers and the right field contains data about those features. These values files have an ".avl" file extension.

access

A database file in Microsoft Access format having an ".mdb" extension. This format is the current resident database format supported by IDRISI, and is supported by Database Workshop. It is also the format used in the creation and display of vector collections.

In the case of the simple ASCII form ("avl" file), format information is unimportant, and thus reads 0 in this example. Spaces or tabs can be used to separate fields. With fixed length ASCII and database files, format information is essential. The format line simply indicates the number of character positions occupied by the field.

byte and integer data

A single number to indicate the number of character positions to be used.

character string data

A single number to indicate the maximum number of characters.

real number data

Two numbers separated by a colon to indicate the total number of columns and the number of those columns to be used for recording decimal values. For example, 5:2 indicates that five columns are to be used, two of which are for the decimal places. Note that the decimal itself occupies one of these columns. Thus the number "25.34" occupies this field completely.

For most other entries, the interpretation is the same as for raster image layers. Valid data types include *byte*, *integer*, *real* and *string* (for character data). However, for data tables (.mdb), many of the data types recognized by the Microsoft Access Jet Engine are supported. These include:

real (-3.402823E38 to +3.402823E38 real numbers)
byte (0-255, whole numbers)
integer (-32768 to 32767, whole numbers)
longint (long integer, -2,147,483,648 to 2,147,483,647, whole numbers)
text (character strings)
Boolean (true or false)

IDRISI will document these types automatically, and knows how to convert them when they are used in the context of display (in a vector collection) or when creating an ASCII values file.

Other File Types

While the majority of data files you will work with are those describing layers and their attributes, many others exist within the IDRISI system and some of them are described below. Other more specialized file types are described in the context of specific modules and in the File Structures section of the on-line Help System.

Map Composition Files (.map)

Map composition files store the graphic instructions necessary to create a map composition using data from a set of map layers and associated symbol and palette files. They are described further in the **Display System** chapter in this volume.

Symbol and Palette Files (.sm0, .sm1, .sm2, .smt, .smp)

In order to display map layers, it is necessary to set up an association between vector features or image values and particular graphic renditions. This is done through the use of **Symbol Files** and **Palette Files**. A set of symbol and palette files is included with IDRISI, and may meet most user needs. However, for final output, it is often desirable to create custom symbol and palette files. This is done with the Symbol Workshop utility under the Display Menu. See the **Display System** chapter in this volume for more information about Symbol Workshop.

Symbol files indicate the manner in which vector features should be symbolized (such as the type, thickness and color of lines or the font, style, size and color of text). Each symbol file lists the characteristics of up to 256 symbols that are identified by index numbers from 0 to 255. In all, four kinds of symbol files are used, one each for the vector feature types: point, line, polygon and text. They are stored with file extensions of ".sm0," ".sm1," ".sm2" and ".smt" respectively. As usual, IDRISI always knows the appropriate symbol file type to use. As a result, you never need to specify a symbol file with its extension.

For raster images, graphic renditions are specified by the use of palette files. Like symbol files, palette files also define up to 256 renditions identified by index numbers from 0 to 255. However, in this case, only the color mixture (defined by the relative amounts of red, green and blue primaries) is specified. Palette files are stored with an ".smp" extension.²⁷

Reference System Parameter Files (.ref)

Reference system parameter files record information about specific geographic referencing systems. They include data on the projection, ellipsoid, datum, and numbering conventions in use with a reference system. IDRISI includes over 400 such files. The user can modify these and also create new reference system parameter files with the Metadata utility. See the **Georeferencing** chapter in this volume for more information about these files.

27. The MS-DOS version of IDRISI used a different palette file structure with a ".pal" extension. The PALIDRIS module in the import group can read and convert these files.

Display System

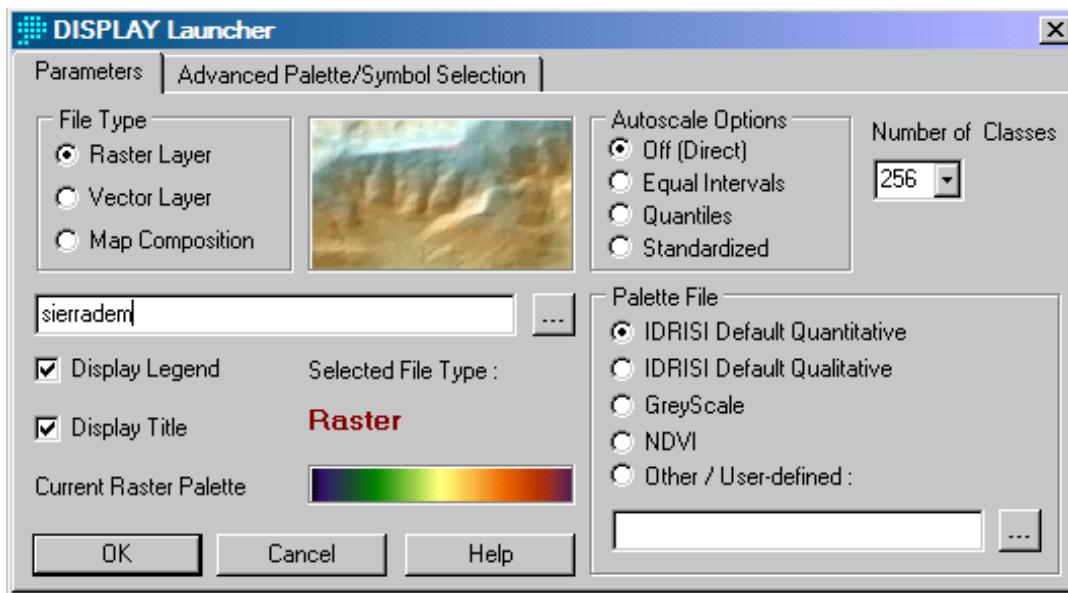
Introduction

The previous chapter described map layers as each representing a single elementary theme. By combining map layers and giving them graphic renditions, we create a *map*. Map compositions may contain as few as one layer and as many as 32 layers. Map compositions are created as a natural consequence of working with the IDRISI display system. As you work, IDRISI keeps track of all changes and additions you make to the composition. You can then save the composition at any time. The composition is stored in a file with a ".map" extension and is simply called a map file.

The display system in IDRISI consists of several separate but interdependent program components, each of which is described in this chapter.

DISPLAY Launcher

DISPLAY Launcher is used to open a new display window. It begins the map composition process, and is always the first operation required to create a new map display. DISPLAY Launcher can be accessed from its toolbar icon (shown above) or by choosing it from the Display menu. Doing so opens a dialog box with options to display a raster layer, a vector layer, or an existing map composition.



When you select a raster or a vector layer, IDRISI uses a set of decision rules based on the values in the layer to suggest an appropriate palette or symbol file. You may change this selection. You must also specify if the layer should be displayed with a direct relationship between numeric values and symbol codes, or should be autoscaled (see below). In the case of a map composition, you will only be required to specify its name, since all display parameters are stored in the map file. See the chapter **System Overview** in this volume for options on choosing filenames from the pick list or typing them in

directly. Click OK to display the map layer or composition.

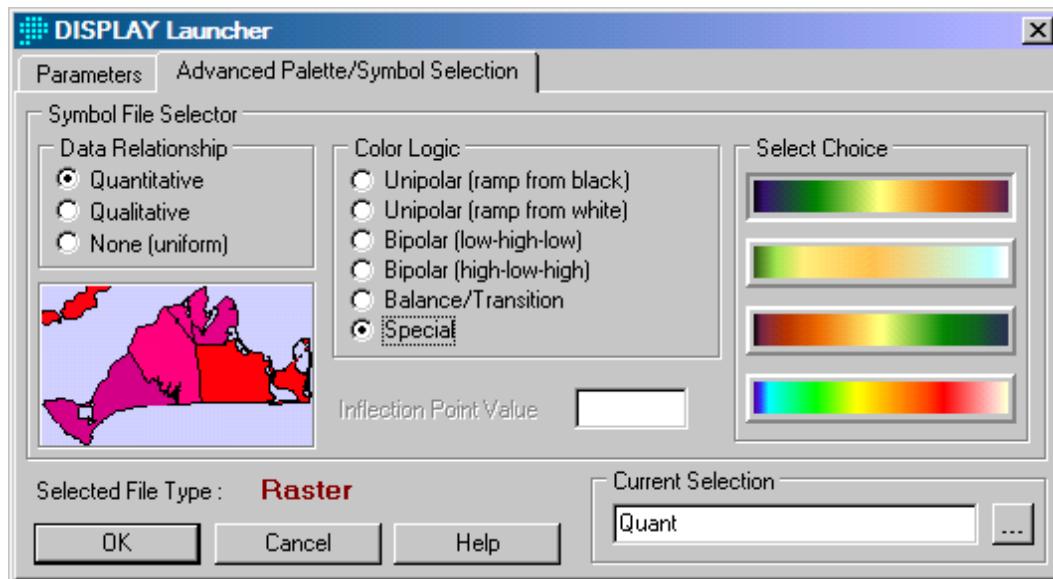
Palette and Symbol Files

Palette files define the way the information stored in image pixel values will be displayed on the screen. Similarly, symbol files define the way vector features with particular ID's or linked attribute values will appear. Each stores the character of up to 256 graphic renditions referenced by index numbers 0-255. Palette files store the Red, Green and Blue (RGB) color mixture for each index while symbol files store a variety of parameters related to specific vector object types (e.g., point size, line width, color, etc.). The section below on Symbol Workshop lists the parameters that may be defined for each type of symbol file.

IDRISI comes with a set of standard palettes and symbol files. These are installed in the symbols folder of the Idrisi Kilimanjaro program directory (e.g., C:\Idrisi Kilimanjaro\Symbols). You may wish to extend that set, either by modifying existing palettes and symbol files, or by creating entirely new ones. Symbol Workshop can be used for this purpose. User-created palette and symbol files may be stored anywhere, including the Idrisi symbols directory. Typically, users store commonly-used symbol files in the Symbols directory. Files that are specific to particular data sets are usually stored with those data.

Advanced Palette/Symbol Selection

DISPLAY Launcher also offers an Advanced Palette/Symbol Selection tab that provides simple access to over 1300 palette and symbol files.



You should first indicate whether your data express quantitative or qualitative variations. Quantitative data express differences of degree (such as elevation) while qualitative data express differences of kind (such as land cover categories). Alternatively, you can indicate that you wish to symbolize all features with a uniform symbol. You will also need to decide on the color logic. For each color logic, you are provided four choices which can be selected by pressing the appropriate button. For Qualitative and Uniform symbol schemes, the choices of the color logic will be straightforward. However, the choices for quantitative data warrant additional explanation:

- Unipolar schemes are used to symbolize data that progress either from low to high or high to low.
- Bipolar schemes have two high or two low ends, with a distinct inflection somewhere in between. For example, population change data may range from +20% to -20%. In this case we have two high ends - high positive change and high negative change, with a clear inflection point at 0%. The colors in bipolar schemes are designed to be perceived by the visual system as falling into two qualitatively different groups while maintaining a quantitative relationship in sequence.
- Balance schemes are less common. These are used in cases where two variables perfectly co-vary - i.e., declines in one are perfectly balanced by increases in the other. For example, imagine a country in which two languages are used. Districts with 40% of one language group would thus have 60% of the other. The colors in balance schemes are designed to be seen as varying mixtures of two distinctive components.

Autoscaling

Autoscaling concerns the relationship between raster cell values and palette indices (and, similarly, vector ID's or linked attribute values and symbol indices). By default, a direct relationship between them is assumed (i.e., a cell with a numeric value of 12 should be displayed with the color defined by palette index 12). However, not all images contain integer values that fall nicely within the allowable range of values for palette indices (0-255). As a result, it is often necessary to scale the actual range of values into this more limited range. For example, we might have an image layer of temperature anomalies ranging from (-7.2) degrees to (+4.6) degrees. These values cannot be directly displayed because there is no palette index of (-7.2). Autoscaling offers a solution to this problem.

Four options for autoscaling are provided: Off (Direct), Equal Intervals, Quantiles and Standard Scores.

- Off (Direct). When autoscaling is off, it is assumed that there is a direct relationship between the numeric values of pixels or features in the layer and the numeric codes in palettes and symbol files. Thus if a pixel contains the number 5 in a layer, it is symbolized with the palette color 5. This option is only permitted in cases where all pixels or features have integer values between 0-255.
- Equal Intervals. This is the default for all cases where direct scaling is not possible - i.e., for cases where some data values are less than zero or greater than 255, and all cases where the layer contains real numbers. In this case, the range of numeric values is automatically scaled (hence the name autoscaling) into a series of equal-width data classes that are subsequently assigned to symbols. Thus, for example, a layer containing elevations from 0 to 5000 meters might be automatically scaled into 10 classes, 500 meters in width. Therefore, the values from 0-499 would be assigned to symbol 0, 500-999 to symbol 1, 1000-1499 to symbol code 2, etc. DISPLAY Launcher will make an initial determination of the number of classes. However, you have the ability to specify any number from 2-256. Remember, however, that all numeric series start from 0 in IDRISI. Thus if you were to choose 256 classes, they would be understood to be symbolized by symbol codes 0 through 255. Note that this process of dividing the numeric data range into a series of symbolization classes is also referred to as classification by cartographers.
- Quantiles. A quantiles classification scheme places an equal number of pixels or features into each class, first by rank ordering the pixels or features and then assigning them to classes in rank order. Some quantiles schemes are known by special names. Thus, for example, a quantiles scheme with four classes is known as a quartiles scheme. Quantiles are a good choice whenever the data are strongly skewed or disproportionately loaded on a small number of equal interval classes.
- Standardized. A standardized classification scheme divides the data into symbol classes based on standard deviation units. With an even number of classes, the mean will represent the central boundary between classes, while with an odd number of classes, the mean will be at the center of the middle class. All classes are one standard deviation in width and the end classes always include all cases below its defining boundary. Thus, for example, the default case of 6 classes will be defined as:

$\leq -2\text{sd}$

-2sd to -1sd
-1 sd to mean
mean to +1sd
+1sd to +2sd
 $\geq +2$ sd

whereas a selection of 5 classes would yield:

≤ -1.5 sd
-1.5sd to -0.5sd
-0.5sd to +0.5sd
+0.5sd to +1.5sd
 $\geq +1.5$ sd

Standardized classes are appropriate whenever the data are approximately normally distributed and one wishes to differentiate unusual from common cases.

There are several further issues that should be noted about autoscaling. First, Equal Intervals autoscaling is automatically invoked whenever real numbers or integer values outside the 0-255 range must be displayed. Second, while the maximum range of palette and symbol indices is 0-255, some palette or symbol files may use a more limited range of values for the purpose of autoscaling (e.g., 1-100). The autoscaling range of a palette or symbol file can be inspected and changed with Symbol Workshop. Third, while Equal Intervals autoscaling works well for many images, those with extremely skewed distributions of data values (i.e., with a very small number of extremely high or extremely low values) may be rendered with poor contrast. In these cases, you can alter the display min and display max values (known as the saturation points) using the Layer Properties option of Composer (see below). Alternatively, Quantiles autoscaling can be used. Finally, note that autoscaling does not change the values stored in a data file; it only alters the display. To create a new raster image with contrast-stretched values, use the module STRETCH.

Automatic Display

IDRISI includes an automatic display feature that can be enabled or disabled from User Preferences, under the File menu. With automatic display, the results of analytical operations are displayed immediately after the operation has finished. The system will determine whether to use the Default Qualitative or Quantitative palette (both of which are user-defined in the Preferences dialog) for display and if autoscaling should be invoked. The artificial intelligence that is used to make these determinations is not fool-proof, however. Palette and autoscaling choices may be quickly corrected in the Layer Properties dialog, if necessary. The automatic display feature is intended as a quick-look facility, not as a substitute for DISPLAY Launcher.

Launching Maps and Layers from IDRISI File Explorer

You will notice that the IDRISI File Explorer has a series of display options. The default is to view the file as a graphic rendition in a map window. This uses the automatic display feature to display any listed data layer or map composition. However, it should be noted that automatic display has limited information about the layer, and thus it may not produce

the best looking display. Again, this is intended only as a quick look facility and not as a substitute for DISPLAY Launcher. Map compositions displayed from IDRISI File Explorer are displayed exactly as they have been saved, since all palette and symbol file information is included in the map file.

Map Windows and Layer Frames

A map layer is displayed in a *Layer Frame*. This and all other components of the map composition (e.g., north arrow, legend) are placed in the *Map Window*. A single Map Window includes a single Layer Frame. The other components that may be placed in a Map Window are described below in the section on Map Properties. The Map Window may be thought of as the graphic "page" upon which a map composition is arranged. Its color is set in the Map Properties dialog and it can be maximized or resized interactively by placing the cursor on the window border until it changes to a double arrow, then dragging it to the desired position.

A layer frame can be moved by first double clicking upon it (you will notice a set of sizing buttons then appear) and then dragging it to its new position.²⁸ It can also be resized by grabbing one of the visible sizing buttons and moving the layer frame border. In either case (moving or resizing), the action will be completed by clicking on any other map component or the map window banner. This will cause the sizing buttons to disappear and the layers to resize (preserving the original aspect ratio) to fit into the layer frame.

Note that there are several tool bar buttons that are useful in layer frame manipulations. These are shown in the table below. The first causes the layer frame to close up around the currently displayed layers. This is particularly useful after resizing, when the new layer frame is not exactly the right shape to hold the layers. The second causes the map window and layer frame to expand as much as possible while still being fully visible (the button can also be activated by pressing the End key). Note that in doing so, the system reserves space for Composer. If you wish to ignore Composer, hold the shift key while pressing the End key. The third button returns the map window and layer frame to their initially-displayed state. This action can also be triggered by pressing the Home key.

Toolbar Icon	Keyboard	Action
		Fit Map Window to Layer Frame
	END	Maximize Display of Layer Frame (leave space for Composer)
	SHIFT+END	Maximize Display of Layer Frame (ignore Composer)
	HOME	Restore Original Window

Composer

As soon as the first IDRISI map window has been opened, the Composer dialog box appears on the screen. Composer can be considered a cartographic assistant that allows one to:

- a) add or remove layers from the composition;

28. To drag a component, place the cursor over the component (not on the sizing buttons) and hold the left mouse button down while moving the cursor to the desired position. "Drop" the component in place by releasing the mouse button.

- b) change the order in which layers are drawn (called the priority of a layer);
- c) set a layer to be part of a color composite;
- d) set a layer to have a transparent background or blend with the layer below it;
- e) temporarily toggle a layer to be invisible (i.e., hide it) without removing it;
- f) examine and alter the properties of a layer, including the symbol or palette file in use, display saturation points, and autoscaling;
- g) add, delete and modify a variety of map components in the map window;
- h) activate Cursor Inquiry mode to examine feature properties of any layer;
- i) save the current composition (as displayed) as a MAP file; and
- j) print the composition.

If you have several map windows open, you will notice that Composer displays the information for the map window that has *focus* (or the last one to have received focus if a non-map window, such as a dialog box or other window, currently has the focus). Focus refers to the ability of a window to receive input messages from the mouse and keyboard. Windows designates the window that has focus by displaying its banner with a specific color.²⁹ To bring a window into focus, click on any part of it. To change the characteristics of a particular map composition, first give it focus.

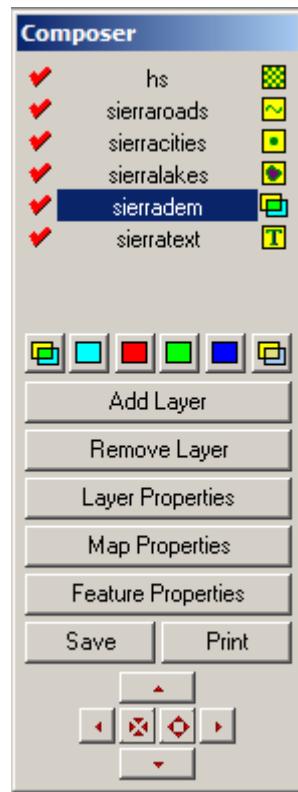
Add Layer

To add a layer, click the Add Layer³⁰ button on Composer and choose the desired file.³¹ You will then be presented with a similar dialog to indicate the palette or symbol file to be used and whether the new layer should be autoscaled. All layers in a map composition must share a common reference system. If this is not the case, a warning message will appear indicating this and warning that the layer may not display properly. The bounds of files do not need to match to be displayed together in a map composition.

When a new layer is added, it is automatically given the highest priority (i.e., it is drawn on top of the other layers). The first layer displayed will thus, by default, have the lowest priority (priority 0) and will appear to be on the bottom of the composition. The layer with priority 0 is often referred to as the *base layer* in this documentation.

When multiple layers are present, their priority can be changed by dragging the name of the layer concerned, and dropping it in the desired position. Whenever a map window is redrawn, the layers are drawn in their priority order, starting with 0 and progressing to the highest value.

Note that if a raster layer is added it will obscure all the layers beneath it unless it is designated to have a transparent background or to blend with the layers below it (see the next section for further details).



29. This is only one of many settings in the Windows system display that is defined when you choose a color scheme or define the display characteristics of individual Windows components.

30. Alternatively, pressing Ctrl-R when a map window has focus will bring up the Add Layer dialog set for adding a raster layer, and pressing Ctrl-V will bring up the Add Layer dialog set for adding a vector layer.

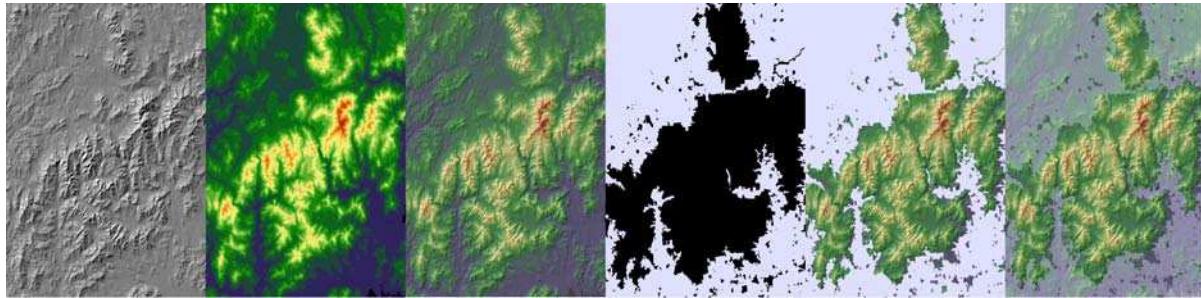
31. Only one layer frame is present in any map window, so all layers are added to the same layer frame. Note that layers may be added from several paths for display. However, if the composition is to be saved as a map composition file, all the layers must exist in the Working Folder and/or Resource Folders of the project from which it is displayed again.

Layer Interaction Buttons

Immediately above the Add Layer button is a set of buttons that control layer interaction effects. The left-most is the Blend button. If the currently highlighted layer is raster, clicking it will cause the layer to blend with the visible layer group below it by 50%. To remove the blend effect, highlight the blended layer and click this button again. The button to the far



right is the Transparency button. If the highlighted layer is raster, clicking it will cause the background color (whatever is color 0 in the palette) to become transparent. Note that layers can be both blended and transparent as illustrated below.



This sequence shows a hillshading layer and a digital elevation model being blended (third frame). In the fourth frame a mask layer has been added with black as color 0 and grey as color 1. In the fifth frame, the mask is made transparent allowing the elevation model to show through the background color. Finally, in the last frame, the mask is also set to blend allowing the topography to be partially visible through the grey mask.

The remaining buttons are for creating various color composites. The Red, Green and Blue buttons allow you to designate layers as the red, green and blue primary color components of a full color image. Note that the layers need to be adjacent to each other for this to work, but that the specific order is unimportant. The Cyan button (second from the left) is used to create three dimensional anaglyphs. The creation of an anaglyph requires stereo images -- two views of the same



landscape taken from different locations, and a set of anaglyphic 3-D glasses. When these are superimposed using Add Layer, assign red primary to the left view and the cyan primary to the right view (this order may need to be reversed depending upon your glasses). Again, the layers should be next to each other in their priority for this to work.



landscape taken from different locations, and a set of anaglyphic 3-D glasses. When these are superimposed using Add Layer, assign red primary to the left view and the cyan primary to the right view (this order may need to be reversed depending upon your glasses). Again, the layers should be next to each other in their priority for this to work.

Remove Layer

To remove a layer, select its name in the list of layers shown on Composer, then click the Remove Layer button. If you wish to only temporarily hide the layer, do not remove it, but rather, change its visibility.

Layer Names, Types, and Visibility

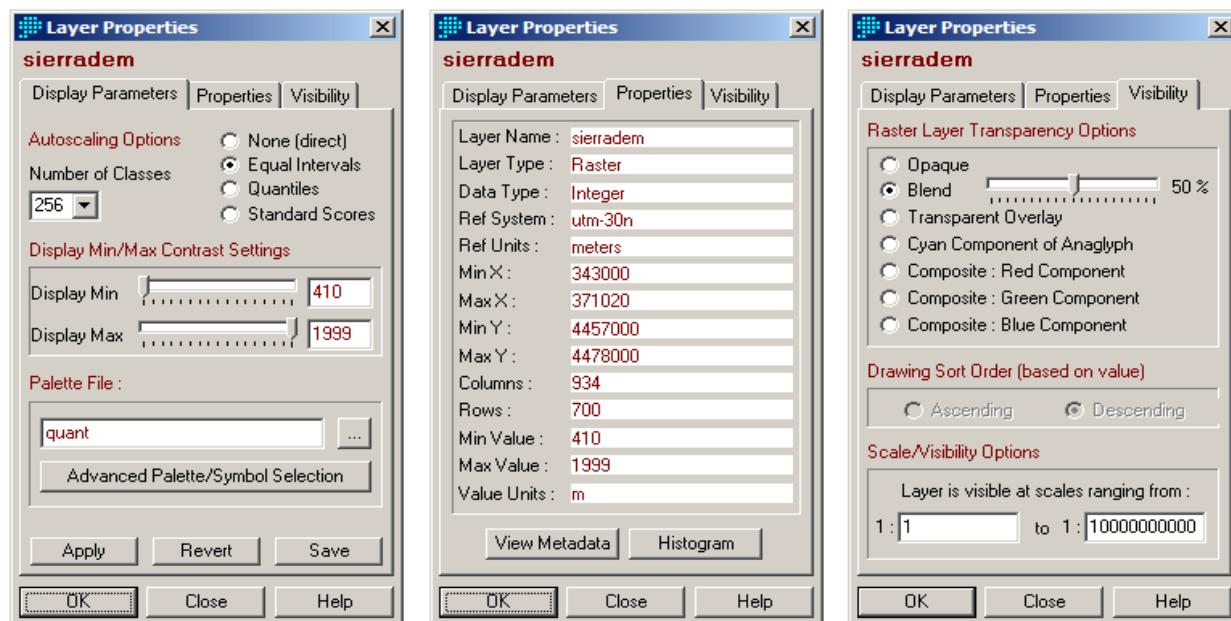
Composer displays one row for each layer in the composition. You can toggle a layer visible or invisible by clicking on the button to the left of its name. Note that toggling a layer off does not remove it from the composition. Rather, it simply makes it temporarily hidden. This can greatly facilitate visual analysis in some instances.

To the right of the layer name is a graphic symbol which indicates the layer's type: raster, vector point, vector line, vector polygon or vector text. These are illustrated in the table below. This information is used, for example, to indicate which file is which when a raster layer and a vector layer are in the same composition.

	Layer Type	Icon on Composer
	Raster	[Raster icon]
	Vector Point	[Vector Point icon]
	Vector Line	[Vector Line icon]
	Vector Polygon	[Vector Polygon icon]
	Vector Text	[Vector Text icon]

Layer Properties

The properties of any layer, including key documentation information and display parameters, are shown in the Layer Properties dialog. To view this, highlight the layer concerned with a single mouse click on the layer name in Composer, and then click the Layer Properties button. This will display the Layer Properties dialog associated with that layer.



The Layer Properties dialog summarizes several important elements of the layer's documentation file organized into three tab views. The first tab allows you to change the manner in which the image is visually scaled for display including the number of classes and the classification scheme. In addition, it allows you to change the contrast of the image if it has been autoscaled using Equal Intervals. This tab also allows you to change the palette or symbol file and gives access to the Advanced Palette/Symbol Selection page (see the section on DISPLAY Launcher).

The second tab highlights key properties and gives you the option of accessing the Metadata utility from which you may view and alter all the documentation file values. If the layer is raster, a histogram of the layer may be launched using the Histogram button. If the layer is a vector feature definition file linked to an attribute table field, the full data table may be displayed using the Access Table button.

The third tab accesses the visibility properties, including the ability to set specific blends other than the default 50% accessible from the Blend button on Composer. The drawing sort order affects vector layers only and controls the order in which vector features are drawn. This is used to establish the proper visual priority when vector features overlap. This dialog also allows you to establish at what scale the layer is visible. As you zoom in and out, layers will toggle off if the scale falls outside the visible range.

Changing the Display Min / Display Max Saturation Points

When Equal Intervals autoscaling is in effect, all values less than or equal to the display min recorded in the documentation file will be assigned the lowest symbol or palette color in the sequence. Similarly, all values greater than or equal to the display max will be assigned the highest symbol or color in the sequence.³² It is for this reason that these values are known as saturation points. Adjusting them will alter the brightness and contrast of the image.

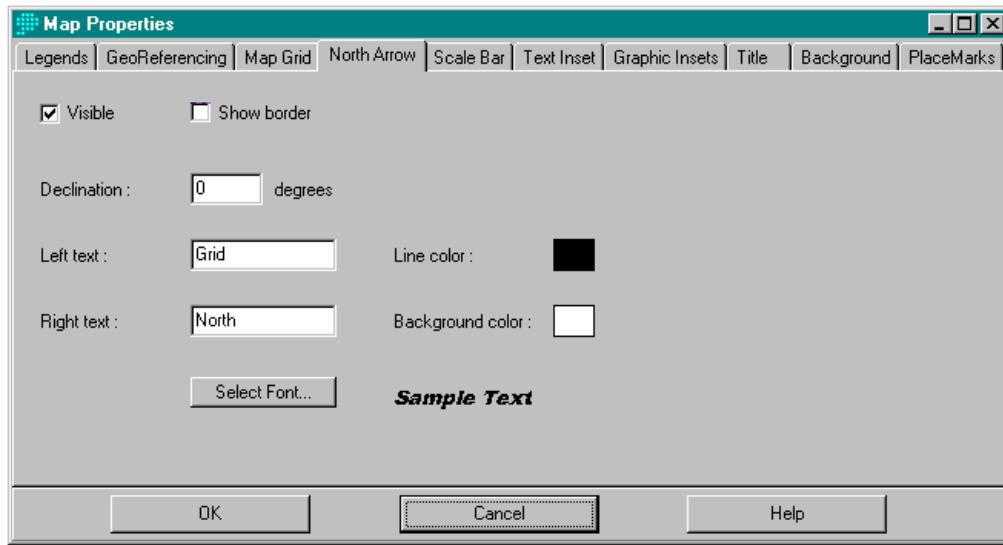
Two options exist for changing these values within the Layer Properties dialog. They can be edited within the appropriate input boxes, or they can be adjusted by means of the sliders. The sliders can be moved by dragging them with the mouse, or by clicking on the particular slider and then pressing either the left or right arrow key. The arrow keys move the slider by very small increments by default, but will yield larger movements if you hold down the shift key at the same time.³³

The best way to become familiar with the effects of changing the saturation points is to experiment. In general, moving the saturation points toward the center from their corresponding minimum and maximum data values will increase the contrast. In effect, you sacrifice detail at the tails of the distribution in order to increase the visibility of detail in the rest of the distribution. If the image seems too dark overall, try moving the display max down. Similarly, if the image seems too bright, move the display min up. Depending on the distribution of the original data values (which can be assessed with HISTO), interactively adjusting the saturation points and observing the effects can be very helpful in visual image analysis. This is particularly the case with 24-bit color composite images.

32. Note that the *lowest* and *highest* symbols or colors in the sequence are those set as the autoscale min and max in the symbol file or palette. These may be adjusted in Symbol Workshop.

33. If you always wish to display a layer with particular display min/max values that are not the actual min/max values of the layer, use Metadata to set the display min and display max fields of the layer documentation file to the desired values. Whenever the file is displayed, these values will be used to set the range of values covered by the palette or symbol file. The actual data values remain unchanged.

Map Properties



While the Layer Properties dialog displays information about a single map layer, the Map Properties dialog describes the entire map composition. It is used to set the visibility and characteristics of map components such as layer legends, north arrow and so forth.

The Map Properties dialog can be brought forward by clicking the Map Properties button on Composer or by clicking the right mouse button in a map window. The right click method is context sensitive. For example, a right click over a legend launches the Map Properties dialog with the legend options visible.

Map Properties is a tabbed dialog. Simply click on the appropriate tab to access the options associated with a particular map component. Ten tabs are provided: Legends, Georeferencing, Map Grid, North Arrow, Scale Bar, Text Inset, Graphic Insets, Titles, Background and Placemarks. The options for each are described below.

You can enlarge the Map Window at any time to increase the size of your graphic page and make space for new map components. In all cases, the components must be set to Visible to be displayed. All components are movable and many may be interactively resized. To do so, double click on the component, make the necessary changes, then click any other element to release move/resize mode.

Legends

Up to five layers may have displayed legends in a map composition. Any raster layer or vector point, line or polygon layer can have a legend. The legend text entries are stored in the layer documentation file and may be entered or altered with the Metadata utility. Up to 256 entries are allowed. By default, the legend will display up to 20 categories (this can be changed in User Preferences under the File menu). However, whenever there are more than 20 legend entries in the documentation file, a vertical scrollbar will be attached to the legend, and you may scroll through the remaining entries.

For a quantitative data layer with byte or integer data type and a range of values of 20 or less, a legend will be formed and labeled automatically (i.e., no legend entries are needed in the documentation file). If the data are real or the data range is greater than 20, a special continuous legend will be displayed with automatically generated representative category labels.

Georeferencing

The Georeferencing tab shows the reference system and units of the current composition, as well as its bounding rectangle. It also shows the actual bounding rectangle of all features within the composition. The former can be changed using the input boxes provided. In addition, a button is provided that will allow you to set the composition bounding rectangle

to the current feature bounds.

The Georeferencing tab is also used to set a very important property regarding vector text layers. Text sizes are set in *points*—a traditional printing measurement equal to 1/72 inch. However, in the dynamic environment of a GIS, where one is constantly zooming in and out (and thus changing scale), it may be more useful to relate point size to ground units. The Convert Text Point Sized to Map Reference Units option on the Georeferencing Tab does this. When it is enabled, text captions associated with text layers (but not map components, such as titles) will change size as you zoom in and out, appearing as if the text is attached to the landscape. The scaling relationship between point size and map reference units is also user-defined. When this option is disabled, the text captions retain their original size as you zoom in and out of the map layers.

Map Grid

A map grid can be placed on the layer frame. The intervals may be specified as well as the starting values in X and Y (all in reference units). You can also choose the color and width of the grid lines and the label font. The grid will be automatically labeled.

North Arrow

A north arrow can be added to a composition. You set the declination of the arrow and may specify any text to be displayed to the left and right of the arrow. This would allow you, for example, to change the text to read "Magnetic North" rather than "Grid North," or to change to a different language.

Scale Bar

Adding a scale bar can also be achieved with the Map Properties dialog. You set the length (in map reference system units), the number of divisions, color properties and text label of the scale bar. The scale bar automatically adjusts in width as you zoom.³⁴

Text Inset

The text inset can be used to display blocks of text. This is done by specifying the name of an ASCII text file. The text inset frame is sizeable and incorporates automatic wordwrap. Font style and background color may be set.

Graphic Insets

Graphic insets can be used to hold inset maps, pictures, logos, or any other graphic image. Any Windows Bitmap ("bmp") or Metafile ("wmf" or ".emf") may be used as a graphic inset. All graphic insets are resizeable. However, if the Stretchable option is disabled, resizing is controlled to preserve the original aspect ratio of the inset.

Titles

A title, subtitle, and caption may be added to a map composition. These are not associated with any particular map layer, but rather belong to the map composition as a whole. However, the title of the base layer, read from its documentation file, will appear as the title text by default. This may be edited and font properties may be set.

Background

The Background tab allows you to change the color of the backgrounds for the Layer Frame and the Map Window. The backgrounds of other components, such as legends and the scale bar, are set in their respective tabs. However, the Background tab also allows you to set the backgrounds of all components to match that of the Map Window. Note that setting

34. This can sometimes prove to be a nuisance since the scale bar may need to expand to be greater than the width of the map window. In these cases, simply set its visibility to "off" using Map Properties. Otherwise, use Map Properties to reset the size of the scale bar to a smaller length.

the layer frame background may not be noticeable if a raster layer is present in the composition (since it may cover up the entire frame). Also note that there is a default background color that is set with the User Preferences dialog under the File menu.

Placemarks

Finally, the Placemarks tab keeps track of placemarks associated with the map composition. A placemark is a particular window of the map layers. The Placemarks tab allows you to define new placemarks, delete existing ones, and go to any specific placemark. Placemarks are described further below in the section on Interactive Display Features.

Feature Properties

The Feature Properties button on Composer invokes the Feature Properties cursor mode, which brings up a table of properties associated with features identified with the mouse. This may also be invoked with the Feature Properties toolbar icon. See the section below on Interactive Display Features for a complete description.

Save Composition

At any point in your use of Composer, it is possible to save your map composition in several formats. The Save Composition button brings up a dialog presenting the format choices. The first is to save the map composition as a MAP file. A MAP file contains a complete record of the composition in a file with a ".map" extension.³⁵ This can then be redisplayed at any time by starting DISPLAY Launcher and indicating that you wish to display a map composition file. With this form of composition storage, it is possible to restore the composition to its exact previous state, and then continue with the composition process.

The next three options allow you to save the composition to either a BMP, WMF or EMF graphics file format. These file types are used in desktop publishing, spreadsheet and word processing software programs. The BMP is a raster bitmap, produced by simply copying the screen. This is often called a *screen dump*, since it copies the exact appearance of the map window, without copying the details of how the composition was formed (i.e., the names of the layers, their symbolization, and so on). The WMF format is the "Windows Metafile" structure used through Windows 3.1. It stores the Windows instructions that can be used to reconstruct both the vector and raster elements of the map window. These can then be imported into a desktop publishing or graphics program. The EMF format is the newer Windows "Enhanced Metafile" structure. This is a follow-on to the WMF structure, and is specifically designed for 32-bit variants of Windows (95, 98/ME, NT, 2000, XP). Whenever you have a choice between WMF and EMF, choose the EMF format.

The next option copies the map window to the Windows clipboard rather than to a file. This facilitates copying compositions immediately into other software programs.

Finally, the last option allows you to save the currently windowed region of the active layer as a new IDRISI layer. If the layer is raster, only the portion of the raster image that is currently displayed will be written to the new raster file. This is thus an interactive version of the WINDOW module. However, if the active layer is vector, the entire file will be copied, but the bounding coordinates will be altered to match the current window. The new file will display the window region, but all the original data outside that window still exists in the file.

Print Composition

To print a map composition, first display it. Ensure that the map window has focus, then click the Print Composition button on Composer. In the Print Composition dialog, you will be able to select the desired printer, access its setup options, choose to fit the map window as large as possible on the page or print to a user-specified scale. Page margins may be set

35. More information about the MAP file structure may be found in the chapter **Map Layers, Collections and Data Structures**. Note especially that the MAP file contains the instructions to construct the map composition, but does not include the map data layers themselves. These separate files are used to recreate the composition.

and the line widths may be scaled. A preview of the page as it will print is shown. All Windows-compatible printing devices are supported.

Navigation

At the bottom of Composer are several small navigation buttons. These allow you to pan (move) to the left, right, top and bottom and to zoom in and out. These are described later in this chapter in the section on Interactive Display Features.

Symbol Workshop

The map layers in a composition are rendered by means of symbol and palette files. While a set of symbol files is included with IDRISI, commonly you will want to develop specific symbol files to optimize the impact of the information presented on your final maps.

Symbol and palette files are created and modified using Symbol Workshop,³⁶ available under the Display menu and through its toolbar icons. In all, five types of files can be created: point symbol files, line symbol files, polygon symbol files, text symbol files and palette files. Symbol files record the graphic renditions for up to 256 symbols, indexed 0-255. For example, a text symbol file might indicate that features assigned symbol index 5 are to have their names rendered using bold, italic, 10 point Times New Roman text in a red color.

From the Symbol Workshop File menu, you can choose to open an existing file or create a new one. If you choose the latter, you will need to indicate the symbol type: point, line, polygon, text, or palette. The 256 symbols are then arrayed in a 16 x 16 grid. To change a symbol, simply click within its symbol grid cell. A symbol-specific dialog will then appear, allowing you to alter any of the following settings:

Symbol File Type	Attribute
Point Symbols	Symbol Type
	Fill Style
	Size
	Fill Color
	Outline Color
Line Symbols	Line Style
	Line Width
	Color
Polygon Symbols	Fill Style
	Color
Text Symbols	Font
	Size
	Color
	Style (normal, bold, italic, underline)
Palette	Color

36. IDRISI for Windows had a separate utility named Palette Workshop for the manipulation of palettes. This was incorporated into Symbol Workshop.

A very important feature of Symbol Workshop is the ability to copy or blend attributes. For example, imagine creating a point symbol file of graduated circles. Click on symbol 0 and set its properties to yield a very small yellow circle. Then move to symbol 255 and set it to be a large red circle. Now set the blend end points to be 0 and 255 and click the blend button. You will now have a sequence of circles smoothly changing in both size and color.

As a companion to the blend option, Symbol Workshop also provides a copy function. With both the blend and copy functions, it is possible to set options that will cause them only to copy or blend specific attributes (such as color or size). To get a sense of how the symbols will look on a particular background, you may also set a background color for the Symbol Workshop display.

Finally, the autoscaling range of a symbol file may be specified or altered in Symbol Workshop. All symbol files contain definitions for 256 symbols. However, by altering the autoscale range, it is possible to create sequences of more limited range. For example, to set up a palette with 8 colors, set the autoscale min and max to be 0 and 7 respectively. Then define these 8 colors. In use, it will then appear that the palette has only 8 colors if the layer with which it is used is autoscaled.

Media Viewer

Media Viewer is a facility for creating and displaying video images composed of a series of IDRISI images. Media Viewer is located under the Display menu. When activated, it will present the basic control dialog with its own separate menu. Click on the File menu to create a new video or to open an existing video. If you choose to create a new video, you will be presented with a new dialog that will require the name of either a Raster Group file ("rgf") or a Time Series file ("ts") that defines the images to be used and their order in the video. You will also be asked to specify a palette to be used and the time delay to be used between each successive image. The result of the operation will be an ".avi" multi-media video file that can be displayed at any time using the Media Viewer controls. Note that the viewer can be sized by dragging its borders. The image can be sized to fit within the viewer by selecting the *Fit to Window* option under the Properties menu. Also note that you can place a video into a continuous loop.

Interactive Display Features

One of the remarkable features of GIS is that map displays are not static. Rather, they provide a highly interactive medium for the exploration of map data. IDRISI includes a number of features that form the basis for this interaction.

Move and Resize

As discussed earlier, all map components (layer frame, title, legend, etc.) can be moved, and most can also be resized. Moving is achieved by placing the cursor over the component in question, double clicking to enter move/resize mode, and then holding down the left mouse button to "drag" that element to its new location. Clicking on any other component, or the map window banner, will exit move/resize mode. While in move/resize mode, resizing is achieved by dragging one of the resizing buttons on the margins of that component.

Cursor Inquiry Mode

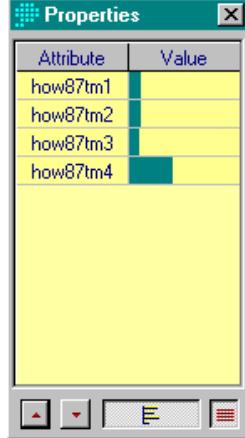
As you move the cursor over a map window, the status bar at the bottom of the screen will indicate the X and Y coordinates of the cursor in the geographic reference system (for raster layers, the column/row position is also shown). You can find out *what* is at that location by using Cursor Inquiry Mode. To enable Cursor Inquiry mode, click on its button on the tool bar. The button will appear to be depressed. When this mode is active, you can query the value at any position on the active layer of any map window. In the case of a raster layer, it will show the numeric value and legend interpretation (if one exists) of the grid cell immediately below the cursor. For a vector layer, it will show the numeric value and legend

interpretation of the nearest feature. Note especially that when a map window contains several layers, the displayed value is for the active layer (the one highlighted in Composer). The active layer can easily be changed by clicking onto the desired layer name in the Composer list. Cursor inquiry can remain active, but you may want to turn it off if you wish move and resize map components. To do so, simply click the Cursor Inquiry button again.

Feature Properties

Cursor Inquiry Mode allows one to view the values of features for a single layer at a time. For layers that are members of a collection, the Feature Properties option allows one to look at a simple tabular view of the values for all members of the collection at the query location. To activate this feature, either click the Feature Properties button on Composer or click the Feature Properties button on the tool bar. Click it again if you wish to turn it off.

For multi-layer query mode to work, the layer must have been displayed or added to the map window as a part of a collection. This is accomplished by either typing in the layer name using the "dot logic" described in the chapter **Map Layers, Collections and Data Structures** in this volume, or by choosing the layer name from within the list of members under the collection filename in the pick list. When Feature Properties is used with layers that are not members of a group, some simple information regarding the layer is presented in the Feature Properties table along with the value and position of the location queried.



The screenshot shows a 'Properties' dialog box with a table titled 'Attribute' and 'Value'. The table contains four rows:

Attribute	Value
how87tm1	
how87tm2	
how87tm3	
how87tm4	

Below the table are standard Windows-style scroll bars and buttons.

Clearly the major advantage of the Feature Properties table is that one can see the values of all group members at one time. In addition, it can display these as a graph. Note that controls are also provided to change the position of the separator line in table mode. To turn off Feature Properties mode, click on the Feature Properties button again, or click its corresponding icon on the tool bar. Cursor Inquiry mode is automatically invoked whenever Feature Properties is active.

Pan and Zoom

Pan and zoom allow one to navigate around the display at varying scales (magnifications). The simplest method of pan and zoom is to use the special buttons at the bottom of Composer, though the keyboard may also be used. Functions are summarized in the table below. The easiest way to understand the actions of these keys is to imagine that you are in an airplane. Zooming in lowers your altitude, thus increasing your scale (i.e., you see less area, but with greater detail). Similarly, zooming out increases your altitude, thus reducing your scale (i.e., you will see more, but with less detail). The representative fraction (RF) in the status bar changes as you zoom in and out. A similar logic is associated with the arrow keys. Pressing the right arrow key (Pan Right) moves your imaginary airplane to the right, causing the scene to appear to move to the left.

Composer Button	Keyboard	Action
	Left Arrow	Pan Left
	Right Arrow	Pan Right
	Up Arrow	Pan Up
	Down Arrow	Pan Down
	PgDn	Zoom In
	PgUp	Zoom Out

In addition to these continuous zoom and pan options, the tool bar also offers a zoom window option. This feature allows you to draw a rectangle around the area you wish to zoom into. To do so, click the Zoom Window button (shown below) and move your cursor to any corner of the area to be zoomed into. Then hold down the left button and drag out the rectangle to the corner diagonally opposite the one you started with. When you release the left mouse button, the zoom will take place. The Restore Original Window button on the tool bar or the Home key will revert the map window and layer frame back to their original size.

Toolbar Icon	Keyboard	Action
		Zoom Window
	HOME	Restore Original Window

All zoom and pan operations take place within the context of the layer frame. See the earlier section of this chapter on Map Windows and Layer Frames for information about enlarging both of these.

Collection Linked Zoom

When a layer is part of a collection, it is possible to have all displayed members of the collection zoom and pan simultaneously and identically. To activate this, click the Collection Linked Zoom button on the tool bar. Notice that the button depresses and remains so until clicked again. This feature is especially useful when ground truthing remotely sensed imagery, particularly when used with the GPS link (see below).

The Collection Linked Zoom works only when layers are launched with their complete collection reference. This is not an issue for linked-table collection members as they can only be launched with their full collection reference. However, for layers belonging to raster group file collections, these layers exist in isolation as well as being referenced as part of a group. Thus, if you wish to use the collection linked zoom feature, be sure to enter the full collection reference (e.g., "spotxs.band3" rather than simply "band3").

Placemarks

Placemarks are the spatial equivalent of bookmarks. Any particular view of a composition can be saved as a placemark. To do so, either launch the Map Properties dialog and click the Placemarks tab or click the Placemarks icon on the tool bar. Then choose the appropriate option to name, rename, delete, or go to any placemark. Note that placemarks are saved with map compositions. They will not be saved if the composition is not saved.

GPS Support

IDRISI also provides real-time GPS support, intended for use with a laptop computer. A Global Positioning System receiver receives continuous position updates in latitude/longitude³⁷ from the system of 21 active GPS navigation satellites. When the IDRISI GPS link is active, this position is shown graphically with a blinking cursor on all map windows that cover your current location and have a common reference system. IDRISI automatically projects the incoming positions to the reference system specified (so long as the specified projection is supported by PROJECT).³⁸ In addition, IDRISI will automatically save your route and allow you to save *waypoints*—positionally tagged notes.

Most receiver units available today support communication with a computer over an RS232C communications channel

37. Actually, the native reference system is a three-dimensional datum known as WGS84. However, the NMEA interface for communication of fixes converts these to latitude/longitude and elevation readings.

38. See the chapter **Georeferencing** in this volume for a discussion of projections and reference systems.

(e.g., the COM1 or COM2 port on your computer), and provide support for the NMEA (National Marine Electronics Association) communications protocol. In most cases, using such a GPS with IDRISI is remarkably simple, and only involves the following steps:

1. Set the GPS to output NMEA data (this is usually an option in its setup menu and can remain as a permanent setting, avoiding this step in future use).
2. Connect the special communication cable that is designed for your GPS to one of the serial communication ports on your computer. For a laptop, this is typically a 9-pin port known as COM1. By default, IDRISI expects communication over COM1, but this can be changed (see below).
3. Display an image or map layer that includes your current location. Be sure that it has focus and then click the GPS button on the tool bar. You will then see the blinking position cursor appear simultaneously in all map windows that use the same reference system (as long as your actual position is within the currently displayed area).

When you have finished a GPS session, click the GPS button again to turn it off. At that point, it will give you the option of saving your route as a vector line layer and your waypoints (if any) as a vector point layer.

Saving Routes and Waypoints

While the GPS is connected and communicating with IDRISI, it automatically saves the route that was started when the GPS button was first clicked. It also keeps a record of any waypoints that are entered along the way. When you click the GPS button again to terminate GPS communication, you have the option of saving these as layers. The route will be saved as a vector line layer and the waypoint file will be stored as a text layer.

To save waypoints, simply press the "w" key at those positions for which you wish to record information. A dialog will then appear, allowing you to enter a text description. IDRISI will keep track of both the location and the text descriptor. Note that as a text layer, your waypoint information is easily displayed after the file has been saved. However, you will probably want to keep these waypoint descriptors very brief. Position readings continue to be recorded while waypoint information is being entered.

Finally, note that IDRISI is not intended as a spatial database development tool. Thus the route and waypoint saving features are more suited to simple ground truthing operations rather than database development. For much more extensive capabilities, we recommend CartaLinx, which provides complete database development facilities with GPS support.

How It Works

When you click the GPS button, IDRISI checks to see if there is a map layer with focus and with a valid reference system. This can be any system with a Reference System Parameter file that has a projection supported by PROJECT (this does not include the system labeled "plane"). This then becomes the output reference system, and all positions will be automatically converted into that form.

Next, IDRISI launches (if it is not already active) a special GPS server program named IDRNMEA.EXE from the folder called GPS located in the Idrisi Kilimanjaro program folder.³⁹ Then it establishes communication with the GPS using communication parameters stored in a special file named "IDRNMEA.CFG," also found in the GPS folder of the Idrisi Kilimanjaro program folder. The default settings stored in this file (1,4800,n,8,1) are probably correct for your unit (since they assume the use of serial port 1 and comply with the NMEA standard) and can, in most cases, be used without modification. However, if you have trouble communicating with the GPS, this file can be edited using an ASCII text editor such as the IDRISI Edit module or Windows Notepad. Details can be found in the on-line Help System.

Once communication has been established, IDRISI converts the native latitude/longitude format of the NMEA GPS fix

39. This is the same system used by CartaLinx. The GPS server program can simultaneously serve multiple applications. Thus, if it is already loaded for use by CartaLinx, IDRISI does not load another copy, but simply registers itself as another user.

to the designated IDRISI reference system, polls all map windows to find which ones are using this system, and then moves a blinking cursor to that location on each.

Interactive Screen Digitizing

Another very important interactive capability that IDRISI provides is the ability to digitize on screen. It is important to recognize, however, that this facility is largely intended as a means of undertaking simple digitizing tasks such as the delineation of training sites for the classification of remotely sensed imagery, or creating text vector layers in a very rapid fashion. For larger and more complex tasks, we recommend CartaLinx, a full spatial database builder software, also available from the Clark Labs.

The Digitize button on the toolbar is shaped like a cross within a circle, and can be found among a group of three related digitizing buttons.

Toolbar Icon	Action
	Digitize
	Delete Feature
	Save Digitized Data

To digitize on screen, make sure that the appropriate map window has focus, and then click on the Digitize button. If the active layer (the one highlighted in Composer) is a raster layer, you will be presented with a dialog box in which you can define the new vector layer to be created. If the active layer is a vector layer, you will first be asked whether you wish to create a new vector layer for the digitized features or append the new features to those existing in the active layer. If you choose to append to an existing layer, you will notice that the file information is already filled out and you only need enter the ID or value. If you are digitizing a new file, you will be asked to specify the following:

Data Type

The data type refers to the attribute stored for each feature. This can be either integer or real. If you are digitizing identifiers to be associated with data in a table, this must be integer.⁴⁰ In all other cases, use integer for coding qualitative attributes and real for recording quantitative data (where a fractional number may be required), or whenever the value is expected to exceed the integer range.⁴¹

Layer Type

Specify the nature of the features you wish to create. Note that if you choose to append to an existing layer, both the layer type and data type are predetermined to match those of the existing layer and you will not be able to change those settings. Layer type options include points, lines, polygons or text.⁴²

Automatic Index

This option is for those cases where you are storing identifiers for features (rather than some attribute directly). When

40. With vector data, no distinction is made between different storage formats of integer. Vector integer data can have values between $\pm 2,147,483,647$.

41. Although vector layers can carry integer attribute values within a $\pm 2,147,483,647$ range, integer raster layers are restricted to a range of $\pm 32,767$. Thus, if you intend to rasterize your digitized data at a later stage, you may wish to specify real if the range is outside the $\pm 32,767$ range.

42. When polygons are chosen, an option to digitize a flood polygon is presented. This is useful in image classification for delineating training sites. See the **Classification of Remotely Sensed Imagery** chapter for more information.

checked, this option will automatically increment the identifier from one feature to the next. This offers speed when digitizing features whose ID's increment in this simple manner.

ID or Value / Index of First Feature

This allows you to specify the ID or attribute of the feature to be digitized. When the Automatic Index feature is specified, this will be used as the starting value for the numeric sequence.

Once you click OK to this startup dialog, a new layer will be added to your composition (unless you chose to append to an existing layer), and the digitizing cursor will appear in the Layer Frame. Note that if you were in Cursor Inquiry mode or Feature Properties mode, these will be temporarily disabled until you exit digitizing mode.

Mouse Button Functions When Digitizing

Once the Digitize dialog has been completed and you click on the OK button, you may begin digitizing. To digitize, use the left mouse button to identify points that define the position, course or boundary of a feature. You will notice it being formed as you digitize. To finish the feature (or the current sequence of points in the case of point features), click the right mouse button.

Deleting Features

The Delete Feature button to the immediate right of the Digitize button on the toolbar allows you to select and delete vector features from the active vector layer. First click on the Delete Feature icon, then select the feature to be deleted (the cursor will become a hand). When the feature is selected, it will turn red. To delete it from the file, press the Delete key on the keyboard.

Digitizing Additional Features Within a Single File

Once you have finished a feature by clicking the right mouse button, you can continue to add further features to the same vector layer data file. Make sure the map window still has focus, and that the layer to which you wish to append another feature is highlighted, and then click the Digitize button on the toolbar. You will be asked whether you wish to append the new feature into the designated layer.

Saving The Vector Layer

As you digitize features, they will each in turn be added to the vector layer designated. However, these data are not committed to disk until you click on the Save Digitized Data button. This can be done repeatedly throughout a session to save your work incrementally. If a map window is closed before a digitized layer is saved, a message will ask whether or not to save the layer or changes made to the layer.

Digitizing Multiple Layers

You may have multiple vector layers open for digitizing at the same time. Any actions you take will apply only to the active layer, the name of which is highlighted in Composer.

A Special Note About Digitizing Point Features

With both line and polygon layers, you digitize a single feature at a time, right click, then click the Digitize button again to digitize the next feature. However, with point vector layers, you can digitize multiple features before ending a sequence with the right mouse button. To do this, select the Automatic Indexing option.

A Special Note About Digitizing Polygon Features

Polygons are defined by lines that join up to themselves. When digitizing polygonal features, clicking the right button not only terminates the definition of the feature, but it also adds a final point which is identical to the first, thereby closing the

feature perfectly. As a result, it is not necessary to try to close the feature by hand—it will be done automatically.

A Final Note

Perhaps the best way to learn more about the IDRISI display system is to work through the first few tutorial exercises in the **Tutorial**. These give step-by-step instructions and guide you through the basics of map layer display and map composition in IDRISI.

Database Workshop

Database Workshop is IDRISI's relational database manager, and lies at the heart of IDRISI's support for layer collections that link vector feature definition files to database tables. IDRISI uses the Microsoft ADO and Access Jet Engine as the basis for Database Workshop. With this facility, one can undertake a wide variety of database operations. However, more importantly, one can interact directly with linked-table collections: database queries can be shown immediately on the associated map layer, and map layer queries can be directly linked to the data table. In addition, database field values can be assigned or extracted from raster layers. Each of these is discussed below.

Working with Linked-Table Collections

A linked-table collection consists of a vector feature definition file, a database table and a link file associating the two. The collection is defined using the Collection Editor from the IDRISI File menu or from the Establish Display Link menu entry in Database Workshop under the Query menu.⁴³ The link file contains information about the vector file, database file, table of the database file (a database file may have several tables), and link field for the collection. In a linked-table collection each field (column) in the database, linked to the geographic definition of the features in the vector file, becomes a map layer. These can each be displayed using DISPLAY Launcher by selecting the layer of interest from below the collection file name, or by typing in the full "dot-logic" name of the layer. Database Workshop offers several additional ways to examine these data. Once a display link is made, one can either query the features in a linked map layer to highlight the records in the database, or select a record in the database to highlight that feature in the vector map layer.

Launching Database Workshop

To launch Database Workshop, either click its icon on the toolbar or select its entry in the Data Entry or Analysis/Database Query menus. When launched, if the selected layer of the map window with focus is from a linked-table collection, Database Workshop will automatically open that table. Otherwise, use the File/Open menu option on the Database Workshop menu to select the desired database file and table.

Displaying Layers from Database Workshop

Simply click the mouse into any record (row) of the field (column) you wish to view and then click the Database Workshop Display icon from the Database Workshop toolbar. The selected field will then be displayed using autoscaling and the IDRISI default symbol file. Note that each such action launches a new map window,⁴⁴ so it is very easy to overload the display system if you do not have a great deal of RAM. To avoid this, close windows periodically. The first time you display a layer you will be prompted to indicate the link file to use.

Database Query using an SQL Filter

Database query by attribute is accomplished in Database Workshop by *filtering* the database. This is simply the identification of which records have attributes that meet our query (i.e., filter) criteria. To query the active database table, click on the Filter Table icon or choose the Filter Table option from the Query menu. This opens the SQL Filter dialog which provides a simple interface to the construction of a Structured Query Language (SQL) statement.

43. For more about collections, see the chapter **Map Layers, Collections and Data Structures**.

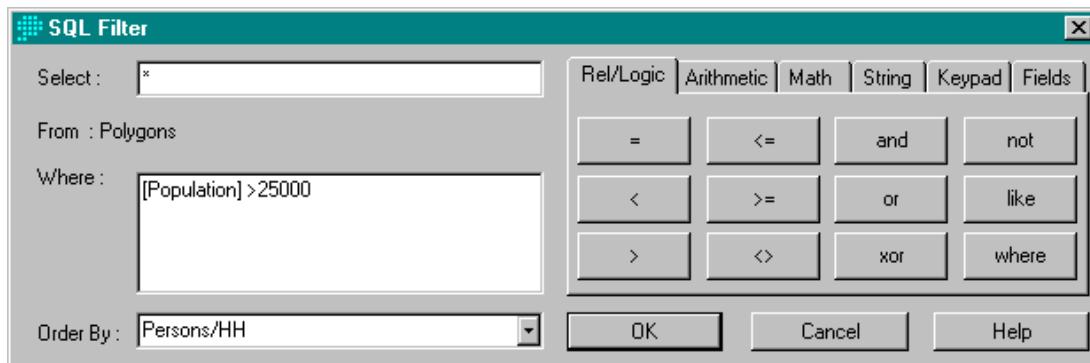
44. This may not be evident since each map window will exactly overlay the previous one. We recommend moving each new window to an unused area of the screen as it is created so that all can be seen.

The *Select* option at the top of the filter dialog specifies which fields to display in the result. The default asterisk indicates all fields and is fine in most instances. To specify a subset of fields, type their names into this input box separated by commas. Remember that all field names require square brackets around them if they contain spaces in the names. (To avoid ambiguity, it is a good habit to place square brackets around every field name.)

The *Where* input box is where the main part of the filter is constructed. The tabbed options to the right facilitate the placement of filter elements. SQL requires spaces on either side of each operator. If you select elements rather than typing them in, IDRISI will ensure that this is so. Note also that any valid SQL clause can be typed in; you are not restricted to the options shown on the tabs.

The *Order By* input box is optional. It simply causes the results of the query to be sorted according to the field chosen.

Clicking OK causes the filter to be executed. Database Workshop will then show only the records that meet the filter criteria. The records that do not meet the criteria are still in the database, but they are hidden. (Remove the filter to restore the full database—see below.)



Mapping the Filtered Records

When a filter is executed, IDRISI checks all open map windows to see if any contain an active layer that is linked to the database that was filtered. If so, it will automatically display the results of the query as a Boolean map with features that meet the filter criteria shown in red and all others shown in black.

Removing the Filter

To remove any filter, choose the Remove Filter icon from the toolbar or choose the option from the Query menu.

Query by Location

When a map window contains an active layer (the one highlighted in Composer) linked to a database, you can use Cursor Inquiry Mode (from the IDRISI toolbar) to perform database query by location. When you click on a feature in the map display, Database Workshop will automatically locate the corresponding record in the database. The located record is indicated with a triangular marker at the left edge of the record in the table display.

Other Database Operations

Calculating Field Values

In addition to querying the database, it is sometimes necessary to create new fields, either through importing external val-

ues or calculating new values from existing fields. For example, one might calculate a new field of population density values based on existing fields of population and area. The Calculate Field Values option of the Query menu (also accessed by clicking its icon on the Database Workshop toolbar) produces an SQL dialog area similar to that of the SQL Filter. In this case, it facilitates the construction of an SQL UPDATE SET operation to calculate new values for a field as a function of a mathematical or logical equation. In the SET input box, select the field to be calculated. Then enter the equation into the main input box after the "=" sign using the tabbed options as an aid. As with Filter, any valid SQL clause can be entered—you are not restricted to the options specified in the tabbed control.

Advanced SQL Queries across relational tables

An Advanced SQL editor is also available under the Query menu. It can be used to make more complicated SQL commands in the active table and across tables in the active database file. Queries can also be saved to a text file. Consult with an SQL text for advanced commands.

Finding Specific Records

The Find Next option of the Query Menu (also accessed by clicking the Find Next icon on the Database Workshop toolbar) provides a simple way to search for records. The "=" option looks for the next exact match while the "like" option looks for approximate matches. This latter option is only valid on text string fields and makes use of wildcard characters. Here are some examples:

like "*ks"	finds next record ending with "ks"
like "ks*"	finds next record beginning with "ks"
like "*ks*"	finds next record with "ks" anywhere within the text string
like "k?s"	allows any character in second position
like "k#s"	allows any digit (but not a letter) in second position
like "[a-d]*"	finds next record starting with a letter from a through d
like "[!a-d]*"	finds next record <i>not</i> starting with a letter from a through d

Sorting

To sort the records of the database according to the values of a particular field, click the mouse into any record of that field then click either the ascending or descending sort button on the Database Workshop toolbar.

Entering or Modifying Data

You will specifically need to enter Edit Mode before any cell value in the database can be entered or modified. This guards against accidental changes to the data values. Enter edit mode by choosing the option from the Edit menu, or by clicking onto the Edit Mode status button. The grid changes color when you are in edit mode. Several toolbar options are disabled until edit mode is turned off. You should therefore exit edit mode as soon as you have finished entering data. To do so, choose the option under the Edit menu, or click onto the Edit Mode status button.

Modifying the Table Structure

The table structure can be modified (i.e., add, rename or remove fields, add or delete records) from the Edit menu. However, this cannot be done if other "users" have access to the table. Any map windows that are linked to this database are considered to be users. Thus you will need to close down all of these map windows before the table structure can be altered.

Assigning Data To and Extracting Data From Raster Layers

Database Workshop provides a very simple means of assigning field data to a raster layer, or extracting data from a raster layer into a database field. To assign field data to a raster layer, use the Export/Raster Image command from the File menu in Database Workshop. A link must be established, and row and column information will be need to be specified. By default, the x and y coordinates in the new raster image will be taken from the linked vector file. To extract data from a raster image, use the Import/Raster Image command from the File menu in Database Workshop. A raster feature definition file will need to be specified representing the ID's in the raster feature definition image that was used in the extraction. These identifiers must match one field (the link field) in the database.

Assigning Data To and Importing Data from Vector Layers

Vector files can be either created or imported from the File menu in Database Workshop. A display link must first be established. Then, to import vector files, simply select the Import/Vector File command from the menu. A new table will be added to the open database containing the values in the vector file. A new ID field will also be created. To export a field to a vector file, simply highlight the field to export and select the Export/Vector File command from the menu. The vector features created will be based on the linked vector file specified in the vector collection file.

Export and Import

Database Workshop also provides selected import and export options under the File menu. Default formats supported are xBase, Microsoft Excel, and comma delimited (.csv) and text files.

IDRISI Modules

The purpose of this chapter is to give a brief overview of the functionality of each of the IDRISI modules and their typical application, and also to outline the logic of the IDRISI menu structure. For much more detailed information about the modules and their operation, please refer to the on-line Help System.

The IDRISI Main Menu is divided into seven headings, each of which is described below. Menu entries followed by an arrow lead to submenus with further choices. Because some modules are used in different contexts, they may appear in more than one place in the menu.

The names of IDRISI modules that can be used in macro mode are shown in the menu in all capital letters. All other interfaces or modules are written with only first letters capitalized.

The File Menu

Data Paths

IDRISI File Explorer

Metadata

Collection Editor

Run Macro

Shortcut On/Off

User Preferences

Import
Export

IDRISI File Conversion (16/32)

Exit

The **Data Paths** command allows you to set the paths from which input data will commonly be selected and output data will commonly be written. You may set a main working folder as well as a set of resource folders. Access to these folders is facilitated in the pick list for input and output filename boxes in all module dialog boxes. At any time, however, the user may easily browse beyond these specified paths to access the entire folder structure.

The **IDRISI File Explorer** provides a familiar interface for managing and viewing IDRISI-specific files. Use IDRISI File Explorer for regular file maintenance activities, such as copying or renaming files. The contents of IDRISI files may be viewed in a variety of formats from this utility as well.

The **Metadata** command provides the ability to create, view, and edit reference system parameter files and documentation files for raster, vector, and attribute values files.

The **Collection Editor** facilitates the creation of raster group files and vector link files. Both are ASCII text files that define sets of layers to be treated as a collection for some processes, such as cursor query.

Run Macro allows you to run an IDRISI macro. Macros contain all the information needed to run many IDRISI modules in a specific order. See the chapter on **Creating Macros** for detailed information about the construction and execution of IDRISI macros.

Toggle the **Shortcut** command on the File menu to open an alphabetical listing of IDRISI command modules. Type in the name of the module you wish to use, or select it from the list. The Shortcut dialog will remain open at the bottom of the IDRISI screen until it is toggled off in the menu.

Through **User Preferences**, you may customize your IDRISI working environment. Overwrite protection may be toggled on and off in the User Preferences dialog box. If overwrite protection is enabled, you will be prompted to cancel or continue the operation each time you provide an output filename that already exists. If overwrite protection is disabled, the existing file will be automatically overwritten without warning. You may also specify a prefix for automatically-generated output file names. Several options governing the nature of the default display choices are also set in the User Preferences dialog box.

The **Import** and **Export** entries of the File menu activate sub-menus. The **Import** tools are organized into four groups. **General Conversion Tools** leads to a list of modules that may be used alone or in combination to change files to an IDRISI format, or to convert them to a format that is compatible with a specific IDRISI import module. The **Government/Data Provider Formats** group includes modules that import the most commonly used government and agency data formats. Similarly, the **Desktop Publishing Formats** group includes modules that import graphic exchange data formats typically used in desktop publishing software. Finally, the **Software-Specific Formats** group provides the modules used to import files from many GIS and related software packages. For detailed information on specific import procedures, see the on-line Help System. For an overview of the issues surrounding database development, including data import, see the **Database Development** chapter.

The **Export** tools are organized into three groups: **General Conversions**, **Desktop Publishing Formats** and **Software-Specific Formats**. The specific commands of both the import and export submenus are listed below.

General Conversion Tools Submenu

IMPORT AND EXPORT

CRLF adds or removes carriage returns or line feeds, and is also commonly used with USGS DEM and DLG files, or in converting data files between DOS or Windows and UNIX or Macintosh systems.

XYZIDRIS is used to import X,Y,Z coordinate data such as might be collected by a GPS unit or might be entered by hand into a spreadsheet or text file. The imported result is a vector point file with the Z value as the point identifier. A vector point file may also be exported to an ASCII XYZ file with this module. The vector X,Y coordinates are written as the X and Y values and the point identifiers are written as the Z values in the output file.

IMPORT ONLY

GENERICRASTER pulls apart band-interleaved-by-line and band-interleaved-by-pixel files into separate images, respectively. It can create a proper IDRISI documentation file for band-sequential (BSQ) files. It may also be used to remove (pare off) a header from the beginning of the file or a trailer from the end of the file. It can also be used when the byte order of two-byte integer files needs to be reversed. This is often the case when data are transferred from UNIX or Macintosh systems to DOS or Windows.

VAR2FIX changes variable-length ASCII files to fixed-length files, with or without end-of-record markers. This module is often used to prepare DLG files (that are not in the standard optional format) for import with the DLG module.

Finally, **SSTIDRIS** is used to import spreadsheet data when the cells of the spreadsheet are to be interpreted as cells in the resulting image.

Government/Data Provider Formats Submenu (Import Only)

Currently supported Government and Data Provider file types are:

Landsat Enhanced Thematic Mapper satellite data in NLAPS, FAST, GEOTIFF or HDF format	imported with Landsat ETM ;
SPOT satellite data in GEOTIFF, SPOT Scene (CAP), or GEOSPOT - SPOTView formats	imported with SPOT ;
GEOTIFF/TIFF (geo-referenced TIFF) format	imported with GEOTIFF ;
HDF 4 (HDF 4 and HDF-EOS 4) formats	imported with HDFEOS ;
SAC-C satellite data from Argentina	imported with SACIDRIS ;
RADARSAT International data	imported with RADARSAT ;

Global AVHRR 10-day composite data from the USGS NASA DAAC in Goode's Homosoline projection	imported with GOODE2LL ;
USGS Raster Spatial Data Transfer Standard (SDTS)	imported with SDTS ;
USGS Digital Line Graphs (Optional Format)	imported with the module DLG ;
USGS Composite Theme Grid data	imported with CTG ;
and USGS Digital Elevation Models	imported with DEMIDRIS .

Desktop Publishing Formats Submenus

These submenus include modules that import and/or export some of the most popular file types used in Desktop Publishing for graphic data exchange. These include:

Windows Bitmap files (BMP)	imported/exported with BMPIDRIS ;
CAD DXF files	imported/exported with DXFIDRIS ;
Tagged Information File Format files (TIFF)	imported/exported with GEOTIFF/TIFF ;
JPEG files	imported/exported with JPGIDRIS ;

Note that the ability to save the currently-displayed map to a Windows Metafile (WMF) or an Enhanced Windows Metafile format (EMF) is available through the Display System Composer's Save Composition dialog box.

Software-Specific Formats Submenus (Import and Export)

IDRISI facilitates import and/or export of many common GIS and related software file structures. Several of these are ESRI format files and these are grouped under the submenu ESRI Formats. The other modules are listed in the menus alphabetically, and are a mixture of raster and vector file types. IDRISI imports raster files to the IDRISI image file format and vector files to the vector file format. The following formats are supported:

ESRI Formats Submenu

ArcView Shape files	imported/exported with SHAPEIDR ;
Arc/Info Raster Exchange Format files	imported/exported with ARCRASTER ;
Arc/Info GENERATE/UNGEN files	imported/exported with ARCIDRIS ;
Atlas*GIS BNA files	imported/exported with ATLIDRIS ;
Erdas LAN and GIS files	imported/exported with ERDIDRIS ;
ER Mapper files	imported/exported with ERMIDRIS ;
GRASS raster files	imported/exported with GRASSIDR ;
Map Analysis Package files	imported/exported with MAPIDRIS ;
MapInfo Interchange Format files	imported/exported with MIFIDRIS ;
Surfer GRD files	imported/exported with SRFIDRIS ;
S-Plus files	imported/exported with SPLUSIDRIS ;
Statistica files (to and from IDRISI images or values files)	imported/exported with STATIDRIS .

Palette files may be imported and exported using the module **PALIDRIS**. Import is supported for Idrisi for Windows (.smp), Idrisi for DOS (.pal), ERDAS (.rnb), ILWIS (.col), IAX (.iax), IMDISP (.pal) and Adobe palettes. Export is supported from the current version of Idrisi to Idrisi for DOS, ERDAS, ILWIS and IAX.

Idrisi Vector Export Format files (.vxp) may be imported/exported with **Vector Export Format** utility.

Use the **Idrisi File Conversion (16/32)** to convert between earlier 16-bit versions of IDRISI files and the current 32-bit version.

Finally, choosing **Exit** will close IDRISI.

The Display Menu

Display Launcher

ORTHO

Fly Through
Media Viewer

Symbol Workshop

COMPOSITE
SEPARATE
ILLUMINATE

HISTO
STRETCH

DISPLAY Launcher is the entry point into IDRISI's extensive display and map composition system. It allows you to either redisplay an existing composition or launch a new composition by displaying either a raster image or a vector layer. Whenever a map composition is displayed, Composer, an interactive facility for the development and modification of map compositions, automatically appears. The operation of Composer is detailed in the chapter **Display System**.

ORTHO is a facility that creates orthographic perspective (3-D) displays of digital elevation models (DEMs) or any continuous raster image. It also provides the ability to drape an image on top of the perspective model, yielding a perspective view of the draped image. The view direction, height above horizon (azimuth), and the vertical exaggeration are user-defined.

Fly Through is an interactive 3-D viewer that allows users to simulate movement through space using existing IDRISI images.

Media Viewer is a presentation utility that can play Windows video (AVI) files as well as WAV audio files. In addition to playing video and sound, the Media Viewer can create AVI video files from a sequence of IDRISI images. This can facilitate the demonstration of time series data.

Symbol Workshop allows one to create and modify symbol and palette files. Symbol files for vector point, line, polygon, and text files indicate how features should be graphically rendered (e.g., their size, color, shape, etc.). For raster image palettes, the color scheme as well as the endpoints used in autoscaling are specified. In all cases, utilities to copy settings and to blend (interpolate) attributes such as size and color across multiple symbol indices are provided.

COMPOSITE produces a 24-bit color composite image from three bands of byte binary imagery. A 24-bit composite is displayed in true 24-bit color and Composer provides independent on-the-fly adjustment of the three input bands for such images. An 8-bit composite image should be displayed with the color composite palette.

SEPARATE performs color separation of palette images into RGB components. This is a handy utility for the preparation of print-ready material.

ILLUMINATE is a hillshading merge facility that uses the color separation process performed in SEPARATE, and the color space transformations of COLSPACE for a full merge of hillshading to any palette color image. It creates a truly dramatic effect for display.

HISTO provides a frequency histogram of the cell values within an image. In addition, HISTO calculates the mean and standard deviation for the entire image and the specified data range. Two forms of output are supported, graphic and numeric. Line, bar, and area graph types are offered, as well as cumulative and noncumulative options.

STRETCH is located in this menu as it is used to prepare images for display. It may be used to increase the contrast in

an image and thereby enhance visual interpretation. Three types of stretches are offered: linear, linear with saturation, and histogram equalization. Note that using STRETCH creates a new image with stretched values. A linear stretch of the display is available in Composer for any currently-displayed image, but no new image is created with this process.

The GIS Analysis Menu

Database Query >
Mathematical Operators >
Distance Operators >
Context Operators >
Statistics >
Decision Support >
Change / Time Series >
Surface Analysis >

At the very heart of GIS is the ability to perform analyses based on geographic location. Indeed, no other type of software can provide this. IDRISI offers a wealth of analytical tools for geographic analysis.

The GIS Analysis Menu contains eight submenus. The first four are organized by toolset type, following the logic presented in the **Introduction to GIS** chapter. The remaining submenus are organized according to the particular type of analysis to be performed. These groupings are primarily for organizational convenience. Most analyses will require the use of tools from multiple submenus.

This section contains, in addition to the general descriptions of the individual modules and their typical usage, further information about the operation and application of the modules to particular problems.

The Database Query Submenu

RECLASS
OVERLAY
CROSSTAB

Edit
ASSIGN
EXTRACT

HISTO
AREA
PERIM
PROFILE
QUERY
PCLASS

Database Workshop
Image Calculator

Database Query is the most fundamental of GIS operations. Following the module descriptions is a section called Performing Database Query with IDRISI, which further describes this procedure.

RECLASS produces a new map image by reclassifying the values of an input image. Ranges as well as individual values may be specified from the original image, while the output values must be integers and must be given individually. Any original values not previously mentioned will be rounded to integers, but will otherwise remain unchanged.

OVERLAY performs operations between two images, producing a single output image. There are nine options for OVERLAY, but in the case of database query, the multiply and maximum options are most often used. In cases where each attribute of interest is represented by a Boolean image, the multiply option will give the Boolean AND, or INTERSECTION result, while the maximum option will give the Boolean OR or UNION result.

CROSSTAB compares two maps with qualitative data. The resulting image contains a unique value for each unique combination of input values. Statistics about the similarity of the two input maps may also be generated.

Edit allows access to the IDRISI text editor and may be used to create a values file for use with **ASSIGN** to assign new values to an image. The values file lists the values from the original image on the left and the new values to be assigned on the right. Any old value not present in the values file will be automatically assigned the new value zero. The original values must be integers, but real numbers may be assigned as new values. The values specified in the values file are assigned to the original values in a feature definition image. The equivalent of **ASSIGN** may also be used from Database Workshop to assign the values of a database field to an image.

EXTRACT develops summary statistics for the features in a feature definition image, given an input image. The minimum, maximum, sum, average, range, population standard deviation, and sample standard deviation may be calculated. The results may be saved as an attribute values file, or as a simple text table. The equivalent of **EXTRACT** may also be accessed from Database Workshop to fill a field with values summarized from an image.

HISTO displays a histogram of the values in an image file. The histogram may be graphic or numeric. Some summary statistics are also calculated and displayed.

AREA creates a new image by giving each output pixel the value of the area of the class to which the input pixel belonged. Output can also be produced as a table or an attribute values file in a range of measurement units.

PERIM creates a new image by giving each output pixel the value of the perimeter of the class to which the input pixel belonged. Output can also be produced as a table or an attribute values file in a range of measurement units.

PROFILE creates profiles over space by querying the values along a linear transect across an image, or over time by querying the value of the same location in several monthly images. The values in the displayed profile may be saved in an attribute values file.

QUERY extracts pixels designated by an independent mask into a sequential file for subsequent statistical analysis. The resulting file has the image file format, complete with documentation file, but cannot be displayed as an image in a meaningful way. The file is meant to be used with non-spatial statistical analyses modules, such as HISTO, or to be moved into a statistical package or spreadsheet software.

PCLASS may be used as an alternative to RECLASS when information about the level of uncertainty in an image is recorded in the Value Error field of the documentation file. In running PCLASS, you are asked to indicate a threshold, and whether you would like to evaluate the probability that each pixel exceeds the threshold or is exceeded by the threshold. The resulting image is one of probabilities, ranging between 0 and 1. RECLASS may then be used on the image to identify those areas that have at most (or at least) a given probability.

Database Workshop may be used for database query on attributes in database files. The results may be linked to a vector file for display. The equivalent of ASSIGN may be accessed from Database Workshop to apply the results of queries to image files, and the equivalent of EXTRACT may also be accessed to move extracted statistics into the database. In addition, the Calculate function of Database Workshop may be used to create new attribute fields in the database. The Filter function may be used to perform the equivalent of RECLASS and OVERLAY in the database. Both the Calculate and Filter operations are supported through the use of Structured Query Language (SQL) in Database Workshop. For more information, see the chapter on **Database Workshop** in this volume.

Finally, **Image Calculator** is an interactive mathematical modeling tool that allows you to enter a model as a full algebraic equation using a calculator-like interface. It incorporates the functionality found in the modules OVERLAY, SCALAR, TRANSFORM, and RECLASS with considerably less work and without the use of macros. Image Calculator allows for two types of expressions, Mathematical Expressions and Logical Queries, the latter of which is used in database query.

Performing Database Query in IDRISI

Perhaps the most fundamental of analytical operations undertaken in GIS is simple database query, in which we ask questions of the database and examine the results as a map. With a spatial database, two types of questions may be posed—"What locations have this attribute?" and "What is the attribute at this location?" The first is known as *query by attribute*, while the second is called *query by location*.

Query by attribute may be performed several ways, depending on the geography of the layers. If you are working with a single geography (e.g. farm fields, provinces) defined by a vector file for which you have multiple attributes in a database, the database query may be accomplished entirely in Database Workshop using an SQL filter. The results may then be linked to a vector file for display or they may be assigned to a raster feature definition image for subsequent display.

For example, if you had a map of the countries of the world, and multiple attributes for each country stored in a database, then you could perform a query such as, "Find all the countries where the median per capita annual income is less than \$5000, but the literacy rate is higher than 60%." The query conditions could be used in an SQL filter in Database Workshop, and the result linked for display to the original vector feature definition file in DISPLAY Launcher. If a new raster image file should be made, the attributes may be assigned to a raster feature definition image from Database Workshop.

However, if the geographies of the attributes of interest are not the same, or if the attributes exist only as image layers, then two steps are involved. First, the features meeting the conditions specified are selected in each layer. This normally involves the use of RECLASS or ASSIGN. Then those selected data are used in an overlay operation, provided through the module OVERLAY, to find the locations that meet all the conditions. (Both steps may be carried out with a single command in Image Calculator, but behind the interface, the individual steps are still carried out in sequence.)

For example, you might ask, "Where are all the locations that have residential land use and are within a half mile of the primary path of planes taking off from the proposed airport?" In this case, the geography of land use and that of the flight paths for the airport are not the same. A Boolean image (zeros and ones only) would be made for each condition using RECLASS or ASSIGN, then these would be combined in an OVERLAY multiply operation. The resulting image would have the value of one only where both conditions are found:

Input Image 1	X	Input Image 2	=	Output Image
0		0		0
1		0		0
0		1		0
1		1		1

Reclassification and overlay are fundamental to query by attribute in GIS. In IDRISI, RECLASS and ASSIGN are the tools used to perform database queries on single attributes, and may be used to produce Boolean images either directly or through the Image Calculator.

While RECLASS and ASSIGN may be used to produce similar results, there are several important differences between these modules. Even in cases where either may be used, generally one will be easier to use than the other. The choice will become more apparent as you become familiar with the characteristics of the two modules.

RECLASS works on an image file. The original image may have byte, integer or real values. However, the new values assigned may only be byte or integer. Original values may be specified as individual values, or as ranges of values. This information is entered in the RECLASS dialog box. Any values left out of the specified reclassification ranges will remain unchanged, except that real values will automatically be rounded to the nearest whole number.

With ASSIGN, a feature definition image file and an attribute values file are required. The latter is commonly created with Edit or imported from a spreadsheet or statistical software package. The data values in the feature definition image must be byte or integer. However, the new value to be assigned may be byte, integer or real. Both old and new values must be specified as single numbers, not as ranges. The old and new values are entered in a values file, rather than in the ASSIGN dialog box. Any original values not specified in the values file will automatically be assigned the new value zero in the output image.

Whenever the query involves more than one attribute, it is necessary to use OVERLAY. (Again, the user may choose to use OVERLAY directly, or through the Image Calculator.) For example, to find all agricultural land on soil type 6 requires that we first isolate soil type 6 as a Boolean image from the soils layer, and the agricultural land as a Boolean image from the land use layer. These two Boolean images are then overlaid, using the multiplication operation, to find all cases where it is soil type 6 AND agricultural.

Similarly, the maximum option of OVERLAY may be used to produce the Boolean OR result:

Input Image 1	Max	Input Image 2	=	Output Image
0.00		0.00		0.00
1.00		0.00		1.00
0.00		1.00		1.00
1.00		1.00		1.00

All of the logical operations can be achieved similarly through simple operations on Boolean images. For example, the Boolean XOR (exclusive OR) operation can be performed with an addition operation, followed by a reclassification of all values not equal to 1 to 0. In developing models that require this kind of logic, it is often helpful to construct a table such as the ones above in order to determine the type of IDRISI operations needed.

In Image Calculator, these analyses are built as logical expressions. While Image Calculator often provides a faster and easier interface, there are advantages, particularly to those new to GIS, to using the modules directly and performing each step individually. Doing so allows each step in the process to be evaluated so that errors in logic may be detected more easily. Using the modules individually also allows the user to become more familiar with their operation, facilitating the use of these modules outside the limits of database query.

Query by location is most easily accomplished in IDRISI with the Cursor Inquiry tool in the Display System. Select the Cursor Inquiry icon and with your cursor placed on the location in question, click the left mouse button. The underlying data value for that location will be displayed on the screen.

Query by location can be extended to include query across multiple raster files by simply creating a raster image group file (.rgf) that contains all files pertaining to a particular group. A query by location in any of the grouped images will bring up information about the pixel value at that location for all the images in the group. Similarly, query by location in a vector file that has associated database and vector links files (.vlx) will bring up all the linked database field values for the queried object. Group and link files are created with the Collection Editor, under the File menu.

Other tools for database query by location include PROFILE, QUERY, WINDOW and EXTRACT. Each of these gives results based on the attributes found at the location of input features.

The Mathematical Operators Submenu

OVERLAY SCALAR TRANSFORM

Image Calculator

IDRISI, like most raster geographic analysis systems, provides a set of mathematical tools necessary for complete map algebra.

OVERLAY performs mathematical operations between two input images to produce a single output image. The options in OVERLAY are:

- | | |
|---------------------|---|
| 1. Add | Image 1 + Image 2 |
| 2. Subtract | Image 1 - Image 2 |
| 3. Multiply | Image 1 x Image 2 |
| 4. Ratio (Divide) | Image 1 / Image 2 |
| 5. Normalized Ratio | (Image 1 + Image 2) / (Image 1 - Image 2) |
| 6. Exponentiate | Each pixel of Image 1, raised to the power in Image 2 |
| 7. Minimize | The minimum of Image 1 and Image 2 |
| 8. Maximize | The maximum of Image 1 and Image 2 |
| 9. Cover | Image 1 covers Image 2, except where zero |

All OVERLAY operations work on a cell-by-cell basis. The following illustrates the result of adding two images.

<table border="1"> <tr><td>3</td><td>2</td><td>5</td></tr> <tr><td>4</td><td>6</td><td>7</td></tr> <tr><td>6</td><td>5</td><td>9</td></tr> </table>	3	2	5	4	6	7	6	5	9	+	<table border="1"> <tr><td>5</td><td>4</td><td>1</td></tr> <tr><td>4</td><td>3</td><td>3</td></tr> <tr><td>2</td><td>1</td><td>0</td></tr> </table>	5	4	1	4	3	3	2	1	0	=	<table border="1"> <tr><td>8</td><td>6</td><td>6</td></tr> <tr><td>8</td><td>9</td><td>10</td></tr> <tr><td>8</td><td>6</td><td>9</td></tr> </table>	8	6	6	8	9	10	8	6	9
3	2	5																													
4	6	7																													
6	5	9																													
5	4	1																													
4	3	3																													
2	1	0																													
8	6	6																													
8	9	10																													
8	6	9																													

While OVERLAY performs mathematical operations between two images, **SCALAR** undertakes arithmetic operations between a constant and a single image. Its options are: add, subtract, multiply, divide, and exponentiate. In all cases, all cells in the image are identically operated upon by a single value. For example, exponentiating an image by two will produce a new image where each cell is the square of its previous value. This module is very useful in evaluating image-wise regression equations.

TRANSFORM undertakes mathematical transformations on the attributes of a single image. Options include natural logarithms and antilogs, a logit transformation, reciprocal, square and square root, absolute value, and all of the trigonometric operations. The log transform, for example, produces a new image where each cell is the natural logarithm of the corresponding cell value in the input image.

In summary, OVERLAY operates between two images, SCALAR operates between an image and a constant value, and TRANSFORM operates only on a single image.

While database query is the most fundamental of GIS operations, map algebra probably represents the second level. Indeed, map algebra is the core of derivative mapping. For example, from a set of input layers including soil type, land use, and slopes, we can derive through various mathematical and database query operations a new map of soil erosion potential.

As with Database Query, these fundamental mathematical operations are also available in **Image Calculator**. In its application to map algebra, the mathematical expression option of Image Calculator is used.

The Distance Operators Submenu

DISTANCE
SPDIST
COST
BUFFER

VARCOST
DISPERSE
RESULTANT
DECOMP

PATHWAY
ALLOCATE
RELOCATE
THIESSEN

The third submenu of analytical tools consists of those that may be called distance operators.

DISTANCE calculates the true Euclidean distance of each cell to the nearest of a set of target cells as specified in a separate image. Distances are output in the reference units specified in the target feature image documentation file.

SPDIST is the equivalent of the DISTANCE module, except that it accommodates the special case of spherical distance units (degrees, radians). SPDIST calculates spherical distances on the surface of the earth from the designated target features using spherical trigonometry.

COST calculates a distance/proximity surface where distance is measured as the least cost distance in moving over a friction surface. For example, the cost distance from a given location can be calculated based on a slope image to gauge the difficulty of crossing terrain. COST requires a target feature image and a friction image as input. IDRISI provides two choices of algorithm to use in COST distance calculations. The COSTPUSH algorithm uses a pushbroom logic and is suitable for friction surfaces that are not overly complex. COSTGROW can accommodate complex friction surfaces, as well as absolute barriers to movement, but requires a longer processing time.

BUFFER creates buffers around any set of specified features in an image. The user specifies the buffer width from the

features in reference units and indicates the output values for the original features, the buffer zones and areas outside the buffer zones.

The next set of four modules are used when frictions act with different strengths depending on the direction of movement. For a detailed discussion, consult the chapter on **Anisotropic Cost Analysis**.

VARCOST computes an anisotropic cost surface for movement having motive energy behind it. An example might be a person walking across a landscape where going up a slope acts as an impediment to movement, while going down the same slope acts as a facilitation to movement. Since the apparent friction experienced depends on the direction of travel through it, the friction surface required for input is in the form of a pair of images—a friction magnitude image, and a friction direction image. In addition, a target feature image is required, and a user-defined function relating direction to friction may be entered if the default function is not desired.

DISPERSE is similar to VARCOST except in this case, the phenomenon for which we are modeling movement has no motive force behind it. Rather, it is subject to anisotropic *forces* that cause it to move. An example might be the dispersion of plant pollen across the landscape, which depends on the force and direction of the wind. A force magnitude and direction image pair is required as input, as well as a target feature image. Again, if the default function relating direction to force is not desired, a user-defined function may be input.

RESULTANT computes the resultant force vector (as a magnitude and direction image pair) from two input force vector image pairs. This may be used to incorporate multiple sources of anisotropic frictions or forces for subsequent use with VARCOST or DISPERSE.

DECOMP decomposes a force vector (as a magnitude and direction image pair) into X and Y component images. It also takes X and Y component images and produces a force vector image pair. This module is most commonly used in developing friction image inputs for VARCOST or DISPERSE. However, it is also useful in applications such as change vector analysis, as outlined in the chapter **Change/Time Series Analysis**.

PATHWAY calculates the route of least cost distance between one or more points and the lowest point on an accumulated cost distance surface produced with COST, VARCOST or DISPERSE. This may be used to identify the least cost pathway across a landscape, given a cost surface.

ALLOCATE performs spatial allocation based on a distance or cost-distance image derived from DISTANCE, COST, VARCOST or DISPERSE. For example, if the distance surface were calculated from a set of health centers, ALLOCATE could be used to assign each cell to its nearest health center. The final result of ALLOCATE would be an image with identifiers indicating to which health center each cell is assigned. The result from ALLOCATE is identical to that of THIESSEN where simple Euclidean distance is used. However, if cost distance is used, the result may be quite different.

RELOCATE moves non-zero features in an image to a target set of features in another image based on minimum distance. This might be used, for example, to locate villages at the nearest position on a road network.

THIESSEN produces Thiessen polygons around a set of irregularly distributed points. The polygons are derived by drawing divisions exactly halfway between the original points, such that each pixel is assigned to its nearest source point. A division of space into polygons of this nature is also known as a Voronoi Tessellation.

The Context Operators Submenu

SURFACE
FILTER
PATTERN
TEXTURE
GROUP
VIEWSHED
WATERSHED
HINTERLAND
PIXEL LOCATION

The fourth toolset in the Analysis menu contains *context* operators (also known as local or neighborhood operators). With context operators, each cell in the output image is assigned a value based on its value in the original image and the values of its surrounding neighbors.

SURFACE calculates either the slope or the aspect of surface cells from a given input image of terrain heights (a DEM) or any quantitative and continuous variable. SURFACE makes this calculation by comparing the heights of cells to those of their neighbors. From this information, the gradient of the slope can be determined for the middle cell, as well as the aspect of the slope (the direction of maximum descent). The latter is expressed as an azimuth (an angle measured clockwise from north). SURFACE is also able to use this information to create a shaded relief image through a technique called "analytical hillshading." Slope calculation is not limited to terrain models. It can be quite useful in highlighting areas of rapid change in any quantitative image. A slope image derived from incidence of disease data, for example, would indicate areas of relatively homogenous incidence (low slope) and areas of rapid change in incidence (high slope).

FILTER is most often thought of as an Image Processing function, and in fact appears in that submenu as well. However, it also has application in geographic analysis. FILTER creates a new image by calculating new values using a mathematical operation on the original cell value and its neighbors. The nature of this operation is determined by the values stored in a 3 by 3, 5 by 5, 7 by 7, or a variable sized template or kernel, that is centered over each pixel as it is processed. The following filters are available in IDRISI: mean, Gaussian, median, standard deviation, adaptive box, mode, Laplacian edge enhancement, high pass, Sobel edge detection, and user-defined.

The simplest filter is a mean filter in which the new value is the average of the original value and that of its neighbors. The result is an image that is "smoother" than the original. Mean and Gaussian filters are often used to generalize an image or in smoothing terrain data after interpolation by means of INTERCON. The mode filter, which assigns the most common value to the center pixel, is also commonly used to remove very small areas from a qualitative image, or slivers left after rasterizing vector polygons. A median filter is useful for random noise removal in quantitative images. The adaptive box filter is good for grainy random noise and also for data where pixel brightness is related to the image scene but with an additive or multiplicative noise factor. Edge enhancement filters, such as the Laplacian, accentuate areas of change in continuous surfaces. High pass filters emphasize areas of abrupt change relative to those of gradual change. The Sobel edge detector extracts edges between features or areas of abrupt change. Finally, the user-defined filter option allows the user to specify any kernel size as well as the mathematical operation, and is useful for simulation modeling.

PATTERN computes various numerical pattern indices using a 3 by 3, 5 by 5, or 7 by 7 template. It computes measures that are commonly used in fields such as landscape ecology, based on the values of the template. PATTERN is also useful for processing of remotely sensed imagery, especially RADARSAT data. Relative richness, diversity, and fragmentation are just a few examples of the measures provided.

TEXTURE provides options to calculate several measures of variability, fractional dimension, class frequency, and edge analysis. All these are calculated based on a 3 by 3, 5 by 5, or 7 by 7 template.

GROUP finds polygons in an image by looking for contiguous groups of pixels sharing the same attribute. The groups are numbered consecutively as they are found, such that each contiguous group of pixels is assigned a unique attribute. This could, for example, be used to isolate distinct stands of vegetation, where all stands have the same value in the original image. The user may designate whether pixels touching only at a corner should be considered neighbors.

VIEWSHED calculates all cells directly in view of a set of target cells specified on a separate image. To do so, VIEWSHED extends visual rays in all directions and traces the line of sight to the height of cells to determine whether or not they are in view.

WATERSHED calculates all cells belonging to the watersheds of one or more target cells. It does so by working from the target cells outward (i.e. uphill), evaluating each pixel to determine whether or not it receives flow from a neighboring

cell. If a pixel receives flow, it is classified as part of the watershed. This process is continued until a divide is reached.

HINTERLAND determines the supply area dominated by point demand centers. Given a supply map and a second point demand map, HINTERLAND allows the demand centers to use up supplies beginning at the demand centers and moving outward, until either their demands are met, or there are no more supplies available. Positive values in the output image indicate pixels with a surplus, negative values indicate a deficit, and zeros indicate a balance between supply and demand.

PIXEL LOCATION creates new images representing the X and/or Y coordinate of each cell center. For example, if the input image were in the Lat/Long reference system, one might use PIXEL LOCATION to create an image of the longitude of each cell center.

The Statistics Submenu

HISTO
EXTRACT
PATTERN
COUNT

REGRESS
MULTIREG
LOGISTICREG
TREND

AUTOCORR
QUADRAT
CENTER
CRATIO

CROSSTAB
BALIDATE
ROC

SAMPLE
RANDOM
STANDARD

SPLUSIDRIS
STATIDRIS

Statistics is a field that provides tools for describing groups of numbers. In IDRISI, the Statistics Submenu provides a series of tools for performing both traditional statistical analysis and specialized spatial statistics routines.

HISTO is perhaps the most often used module of this set. It provides a frequency histogram of the cell values within an image. In addition, HISTO calculates the mean and standard deviation for the entire image and the specified data range. Two forms of output are supported, graphic and numeric. There are several options for the graphic histogram including line, bar and area graphs, and cumulative and non-cumulative options. The mode may be easily determined from the graphic display. The numeric display provides a summary of frequencies within each class along with cumulative and proportional frequencies. From this, the median can easily be determined, as well as any other percentile range.

EXTRACT calculates data summaries for specific features. The features are described in an image known as a *feature definition file*, which contains integer identifiers. The summaries are then output either as a table or as an attribute values file. Summary statistics include minimum, maximum, total, average, range, mean, and standard deviation.

PATTERN performs pattern analysis based on a comparison of cell values with their surrounding cells in any of a 3 by 3, 5 by 5, or 7 by 7 pattern. Pattern measures include relative richness, diversity, the dominance index, the fragmentation index, the number of different classes (NDC), center versus neighbor (CVN), and a binary comparison matrix (BCM). These measures are very commonly used in landscape ecology studies.

COUNT calculates a relative frequency probability image derived from a set of input Boolean images. The probability is based on the occurrence of non-zero values over multiple images.

REGRESS undertakes linear regression analysis on either two images or two attribute values files. Output consists of a graphic scatter diagram and trend line, a tabular summary of the regression equation, the correlation coefficient, and several tests of significance.

Note that extreme care should be exercised in the use of the REGRESS module with image data. The regression model assumes that the observations are independent of one another. With image data, this is rare. As a result, the degrees of freedom may not accurately reflect the true number of independent observations, leading to a tendency to commit a Type 1 error (accepting the hypothesis when it is not true). A possible approach here is to use SAMPLE or CONTRACT with the thinning option, to extract a sample of cells that are far enough apart to be considered independent prior to doing the regression. AUTOCORR can be used to help verify this independence.

MULTIREG performs a multivariate regression analysis between one dependent variable and two or more independent variables. The analysis can be performed on either images or values files. The same assumptions of independence mentioned in REGRESS above also apply. This analysis is useful for identifying the contribution of factors for any particular

outcome. MULTIREG outputs a table of regression coefficients and statistics as well as predicted and residual images.

LOGISTICREG is a logistical regression analysis performed on images or values files. Unlike that in REGRESS or MULTIREG, the independent variable expresses the probability that an underlying dichotomous variable is true. This analysis can be useful in explaining the "occurrence" or "nonoccurrence" of phenomena.

TREND is perhaps the most obviously spatial statistical routine. TREND can be considered the spatial equivalent of REGRESS. TREND calculates the relationship between pixel values and their positions within the image. Three types of polynomial regression are supported—linear, quadratic, and cubic.

TREND is most often used to determine whether a significant spatial trend can be found in the attribute values of an image over space. For example, the number of cases reported of a particular disease may be available for villages over a region. TREND might then be used to determine whether a trend occurs, in the hopes of isolating the spatial direction of the progress of that disease. Given this relationship, TREND is then able to interpolate a full surface. TREND can also be used as a generalization routine, decomposing complex surfaces into simpler underlying trends. Additionally, TREND may be used to interpolate a surface from point data.

AUTOCORR is also distinctively spatial in character. AUTOCORR calculates the first-lag autocorrelation coefficient of an image using Moran's "I" statistic. The area of analysis may be limited by a mask if desired. The user may choose to calculate autocorrelation based on a pixel's four immediate neighbors (rook's case) or on the eight surrounding neighbors (king's case). The autocorrelation statistic is essential for the examination of the spatial dependence of attribute values. While the routine itself can only calculate first-lag (i.e., immediate neighbor) autocorrelation, CONTRACT can be used with its thinning option to create images representing other lags. Thus, several runs can be used to examine how the autocorrelation changes with distance. This can then be used, for example, to determine the distance necessary for spatial independence to be assumed. The geostatistics tools available through the Surface Analysis submenu also provide means for assessing spatial correlation and independence.

QUADRAT performs quadrat analysis, a useful technique for determining the character of a point set's pattern. Point patterns may be distributed, random or clustered. One of the outputs of QUADRAT is the variance/mean ratio which can be interpreted to indicate one of these three alternative patterns. QUADRAT also indicates the density of the point set.

CENTER calculates the mean center (weighted or unweighted) and standard radius for a set of points. The mean center can be considered the "center of gravity" for a point set while the standard radius is directly analogous to the standard deviation for non-spatial data. It is used as a measure of the dispersion of points from a most probable location.

CRATIO measures the compactness ratio of defined polygons. This statistic is useful in many habitat studies that wish to examine the spatial compactness of land cover units such as forest stands. The measure compares the area/perimeter ratio to that of a circle (the most compact polygon possible).

CROSSTAB provides a comparative technique for qualitative data (e.g. landcover class images). The most fundamental output of CROSSTAB is a crosstabulation table which lists the frequency with which each possible combination of the categories on the two images occur. Marginal totals are also provided. A Chi Square statistic (to judge the likelihood of association), degrees of freedom, and Cramer's "V" statistic (to measure the degree of association) are reported. In cases where the categories on the two images are identical, CROSSTAB also outputs a Kappa Index of Agreement (KIA—also known as "KHAT") measure to compare with Cramer's "V". The KIA measure can be evaluated both as an overall value and on a per category basis—an excellent means of examining change in qualitative data sets.

In addition to this table, CROSSTAB can also create a crosscorrelation image. A crosscorrelation image has new categories to illustrate all existing combinations of the categories on the two input maps. It is thus directly related to the crosstabulation table. In essence, the crosscorrelation image can be considered as a *multiple AND* overlay showing all possible combinations of the categories on one image AND those on the other. CROSSTAB also appears in the Change/Time Series submenu.

VALIDATE provides several statistics for measuring the similarity between two qualitative images. These include spe-

cialized Kappa measures that discriminate between errors of quantity and errors of location.

ROC, the Relative Operating Characteristic, provides a measure of the correspondence between a quantitative modeled image showing the likelihood that a particular class exists and a Boolean image of that class as it actually occurs (i.e., “reality”).

SAMPLE produces a vector file of points according to a systematic, random or stratified random sampling scheme. It is useful in many accuracy assessment studies and also provides a general technique for creating samples to be submitted to statistical analyses. Once a sample is created, variables for those points may be extracted from raster images with EXTRACT, either directly or through Database Workshop.

RANDOM is useful for simulation studies. It creates a new image of specified dimensions with random values that obey either a rectilinear, normal, or lognormal distribution, according to a user-specified mean and standard deviation. This might be used, for example, to produce a probability surface to determine the likelihood of certain events. RANDOM also has application in error analysis and thus is also found under the Decision Support submenu.

STANDARD is used to convert the values in an image to standard scores. To accomplish this, it subtracts the image mean from each pixel, then divides by the image standard deviation. Converting images to standard scores may be desirable before comparing them.

SPLUSIDRIS is used to import and export between IDRISI and S-Plus. Images or values attribute files can be converted.

STATIDRIS is used to import and export between IDRISI and Statistica. Images or values attribute files can be converted.

The Decision Support Submenu

Decision Wizard

WEIGHT
MCE
RANK
MOLA

STANDARD
FUZZY

COUNT
MDCHOICE

PCLASS
BAYES
Belief
RANDOM

SAMPLE
ERRMAT

One of the most important applications of GIS is that of decision support. In fact, many of the analyses performed with the modules in the other menus of IDRISI are intended to support decision making. The modules in this menu are unique in that they specifically address multi-objective, multi-criteria resource allocation decision problems, as well as problems of assessing and incorporating uncertainty in the decision making process. For an in-depth discussion of these issues, refer to the chapters on **Decision Support**.

The **Decision Wizard** is an automated assistant that steps you through single- or multi-objective multi-criteria evaluation problems. The Wizard facilitates your use of WEIGHT, MCE, RANK and MOLA. Your decision rules are recorded at each step so you may return to change parameters at any time.

WEIGHT computes a best-fit set of weights by calculation of the principal eigenvector of a pairwise reciprocal comparison matrix in which each factor in a multi-criteria evaluation is compared to every other factor. Information on consensus and procedures for resolving lack of consensus are provided. Weights derived in this manner sum to one. The weights are then used with MCE to create a multi-criteria suitability image.

MCE computes a multi-criteria evaluation image by means of either a Boolean analysis, Weighted Linear Combination (WLC) or Ordered Weighted Averaging (OWA) of factor images. Using WLC, each standardized factor image is multiplied by its weight, then the results are summed. OWA also works with standardized factor images and employs a variant of the WLC. It takes into account the risk associated with the decision and degree of tradeoff associated with the variables in the analysis. In both cases, Boolean constraint maps can be applied to limit areas under consideration in the final analysis. Finally, a strictly Boolean analysis can be employed using Boolean factor maps to produce a crisp result.

RANK orders the cells in a raster image. Ties may optionally be resolved by using the rank order of a second image. Both

primary and secondary ranks may be in ascending or descending order. The results of this procedure are used extensively in optimization problems such as with RECLASS for single objective decisions or MOLA for multi-objective decisions.

MOLA is an iterative multi-objective land allocation routine. Input maps are ranked suitability maps such as would be produced by MCE, followed by RANK. The procedure uses a decision heuristic to resolve conflicts and is suitable for use with massive data sets. Weights for the objectives are defined by the user.

STANDARD converts an image to standard scores. This type of standardization may be one option for standardizing multiple criteria images to a common scale prior to scaling them to the byte range for combination with MCE or MDCHOICE.

FUZZY evaluates the fuzzy set membership values (possibilities) of data cells based on any of three membership functions: sigmoidal, j-shaped, and linear. A fourth function allows for a user-defined membership. Monotonically increasing, monotonically decreasing, symmetric, and asymmetric variants are supported. FUZZY is used to model transitional changes in class membership in an image and is also used to standardize factor images for MCE using subjective knowledge of suitability. Other Fuzzy Set operations such as CON (concentration), DIL (dilution), AND and OR are covered by the standard modules TRANSFORM and OVERLAY.

COUNT calculates a relative frequency probability image derived from a set of input Boolean images. The probability is based on the occurrence of non-zero values over multiple images.

MDCHOICE resolves conflicts between competing objectives by means of a multiple ideal-point procedure. Axes in the multidimensional decision space can be differentially weighted and minimum suitabilities set for each. Input maps should be standardized either with STANDARD or by means of the histogram equalization procedure in STRETCH.

The remaining modules in this submenu are used in the evaluation and handling of error in geographic analysis.

PCLASS evaluates the probability with which data cells exceed or are exceeded by a specified threshold based on the stated RMS error for the input map. In Bayesian terms, this procedure evaluates the probability of the evidence given a hypothesis in the form of a threshold— $p(e|h)$.

BAYES evaluates Bayes' Theorem. Multiple evidence maps (such as those produced by PCLASS) are permitted as long as they are conditionally independent. Prior probabilities may be input in map form (and thus, like the evidence maps, may vary continuously over space). This is an extension to what is sometimes called a Bayesian Weight-of-Evidence Approach. The user may also specify the confidence in the decision rule, i.e., the belief that the evidence supportive of the hypothesis is truly reflected in the evidence at hand— $p(e|e')$.

Belief evaluates the degree to which evidence provides concrete support for a hypothesis (belief) and the degree to which that evidence does not refute the hypothesis (plausibility). Belief employs the Dempster-Shafer Weight-of-evidence Procedure in the evaluation. Unlike Bayesian Probability Analysis, Belief explicitly recognizes the possibility of ignorance in the evaluation, i.e., the incompleteness of knowledge or evidence in the hypothesis.

RANDOM creates random images according to rectilinear, normal or log-normal models. This module is particularly important in the development of Monte Carlo simulations of error propagation.

SAMPLE creates systematic, random, and stratified random point sampling schemes.

ERRMAT produces an error matrix analysis of categorical map data compared to ground truth information. It tabulates errors of omission and commission, marginal and total errors, and selected confidence intervals. Per-category Kappa Index of Agreement figures are also provided.

The Change / Time Series Submenu

IMAGEDIFF
IMAGERATIO
CVA
CALIBRATE
CROSSTAB

PROFILE
TSA
CORRELATE
Media Viewer

MARKOV
STCHOICE
DISAGGREGATE
NORMALIZE
LOGISTICREG

CELLATOM
CA_MARKOV
GEOMOD

VALIDATE
ROC

Change and time series analysis is an important application area for GIS and Image Processing. There is an ongoing need to identify and quantify change, as well as to predict the effects of change on the environment, at scales ranging from local to global. For more information about this application area, see the chapter on **Change and Time Series Analysis**.

The simplest type of change analysis is a comparison between images from two dates.

IMAGEDIFF compares two quantitative images of the same variable for different dates. Output may be a simple difference image, a percentage change ((2nd image - 1st image) / 1st image), a standardized difference image showing z values or a standardized class image in which z values have been reclassified into 6 classes. A mask file may be specified for each image to limit the area under analysis.

IMAGERATIO compares two quantitative images of the same variable for different dates through ratioing. Options are offered to create either a ratio image (later image / earlier image) or a log ratio image ($\ln(\text{later image} / \text{earlier image})$).

CVA (Change Vector Analysis) provides a means for comparing two-band sets of images for two dates. Both the magnitude and direction (character) of change are derived.

CALIBRATE may be used to address problems of non-comparability due to sensor changes between two images from different dates. Three types of information may be used to calibrate the input image: regression with a reference image, user-defined offset and gain, and user-defined mean and standard deviation. Regression with a reference image is the

most common, with the earlier image as the input file and the later image as the reference file. The module computes a linear regression equation relating the two images. This equation is then used with the earlier image to produce a predicted later image. This predicted later image is really an adjusted earlier image, accounting for systematic sensor differences. It can then be compared with the actual later image through quantitative change analysis techniques.

CROSSTAB may be used to compare two qualitative images, such as land cover images, from two dates. A new image may be produced in which a unique identifier is assigned to every combination of original values. RECLASS or ASSIGN might then be used to create an image with change and no change categories. A cross-classification matrix and statistics on the similarity of the two images may also be generated. In cases where the categories of both images are the same, per-category Kappa Index of Agreement figures will be calculated. This provides important information about the types of change occurring in the image. The overall Kappa, as well as the Chi Square and Cramer's V, give information about the degree of change the entire image has experienced.

To analyze change over multiple dates, the following four modules may be used.

PROFILE, when used in the context of change and time series analysis, is used to extract statistics for specific areas of an image over multiple dates. For example, one might create a polygon of a forested area, then extract the average NDVI (a measure of the amount of biomass present) over several dates. The trend of these data would give some indication of the type of change occurring in the area. PROFILE will plot the data as a graph, and can also write the values to a file for subsequent analysis.

TSA stands for Time Series Analysis, and provides standardized Principal Components Analysis for time series data. The input to TSA is in the form of an IDRISI time series file, which simply lists, in order, the images to be analyzed. TSA will accept up to 256 input images and can produce up to an equal number of component images. Loadings graphs are output either as values files or as DIF-format data files that can be used with virtually all spreadsheet software systems. Standardized Principal Components Analysis has been shown to be quite effective in extracting elements of change from time series data. The first component produced is the typical condition over the entire time series, then subsequent images represent different change events. Graphs of the loadings yield information about the temporal patterns of change events.

Input a mask image to limit the area to be analyzed.

CORRELATE calculates the Pearson Product Moment Coefficient of Correlation between a set of values in an attribute values file and the values through a time series of images for each pixel of an image. The main use of the module is to identify areas that correlate well with a particular temporal pattern of interest. The pattern of interest is defined by the values in the attribute values file. In the resulting image, high values show areas that better match the pattern of the values file.

Media Viewer may be used to create video clips of the images in a time series. This is a powerful device for visual change detection.

The following six modules are used in modeling future change.

MARKOV analyzes two qualitative land cover images from different dates and produces a transition matrix, a transition areas matrix, and a set of conditional probability images. The transition matrix records the probability that each land cover category will change to every other category while the transition areas matrix records the number of pixels that are expected to change from each land cover type to each other land cover type over the specified number of time units. The conditional probability images report the probability that each land cover type would be found at each pixel after the specified number of time units and can be used as prior probability images in Maximum Likelihood Classification of remotely sensed imagery.

STCHOICE (Stochastic Choice) creates a stochastic land cover map by evaluating the conditional probabilities that each land cover can exist at each pixel location against a rectilinear random distribution of probabilities. It does so by generating a random number. The conditional probabilities for each class are then summed beginning with the first image listed in the group file. The class represented by the image that makes the sum exceed the random threshold is assigned. The resulting image will appear to be quite pock-marked. Because the random element is introduced, a different output image may result each time the module is run.

DISAGGREGATE generates a sequence of calls to other IDRISI modules in order to redistribute the conditional probabilities of a particular land cover type according to a designated pattern. The user provides a model for the disaggregation. Note that the output image is designed to have the same sum of probabilities as the original image (on a per-category basis).

NORMALIZE linearly adjusts the values for a set of quantitative images so the values sum to 1.0 at each pixel. For example, a set of images showing monthly rainfall for a year might be normalized so the resulting images show the percent of annual rainfall accumulated each month.

CELLATOM provides for cellular automata in IDRISI. This is typically used in dynamic modeling where the future state of a pixel depends upon its current state and the states of its neighbors. The rules for changing states are governed by a filter file and a reclass file that may be modified. The user specifies the number of iterations to perform. With each iteration, the filter is applied to the image, then the resulting image is reclassified according to the reclass file. This output image is then used as input for the next iteration.

CA_MARKOV is a combined cellular automata / Markov change land cover prediction procedure that adds an element of spatial contiguity as well as knowledge of the likely spatial distribution of transitions to Markov change analysis.

GEOMOD is a landuse change simulation model that predicts the locations of grid cells that change over time. The simulation can occur either forward or backward in time.

VALIDATE provides several statistics for measuring the similarity between two qualitative images. These include specialized Kappa measures that discriminate between errors of quantity and errors of location.

ROC, the Relative Operating Characteristic, provides a measure of the correspondence between a quantitative modeled image showing the likelihood that a particular class exists and a Boolean image of that class as it actually occurs (i.e., “reality”).

The Surface Analysis Submenu

Interpolation >
Geostatistics >
Topographic Variables >
Feature Extraction >

The Surface Analysis submenu contains four headings, each leading to further submenus. While the descriptions of the modules in these submenus often refer to elevation data and digital elevation models as examples, the modules available in the Surface Analysis submenu provide a powerful set of analytical techniques that can be applied to any continuous quantitative data.

Interpolation Submenu

INTERPOL
INTERCON
TIN Interpolation >
Kriging
THIESSEN
TREND

The first of the Surface Analysis submenus is Interpolation. The issues surrounding surface interpolation as well as the options available in IDRISI are discussed in detail in the chapter **Surface Interpolation**.

INTERPOL interpolates a surface, according to either a distance-weighted average or a potential model, given a vector input file of points, and the values of the variable at those points. With both models, the exponent associated with the distance weight is user-defined.

For example, if the user specifies 2, the interpolation process uses $1/d^2$. A variable search radius is used that aims to use the 6 closest known points from the vector point in undertaking the interpolation of any grid cell, and ensures that no fewer than 4 and no more than 8 are ever used. The search radius can be disabled, in which case all control points are used in the interpolation of each point.

INTERCON interpolates a surface from a set of digitized lines. This module was designed specifically to facilitate the creation of Digital Elevation Models from contour lines. One starts with digitized contours such as might be created with CartaLinx or purchased from a data vendor. Then the vector contours are rasterized (with the use of INITIAL and RASTERVECTOR). INTERCON is then run on the rasterized line file to produce the interpolated model. Finally, it is common to run FILTER to smooth the surface (using a mean filter) since the linear interpolation procedure tends to create a slightly faceted surface.

TIN Interpolation is discussed in detail in the chapter **Triangulated Networks and Surface Generation**. The modules used to prepare data for TIN generation, to create and optimize the TIN, and to interpolate a full surface from a TIN are all found in this submenu.

TIN creates a triangulated irregular network from point data. The input data may be true point data or isoline data. In the latter case, the vertices of the lines are used in the triangulation. TIN offers the options to constrain the TIN to isolines and to optimize the TIN for features known as bridge and tunnel edges. **TINSURF** uses the TIN and the original point attribute data to interpolate a full raster surface model. Selecting the create raster surface option on the TIN dialog will automatically launch the TINSURF dialog. The other two modules in this submenu, **GENERALIZATION** and **TINPREP** are used to prepare the input vector data for triangulation. **GENERALIZATION** creates a point vector file from the vertices of an input line file. This is useful in visually assessing the density of the points in the line file. **TINPREP** adds or removes points along an isoline given a user-specified tolerance distance. **GENERALIZATION** also thins vector point data according to a user-defined radial search distance.

The **Kriging** submenu leads to three interfaces to the Gstat geostatistical modeling software package.⁴⁵ The chapter **Geostatistics** gives background on the field of geostatistics and the functions provided through these interfaces. In the **Spatial Dependence Modeler** interface, the user employs a wide range of tools to learn about the patterns of spatial dependence in the sample data set. In the **Model Fitting** interface, the user defines mathematical models to describe the covariance relationships among sample data. In the **Kriging and Simulation** interface, full raster surfaces may be created

45. Idrisi32 provides a graphical user interface to Gstat, a program for geostatistical modeling, prediction and simulation written by Edzer J. Pebesma (Department of Physical Geography, Utrecht University). Gstat is freely available under the GNU General Public License from <http://www.geog.uu.nl/gstat/>. Clark Labs' modifications of the Gstat code are available from the downloads section of the Clark Labs website at <http://www.clarklabs.org>.

from sample data and the models developed through the other interfaces. These interfaces are presented here as tools for surface interpolation. The use of these geostatistical techniques is broader than interpolation and they are therefore also available through the Geostatistics submenu.

THIESSEN creates Thiessen polygons around a set of irregularly distributed points, such that every pixel in the resulting image is assigned the identifier of the point to which it is nearest. THIESSEN does not produce a smooth surface as the other modules in this submenu do, but rather a stepped surface.

TREND calculates linear, quadratic, and cubic trend surfaces from an irregular set of points, and produces an interpolated surface as a result. TREND may be used as an interpolator in cases where it is known that some error exists in the control points and the degree of variation in the surface is small. TREND also appears in the Analysis/Statistics submenu.

Geostatistics Submenu

Spatial Dependence Modeler
Model Fitting
Kriging and Simulation

The second submenu in the Surface Analysis submenu is Geostatistics. The field of geostatistics has a broad range of applications to many types of data and analyses. The chapter on **Geostatistics** gives a broad overview of Geostatistics and the functions available in IDRISI through the three interfaces to Gstat (see footnote on previous page).

In the **Spatial Dependence Modeler** interface, the user employs a wide range of tools to learn about the patterns of spatial dependence in the sample data set. In the **Model Fitting** interface, the user defines mathematical models to describe the covariance relationships among sample data. In the **Kriging and Conditional Simulation** interface, full raster surfaces may be created from sample data and the models developed through the other interfaces.

Topographic Variables Submenu

SLOPE
ASPECT
HILLSHADE
CURVATURE
FRACTAL

The Topographic Variables submenu is the third submenu of the Surface Analysis group and contains modules that operate on surface images to calculate a variety of measures. The **SLOPE**, **ASPECT** and **HILLSHADE** entries all open the SURFACE module interface. SURFACE calculates these variables by comparing the heights of cells to those of their neighbors (thus, it is also found in the Context Operators Submenu). From this information, the gradient of the slope can be determined for the middle cell, as well as the aspect of the slope (the direction of maximum descent). The latter is expressed as an azimuth (an angle measured clockwise from north). SURFACE is also able to use this information to create a shaded relief image through a technique called "analytical hillshading."

The **CURVATURE** module calculates the maximum rate of change of a curve fit through a pixel in both the direction of aspect and also in the direction orthogonal to aspect. The values of the pixel and its immediate neighbors are used in the calculation.

FRACTAL calculates the fractal dimension of a surface using a 3 by 3 neighborhood. This calculation is also available through the TEXTURE module.

Feature Extraction Submenu

CONTOUR
TOPOSHAPE
PIT REMOVAL
RUNOFF
FLOW
RUSLE
WATERSHED
SLOPELENGTH
SEGMENT

The final submenu in the Surface Analysis group, Feature Extraction, includes two modules.

CONTOUR creates vector isolines (i.e., lines of constant value) from a continuous surface raster image. The user specifies the contour interval to use. Options are available for cartographic generalization of the output vector file. The resulting vector file is most typically used to enhance visualization of continuous surfaces or for export to other software packages.

TOPOSHAPE classifies a surface into eleven different features: peak, ridge, saddle, flat, ravine, pit, convex hillside, saddle hillside, slope hillside, concave hillside, and inflection hillside. Any pixels not assigned to these classes are assigned to the “unclassified” class.

PIT REMOVAL creates an adjusted "depressionless" DEM in which the cells contained in depressions are raised to the lowest elevation value on the rim of the depression. This is an important preparation step in many flow models.

RUNOFF calculates the accumulation of rainfall units per pixel as if one unit of rainfall was dropped on every location. Using the RECLASS module, a threshold can be applied to the output to produce drainage networks.

FLOW calculates the flow direction from each pixel into its next “downhill” neighbor. Unlike ASPECT, which gives a continuous output of directions, the output of FLOW is restricted to the eight neighboring cells for each pixel. This output is useful in many flow and hydrological applications.

RUSLE evaluates farmland and rangeland nonchannelized soil loss by water. It is an implementation of the Revised Universal Soil Loss Equation (RUSLE) described in the USDA Agricultural Handbook, Number 703.

WATERSHED identifies areas that contribute flow to a part of the drainage network. The user either indicates the minimum size of watershed desired and the module breaks the entire image into watersheds of this size or larger or the user indicates one or more “seed” features for which to identify watersheds.

SLOPELENGTH calculates the longest slope length in a given raster region. Using the surface image, SLOPELENGTH calculates each pixel’s slope length within a region and the longest slope length is assigned to that region.

SEGMENT produces an image of homogeneity. The procedure is based on a standard deviation filter to produce categories of similarity.

The Modeling Menu

Macro Modeler
Image Calculator

Run Macro
Edit

COM / API Documentation

RUSLE
CA_MARKOV
GEOMOD

The items found under the Modeling menu unleash the power of doing raster analysis in IDRISI. Repetitive tasks can be simplified with routines such as **Run Macro**, **Macro Modeler**, or **Image Calculator**. Image analysis and simulation can be accomplished with **Macro Modeler**, **GEOMOD**, **RUSLE**, or **CA_MARKOV**. As a fully COM compliant technology, all IDRISI analytical routines can be accessed through the Windows API and the IDRISI API, to integrate new routines or simply to access IDRISI functionality.

Macro Modeler is a graphical environment for building and executing multi-step models. Facilities are included for batch processing (running many inputs through the same model to produce many outputs) and for dynamic modeling (using the output of one iteration of a model as an input into the next iteration).

Image Calculator is an interactive mathematical modeling tool that allows you to enter a model as a full algebraic equation using a calculator-like interface. It incorporates the functionality found in the modules OVERLAY, SCALAR, TRANSFORM, and RECLASS with considerably less work and without the use of macros. Image Calculator allows for

two types of expressions, Mathematical Expressions and Logical Queries, the latter of which is used in database query.

Run Macro allows you to run an IDRISI macro. Macros contain all the information needed to run many IDRISI modules in a specific order. See the chapter on **Creating Macros** for detailed information about the construction and execution of IDRISI macros.

Edit is a very basic text editor that may be used to create or alter any ASCII file. Its primary function in data entry is in the creation of attribute values files for use with the ASSIGN module. Edit may also be used to create a variety of other IDRISI files, such as correspondence files and group files.

COM/API Documentation links to a PDF documenting the IDRISI API. If you want to script macros, develop and integrate new modules, or simply manipulate IDRISI from another application, you will want to access information in the IDRISI API. IDRISI has been designed as an OLE Automation Server using COM Object technology (i.e., a COM Server). As a consequence, it is possible to use high-level development languages, such as Delphi, Visual C++, Visual Basic, or Visual Basic for Applications (VBA) as macro languages for controlling the operation of IDRISI. In addition, you can create sophisticated OLE Automation Controller applications that have complete control over the operation of IDRISI.

RUSLE simulates farmland and rangeland nonchannelized soil loss by water. It is an implementation of the Revised Universal Soil Loss Equation (RUSLE) described in the USDA Agricultural Handbook, Number 703.

CA_MARKOV is a combined cellular automata / Markov change land cover prediction procedure that adds an element of spatial contiguity as well as knowledge of the likely spatial distribution of transitions to Markov change analysis.

GEOMOD is a landuse change simulation model that predicts the locations of grid cells that change over time. The simulation can occur either forward or backward in time.

The Image Processing Menu

Restoration >
Enhancement >
Transformation >
Fourier Analysis >
Signature Development >
Hard Classifiers >
Soft Classifiers / Mixture Analysis >
Hyperspectral Image Analysis >
Accuracy Assessment >

Alongside the geographic analytical operators found in IDRISI, the Image Processing capabilities round out a full suite of tools for the processing of spatial data. The Image Processing functions fall into ten categories: restoration, enhancement, transformation, Fourier analysis, signature development, hard classifiers, soft classifiers, hardeners, hyperspectral analysis and accuracy assessment.

For background information, consult the **Introduction to Remote Sensing and Image Processing** chapter. Also, several chapters provide in-depth discussions of particular image processing tasks. The

Image Restoration chapter discusses issues of geometric and radiometric correction and the IDRISI modules designed for these purposes. The **Classification of Remotely Sensed Imagery** chapter provides more detailed information on the classification process and a tour of the IDRISI classification operators. The **Fourier Analysis** chapter gives detailed information about the use and function of FOURIER and its companion modules in the Fourier Analysis submenu. Finally, the **Tutorial** includes an extensive set of exercises covering many aspects of image processing.

Restoration Submenu

RESAMPLE
LOCALAFFINE
MOSAIC
DESTRIPE
RADIANCE
ATMOSC
NDVICOMP
SCREEN

Image Restoration is the manipulation of remotely sensed images in an attempt to remove known value distortions. Restoration can be geometric or radiometric. The first module performs geometric corrections which are used to reduce distortion at the edges of the image and to register the image to a coordinate system. The other three modules perform radiometric restoration for the removal or diminishment of distortions in the data values of the images.

RESAMPLE performs geometric restoration of images, especially in the case of remotely sensed imagery which, in its raw state, is without real world coordinates. RESAMPLE is used to georegister an image or vector file to a reference system or to another file. It takes the coordinates of a set of control points in an existing file, and in the desired new reference system, and converts the file to the new reference system by means of either a linear, quadratic or cubic polynomial mapping function. (In the case of satellite imagery, a simple linear resampling is sufficient in most instances.) With raster images, cells in the new grid rarely match up in any simple way with the original grid. Therefore, new cell values are estimated by resampling the old grid (hence the name of the module). When an image or vector file is already georeferenced to a known reference system but must be transformed to another reference system, the PROJECT module in the Reformat menu is more appropriate.

LOCALAFFINE is used to rectify images that have embedded in them a grid of control points with precise known locations.

MOSAIC automates color balancing when adjacent overlapping images are joined into a single larger image. The areas of overlap are used to determine the color adjustments.

DESTRIPE removes the striping caused by variable detector output in scanned imagery. It works by calculating a mean and standard deviation for the entire image and for each detector separately. The output from each detector is scaled to match the mean and standard deviation of the entire image.

RADIANCE converts raw Landsat data values to calibrated radiance using lookup tables of gain and offset values. Conversion to radiance is used to facilitate comparisons between images from different dates.

ATMOSC performs the calculations necessary to correct remotely sensed images for atmospheric effects. Four models are provided: a Dark Object Subtraction (DOS) model, Chavez's Cos(t) model, a full radiative transfer equation model (FULL), and an Apparent Reflectance Model (ARM).

NDVICOMP is intended for the production of temporal composite images of NDVI imagery, most typically for purposes of cloud removal. Two formats are provided. The first produces maximum value composites in which the final output value of each cell represents the maximum of the values in a set of input images. The other produces output cells equal to the quadratic mean of the input cells.

SCREEN is used to screen a hyperspectral series of images for the presence of significant atmospheric noise (i.e., scattering). It achieves this by calculating the first lag spatial autocorrelation coefficient of each image.

Simple haze removal can be accomplished with the SCALAR module, which is found in the Analysis/Mathematical Operators submenu. The linear with saturation option in STRETCH may often be used to produce the same result.

When DESTRIPE is not applicable or does not perform well, Principal Components Analysis (PCA) or Fourier Analysis may provide solutions for destriping satellite imagery. See the chapter on **Image Restoration** for details.

Enhancement Submenu

STRETCH
COMPOSITE
FILTER

Image enhancement is the modification of image values to highlight information within the image. Most often these enhanced images are used in visual analysis only, while the original images are used for automated analyses. The IDRISI display system includes some facilities for enhancement of the screen display. These include the ability to interactively set the endpoints used in applying the color

palette to images. No new files are created with the Display tools, however. To create new images that are enhanced, the following three modules are often used.

STRETCH is a contrast stretch utility used to visually enhance images. Options include linear stretch, linear stretch with saturation, and histogram equalization. A linear stretch takes the maximum and minimum values for the image as a whole and assigns the intervening values to a user specified number of classes. A linear stretch with saturation uses a user-specified percent saturation, and forces that percent of the distribution at the upper end to have the maximum value, and that percent at the lower end to have the minimum value. The intervening values are assigned to the user-specified number of classes. A histogram equalization stretch divides the histogram into classes containing an equal number of pixels (or as close to that as possible), given the condition that pixels with the same original value may not be broken into different new classes.

COMPOSITE produces a color composite image from three bands of byte binary imagery. To create a composite image, you must have three bands of imagery that are registered to each other. Options are available to create 24-bit composites (primarily to visualization) or 8-bit composites (primarily as input to the CLUSTER unsupervised image classification routine). A 24-bit composite is displayed in true 24-bit color and Composer provides independent on-the-fly adjustment of the three input bands for such images. An 8-bit composite image should be displayed with the color composite palette.

FILTER changes the values of all pixels in an image, based on each pixel's original value and those of its neighbors. The nature of this operation is determined by the values stored in a variable sized template, or kernel. The pixel and its neighbors are multiplied by the values stored in the corresponding positions of the template, and the resulting values are summed to arrive at a new value for the center pixel. The following filters are available in IDRISI: mean, Gaussian, median, standard deviation, adaptive box, mode, Laplacian edge enhancement, high pass, Sobel edge detection, and user-defined.

The simplest and the most commonly used filter in Image Processing is a mean filter. In operation, the new value is the average of the original value and those of its neighbors. The result is an image that is "smoother" than the original. Mean and Gaussian filters are often used to generalize or "smooth" an image and are effective for random noise removal. The mode filter assigns the most common value to the center pixel of the kernel. A median filter is most commonly used for random noise removal. The adaptive box filter is good for grainy random noise and also for data where pixel brightness is related to the image scene but with an additive or multiplicative noise factor. Edge enhancement filters, such as the Laplacian, accentuate areas of change in continuous surfaces. High pass filters emphasize areas of abrupt change relative to those of gradual change. The Sobel edge detector extracts edges between features or areas of abrupt change. Finally, the user-defined filter option allows the user to specify any kernel size as well as the mathematical operation, and is useful for simulation modeling.

Transformation Submenu

PCA
CCA
TSA
COLSPACE
TEXTURE
THERMAL
VEGINDEX
TASSCAP

PCA provides both standardized and unstandardized principal components analysis. Principal components analysis is a mathematical technique closely related to factor analysis that transforms an original set of image bands, typically from Landsat or SPOT, into a new set of *components* that are uncorrelated and are ordered in terms of the amount of the original variance explained. Often the first two or three components explain virtually all of the variance because of substantial correlation between the bands. As a result, unstandardized PCA is commonly used to determine which bands are carrying the most information in a data set and for data compaction. PCA can also be used to remove image correlation. Standardized PCA has been shown to be a powerful tool for time series analysis, and is offered for that purpose in the module **TSA**. Using TSA is the same as the standardized option in PCA, except that PCA is limited to 12 input images while TSA can work with up to 84 input images.

CCA performs a canonical transformation on images specified from input signatures. Similar to principal component analysis, it can be used as a feature reduction process.

COLSPACE performs Hue/Lightness/Saturation (HLS) to Red/Green/Blue (RGB) and vice versa color space trans-

formations. This is especially useful for merging high resolution panchromatic imagery with lower resolution multispectral imagery. Three bands of the original multispectral imagery are input as RGB and output as HLS. The Lightness band is then discarded and the (often enhanced) panchromatic imagery is substituted. The three bands are registered to each other, then transformed from HLS to RGB. The resulting three images may then be combined in a color composite image. The resulting image retains much of its spectral resolution and has a higher spatial resolution than the original multi-spectral bands. COLSPACE may also be used to integrate a hillshaded model into a composite image following the same procedure as outlined above, substituting a hillshade model rather than a panchromatic image.

TEXTURE performs texture analysis of an image and is useful for the processing of remotely sensed imagery. The TEXTURE analysis provided includes variability, fractal dimension, and convolution filtering.

THERMAL converts Landsat TM Band 6 raw data values to blackbody temperatures, either Celsius, Kelvin or Fahrenheit.

VEGINDEX computes a large variety of vegetation indices from remotely sensed images. See the Chapter on **Vegetation Indices**.

TASSCAP performs the Tasseled Cap transformation. This transformation yields a variety of useful indices for assessing vegetation biomass and moisture conditions.

Fourier Analysis Submenu

FOURIER
ZEROPAD
FILTERFQ
FREQDIST
DRAWFILT

The modules in the Fourier Analysis submenu support the application of Fourier Analysis, a transformation between spatial and frequency domains. The chapter **Fourier Analysis** provides detailed information on the use of FOURIER and its companion modules as well as the interpretation of results.

FOURIER allows for the transformation of images from the spatial domain to the frequency domain and back again.

FOURIER requires that the number of rows and columns in an image be a power of two. **ZEROPAD** is used to prepare images that do not meet these conditions so they may be used with FOURIER.

FILTERFQ, **FREQDIST** and **DRAWFILT** all facilitate the creation of filters to be applied to frequency domain images to enhance, suppress or remove particular frequencies prior to performing a reverse Fourier Transform. FILTERFQ offers 26 types of filters, each with several user-defined options. FREQDIST creates a frequency distance image that may then be manipulated with RECLASS or FUZZY. DRAWFILT provides an interactive display utility in which the user may trace with the cursor particular frequencies to be masked out.

Signature Development Submenu

MAKESIG
Endsig
FUZSIG
PURIFY

HYPERSIG
HYPERRAUTOSIG

SIGCOMP
SEPSIG
SCATTER

Signature Development is most often associated with the first stages of supervised classification and typically requires two steps—the creation of training sites and the creation of signature files from the training sites. Training sites are examples of informational classes, e.g., forests, urban or rangeland, which can be characterized across all bands of imagery. These characterizations are then used to create signatures or spectral response patterns for each informational class. Delineation of training sites is often accomplished through on-screen digitizing in the IDRISI Display System or through the import of GPS data collected in the field. The second step, signature development, is accomplished with the use of the modules in this submenu.

MAKESIG creates signature files for each informational class for which you have created training sites. It does so by extracting and analyzing the pixels identified in the training sites for each informational class for each band being used. Statistics such as the mean, minimum, maximum, variance, and covariance between the pixels in the separate bands are developed.

Endsig is used to create end-member (i.e., pure) signatures for use with UNMIX.

FUZSIG is a variant of MAKESIG. Unlike MAKESIG which assumes the signature data are from areas that are pure examples of any particular class, FUZSIG produces signatures from data that are assumed to be inherently fuzzy or ambiguous in character. This is often used in the case of mixed pixel analysis.

PURIFY performs a parametric or a nonparametric purification on existing training site data. The parametric purification process uses the Mahalanobis distance to the mean of existing classes to purify. The nonparametric process uses an unsupervised clustering technique to calculate areal proportions of new classes to purify. The resulting purification produces a new training site file and a new set of signature files.

HYPERSIG extends the logic of signature development to the special case of hyperspectral data. HYPERSIG creates and displays hyperspectral signatures either from training site data or from spectral curve library files.

HYPERAUTOSIG automatically develops signatures for hyperspectral image data based on the Linear Spectral Unmixing logic. Given a raster group file of hyperspectral images, a set of hyperspectral signature files (.hsg) and a signature group file (.sgf) are produced. The number of signatures to develop is set by the user and must be less than or equal to the number of bands in the hyperspectral image set.

SIGCOMP graphically displays the signatures created with MAKESIG for comparison purposes. You may choose which signatures to display and whether to display the mean, minimum and maximum, or all three. Overlapping signatures may indicate heterogeneous training sites. With a parallelepiped classification, at least one of the bands should show completely separated minimum/maximum ranges. If not, ambiguity in the classification exists. Likewise, signatures that are nearly coincident in their means will be difficult to separate with the Minimum Distance to Means and Maximum Likelihood classifiers.

SEPSIG provides statistical measures of the separability of signatures over a given set of bands. It is useful in determining which bands should be used to best differentiate particular cover types.

SCATTER is used as a further means to compare the relationship of signatures in relation to their bands. SCATTER creates a scattergram of the band space between two images. The signatures can be plotted in the band space to further evaluate signature spacing.

Hard Classifiers Submenu

PIPED
MINDIST
MAXLIKE
FISHER (LDA)

CLUSTER
Isoclust
MAXSET

NeuralNet

There are two basic approaches to the classification process: supervised and unsupervised classification. IDRISI provides the tools for both processes with both hard and soft classifiers available. This submenu lists the hard classifiers available for both supervised and unsupervised classification.

PIPED is a Parallelepiped classifier. It is generally considered to be the fastest but least accurate of the classifiers available in the IDRISI set. For parallelepiped classification, pixels are assigned to a given informational class if either the reflectance values on each band fall within the minimum and maximum values recorded for that signature or the values fall within a specified number of standard deviations. This procedure provides no assurance that a pixel will not fall into several classes, which often happens. The class assigned to a pixel is the last signature encountered that meets these criteria. To make use of this, specify your signatures in reverse order of importance or likelihood, starting with the least important or likely ones, and ending with those that are most important or likely.

MINDIST is a Minimum Distance to Means classifier in which pixels are assigned to the class they are *closest* to in band space, a coordinate system defined by orthogonal axes representing each band. It is the second fastest routine, and produces the best results when training sites are of less than perfect quality. The procedure will, by default, classify each pixel with no ambiguity. You can specify a maximum distance, either in raw Dn units (0-255) or in standard deviations, so that pixels beyond that radius are left unclassified. The use of standard deviations allows the system to accommodate differences in spectral variance between informational classes.

MAXLIKE is a Maximum Likelihood classifier. When using MAXLIKE, a full multidimensional probability function is evaluated to determine the likelihood that any pixel belongs to a given class. By default, each pixel is assigned to the most likely class, regardless of how likely or unlikely that maximum may be. This is the most advanced technique, but also the slowest, for image classification in IDRISI. This technique produces the best result when training sites are well sampled, i.e., each site has 50-100 times as many pixels as there are bands in the data set, and the site is also strongly homogeneous. As with MINDIST, you have the opportunity of leaving any pixel unclassified where the probability that it belongs to any particular class falls too low. With this procedure, you have the option to enter prior knowledge of the likelihood with which each class occurs in the image. These prior probabilities can either be entered as values, as images, or as a combination of values and images for each signature. Without *a priori* knowledge, the system assumes all classes are equally likely for the purposes of classification.

FISHER provides image classification based on discriminant analysis.

CLUSTER performs an unsupervised classification and uses a variant of the histogram peak technique to create a new image of like clusters from an 8-bit composite image created with COMPOSITE. You have a choice of broad or fine classifications. The broad classification gives a general picture of spectral classes while the fine classification gives intricate details. The clusters are ranked in terms of how much of the image they describe. The resulting cluster image must then be interpreted by the analyst, matching the clusters to their appropriate land cover categories.

Isoclust is an iterative self-organizing cluster analysis procedure and is a variant on the ISODATA routine. Like CLUSTER, ISOCLUST is an unsupervised classifier. However, in contrast to CLUSTER, input to ISOCLUST may include up to seven image bands. With ISOCLUST, the user determines, or seeds, the number of clusters to be uncovered in the data. Through an iterative process, cells are assigned to their nearest cluster in band space.

MAXSET is a hard classifier that assigns to each pixel the class with the greatest degree of commitment based on a full Dempster-Shafer hierarchy describing all classes and their hierarchical combination. MAXSET is an unsupervised classification in that it can assign a pixel to a class for which no exclusive training data have been supplied. MAXSET can be instrumental in identifying mixed-class pixels.

NeuralNet undertakes the classification of remotely sensed imagery through an artificial neural network classifier using the back propagation (BP). The calculation is based on information from training sites.

Soft Classifiers / Mixture Analysis Submenu

BAYCLASS
MAHALCLASS
BELCLASS
FUZCLASS

UNMIX

HYPERSP
HYPEROISP
HYPERUNMIX
HYPERABSORB

MIXCALC
Belief

HARDEN

Unlike hard classifiers, soft classifiers defer making a definitive judgment about the class membership of any pixel and instead make groups of statements about the degree of membership of any given pixel in each of all possible classes. Furthermore, with soft classifiers, the result is not a single image or classification but a set of images (one per class) that express the degree of membership each pixel possesses in a particular class. These modules are very important in developing robust signatures and evaluating classification techniques, and are also used for sub-pixel classification and mixture analysis. Several modules are available, each employing a different set membership metric to express the degree of membership of any pixel to any class. Modules for use with multispectral and hyperspectral image sets are provided.

BAYCLASS employs Bayesian probability theory to express the degree of membership of a pixel to any class.

MAHALCLASS calculates Mahalanobis distance to produce a new set of signature classes.

BELCLASS employs Dempster-Shafer theory to express the degree of membership of a pixel to any class.

FUZCLASS employs Fuzzy Set theory to express the degree of membership of a pixel to any class.

UNMIX is used to classify remotely-sensed images using Linear Spectral Unmixing (LSU—also called Linear Mixture Modeling). The approach assumes that a pixel value is a combination of the means of the signatures of all the classes present in the pixel. It thus provides information about mixed pixels and is used for sub-pixel classification. This approach is most effective when the cover types being considered are few and are distinct, e.g., for target detection.

HYPERSUP provides unsupervised classification for hyperspectral image data. The module operates using a combined process of signature development using the same logic as HYPERAUTOSIG and linear spectral unmixing using the same logic as HYPERUNMIX.

HYPEROSP provides for hyperspectral image classification through an orthogonal subspace projection approach.

HYPERUNMIX extends the capabilities of UNMIX to hyperspectral data sets.

HYPERABSORB provides for hyperspectral image classification based on library spectra and continuum removal of absorption areas and the correlation of these areas in terms of fit and depth between the library spectrum and the spectra from an imaging data set.

MIXCALC extends the functionality of BELCLASS in the statement of degree of membership that each pixel exhibits for each of the classes for which training data has been provided. Using the logic of Dempster-Shafer theory, a whole hierarchy of classes can then be recognized, made up of the indistinguishable combinations of these classes. MIXCALC can be used to deconstruct or calculate the possible mixture of any one of these non-unique classes.

Belief performs a Dempster-Shafer Weight-of-Evidence classification and extends the logic of mixture analysis, allowing for the ability to combine new evidence with existing knowledge. The outputs of both BELCLASS and MIXCALC can be used as inputs into Belief.

HARDEN produces hard decision images from the soft classifier outputs of BAYCLASS, UNMIX, FUZCLASS, BELCLASS, or MAHALCLASS by choosing the class that has the maximum value.

Hyperspectral Image Analysis Submenu

HYPERSIG
HYPERAUTOSIG
SCREEN

HYPERSAM
HYPERMINT

HYPERSUP
HYPEROSP
HYPERUNMIX
HYPERABSORB

HYPERSIG extends the logic of signature development to the special case of hyperspectral data. HYPERSIG creates and displays hyperspectral signatures either from training site data or from spectral curve library files.

HYPERAUTOSIG automatically develops signatures for hyperspectral image data based on the Linear Spectral Unmixing logic. Given a raster group file of hyperspectral images, a set of hyperspectral signature files (.hsg) and a signature group file (.sgf) is produced. The number of signatures to develop is set by the user and must be less than or equal to the number of bands in the hyperspectral image set.

SCREEN is used to screen a hyperspectral series of images for the presence of significant atmospheric noise (i.e., scattering). It achieves this by calculating the first lag spatial autocorrelation coefficient of each image.

HYPERSAM is a spectral angle mapper hard classifier for hyperspectral data. The spectral angle mapper algorithm is a minimum-angle procedure specifically designed for use with spectral curve library data.

HYPERMINT is a minimum-distance hyperspectral hard classifier specifically intended for use with image-based signatures developed using training sites. It employs the same logic as MINDIST using standardized distances.

HYPERSUP provides unsupervised classification for hyperspectral image data. The module operates using a combined process of signature development using the same logic as HYPERAUTOSIG and linear spectral unmixing using the same logic as HYPERUNMIX.

HYPEROSP provides for hyperspectral image classification through an orthogonal subspace projection approach.

HYPERRUNMIX extends the capabilities of UNMIX

HYPERRABSORB provides for hyperspectral image classification based on library spectra and continuum removal of absorption areas and the correlation of these areas in terms of fit and depth between the library spectrum and the spectra from an imaging data set.

Accuracy Assessment Submenu

SAMPLE ERRMAT

Accuracy assessment is an important final step in both unsupervised and supervised classifications. Its purpose is to quantify the likelihood that what you mapped is what you will find on the ground. This is useful in comparing classification techniques, and determining the level of error that might be contributed by the land cover image in further analyses in which it is incorporated.

SAMPLE generates a random set of locations to be used for ground verification and the assessment of the level of error associated with the classification process result. SAMPLE produces a vector file of points according to a systematic, random or stratified random sampling scheme. It is useful in many other accuracy assessment studies as well as in providing a general technique for sampling over space.

ERRMAT produces an error matrix analysis of categorical map data compared to ground truth information. It tabulates errors of omission and commission, marginal and total errors, and selected confidence intervals. Per-category Kappa Index of Agreement figures are also provided. This information is used to assess the accuracy of the classification procedure undertaken and may suggest ways to improve the classification.

The Reformat Menu

CONVERT

Items in the Reformat Menu allow you to change the data and file type of a file, reorient an image or vector file, change the extent of the study area, change resolution, generalize the level of detail in the file, join files together, and convert files from raster to vector and vice versa.

PROJECT RESAMPLE

CONVERT may be used to change the data type or file type of an image or vector file. Valid data types are byte, integer, and real, while valid file types are ASCII, binary, and packed binary.

WINDOW EXPAND CONTRACT CONCAT TRANSPOSE

PROJECT is used to change the reference system of image or vector files. Supported projections in this version are: Albers Equal Area Conic, Hammer Aitoff, Lambert Azimuthal Equal Area (North and South Polar Aspects, Transverse and Oblique), Lambert Conformal Conic, Mercator, Plate Carree, Stereographic (North and South Polar Aspects, Transverse and Oblique), Transverse Mercator, and Gauss-Kruger. By using reference system parameter files, a limitless number of reference system conversions can be undertaken. PROJECT

takes a file currently in one of the supported systems, back projects to spherical or geodetic coordinates, then undertakes the forward projection to the desired referencing system. For a detailed discussion see the chapter on **Georeferencing**.

RESAMPLE is used to georegister an image or vector file to a reference system or to another file. It takes the coordinates of a set of control points in an existing file, and in the desired new reference system, and converts the file to the new reference system by means of either a linear, quadratic or cubic polynomial mapping function. In the case of raster images, cells in the new grid rarely match up in any simple way with the original grid. Therefore new cell values are estimated by resampling the old grid (hence the name of the module). When an image or vector file is already georeferenced to a known reference system but must be transformed to another reference system, the PROJECT module in the Reformat menu is more appropriate. For a detailed discussion see the chapter on **Georeferencing**.

WINDOW extracts a rectangular sub-area of a larger image to create a new smaller image. The same area, defined by

either row and column positions or reference system coordinates, may be extracted from multiple images in one operation. A similar function which allows the currently-displayed window to be saved to a new image is available through the Save Map Composition dialog box of Composer in the Display System.

EXPAND and **CONTRACT** may be used to alter the resolution of raster images (by increasing or decreasing the frequency of cells) as long as the change is by some integer multiple or division (e.g., by 3 times, but not by 3.2 times—for that you need RESAMPLE). EXPAND works by pixel duplication while CONTRACT offers the options of contraction by pixel thinning or by pixel aggregation. With thinning, pixel values are dropped out, while with aggregation, the resulting value is the mean of the original values. CONTRACT may thus be used to generalize the information in an image.

CONCAT joins together multiple images or multiple vector files into a single image or vector file. Images may overlap when concatenated, and you may choose an opaque process or a transparent process. All files must be in the same reference system. For vector files, the object type must be the same for all files.

TRANSPOSE can be used if a rotation is required. TRANSPOSE can rotate an image by 90 degrees in either direction. It can also reverse the order of rows or columns—a process known as reflection.

RASTERVECTOR converts data between raster and vector formats. Point, line, and polygon data can be converted in either direction.

GENERALIZATION is used to generalize data vector point and line data. It can also generalize raster data by merging smaller regions into neighboring regions based on a given threshold.

The Data Entry Menu

CartaLinx

Edit ASSIGN

INITIAL UPDATE UTMRef

Surface Interpolation

Database Workshop Collection Editor

Data entry is often a very time consuming element when working with GIS. In addition to the data entry modules in this menu, you will find conversion utilities for existing data that are in non-IDRISI formats in the File/Import submenu. The chapter on **Database Development** also discusses issues of data entry.

CartaLinx is a spatial database development tool also developed and distributed by Clark Labs. It provides tablet as well as on-screen digitizing capabilities and a wide range of data editing tools. If CartaLinx is installed, choosing this menu item will launch CartaLinx. If it is not installed, choosing this menu item will display information about CartaLinx.

Edit is a very basic text editor that may be used to create or alter any ASCII file. Its primary function in data entry is in the creation of attribute values files for use with the **ASSIGN** module. Edit may also be used to create a variety of other IDRISI files, such as correspondence files and group files.

ASSIGN is used to change the attributes of a raster image. It does this by using an attribute values file, in which the first column contains the original value and the second column contains the new values to be assigned. The equivalent of **ASSIGN** may also be accessed from Database Workshop.

INITIAL is used to create an image containing a single value. While there are some analytical operations where a single value image might be needed, more commonly it will be used to set up a blank image initialized with zeros for the rasterization modules (of the Reformat Menu) to update. The parameters of the new file, such as rows and columns, reference coordinates and so forth, may be entered interactively, or may be copied from a specified existing image.

UPDATE assigns single values to specific cells or rectangular groups of cells.

UTMRef is used to facilitate the creation of reference system parameter files based on the Universal Transverse Mercator

system, for subsequent use with PROJECT.

The options in the **Surface Interpolation submenu** are identical to those of the Analysis / Surface Analysis / Interpolation submenu and are described in that section above. The issues surrounding surface interpolation as well as the options available in IDRISI are discussed in greater detail in the chapter **Surface Interpolation**.

Database Workshop provides full access to dBase, Access, and FoxPro database files. Edit mode may be activated so the user may update or enter database values directly.

The **Collection Editor** is used to define members of a raster group file collection or a linked-table collection.

Window List

The Window List menu item provides a listing of all open windows. Open dialogs are listed with the module name (e.g., Composer) and open map windows are listed with the filename that appears in the banner of the map display window. Clicking on a window name in the list will bring that window into focus. All open map layers and dialogs can be closed from the Window List menu.

The Help Menu

Contents

Using Help

ESRI Quick Start >

What's New in Kilimanjaro

IDRISI Manual

IDRISI Tutorial

Clark Labs Home Page

IDRISI Technical Support

About IDRISI Kilimanjaro

The Help menu gives you access to the IDRISI on-line Help System.

Contents leads you directly to the IDRISI Help System. Here you will find Contents, Index and Search tabs.

Using Help describes the IDRISI Help System. Here you will find out how to access and navigate through the Help System. Also provided are general information on the Help System screen and functions as well as how a typical program module's entry is organized within the Help System.

What's New in Kilimanjaro gives an overview of the newest functionality since the last version, Idrisi32 Release 2.

The **IDRISI Manual** and **IDRISI Tutorial** lead you directly to the respective pdf files.

The **Clark Labs Home Page** will launch the Clark Labs web site. The **IDRISI Technical Support** will lead you to an on-line form on our web site where you can describe your technical problem and send it as an email to our Technical Support Staff.

About IDRISI Kilimanjaro provides licensing and copyright information as well as general information, including contact addresses and telephone numbers.

ESRI Quick Start Submenu

Help Using ArcView with IDRISI

**SHAPEIDR
ARCRASTER
ARCIDRIS**

**GEOTIFF/TIFF
JPGIDRIS**

RASTERVECTOR

The **ESRI Quick Start** section of the help menu gives immediate access to information about Using ArcGIS/ArcView with IDRISI and to the Help System for modules commonly used to transfer data between the two systems.

IDRISI Modeling Tools

One of the most fundamental roles for GIS is in the development, testing and utilization of models—suitability models, soil erosion models, urban growth models, and the like. IDRISI provides an extensive set of tools for modeling, accommodating a range of levels of expertise. The most fundamental is Macro Modeler, a graphical modeling environment that combines the strengths of extensive capabilities and ease of use. For simpler equation-based modeling using GIS layers, Image Calculator provides rapid equation entry using a familiar calculator interface. A third facility is a macro scripting (.iml) language that is provided largely for legacy applications (this was the original form of modeling tool provided in an early release of IDRISI). Finally, there is the IDRISI API, an industry standard COM interface that provides access to the internals of the IDRISI system for the most demanding applications and interface development. The API requires a COM-compliant programming environment such as Visual C++, Delphi, or Visual Basic.

IDRISI Macro Modeler

The IDRISI Macro Modeler is a graphic environment in which you may assemble and run multi-step analyses. Input files, such as raster images, vector layers, and attribute values files, are linked with IDRISI modules that in turn link to output data files. The result is a graphic model, much like the cartographic models described in the Introductory GIS Exercises of the **Tutorial**. A model may be as simple or complex as desired. Figure 9-1 shows a simple suitability mapping model.

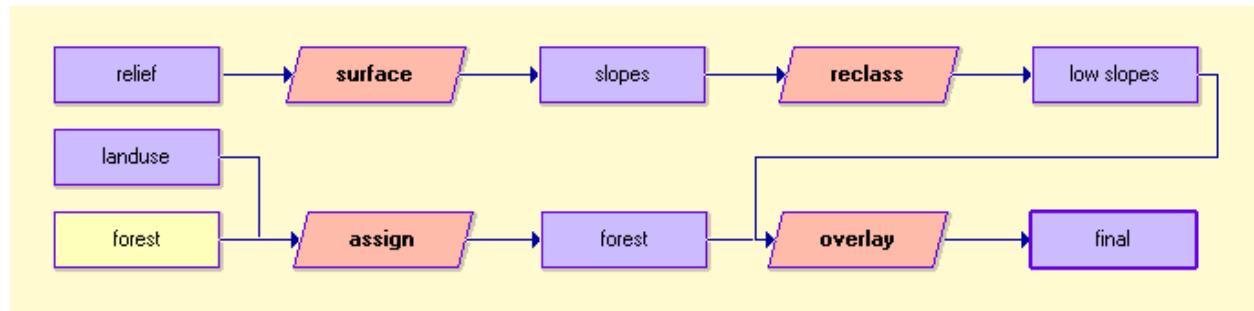


Figure 9-1

Model Construction

To work with Macro Modeler, either click on its icon on the tool bar, or select it from the Modeling Menu. This yields a special workspace in the form of a graphic page along with a separate menu and tool bar. Constructing a model involves placing symbols for data files and modules on the graphic page, then linking these model elements with connectors. To place an element, click on its icon (shown in Figure 9-2) or choose the element type from the Macro Modeler menu, then choose the specific file or operation from the supplied pick list. (Note that not all IDRISI modules are available for use in the Macro Modeler.⁴⁶) To connect elements, click on the connector icon, then click on one of the elements and drag the cursor onto the other element and release. It is necessary to connect elements in the proper order because the Macro Modeler assumes that the process flows *from* the element you first clicked *to* the element upon which you released the cur-

46. The modules of the image processing menu as well as any module that does not create an output file, such as REGRESS, do not work in the Macro Modeler.

sor. To delete any element, first click on that element, then click the delete icon or press the delete key on the keyboard.

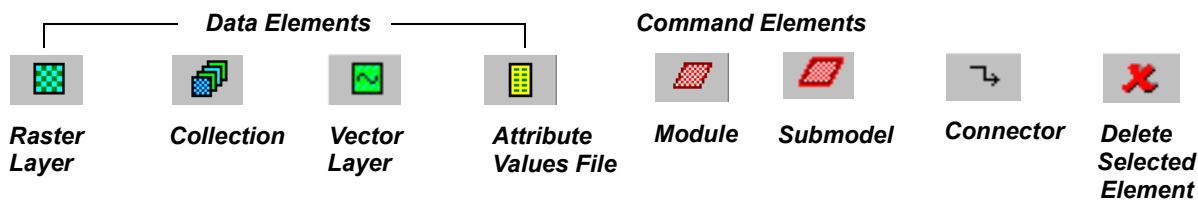


Figure 9-2

Whenever a module is placed in the model, it is placed with the appropriate output data element for that operation already linked. Default temporary filenames are assigned to output files. Right click on a data file symbol to change the filename. Long filenames are left-justified. To see the entire filename, pause the cursor over the element graphic to cause a balloon to appear with the entire element name. Filenames are stored without paths. Input files may exist in any folder of the current project (specified with the Data Paths module) while intermediate and final output files are always written to the Working Folder.

To specify the parameters for the module, right click on the module symbol. This opens the module parameters dialog. Click on any parameter to see and choose other options. Access the Help System for detailed information about the various options by clicking the Help button on the parameters dialog.

Submodels are user-constructed models that are subsequently encapsulated into a single command element. When a model is saved as a submodel, a parameters dialog is created for the submodel in which the user provides captions for all the model inputs and outputs. A submodel consists of a submodel parameter file (.ims) and its macro model (.imm) file. When a submodel is placed as a command element in a model, the Macro Modeler knows how many and what types of input files are required to run the submodel and what types of output files should be produced. For example, you might create a submodel that stretches a single image then exports it to a JPG image. The model would include both STRETCH and JPGIDRIS, but the submodel would only require the input image and the output image. The user may change the input and output data elements of a submodel through the parameters dialog, but settings for module commands that are part of a submodel must be changed by opening the model from which the submodel was created, making the changes, then resaving the submodel. Submodels not only simplify multi-step processes, but are important elements when models have sections that should loop. This is discussed in further detail below.

Models are saved to an IDRISI Macro Model file (.imm). This file preserves all aspects of the model, including the graphic layout. The file may be opened and modified with the Macro Modeler. The graphic description of the model may be copied (as a .bmp file) to the operating system clipboard then pasted into other word processing and graphics software. The graphic may also be printed. The toolbar icons for Macro Modeler file management are shown in Figure 9-3.



Figure 9-3

Models may be run at any stage of completion. The Macro Modeler first checks to see if any of the output images already exist. If so, it shows a message indicating which file exists and asks if you wish to continue. Answering Yes (or Yes to All) will cause the existing output file to be deleted before the model runs. Answering No will prevent the Macro Modeler from running the model. As the model runs, the module element that is currently being processed turns green. To stop a model at any point while it is running, click the stop icon. The model will finish the process it is currently doing and will then stop.

The last output created by the model will be automatically displayed. Intermediate images may be displayed by clicking on the display icon on the Macro Modeler toolbar, then on the data layer symbol. The metadata may be accessed for any data layer by clicking the Metadata icon on the Macro Modeler toolbar, then the data layer symbol. Toolbar icons for these functions are shown in Figure 9-4.



Figure 9-4

DynaGroups

Two distinctive capabilities of the Macro Modeler are its use of DynaGroups to facilitate running the same model on multiple data layers and DynaLinks to construct iterative models in which the output of a process becomes an input for the next iteration of the process.

A DynaGroup is a raster group file or time series file that is used in the Macro Modeler so each member of the group is used to produce an output. Contrast this with the use of a group file as a regular data input to the Macro Modeler in which a group file used as input produces a single output as with the module COUNT. More than one DynaGroup may be used in a process. The module OVERLAY, for example, requires two raster images as input and produces a single raster image output. If DynaGroups are used as the two inputs, then Macro Modeler will first verify that the same number of members is in each group, then it will run OVERLAY using corresponding images from the two group files. If a single raster layer is used as one input and a DynaGroup is used as the other input to OVERLAY, then the modeler will run OVERLAY with the single raster image and each group member in turn. In all cases the number of output images is equal to the number of members in the DynaGroup file. Several options for naming the output files are described in the Help System. In addition, a group file of the output files is automatically constructed. In the model shown in Figure 9-5, two Dynagroups are used as input to an overlay operation. The output will be the number of raster images contained in the input DynaGroups and a raster group file.

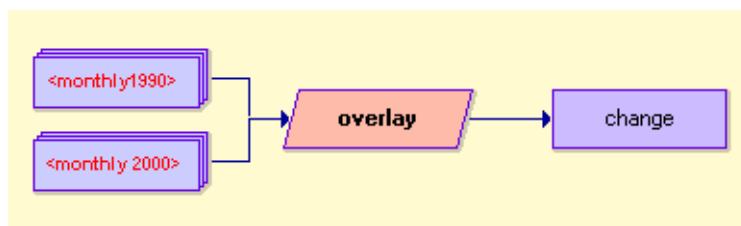


Figure 9-5

DynaLinks

DynaLinks are used to run a model iteratively. An output data element is joined to an initial input data element with a DynaLink. This indicates that when the model performs the second iteration, the linked output filename should be substituted in place of the linked input file name. A dynalink may only be connected to an initial data layer, i.e., one which has no connector joining to its left-hand side. Macro Modeler asks how many iterations are to be performed and whether each terminal output of the model or only the terminal output of the final iteration should be displayed. The final output of a model that includes a DynaLink is a single raster layer. The model shown in figure 9-6 uses a DynaLink. When the model

is run, the Macro Modeler asks for the desired number of iterations. Let us assume that in this case we want 5 iterations. The input file called Band3 is filtered to create an output image that is called filtered x5_01. On the second iteration, the file filtered x5_01 is filtered to produce the file filtered x5_02. This file is then substituted back as the input for the next filter operation and so forth. At the end of the 5th iteration, the final image is named filtered x5. Intermediate images (e.g., filtered x5_03) are not automatically deleted until the model is run again.

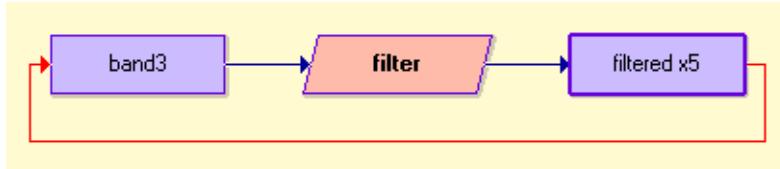


Figure 9-6

Note that the entire model is run during each iteration, even if the substitution occurs near the end of the model. To run just a portion of a model iteratively, use a submodel that contains a DynaLink (see below).

Using DynaGroups, DynaLinks and Submodels Together

When DynaGroups and DynaLinks are used in the same model, the number of members in the DynaGroup defines the number of iterations to be run. Each iteration of the model feeds in a new DynaGroup member. The output of a model that contains both elements is a single data layer. Figure 9-7 shows a simple model in which a group of raster files is summed through the use of a DynaGroup and a DynaLink. In the first iteration, the initial layer BLANK (which is simply an image with value 0 everywhere) is used with the first DynaGroup member in an Overlay Addition operation. This produces the temporary file SUM_01. SUM_01 is then substituted back into the place of BLANK for the second iteration. The number of iterations equals the number of images in the DynaGroup.

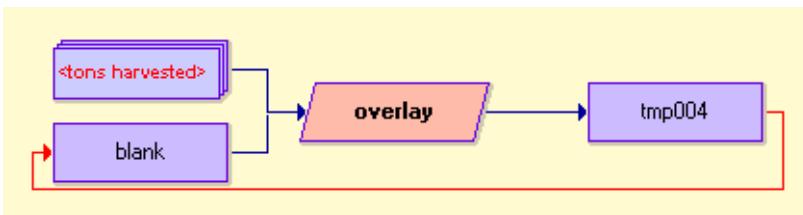


Figure 9-7

To iterate or loop only a portion of the model, create a submodel for the portion that should loop. When you run the model, a dialog box for each submodel will appear asking for the number of iterations for that submodel.

Image Calculator

Image Calculator is a calculator tool for quickly evaluating equations using raster image layers. Equations can be saved and edited at a later time. It is extremely easy to use since it works exactly like a scientific calculator and offers the functionality typically associated with calculators. The only special requirement is that all image files must be specified using square brackets as delimiters. For instance, in the example shown in Figure 9-8, a calibrated image band is created by multiplying its values by a gain of 0.9421 and then adding an offset of 0.0168. It can be accessed through its icon on the tool bar, or from several menu entries: under the Mathematical Operators and Database Query sections of the GIS Analysis menu,

and the Modeling menu.

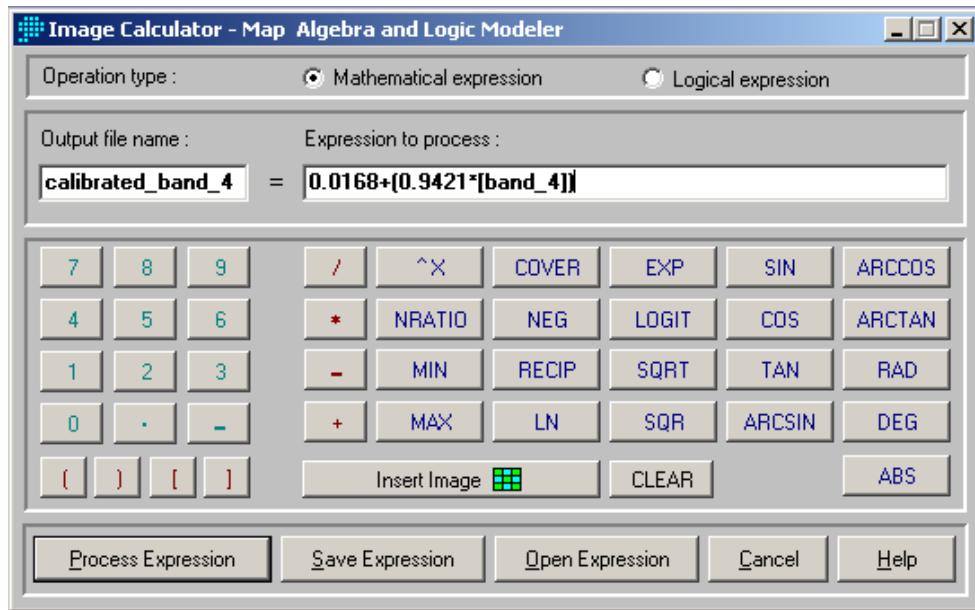


Figure 9-8

Command Line Macros

Macro Modeler and Image Calculator provide very direct and simple interfaces for creating models. However, command line macro scripts are also supported, primarily to support legacy applications from early versions of IDRISI. A command line macro is an ASCII file containing the module names and parameters for the sequence of commands to be performed. It provides no automatic batch capability (though you may quickly copy, paste and edit to achieve this) nor looping.

Specific instructions on the macro command format for each module can be found in the on-line Help System. These instructions can then be placed into an ASCII file with an ".IML" (an acronym for "IDRISI Macro Language") extension. Typically, this will be done with the Edit module. Choosing to save as type Macro file will automatically add the correct file name extension.

Note that some modules do not have a macro command version. These are typically modules that do not produce a resulting file (e.g., IDRISI File Explorer) or modules that require interaction from the user (e.g., Edit). In the menu, any module written in all upper-case letters may be used in a macro.

IDRISI records all the commands you execute in a text file located in the working folder. This file is called a LOG file. The commands are recorded in a similar format to the macro command format. It may sometimes be more efficient to edit a LOG file to have the macro format rather than typing in macro commands from scratch. To do so, open the LOG file in Edit and alter it to have the macro file format. Save it as a macro file.

Note also that whether from a dialog box or a macro, the command line used to generate each output image is recorded in that image's Lineage field in its documentation file. This may be viewed with the Metadata utility and may be copied and pasted into a macro file using the CTL+C keyboard sequence to copy highlighted text in Metadata and the CTL+V keyboard sequence to paste it into the macro file in Edit.

The Run Macro dialog box, under the File menu, will ask for the name of the macro to execute along with any command line parameters that should be passed to the macro file itself. This latter box can be left blank if there are none.

Each line in a macro is completed in sequence because the resulting image of one command is often used as input to a later command. In addition, if more than one macro is launched, the first macro launched will be completed before the second is begun.

Macro File Structure

IDRISI Macro files support the following syntax.

1. IDRISI modules in command line mode. The following is an example of a valid line in an IML file:

```
OVERLAY x 3*soilsuit*slopsuit*suitland
```

The x after the module name indicates that command line mode is invoked. *All parameters after the x are separated by asterisks.* Short file names may be given, as in the example above. In this case, the macro will look for the appropriate file type first in the working folder, then in each resource folder in the order in which they are listed in the current project file. Long file names, as well as full paths, may also be given in the command line, e.g.:

```
OVERLAY x 3*c:\data\soil.rst*c:\data\slopes.rst*c:\output\suitable_land.rst
```

2. The macro processor supports command line variables using the %## format (familiar to DOS users). For example, the above line could be modified as such:

```
OVERLAY x 3%1%2%3
```

In this modification, %1, %2 and %3 are replaced with the first three command line parameters specified in the Macro Parameters text box of the Macro dialog box.

3. External Applications. Any non-IDRISI application can be called from an IML file. The syntax is as follows:

CALL [Application Name] [Command Line Parameters]

where [Application Name] should specify the full path to the application. For example, to call an application named SOILEROD.EXE in a folder named MODELS on drive C, and pass it the name of an input file named SOILTYPE.RST, the command would be:

```
CALL c:\models\soilerod.exe soiltpe.rst
```

4. Other IML files. It is possible to execute other IML files from within a main IML file by using the BRANCH command. The syntax for such an operation is:

BRANCH [IML File Name] [Command Line Parameters]

The IML file name is the short name of the IML file to call (i.e., without extension). Command line parameters are optional.

The IDRISI API

IDRISI has been designed as an OLE Automation Server using COM Object technology (i.e., a COM Server). As a consequence, it is possible to use high-level development languages, such as Delphi, Visual C++, Visual Basic, or Visual Basic for Applications (VBA) as macro languages for controlling the operation of IDRISI. In addition, you can create sophisticated OLE Automation Controller applications that have complete control over the operation of IDRISI. Thus you would use the API in instances where you wish to:

- create complex models that require the extensive control structures of a high level computer language (such as many dynamic models);
- create custom applications, perhaps to be distributed to others;
- create custom interfaces.

The OLE Automation Server feature of IDRISI is automatically registered with Windows when IDRISI is first run on your system. Thus, if you have installed IDRISI and run it at least once, you automatically have access to the full API. The IDRISI OLE Automation server provides a wide range of functions for controlling IDRISI, including running modules, displaying files, and manipulating IDRISI environment variables. With visual programming environments such as Delphi, Visual C++ and Visual Basic, access to these functions is very easy. Specific instructions as well as a complete reference manual can be accessed from the Modeling menu.

While the API provides a wide range of tools, most users only require a few - typically the RunModule operation that can run any module, the Display File operator (for construction of automatically displayed outputs), and a couple of procedures for accessing the project folder structure.

Instructions are also provided in the API help file (under the Modeling menu) for how to change the scripts for the IDRISI menu system so that you can develop applications and incorporate them as direct extensions to IDRISI itself.

Database Development

Introduction

As illustrated in the **Introduction to GIS** chapter, the database is at the center of the Geographic Information System. In fact, it provides fuel for the GIS. The tasks of finding, creating, assembling and integrating these data may collectively be termed *database development*. While it is seldom the most interesting part, database development can easily consume 80 to 90 percent of the time and resources allocated to any project. This chapter discusses many of the issues encountered in database development and also presents a summary of the database development tools included in IDRISI. Also presented are techniques and tips for importing raster data, particularly satellite imagery.

Collecting Data

The first stage in database development is typically the identification of the data layers necessary for the project. While it is tempting to acquire every available data layer that coincides with the study area, it is usually more efficient to identify what is needed prior to assessing what is available. This helps to avoid the problems of data-driven project definitions and encourages a question-driven approach.

In addition to identifying the themes that are necessary (e.g., elevation, landcover), it is also important to determine what resolution (precision) and what level of accuracy are required. These issues will be discussed more thoroughly below.

Once the necessary data requirements for the project have been specified, the search for data can begin. There are five main ways to get data into the database:

1. Find data in digital format and import it.
2. Find data in hard-copy format and digitize it.
3. Collect data yourself in the field, then enter it.
4. Substitute an existing data layer as a surrogate.
5. Derive new data from existing data.

Find Data In Digital Format and Import It

Where To Find Data

In many countries, governmental organizations provide data as a service. In the United States, these data are often free for electronic download or are available for a small fee. The United States Geological Survey, the National Oceanic and Atmospheric Administration and the Census Bureau are good examples of governmental agencies in the United States that have a mandate to create and provide digital data. All these agencies have web sites that provide information about the availability of digital data. In addition, it is becoming more and more common for states (or provinces) to have a particular agency or department that is responsible for creating, maintaining and distributing GIS data layers. In the state of Massachusetts, for example, the state office MASSGIS has this mandate. For many, the web is fast becoming the preferred method for finding and acquiring government-provided data.

Commercial companies also provide digital data. These data might be generic sets of commonly-used base layers, such as transportation networks, or they might be customized to the needs of specific users. Commercial companies often begin

with free government data and "add value" by converting it to a specific software format or updating it.

With the proliferation of GIS and image processing technologies, many non-governmental and academic institutions may also possess data layers that would be useful to your project. While these organizations seldom create data specifically to share with others, they will often make available what they have collected. Therefore, it is usually worthwhile to identify other researchers or institutions working in the same study area to inquire about their data holdings. Electronic discussion forums, such as IDRISI-L, also provide a venue to request particular data layers. The Clark Labs Web Site, www.clarklabs.org, includes a page of links to GIS and image processing related sites, many of which provide data.

Once you have located electronic data useful to your project, you need to import it into the software system you are using. Sound simple? Ideally, these format conversions are trivial. In reality, it is not uncommon to find that getting from what you have to what you want is not straightforward.

Physical Media

Data might come on a CD-ROM, an 8 mm tape, a Zip disk or any number of less common media. When acquiring data, be sure to ask about the medium on which it will arrive and make sure you have the ability to read that medium. The ability to download large data files electronically has lessened this problem substantially, but even with network download, access to the network and appropriate software, such as an FTP client, are necessary.

Data Formats

You must also determine the file format of the data. In general terms, you should know which category it falls within. For example, a data layer of superfund sites could be stored as a raster, vector, database or spreadsheet data file. More specifically, you will need to know if it is in a particular software format, an agency format, or some sort of data interchange format. IDRISI provides for direct import of a number of standard raster, vector and attribute data formats.

If the particular file format you have is not directly supported by IDRISI import routines, then you may still be able to use it. Raster data tends to have a simpler structure than vector data and it is often the case that a raster file can be imported into IDRISI using a low-level tool, such as GENERICRASTER, to re-order the pixels. Vector data are more complex, and unless the structure is extremely simplistic, it is unlikely that you will be able to import vector data using low-level tools.

For both raster and vector data, translation software can provide the bridge between the current data format and IDRISI. Translation software for GIS file formats vary in the number of formats they support and in price. Your goal in using a translation software is to turn the file you have into something that you can import with IDRISI.

You must also consider data compression when acquiring data. Compression allows files to be stored using less memory than they would normally require. There are two types of compression of which to be aware. The first is compression that is used by the GIS software itself. IDRISI, for example, has a run-length encoded packed binary file type. While IDRISI can use these files analytically, other GIS software will not be able to use them. It is best to avoid sharing files that are compressed with software-specific routines between different GIS software systems.

The second type of compression, sometimes referred to as "zipping", is used for files that are not currently being used. This compression is used to make the transfer and storage of files easier. The files must be uncompressed or "unzipped" prior to use.

Always inquire about the compression software used. If you don't have the same software, get the file in an uncompressed format, or as a self-extracting executable file (a compressed file that will uncompress itself without the compression software installed on your computer).

Information about importing specific file formats is available in the IDRISI on-line Help System.

Find Data In Hard Copy Format and Digitize It

If the data you need is in a hard-copy format, you will need to digitize it (i.e., make it digital) to bring it into your GIS database. Some hard-copy data might be digitized by simply typing it into an ASCII editor or a database table. For example, you might have tabular field notes from sample survey sites that can be typed directly into Database Workshop.

More commonly, hard-copy data is in the form of a map, an ortho-photo, or an aerial photograph. If you want to extract particular features from a map, such as elevation contours, well-head locations or park boundaries, then you will digitize these as vector features using a digitizing tablet. (Alternatively, features plainly visible on a digital ortho-photo can be captured with on-screen digitizing, which is discussed below.)

A digitizing tablet (or digitizing board) contains a fine mesh of wires that define a cartesian coordinate system for the board. Most boards have 1000 wires per inch, which results in a maximum resolution of 1/1000 inch. The user attaches the hard copy map to the digitizing board, then traces features on the map with a digitizing puck (a mouse-like device) or stylus (a pen-like device). The digitizing tablet senses the x,y positions of the puck as the features are traced and communicates these to the digitizing software.

Digitizing software packages vary tremendously in ease of use and capability. CartaLinx, also a product of the Clark Labs, combines an easy user interface with flexible digitizing and post-digitizing editing, and is recommended for use with IDRISI.

Most digitizing software will allow the user to "register" the map on the digitizing tablet. This process establishes the relationship between the tablet's cartesian coordinates and the coordinate system of the paper map. The software compares the tablet coordinates and the map coordinates for a set of control points and then derives a best-fit translation function. This function is then applied to all the coordinates sent from the board to the software. If your digitizing software does not provide this translation, the RESAMPLE module in IDRISI may be used to transform the tablet coordinates to map coordinates after digitizing.

In addition to digitizing using a digitizing tablet, scanners can be used to digitize hard copy images such as maps or aerial photos. Unlike digitizing tablets that produce vector data, scanners produce raster data. Also, scanners do not capture distinct features but measure the relative reflectance of light across a document according to a user-specified resolution, normally specified as dots per inch. Each dot becomes a pixel in the resulting image. Sensors detect the reflection of red, green, and blue (or gray level) and record these as digital values for each pixel. The scanned image is then imported to the GIS software as a raster image. Scanned images are normally imported to IDRISI through either the .BMP or .TIF file formats.

Once scanned, features such as roads may be extracted into a vector file format using specialized software. However, this process is often too costly or inaccurate, and it requires very clean hard copy sources for scanning. Another method to extract vector features from scanned raster images is with on-screen digitizing.

With on-screen digitizing, sometimes referred to as "heads-up" digitizing, the scanned source map or photo is displayed on screen and features are digitized using a standard mouse. The RESAMPLE module may be used to georegister the image before digitizing. Both IDRISI and CartaLinx allow you to capture features as vector format files from a displayed raster image through on-screen digitizing.

Collect Data Yourself in the Field then Enter It

For many research projects, it is necessary to go into the field and collect data. When collecting data in the field for use in a GIS, it is imperative to know the location of each data point collected. Depending upon the nature of the project and level of accuracy required, paper maps may be used in conjunction with physical landmarks (e.g., roads and buildings) to determine locations. However, for many projects, traditional surveying instruments or Global Positioning System (GPS) devices are necessary to accurately locate data points.

Locational coordinates from traditional surveys are typically processed in Coordinate Geometry (COGO) software and may then be transformed into a GIS-compatible vector file type. CartaLinx is able to perform this transformation.

For many GIS applications, however, GPS devices may provide a less expensive alternative. The Global Positioning System is composed of 274 US Department of Defense satellites orbiting the Earth at approximately 20,000 kilometers. Each satellite continuously transmits a time and location signal. A GPS receiver processes the satellite's signals and calculates its position.⁴⁷ The level of error in the position depends on the quality (and price) of the receiver, atmospheric conditions and other variables. Some units provide display of locations only, while others record locational information electronically for later download to a computer.

IDRISI includes a GPS link. When this is active, the position recorded by the GPS receiver is displayed on the active display window and is updated as the position changes. The geodetic coordinates produced by the receiver are automatically projected to the reference system of the display. This GPS link is intended primarily to display and save waypoints and routes. CartaLinx also includes a GPS link and allows for more flexibility in creating data layers from the GPS data.

Most GPS data processing software supports several GIS export formats. Shape files and DXF files are commonly used as a bridge to IDRISI. If those formats are not available or if the data set is small enough to be entered by hand, creation of an IDRISI vector file is easily accomplished through the use of a text editor like the Edit module in IDRISI.

Additionally, GPS software typically exports to several types of ASCII text files. The XYZIDRIS module in IDRISI imports one of the more common types. The input to XYZIDRIS must be a space- or comma-delimited ASCII file in which numeric X, Y, and Z values (where Z = elevation or a numeric identifier) are separated by one or more spaces and each line is terminated by a CR/LF pair. XYZIDRIS produces an IDRISI vector point file.

Substitute an Existing Data Layer as a Surrogate

At times, there is simply no way to find or create a particular data layer. In these cases, it may be possible to substitute existing data as a surrogate. For example, suppose an analysis requires powerline location information, but a powerlines data file is not available and you don't have time or funds to collect the data in the field. You know, however, that in your study area, powerlines generally follow paved roads. For the purposes of your analysis, if the potential level of error introduced is acceptable, you may use a paved roads layer as a surrogate for powerlines.

Derive New Data from Existing Data

New data layers may also be derived from existing data. This is referred to as derivative mapping and is the primary way in which a GIS database grows. With derivative mapping, some knowledge of relationships is combined with existing data layers to create new data layers. For example, if an image of slopes is needed, but none exists, you could derive the slope image (using the IDRISI SURFACE module) from a digital elevation model, if available. Similarly, if you need an image of the relative amount of green biomass on the ground, you might derive such an image from the red and infrared bands of satellite imagery.

Another common form of derivative mapping is the interpolation of a raster surface (e.g., an elevation model or temperature surface) from a set of discrete points or isolines using TIN modeling or Geostatistics, both of which are available in IDRISI. See the **Surface Interpolation** chapter for more information about this form of derivative mapping.

Considerations

Resolution

In all database development tasks, no matter the source of data, there are several issues to consider. The first consideration is the resolution of the data. How much detail do you need? Resolution affects storage and processing time if too fine and limits the questions you can ask of the data if too coarse.

Resolution may refer to the spatial, temporal or attribute components of the database. Spatial resolution refers to the size

47. Several GPS companies (e.g., Trimble, Magellan) have excellent web sites explaining how positions are calculated from the satellite signals.

of the pixel in raster data or the scale of the map that was digitized to produce vector data. Temporal resolution refers to the currency of the data and whether the images in the time series are frequent enough and spaced appropriately. Attribute resolution refers to the level of detail captured by the data values. This might be exemplified by the difference between a landcover map with a single forest class, a landcover map with hardwood and softwood forest classes, and a landcover map with many different forest classes.

Accuracy

Accuracy is the second consideration. While accuracy does have a relationship to resolution, it is also a function of the methods and care with which the data were collected and digitized. Unfortunately, it is not always easy to evaluate the accuracy of data layers. However, most government mapping agencies do have mapping standards that are available. The IDRISI metadata (documentation) file structure includes fields for accuracy information, but many other file formats carry no accuracy information. In addition, even when such information is reported, the intelligent use of accuracy information in GIS analysis is still an avenue of active research. The capabilities of IDRISI in this area are discussed in the chapter on **Decision Support**.

Georeferencing

The third consideration is georeferencing. Data layers from various sources will often be georeferenced using different reference systems.⁴⁸ For display and GIS analysis, all data layers that are to be used together must use the same reference system. Providing that the details of the reference systems are known, they can usually be converted with the IDRISI module PROJECT.

However, it may be possible to download graphic files from the Web, for example, that are not georeferenced. Also, while IDRISI supports many projections, you may find data that are in an unsupported projection. In both cases, you will not be able to use PROJECT. It is sometimes possible to georeference an unreferenced file or a file with an unsupported projection through resampling (the IDRISI module RESAMPLE) if points of known locations can be found on the unreferenced image. All users are encouraged to read the chapter on **Georeferencing** for more information about this key issue. Also, see the section on Data Integration below.

Cost

Finally, the fourth consideration in database development is cost. This must be assessed in terms of both time and money. Other considerations include whether the data will be used once or many times, the accuracy level necessary for the particular (and future) uses of the data, and how often it must be updated to remain useful. Don't underestimate the amount of labor required to develop a good database. As stated in the introduction, database development quite commonly consumes a large portion of the labor allocated to a GIS project.

General Import Tips

IDRISI requires files to be in Intel format. This typically is only an issue when integer data are acquired from Macintosh, UNIX, workstation and mainframe platforms which use the Motorola format. The byte order is different between the two formats. The GENERICRASTER module can be used to reverse the byte order of an integer file.

If you import data from certain media, the files will be written to your hard disk with a read-only attribute. This can be removed with Windows Explorer in the Properties dialog for the file.

48. A reference system is defined by a projection, datum and grid system. See the chapter on **Georeferencing** for more information.

Tools for Import

Files are sometimes in an unspecified format. The following modules are useful for inspecting files and changing file structures.

Edit: Displays and edits an ASCII file. Vector data are often ASCII.

CONVERT: Converts raster and vector ASCII or binary files.

IDRISI File Explorer VIEW STRUCTURE: Displays byte-level contents of any file (in ASCII or binary format). Used to view data files for presence and size of headers, as well as to check files for carriage returns (CR's) and line feed (LF's).

CRLF: Adds or removes carriage returns and line feeds. Used with import modules that require specific fixed record lengths (e.g., DLG's, DEM's).

VAR2FIX: Converts variable length ASCII files to fixed length. Used in conjunction with CRLF.

Metadata: Creates and updates documentation files for data files already in IDRISI format.

GENERICRASTER: Imports, converts and swaps byte order on single or multi-banded raster data in band-interleaved-by-line (BIL), band-interleaved-by-pixel (BIP) or band sequential (BSQ) format.

SSTIDRIS: Converts raster images entered in from a spreadsheet program or any ASCII grid format into an IDRISI image.

Importing Satellite Imagery

IDRISI includes several special import routines for specific satellite image formats, such as LandsatLandsat, HDF-EOS and SPOT. If a specific import routine isn't available for the format of your data, the generic import tools available in IDRISI may be used to bring the satellite imagery into IDRISI. When preparing to import satellite data using the generic tools, there are three primary concerns: the location and contents of the header information (the information that describes the data), the size of the data in bytes, and the format in which the data are stored.

The first task is to locate and read the header information. Commonly, the header will be stored as a separate file in ASCII format or it will be distributed as paper documentation. Use the EDIT module to read this data. If, as a third possibility, the header is attached to the data file, it will be stored in binary format because satellite imagery is usually distributed in binary format. In this case, you can try to read the information from the header with the VIEW STRUCTURE utility of the IDRISI File Explorer which displays binary data as ASCII text. If you know the number of rows and columns in the image, but not the header size, there is a way to figure this out. If the image is in byte binary format, the number of bytes in the image should equal the number of pixels in the image. To find the number of pixels in the image, multiply the number of rows by the number of columns. The extra bytes in the data file should be the size of the header. If the file contains 2-byte integer data, the number of bytes in the image equals rows multiplied by columns multiplied by 2. You can determine the exact file size of a file by using the Properties command in Windows Explorer. For example, if an image has 512 rows and 512 columns, and the data type is binary, the file size with no header must be 262,144. As another example, if we know the file has 512 rows and 512 columns, and the file size is 262,656, we can assume that the header size is 512 bytes.

The contents of the header information will be presented in a variety of formats and with varying levels of detail. This section will tell you what to look for in the header, but it cannot anticipate the format in which you will find that information.

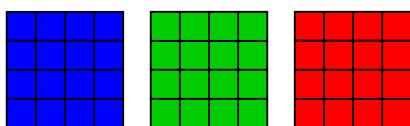
There are three pieces of information you must find in the header (the answers to 1 and 2 will determine the import routine you will use):

- 1) the data file format;
- 2) whether a header file is attached to the data file, and its size; and

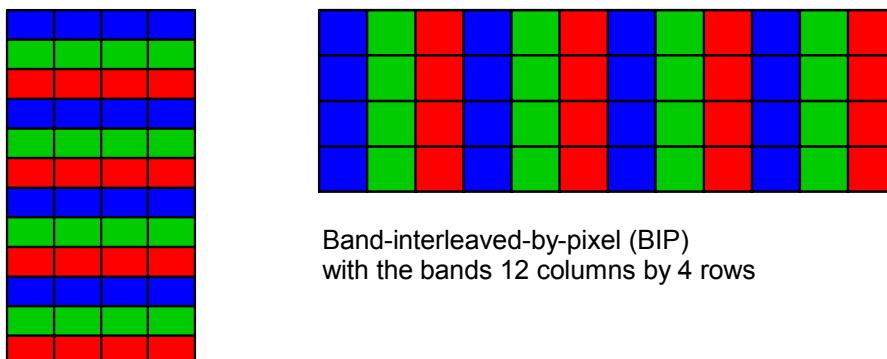
3) the number of rows and columns in the data file.

Most satellite imagery is distributed in one of two file formats: band sequential (BSQ) and band-interleaved-by-line (BIL). Band sequential format is the same as IDRISI image format. The data values are stored as a single series of numbers and each band of imagery is contained in a separate data file. SPOT panchromatic⁴⁹ and Landsat TM and MSS satellite imagery are commonly distributed in this format. Band-interleaved-by-line (BIL) format stores all the bands of multi-spectral imagery in a single data file. The bands are combined by storing the first row of data from the the first band of imagery, followed by the first row of data from the second band and so on for every band of imagery stored in the data file. The data file then continues with the second row of data from the first band, the second row of data from the second band and so on. SPOT multispectral imagery (SPOT-XS) is commonly distributed in BIL format. A third format, band-interleaved-by-pixel (BIP), is uncommon but used with older datasets. The figures below illustrate the three formats.

Satellite Imagery Formats



Band Sequential (BSQ) with three bands 4 columns by 4 rows



Band-interleaved-by-pixel (BIP)
with three bands 12 columns by 4 rows

Band-interleaved-by-line (BIL)
with three bands 4 columns by 12 rows

With BSQ format files, the following two import routines apply:

1. If a header is attached to the data file, use the GENERICRASTER module, specify BSQ and no header. GENERICRASTER will remove the header, rename the data file to have an .RST extension, and create a documentation file. GENERICRASTER will ask for the number of rows and columns, the minimum information necessary to create the documentation file. GENERICRASTER will ask for additional information as well, such as the reference system parameters which you should be able to find in the header. When in doubt, you can try the following values: Minimum X=0,

49. The documentation for the SPOT panchromatic band will indicate that it is in BIL format, but it contains only one band, so there is no interleaving. This is the same as BSQ format.

Maximum X=1 (or # of columns), Minimum Y=0, Maximum Y=1 (or # of rows), Reference System=Plane.

2. If the header file is separate from the data file, then you merely need to rename the data file to have an .RST extension (with the Windows Explorer rename command). Then run Metadata from the File menu to create the documentation file. Again, you must know the number of columns and rows in order to complete the documentation file, but you should also enter any reference system information that is available from the header. If the values are unknown, try entering the values noted in the previous paragraph, with the addition of Reference Units=M and Unit Distance=1.

For all BIL files, with or without a header, the following import routine is used:

If a header is attached, determine its size (in bytes) and then use the GENERICRASTER module. GENERICRASTER will pull apart and create individual IDRISI format image and documentation files for each band of imagery.

There are some common errors during the import process that can give the user some useful information for correction. If the header size is incorrectly specified using GENERICRASTER, the image will not be correctly imported. Because too much or too little is taken out of the file as a header, the actual data left behind is either incomplete or contains extra information. This can lead to an error message informing you that the image is incorrectly sized, or the image will display improperly (e.g., data from the right side may appear moved to the left side). You will need to determine the correct header size and try again.

If an image seems to successfully import, but the display of the image is staggered or skewed to one side, it is possible that you have incorrectly specified the number of rows and columns in the documentation file. Recheck your documentation file for the correct information or use the method discussed above to determine the correct rows and columns based on the image and header size. If an image seems to successfully import but the display of the image contains a venetian blind striping effect, there are two possible causes. If the striping is vertical, try importing the image as a BIP format file using GENERICRASTER. If the striping is horizontal, try importing as a BIL format file. The first figure below left illustrates an import when the incorrect number of bands is specified during a BIL import. The figure in the center illustrates a common result when the header is not calculated correctly. The figure on the right demonstrates a result when the incorrect number of columns is specified.

A common error associated with importing integer data results from data originating from the UNIX environment. If after successfully importing integer data the result shows very maximum and minimum values in the image, most likely the bytes need to be swapped. Run the GENERICRASTER routine again, but check the option to Swap byte order. The byte swapping option is used for the conversion of UNIX and Macintosh files that have the low byte/high byte order reversed from Intel-based systems.

Common errors during import

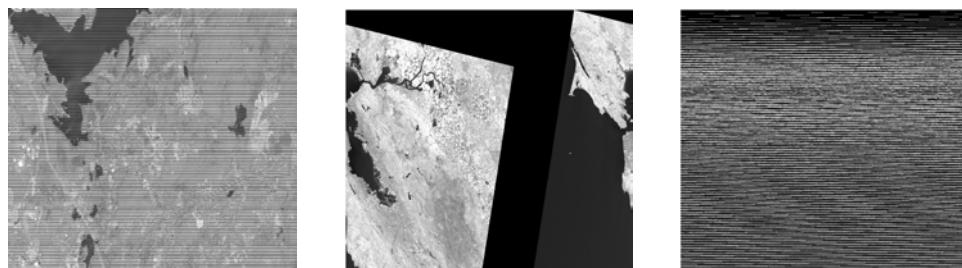


Figure 10-2

Importing GIS Data

Much of the information presented above regarding the import of satellite imagery also applies to the import of raster GIS data. Specific import routines are also available in IDRISI for many common agency and software-specific GIS formats. The procedures for importing common GIS data formats are found in the on-line Help System. Search the Help System index for the name of the format you would like to import. In many cases, GIS data can be saved in a format that can be imported into IDRISI, such as ERDAS Imagine, ESRI ArcRaster, GEOTIFF or GRASS files.

Data Integration

Requirements for Data Integration

Data integration refers to the use of various data layers together in display or analysis. There are several parameters that must match if data layers are to be used together. First, the reference systems must match.⁵⁰ This is important because we are often combining separate themes based on their spatial characteristics. A roads vector layer can be properly overlaid on a raster satellite image, for example, only if both layers have the same reference system. A particular x,y coordinate pair in the vector file must represent exactly the same place on the ground as that same x,y coordinate pair in the raster image.

For raster layers to be used in image-to-image operations, such as OVERLAY, the extent of the images, the number of rows and columns in the images, and the pixel resolution of the images must all match. Note that if any two of these three conditions are met, the third is met automatically.

The steps used to integrate data layers, especially if data is transformed, should be recorded in the lineage field of the documentation file of each data layer. For example, it is trivial to make smaller pixels from larger pixels, and this is often done to achieve data integration. However, it is important to note that the information carried by those smaller pixels still represents data gathered at a coarser resolution. Such information should be recorded in the new image's documentation file so users will be aware that the apparent resolution exceeds that of the information.

Tools for Data Integration

The modules PROJECT and RESAMPLE are the primary tools available in IDRISI for georeferencing and changing reference systems. The chapter on **Georeferencing** details the differences between these two modules and when they should be employed. In addition, the **Tutorial** includes exercises on the use of RESAMPLE and PROJECT.

The WINDOW module is used to change the extent of raster images. It works by saving a subset of a larger image as a new image.

Changing the number of rows and columns in an image and/or changing the image pixel resolution can be accomplished with the modules CONTRACT, EXPAND and PROJECT. CONTRACT and EXPAND work by thinning or duplicating pixels in an image by an integer multiple. For example, an image with 100 rows and 100 columns and a pixel resolution of 30 meters could be contracted to an image of 25 rows and 25 columns by using a contraction factor of 4. Similarly, an image of 100 rows and 100 columns could be transformed to an image of 1000 rows and 1000 columns by using EXPAND and an expansion factor of 10.

If it is necessary to contract or expand an image by a non-integer factor (e.g., 1.5), CONTRACT and EXPAND cannot be used. In these cases, the easiest method to use is PROJECT. Project to the same reference system that the file is currently in, enter the new number of rows and columns, and choose the resampling technique that best suits your data and application. In general, if quantitative data are used, then the bilinear resampling type should be chosen, but if qualitative data are used, the nearest neighbor resampling type should be chosen.

50. The reference system projection, datum and grid system must all match for the reference system to match.

Conclusions

Database development is seldom quick and easy. The quality of the database, in many ways, limits the quality of any resulting analysis to be carried out with the data. There are a significant number of technical pitfalls that are typically encountered during database development and it is easy to become overwhelmed by these. However, despite these difficulties, it is important to maintain a project-driven approach to GIS analysis and to limit the influence data constraints can have on the definition of the problems to be addressed.

Georeferencing

Georeferencing refers to the manner in which map locations are related to earth surface locations. Georeferencing requires several ingredients:

- a logic for referring to earth surface locations—a concern of the field of Geodesy;
- a specific implementation of that logic, known as a Geodetic Datum—a concern of the field of Surveying;
- a logic for referring locations to their graphic positions—a concern of the field of Cartography; and
- an implementation of that logic, known as a data structure—a concern of GIS and Desktop Mapping software, and in this case, IDRISI.

Geodesy

Geodesy is that field of study which is concerned with the measurement of the size and shape of the earth and positions upon it. The most fundamental problem that Geodesists face is the fact that the earth's surface is irregular in shape. For example, imagine two locations (A and B) at either end of a thin straight beach bordered by a steep cliff (Figure 11-1). Clearly the distance one would determine between the two locations would depend upon the route chosen to undertake the measurement. Measuring the distance by the cliff route would clearly be longer than that determined along the coast. Thus irregularities (hills and valleys) along the measuring surface can cause ambiguities in distance (and thereby, location). To alleviate this situation, it has long been a common practice to reduce all measurements to a more regular measuring surface—a *reference surface*.

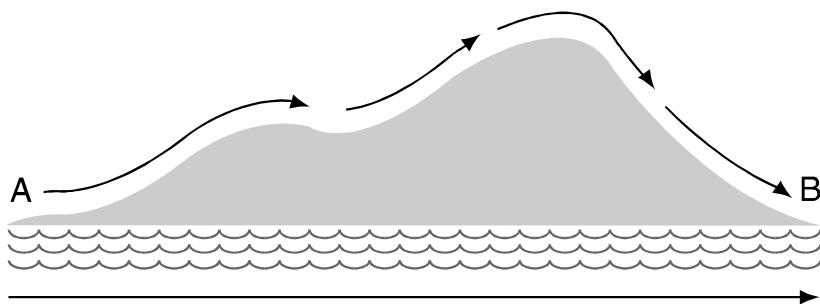


Figure 11-1

Geoid

The oldest reference surface used for mapping is known as the *geoid*. The geoid can be thought of as mean sea level, or where mean sea level would be if the oceans could flow under the continents. More technically, the geoid is an *equipotential surface of gravity* defining all points in which the force of gravity is equivalent to that experienced at the ocean's surface. Since the earth spins on its axis and causes gravity to be counteracted by centrifugal force progressively towards the equator, one would expect the shape of the geoid to be an oblate spheroid—a sphere-like object with a slightly fatter middle and flattened poles. In other words, the geoid would have the nature of an ellipse of revolution—an *ellipsoid*.

As a reference surface, the geoid has several advantages—it has a simple physical interpretation (and an observable position along the coast), and it defines the horizontal for most traditional measuring instruments. Thus, for example, leveling a theodolite or sextant is, by definition, a process of referring the instrument to the geoid.

Reference Ellipsoids

Unfortunately, as it turns out, the geoid is itself somewhat irregular. Because of broad differences in earth materials (such as heavier ocean basin materials and lighter continental materials, irregular distributions such as mountains, and isostatic imbalances), the geoid contains undulations that also introduce ambiguities of distance and location. As a result, it has become the practice of modern geodetic surveys to use abstract reference surfaces that are close approximations to the shape of the geoid, but which provide perfectly smooth reference ellipsoids (Figure 11-2). By choosing one that is as close an approximation as possible, the difference between the level of a surveying instrument (defined by the irregular geoid) and the horizontal of the reference ellipsoid is minimized. Moreover, by reducing all measurements to this idealized shape, ambiguities of distance (and position) are removed.

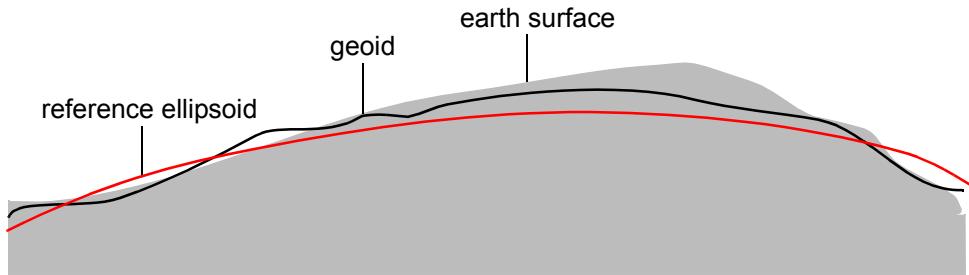


Figure 11-2

There are many different ellipsoids in geodetic use (see *Appendix 1: Ellipsoid Parameters*). They can be defined either by the length of the major (a) and minor (b) semi-axes⁵¹ (Figure 11-3), or by the length of the semi-major axis along with the degree of flattening [$f = (a-b) / a$]. The reason for having so many different ellipsoids is that different ones give better fits to the shape of the geoid at different locations. The ellipsoid chosen for use is that which best fits the geoid for the particular location of interest.

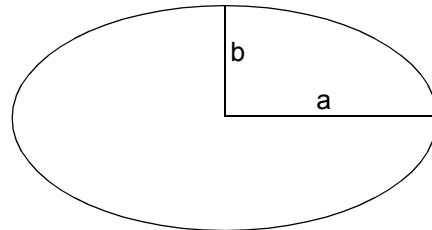


Figure 11-3

Geodetic Datums

Selecting a specific reference ellipsoid to use for a specific area and orienting it to the landscape, defines what is known in Geodesy as a *datum* (note that the plural of datum in geodesy is *datums*, not data!). A datum thus defines an ellipsoid (itself defined by the major and minor semi-axes), an initial location, an initial azimuth (a reference direction to define the direction of north), and the distance between the geoid and the ellipsoid at the initial location. Establishing a datum is the task of geodetic surveyors, and is done in the context of the establishment of national or international geodetic control survey networks. A datum is thus intended to establish a permanent reference surface, although recent advances in survey technology have led many nations to redefine their current datums.

Most datums only attempt to describe a limited portion of the earth (usually on a national or continental scale). For example, the North American Datum (NAD) and the European Datum each describe large portions of the earth, while the Kandawala Datum is used for Sri Lanka alone. Regardless, these are called local datums since they do not try to describe the entire earth. By contrast, we are now seeing the emergence of World Geodetic Systems (such as WGS84) that do try

51. The major and minor semi-axes are also commonly referred to as the semi-major and semi-minor axes.

to provide a single smooth reference surface for the entire globe. Such systems are particularly appropriate for measuring systems that do not use gravity as a reference frame, such as Global Positioning Systems (GPS). However, presently they are not very commonly found as a base for mapping. More typically one encounters local datums, of which several hundred are currently in use.

Datums and Geodetic Coordinates

Perhaps the most important thing to bear in mind about datums is that each defines a different concept of geodetic coordinates—latitude and longitude. Thus, in cases where more than one datum exists for a single location, more than one concept of latitude and longitude exists. It can almost be thought of as a philosophical difference. It is common to assume that latitude and longitude are fixed geographic concepts, but they are not. There are several hundred different concepts of latitude and longitude currently in use (one for each datum). It might also be assumed that the differences between them would be small. However, that is not necessarily the case. In North America, a change is being undertaken to convert all mapping to a recently defined datum called NAD83. At Clark University, for example, if one were to measure latitude and longitude according to NAD83 and compare it to the ground position of the same coordinates in the previous system, NAD27, the difference is in excess of 40 meters! Other locations in the US experience differences in excess of 100 meters. Clearly, combining data from sources measured according to different datums can lead to significant discrepancies.

The possibility that more than one datum will be encountered in a mapping project is actually reasonably high. In recent years, many countries have found the need to replace older datums with newer ones that provide a better fit to local geoidal characteristics. In addition, regional or international projects involving data from a variety of countries are very likely to encounter the presence of multiple datums. As a result, it is imperative to be able to transform the geodetic coordinates of one system to those of another. In IDRISI, the PROJECT option under the Reformat menu incorporates full datum transformation as a part of its operation.

Cartographic Transformation

Once a logic has been established for referring to earth locations and a set of measurements has been made, a means of storing and analyzing those positions is required. Traditionally, maps have been the preferred medium for both storage and analysis, while today that format has been supplemented by digital storage and analysis. Both, however, share a common trait—they are most commonly flat! Just as flat maps are a more manageable medium than map globes, plane coordinates are a more workable medium than spherical (ellipsoidal) coordinates for digital applications. As a result, surveyed locations are commonly transformed to a plane grid referencing system before use.

Projection

The process of transforming spheroidal geodetic coordinates to plane coordinate positions is known as projection, and falls traditionally within the realm of cartography. Originally, the concern was only with a one-way projection of geodetic coordinates to the plane coordinates of a map sheet. With the advent of GIS, however, this concern has now broadened to include the need to undertake transformations in both directions in order to develop a unified database incorporating maps that are all brought to a common projection. Thus, for example, a database developed on a Transverse Mercator projection might need to incorporate direct survey data in geodetic coordinates along with map data in several different projections. *Back projecting* digitized data from an existing projection to geodetic coordinates and subsequently using a *forward projection* to bring the data to the final projection is thus a very common activity in GIS. The PROJECT module in IDRISI supports both kinds of transformation.

Projection of spheroidal positions to a flat plane simply cannot be done without some (and oftentimes considerable) distortion. The stretching and warping required to make the transformation work leads to continuous differences in scale over the face of the map, which in turn leads to errors in distance, area and angular relationships. However, while distor-

tion is inevitable, it is possible to engineer that distortion such that errors are minimized and certain geometrical qualities are maintained.

Of particular importance to GIS are the family of projections known as *conformal* or *orthomorphic*. These are projections in which the distortion is engineered in such a manner that angular relationships are correctly preserved in the near vicinity of any location. Traditionally, the field of surveying has relied upon the measurement of angular relationships (using instruments such as a theodolite) in order to measure accurate distances over irregular terrain. As a result, conformal projections have been important for the correct transfer of field measurements to a map base, or for the integration of new surveys with existing map data. For the same reasons, conformal projections are also essential to many navigation procedures. As a consequence, virtually all topographic map bases are produced on conformal projections, which form the basis for all of the major grid referencing systems in use today.

Grid Referencing Systems

A grid referencing system can be thought of very simply as a systematic way in which the plane coordinates of the map sheet can be related back to the geodetic coordinates of measured earth positions. Clearly, a grid referencing system requires a projection (most commonly a conformal one). It also requires the definition of a plane cartesian coordinate system to be superimposed on top of that projection. This requires the identification of an initial position that can be used to orient the grid to the projection, much like an initial position is used to orient a datum to the geoid. This initial position is called the *true origin* of the grid, and is commonly located at the position where distortion is least severe in the projection (Figure 11-4). Then, like the process of orienting a datum, a direction is established to represent *grid north*. Most commonly, this will coincide with the direction of true north at the origin. However, because of distortion, it is impossible for true north and grid north to coincide over many other locations.

Once the grid has been oriented to the origin and true north, a numbering system and units of measure are determined. For example, the UTM (Universal Transverse Mercator) system uses the equator and the central meridian of a 6-degree wide zone as the true origin for the northern hemisphere. The point is then given arbitrary coordinates of 500,000 meters East and 0 meters North. This then gives a *false origin* 500 kilometers to the west of the true origin (see figure above). In other words, the false origin marks the location where the numbering system is 0 in both axes. In IDRISI, the numbering logic of a grid referencing system is always given by specifying the latitude and longitude of the true origin and the arbitrary coordinates that exist at that point (in this example, 500,000 E and 0 N).

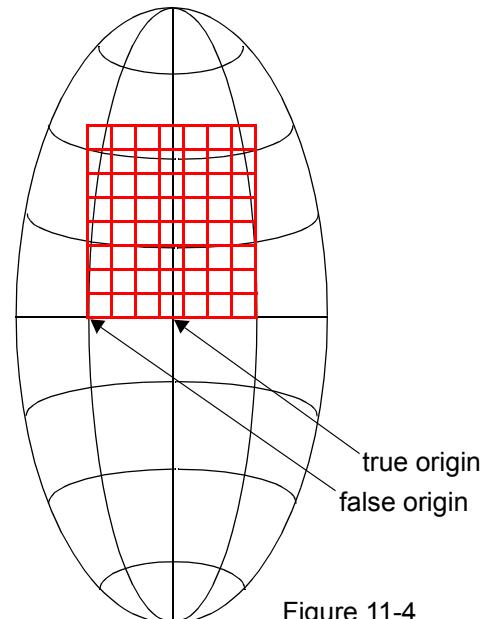


Figure 11-4

Georeferencing in IDRISI

The logic that IDRISI uses for georeferencing is quite simple, and is based on the issues previously described.

All geographic files are assumed to be stored according to a grid reference system where grid north is aligned with the edges of the raster image or vector file. As a result, the minimum X of a raster image is always the left-hand edge, the maximum Y is the top edge, and so on (Figure 11-5). The georeferencing properties of an IDRISI coverage (accessible through Metadata) include an entry specifying the reference system used by that file when referring to geographic locations. The particulars of that reference system (e.g., projection, datum, origin, etc.) are then contained in a *reference system parameter file* (.ref) (see below). Whenever a projection or datum transformation is required, the PROJECT module refers to this information to control the transformation process.

Every grid reference system must have a reference system parameter file. The only exception to this is a system identified by the keyword "plane". Any coverage that indicates a *plane* coordinate referencing system is understood to use an arbitrary plane system for which geodetic and projection parameters are unknown, and for which a reference system parameter file is not provided. A coverage in a plane reference system cannot be projected.

Over 400 reference system parameter files are supplied with IDRISI. However, .ref files can be created for any grid referencing system using the Metadata or Edit modules. Details on the structure of this simple ASCII text file format are provided below and in the on-line Help System.

For simplicity, geodetic coordinates (lat/long) are recognized as a special type of grid referencing system. The true and false origins are identical and occur at 0 degrees of longitude and 0 degrees of latitude. Units are in decimal degrees or radians. Longitudes west of the prime (Greenwich) meridian and latitudes south of the equator are expressed as negative numbers. Thus, for example, Clark University has approximate coordinates of -71.80,+42.27 ($71^{\circ}48' W$, $42^{\circ}16' N$). Although geodetic coordinates are truly spheroidal, they are logically treated here as a plane coordinate system. Thus, they are implicitly projected according to a Plate Carrée projection,⁵² although the projection will actually be listed as "none".

Be aware that there are many possible interpretations of the concept of latitude and longitude, depending upon the datum in use. A single .ref file has been supplied (called LATLONG) for the case where the datum is WGS84. This should be copied and modified for other interpretations.

In IDRISI, the registration point for referencing raster images is in the lower left corner of any cell. Thus, for example, with an image of 10 columns by 5 rows, using a plane reference system of cells 10 meters by 10 meters, the lower left-most corner of the lower left-most cell has the coordinates 0,0 while the upper right-most corner of the upper right-most cell has the position 100.50. Note also that this lower left-most cell is considered to be in column 0 and row 4 while the upper right-most cell is in column 0, row 0.

IDRISI image and vector documentation files contain several fields that describe the reference system of the file. The *min X*, *max X*, *min Y*, and *max Y* entries give the edge coordinates of an image or bounding rectangle of a vector file in reference system coordinates. For image files, the number of *rows* and *columns* is also given.

The reference system name (*ref. system*) identifies the reference system of the file, and may be "plane", "lat/long", or a specific reference system described by a reference system parameter file. In the latter case, the *ref. system* entry in the documentation file should match exactly the name of the corresponding reference system parameter file, without the .ref extension.

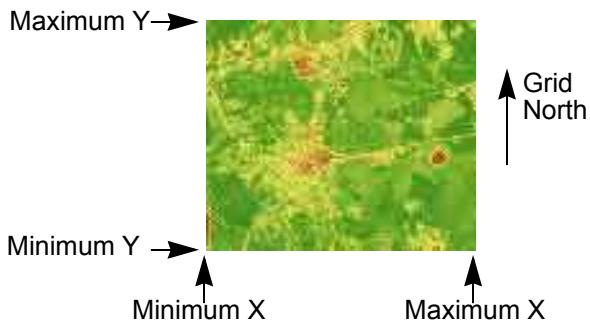


Figure 11-5

52. Note, however, that geodetic coordinates are not considered to be identical to a system based on the Plate Carrée since no new numbering scheme has been undertaken. PROJECT supports transformations based on the Plate Carrée as well, where it is understood that a true plane coordinate system with its own special numbering scheme has been superimposed onto the projected geodetic coordinates.

The *reference units* entry indicates the unit of measure used in the reference coordinate system (e.g., meters).

The *unit distance* refers to the ground distance spanned by a distance of one unit (measured in reference units) in the reference system. Thus, for example, if the hypothetical image in Figure 11-6 had a unit distance of 2.0 and reference units in meters, it would imply that as we move one in the reference coordinates (e.g., from X=99 to X=100) we actually move 2.0 meters on the ground. Simply think of the unit distance parameter as a multiplier that should be applied to all coordinates to yield the reference units indicated. ***The unit distance will be 1.0 in most cases.*** However, the unit distance should be specified as a value other than 1.0 whenever you have reference units that are not among the standard units supported (meters, feet, miles, kilometers, degrees or radians). For example, a data file where the coordinates were measured in minutes of arc (1/60 of a degree) should have reference units set to degrees and the unit distance set to 0.016667.

In IDRISI, as you move the cursor over a displayed image, the row and column position and the X and Y coordinates are indicated on the status bar at the bottom of the screen. Vector files can also be overlaid (using Composer) onto the raster image, provided they have the same reference system. *There is no need for the bounding rectangle of the vector file to match that of the raster image.* IDRISI will correctly overlay the portion of the vector file that overlaps the displayed raster image. Onscreen digitizing will always output coordinates in the same reference system as the raster image being displayed.

Reference System Parameter Files

As previously indicated, IDRISI comes with over 400 reference system parameter files. These include one for geodetic coordinates (latitude/longitude) using the WGS84 datum, 160 for the UTM system (one each for the 60 zones, for both the northern and southern hemispheres) using the WGS84 datum, 32 based on the Gauss-Kruger projection, 40 for the UTM system covering North America using NAD27 and NAD83, and 253 for all US State Plane Coordinate (SPC) systems based on the Lambert Conformal Conic and Transverse Mercator projections. ***Appendix 3: Supplied Reference System Parameter Files*** lists each of these files by geographic location. In CartaLinx, the Georeferencing tab of the *Preferences/Properties/Options* dialog provides a list box in which any specific .ref file can be selected. During program installation, these supplied files are installed in the \Georef subfolder of your IDRISI program folder. The contents of these files may be viewed with Metadata or Edit.

Where .ref Files are Stored

As indicated above, supplied .ref files are stored in a subfolder of the IDRISI program folder called \Georef. If your IDRISI program folder is C:\Idrisi Kilimanjaro, then the supplied reference system parameter files are in c:\Idrisi Kilimanjaro\Georef. When you create new reference system parameter files, we recommend that you store them in this folder, always adding to your master library of reference files. However, reference system parameter files may be stored anywhere. When a reference system parameter file is given without its path in a dialog box or macro, IDRISI always looks first in the working folder, followed by the resource folders in the order they are named in the current project. If the file is not found, IDRISI will next look in the \Georef subfolder.

Creating New .ref Files

Although IDRISI supplies more than 400 reference system parameter files, many users will need to create new .ref files to suit the needs of specific systems. There are two options available to do this:

1. For cases where a different version of the UTM system is required, the module UTMREF can be used. This will simply require that you enter the zone number, the hemisphere (northern or southern), and the name of the datum along with its associated Molodensky constants. The constants can be found in ***Appendix 2: Molodensky Constants for Selected Geodetic Datums*** and are used by PROJECT to undertake datum transformations. If you enter a datum that is not included, you will need to supply the name of the reference ellipsoid and the length of its major and minor semi-axes. These constants for many reference ellipsoids are given in ***Appendix 1: Reference Ellipsoids***.
2. For all other cases, Metadata or Edit can be used to create the appropriate file. Often the easiest procedure is to use the copy function in the IDRISI File Explorer to duplicate an existing .ref file, and then modify it using Metadata or Edit. The section below indicates the structure and contents of a .ref file.

The .ref File Structure

Like all IDRISI documentation files, .ref files are simple ASCII text files that can be modified by means of any ASCII text editor.

The .ref file type follows the conventions of other documentation files by having the first 14 characters of each line devoted to a field description. Here is an example of a reference system parameter file named UTM-19N:

ref. system	:	Universal Transverse Mercator Zone 19
projection	:	Transverse Mercator
datum	:	NAD27
delta WGS84	:	-8 160 176
ellipsoid	:	Clarke 1866
major s-ax	:	6378206.5
minor s-ax	:	6356584
origin long	:	-69
origin lat	:	0
origin X	:	500000
origin Y	:	0
scale fac	:	1
units	:	m
parameters	:	0

The first line is simply a title and has no analytical use. The second line indicates the projection. In the current release of IDRISI, the following projections are supported:

Mercator
Transverse Mercator
Gauss-Kruger (Gauss-Krueger spelling also accepted.)
Lambert Conformal Conic
Plate Carrée
Hammer Aitoff
Lambert North Polar Azimuthal Equal Area
Lambert South Polar Azimuthal Equal Area
Lambert Transverse Azimuthal Equal Area
Lambert Oblique Azimuthal Equal Area
North Polar Stereographic
South Polar Stereographic
Transverse Stereographic
Oblique Stereographic
Albers Equal Area Conic
none (i.e., geodetic coordinates)

Note that each of the names above are keywords and must appear exactly in this form to be understood by the PROJECT module.

The next line lists the geodetic datum. The text here is informational only, with the exception of the keywords NAD27 and NAD83. These latter two cases are the keywords for the two existing implementations of the North American datum. When PROJECT encounters these keywords as elements of the input and output systems, it uses a special (and more precise) datum transformation technique. For all other cases, the Molodensky transform is used.

The next line lists the differences between the center (i.e., earth center) of the datum in use and that of the WGS84 datum (World Geodetic System, 1984). The values express, in meters, the differences in the X, Y and Z axes respectively, relative to WGS84. These are known as the Molodensky constants. **Appendix 2: Datum Parameters** contains a list of these constants for most datums worldwide. They are used by PROJECT to undertake datum transformations except in the case of conversions between NAD27 and NAD83. These conversions use the US National Geodetic Survey's NADCON procedure.

The next three lines give information about the reference ellipsoid. The line listing the name of the ellipsoid is for informational purposes only. The following two lines describe, in meters, the major and minor semi-axes of the ellipsoid used by the datum. These entries are used analytically by the PROJECT module. **Appendix 1: Reference Ellipsoids** contains a list of ellipsoids and the lengths of their semi-axes.

The next four lines describe the origin and numbering system of the reference system. The *origin long* and *origin lat* entries indicate the longitude and latitude of the true origin. The *origin X* and *origin Y* entries then indicate what coordinate values exist at that location in the reference system being described.

The *scale fac* entry indicates the scale factor to be applied at the center of the projection. A typical value would be 1.0, although it is equally permissible to indicate "na" (an abbreviation for "not applicable"). The most likely case in which users will encounter a need to specify a value other than 1.0 is with the UTM system and other systems based on the Transverse Mercator projection. With the UTM system, the *scale fac* should read 0.9996.

The *units* entry duplicates the units information found in raster and vector documentation files and is included here for confirmation by the system. Therefore, in both reference and documentation files, *units* should be considered as analytical entries.

Finally, the *parameters* entry indicates the number of special parameters included with the .ref file information. Of the projections currently supported, only the Lambert Conformal Conic and Alber's Equal Area Conic require special parameters. All others should have a value of 0 for this entry. In cases using either the Lambert Conformal Conic or Alber's Equal Area Conic projections, the *parameters* entry should read 2 and then include the latitudes (in decimal degrees) of the two standard lines (lines of no distortion) on the following two lines. Here, for example, is the .ref file for the Massachusetts State Plane Coordinate System, Zone 1 (SPC83MA1), based on the NAD83 datum:

ref. system	:	Massachusetts State Plane Coordinate System Mainland Zone
projection	:	Lambert Conformal Conic
datum	:	NAD83
delta WGS84	:	0 0 0
ellipsoid	:	GRS80
major s-ax	:	6378137
minor s-ax	:	6356752.31
origin long	:	-71.5
origin lat	:	41
origin X	:	200000
origin Y	:	750000
scale fac	:	na
units	:	m
parameters	:	2
stand ln 1	:	41.72
stand ln 2	:	42.68

Note that the latitude of the "lower" standard line (i.e., that which is nearer the equator) is listed first and that of the lati-

tudinally "higher" standard line is listed second. Future additions of other projections may dictate the need for additional special parameters. However, in this version, only those reference systems based on the Lambert Conformal Conic and Alber's Equal Area Conic projections require special parameters.

Projection and Datum Transformations

The PROJECT option of the COVERAGE menu in CartaLinx provides full transformation between reference systems. As a consequence, it undertakes transformation of both projection and datum characteristics.

Projection transformations for all supported projections are undertaken using ellipsoidal formulas accurate to 2 cm on the ground. As CartaLinx uses double precision floating point numbers (15 significant figures) for the representation of coordinate data, this precision is maintained in all results. However, be aware that the precision of exported coordinates depends on the numeric precision used by the export format being used.

For datum transformations, the primary method used is the Molodensky transformation described in DMA (1987). Whenever the keywords "NAD27" and "NAD83" are encountered in a transformation, however, the more precise US National Geodetic Survey NADCON procedure is used (see Dewhurst, 1990). *This latter procedure should only be used within the continental US.* For other regions covered by the North American Datum, do not use the NAD27 or NAD83 keywords, but rather some other spelling in the datum field, and indicate the correct Molodensky transformation constants from *Appendix 2: Datum Parameters*.⁵³

Algorithms Used by PROJECT

Forward and backward ellipsoidal projection formulae were taken from Snyder (1987). For datum transformations, the Molodensky transform procedure (DMA, 1987) is used. However, for the specific case of NAD27/NAD83 transformations within the continental US, the US National Geodetic Survey's NADCON procedure is used. This procedure is described in Dewhurst (1990) and is accompanied by relevant data files. Briefly, the procedure involves a matrix of corrections to longitude and latitude that are accessed as a two-dimensional look-up table. Final correction values are then interpolated between the four nearest values in the matrix using a bilinear interpolation. To facilitate this, the NADCON data files for the continental US were converted into IDRISI images. These images are contained in the GEOREF subdirectory under the IDRISI program directory. There are two of these files, NADUSLON and NADUSLAT, for the corrections to longitude and latitude respectively.

Further Reading

Geodesy is not a familiar topic for many in GIS. However, as one can see from the material in this chapter, it is essential to effective database development. The following readings provide a good overview of the issues involved:

Burkard, R.K., (1964) *Geodesy for the Layman*, (St. Louis, MO: USAF Aeronautical Chart and Information Center).

Smith, J.R., (1988) *Basic Geodesy*, (Rancho Cordova, CA: Landmark Enterprises).

For projections, there are few texts that can match the scope and approachability of:

53. Note that while a set of constants exist for the North American Datum as a whole, Molodensky constants also exist for more specific portions of the regions covered to enhance the precision of the transformation.

Maling, D.H., (1973) *Coordinate Systems and Map Projections*, (London: George Phillip and Son).

Snyder, J.P., (1987) *Map Projections: A Working Manual*. U.S. Geological Survey Professional Paper 1395, (Washington, DC: US Government Printing Office).

Finally, with respect to the procedures used for datum transformations, please refer to:

DMA, (1987) *Department of Defense World Geodetic System 1984: Its Definition and Relationships with Local Geodetic Datums*, DMA TR 8350.2, (Washington, DC: The Defense Mapping Agency).

Dewhurst, W.T., (1990) *NADCON: The Application of Minimum Curvature-Derived Surfaces in the Transformation of Positional Data from the North American Datum of 1927 to the North American Datum of 1983*, NOAA Technical Memorandum NOS NGS-50, (Rockville, MD: National Geodetic Information Center).

Decision Support: Decision Strategy Analysis

With rapid increases in population and continuing expectations of growth in the standard of living, pressures on natural resource use have become intense. For the resource manager, the task of effective resource allocation has thus become especially difficult. Clear choices are few and the increasing use of more marginal lands puts one face-to-face with a broad range of uncertainties. Add to this a very dynamic environment subject to substantial and complex impacts from human intervention, and one has the ingredients for a decision making process that is dominated by uncertainty and consequent risk for the decision maker.

In recent years, considerable interest has been focused on the use of GIS as a decision support system. For some, this role consists of simply informing the decision making process. However, it is more likely in the realm of resource allocation that the greatest contribution can be made.

Over the past several years, the research staff at the Clark Labs have been specifically concerned with the use of GIS as a direct extension of the human decision making process—most particularly in the context of resource allocation decisions. However, our initial investigations into this area indicated that the tools available for this type of analysis were remarkably poor. Despite strong developments in the field of Decision Science, little of this had made a substantial impact on the development of software tools. And yet, at the same time, there was clear interest on the part of a growing contingency of researchers in the GIS field to incorporate some of these developments into the GIS arena. As a consequence, in the early 1990s, we embarked on a project, in conjunction with the United Nations Institute for Training and Research (UNITAR), to research the subject and to develop a suite of software tools for resource allocation. These were first released with Version 4.1 of the MS-DOS version of IDRISI, with a concentration on procedures for Multi-Criteria and Multi-Objective decision making—an area that can broadly be termed *Decision Strategy Analysis*. Since then, we have continued this development, most particularly in the area of *Uncertainty Management*.

Uncertainty is not simply a problem with data. Rather, it is an inherent characteristic of the decision making process itself. Given the increasing pressures that are being placed on the resource allocation process, we need to recognize uncertainty not as a flaw to be regretted and perhaps ignored, but as a fact of the decision making process that needs to be understood and accommodated. Uncertainty Management thus lies at the very heart of effective decision making and constitutes a very special role for the software systems that support GIS. The following discussion is thus presented in two parts. This chapter explores Decision Strategy Analysis and the following chapter discusses Uncertainty Management.

Introduction⁵⁴

Decision Theory is concerned with the logic by which one arrives at a choice between alternatives. What those alternatives are varies from problem to problem. They might be alternative actions, alternative hypotheses about a phenomenon, alternative objects to include in a set, and so on. In the context of GIS, it is useful to distinguish between policy decisions and resource allocation decisions. The latter involves decisions that directly affect the utilization of resources (e.g., land) while the former is only intended to influence the decision behavior of others who will in turn make resource commitments. GIS has considerable potential in both arenas.

In the context of policy decisions, GIS is most commonly used to inform the decision maker. However, it also has potential (almost entirely unrealized at this time) as a process modeling tool, in which the spatial effects of predicted decision behavior might be simulated. Simulation modeling, particularly of the spatial nature of socio-economic issues and their

54. The introductory material in this chapter is adapted from Eastman, J.R., 1993. Decision Theory and GIS, *Proceedings, Africa GIS '93*, UNITAR, Geneva.

relation to nature, is still in its infancy. However, it is to be expected that GIS will play an increasingly sophisticated role in this area in the future.

Resource allocation decisions are also prime candidates for analysis with a GIS. Indeed, land evaluation and allocation is one of the most fundamental activities of resource development (FAO, 1976). With the advent of GIS, we now have the opportunity for a more explicitly reasoned land evaluation process. However, without procedures and tools for the development of decision rules and the predictive modeling of expected outcomes, this opportunity will largely go unrealized. GIS has been slow to address the needs of decision makers and to cope with the problems of uncertainty that lead to decision risk. In an attempt to address these issues, the Clark Labs has worked in close collaboration with the United Nations Institute for Training and Research (UNITAR) to develop a set of decision support tools for the IDRISI software system.

Although there is now fairly extensive literature on decision making in the Management Science, Operations Research and Regional Science fields (sometimes linked together under the single name *Decision Science*), there is unfortunately a broadly divergent use of terminology (e.g., see Rosenthal, 1985). Accordingly, we have adopted the following set of operational definitions which we feel are in keeping with the thrust of the Decision Science literature and which are expressive of the GIS decision making context.

Definitions

Decision

A decision is a choice between alternatives. The alternatives may represent different courses of action, different hypotheses about the character of a feature, different classifications, and so on. We call this set of alternatives the *decision frame*. Thus, for example, the decision frame for a zoning problem might be [commercial residential industrial]. The decision frame, however, should be distinguished from the individuals to which the decision is being applied. We call this the *candidate set*. For example, extending the zoning example above, the set of all locations (pixels) in the image that will be zoned is the candidate set. Finally, a *decision set* is that set of all individuals that are assigned a specific alternative from the decision frame. Thus, for example, all pixels assigned to the residential zone constitute one decision set. Similarly, those belonging to the commercial zone constitute another. Therefore, another definition of a decision would be to consider it the act of assigning an individual to a decision set. Alternatively, it can be thought of as a choice of alternative characterizations for an individual.

Criterion

A criterion is some basis for a decision that can be measured and evaluated. It is the evidence upon which an individual can be assigned to a decision set. Criteria can be of two kinds: factors and constraints, and can pertain either to attributes of the individual or to an entire decision set.

Factors

A factor is a criterion that enhances or detracts from the suitability of a specific alternative for the activity under consideration. It is therefore most commonly measured on a continuous scale. For example, a forestry company may determine that the steeper the slope, the more costly it is to transport wood. As a result, better areas for logging would be those on shallow slopes — the shallower the better. Factors are also known as *decision variables* in the mathematical programming literature (see Feiring, 1986) and *structural variables* in the linear goal programming literature (see Ignizio, 1985).

Constraints

A constraint serves to limit the alternatives under consideration. A good example of a constraint would be the exclusion from development of areas designated as wildlife reserves. Another might be the stipulation that no development may

proceed on slopes exceeding a 30% gradient. In many cases, constraints will be expressed in the form of a Boolean (logical) map: areas excluded from consideration being coded with a 0 and those open for consideration being coded with a 1. However, in some instances, the constraint will be expressed as some characteristic that the decision set must possess. For example, we might require that the total area of lands selected for development be no less than 5000 hectares, or that the decision set consist of a single contiguous area. Constraints such as these are often called *goals* (Ignizio, 1985) or *targets* (Rosenthal, 1985). Regardless, both forms of constraints have the same ultimate meaning—to limit the alternatives under consideration.

Although factors and constraints are commonly viewed as very different forms of criteria, material will be presented later in this chapter which shows these commonly held perspectives simply to be special cases of a continuum of variation in the degree to which criteria tradeoff in their influence over the solution, and in the degree of conservativeness in risk (or alternatively, pessimism or optimism) that one wishes to introduce in the decision strategy chosen. Thus, the very hard constraints illustrated above will be seen to be the *crisp* extremes of a more general class of *fuzzy* criteria that encompasses all of these possibilities. Indeed, it will be shown that continuous criteria (which we typically think of as factors) can serve as *soft* constraints when tradeoff is eliminated. In ecosystems analysis and land suitability assessment, this kind of factor is called a *limiting factor*, which is clearly a kind of constraint.

Decision Rule

The procedure by which criteria are selected and combined to arrive at a particular evaluation, and by which evaluations are compared and acted upon, is known as a decision rule. A decision rule might be as simple as a threshold applied to a single criterion (such as, all regions with slopes less than 35% will be zoned as suitable for development) or it may be as complex as one involving the comparison of several multi-criteria evaluations.

Decision rules typically contain procedures for combining criteria into a single composite index and a statement of how alternatives are to be compared using this index. For example, we might define a composite suitability map for agriculture based on a weighted linear combination of information on soils, slope, and distance from market. The rule might further state that the best 5000 hectares are to be selected. This could be achieved by choosing that set of raster cells, totaling 5000 hectares, in which the sum of suitabilities is maximized. It could equally be achieved by rank ordering the cells and taking enough of the highest ranked cells to produce a total of 5000 hectares. The former might be called a *choice function* (known as an *objective function* or *performance index* in the mathematical programming literature—see Diamond and Wright, 1989) while the latter might be called a *choice heuristic*.

Choice Function

Choice functions provide a mathematical means of comparing alternatives. Since they involve some form of optimization (such as maximizing or minimizing some measurable characteristic), they theoretically require that each alternative be evaluated in turn. However, in some instances, techniques do exist to limit the evaluation only to likely alternatives. For example, the Simplex Method in linear programming (see Feiring, 1986) is specifically designed to avoid unnecessary evaluations.

Choice Heuristic

Choice heuristics specify a procedure to be followed rather than a function to be evaluated. In some cases, they will produce an identical result to a choice function (such as the ranking example above), while in other cases they may simply provide a close approximation. Choice heuristics are commonly used because they are often simpler to understand and easier to implement.

Objective

Decision rules are structured in the context of a specific objective. The nature of that objective, and how it is viewed by the decision makers (i.e., their motives) will serve as a strong guiding force in the development of a specific decision rule. An objective is thus a *perspective* that serves to guide the structuring of decision rules.⁵⁵ For example, we may have the

stated objective to determine areas suitable for timber harvesting. However, our perspective may be one that tries to minimize the impact of harvesting on recreational uses in the area. The choice of criteria to be used and the weights to be assigned to them would thus be quite different from that of a group whose primary concern was profit maximization. Objectives are thus very much concerned with issues of motive and social perspective.

Evaluation

The actual process of applying the decision rule is called evaluation.

Multi-Criteria Evaluations

To meet a specific objective, it is frequently the case that several criteria will need to be evaluated. Such a procedure is called *Multi-Criteria Evaluation* (Voogd, 1983; Carver, 1991). Another term that is sometimes encountered for this is *modeling*. However, this term is avoided here since the manner in which the criteria are combined is very much influenced by the objective of the decision.

Multi-criteria evaluation (MCE) is most commonly achieved by one of two procedures. The first involves *Boolean overlay* whereby all criteria are reduced to logical statements of suitability and then combined by means of one or more logical operators such as intersection (AND) and union (OR). The second is known as *weighted linear combination* (WLC) wherein continuous criteria (*factors*) are standardized to a common numeric range, and then combined by means of a weighted average. The result is a continuous mapping of suitability that may then be masked by one or more Boolean *constraints* to accommodate qualitative criteria, and finally thresholded to yield a final decision.

While these two procedures are well established in GIS, they frequently lead to different results, as they make very different statements about how criteria should be evaluated. In the case of Boolean evaluation, a very extreme form of decision making is used. If the criteria are combined with a logical AND (the intersection operator), a location must meet *every* criterion for it to be included in the decision set. If even a single criterion fails to be met, the location will be excluded. Such a procedure is essentially risk-averse, and selects locations based on the most cautious strategy possible—a location succeeds in being chosen only if its worst quality (and therefore all qualities) passes the test. On the other hand, if a logical OR (union) is used, the opposite applies—a location will be included in the decision set even if only a single criterion passes the test. This is thus a very gambling strategy, with (presumably) substantial risk involved.

Now compare these strategies with that represented by weighted linear combination (WLC). With WLC, criteria are permitted to tradeoff their qualities. A very poor quality can be compensated for by having a number of very favorable qualities. This operator represents neither an AND nor an OR—it lies somewhere in between these extremes. It is neither risk averse nor risk taking.

For reasons that have largely to do with the ease with which these approaches can be implemented, the Boolean strategy dominates vector approaches to MCE, while WLC dominates solutions in raster systems. But clearly neither is better—they simply represent two very different outlooks on the decision process—what can be called a decision strategy. IDRISI also includes a third option for Multi-Criteria Evaluation, known as an ordered weighted average (OWA) (Eastman and Jiang, 1996). This method offers a complete spectrum of decision strategies along the primary dimensions of degree of tradeoff involved and degree of risk in the solution.

Multi-Objective Evaluations

While many decisions we make are prompted by a single objective, it also happens that we need to make decisions that satisfy several objectives. A Multi-Objective problem is encountered whenever we have two candidate sets (i.e., sets of entities) that share members. These objectives may be complementary or conflicting in nature (Carver, 1991: 322).

55. It is important to note here that we are using a somewhat broader definition of the term objective than would be found in the goal programming literature (see Ignizio, 1985). In goal programming, the term *objective* is synonymous with the term *objective function* in mathematical programming and *choice function* used here.

Complementary Objectives

With complementary or non-conflicting objectives, land areas may satisfy more than one objective, i.e., an individual pixel can belong to more than one decision set. Desirable areas will thus be those which serve these objectives together in some specified manner. For example, we might wish to allocate a certain amount of land for combined recreation and wildlife preservation uses. Optimal areas would thus be those that satisfy both of these objectives to the maximum degree possible.

Conflicting Objectives

With conflicting objectives, competition occurs for the available land since it can be used for one objective or the other, but not both. For example, we may need to resolve the problem of allocating land for timber harvesting and wildlife preservation. Clearly the two cannot coexist. Exactly how they compete, and on what basis one will win out over the other, will depend upon the nature of the decision rule that is developed.

In cases of complementary objectives, multi-objective decisions can often be solved through a *hierarchical* extension of the multi-criteria evaluation process. For example, we might assign a weight to each of the objectives and use these, along with the suitability maps developed for each, to combine them into a single suitability map. This would indicate the degree to which areas meet all of the objectives considered (see Voogd, 1983). However, with conflicting objectives the procedure is more involved.

With conflicting objectives, it is sometimes possible to rank order the objectives and reach a *prioritized* solution (Rosenthal, 1985). In these cases, the needs of higher ranked objectives are satisfied before those of lower ranked objectives are dealt with. However, this is often not possible, and the most common solution for conflicting objectives is the development of a *compromise* solution. Undoubtedly the most commonly employed techniques for resolving conflicting objectives are those involving optimization of a choice function such as mathematical programming (Fiering, 1986) or goal programming (Ignizio, 1985). In both, the concern is to develop an allocation of the land that maximizes or minimizes an objective function subject to a series of constraints.

Uncertainty and Risk

Clearly, information is vital to the process of decision making. However, we rarely have perfect information. This leads to uncertainty, of which two sources can be identified: *database* and *decision rule uncertainty*.

Database Uncertainty

Database uncertainty is that which resides in our assessments of the criteria which are enumerated in the decision rule. Measurement error is the primary source of such uncertainty. For example, a slope of 35% may represent an important threshold. However, because of the manner in which slopes are determined, there may be some uncertainty about whether a slope that was measured as 34% really is 34%. While we may have considerable confidence that it is most likely *around* 34%, we may also need to admit that there is some finite probability that it is as high as 36%. Our expression of database uncertainty is likely to rely upon probability theory.

Decision Rule Uncertainty

Decision rule uncertainty is that which arises from the manner in which criteria are combined and evaluated to reach a decision. A very simple form of decision rule uncertainty is that which relates to parameters or thresholds used in the decision rule. A more complex issue is that which relates to the very structure of the decision rule itself. This is sometimes called *specification error* (Alonso, 1968), because of uncertainties that arise in specifying the relationship between criteria (as a model) such that adequate evidence is available for the proper evaluation of the hypotheses under investigation.

Decision Rule Uncertainty and Direct Evidence: Fuzzy versus Crisp Sets

A key issue in decision rule uncertainty is that of establishing the relationship between the evidence and the decision set. In most cases, we are able to establish a direct relationship between the two, in the sense that we can define the decision

set by measurable attributes that its members should possess. In some cases these attributes are crisp and unambiguous. For example, we might define those sewer lines in need of replacement as those of a particular material and age. However, quite frequently the attributes they possess are fuzzy rather than crisp. For example, we might define suitable areas for timber logging as those forested areas that have gentle slopes and are near to a road. What is a gentle slope? If we specify that a slope is gentle if it has a gradient of less than 5%, does this mean that a slope of 5.0001% is not gentle? Clearly there is no sharp boundary here. Such classes are called *fuzzy* sets (Zadeh, 1965) and are typically defined by a set membership function. Thus we might decide that any slope less than 2% is unquestionably gentle, and that any slope greater than 10% is unquestionably steep, but that membership in the gentle set gradually falls from 1.0 at a 2% gradient to 0.0 at a 10% gradient. A slope of 5% might then be considered to have a membership value of only 0.7 in the set called "gentle." A similar group of considerations also surround the concept of being "near" to a road.

Fuzzy sets are extremely common in the decision problems faced with GIS. They represent a form of uncertainty, but it is not measurement uncertainty. The issue of what constitutes a shallow slope is over and above the issue of whether a measured slope is actually what is recorded. It is a form of uncertainty that lies at the very heart of the concept of factors previously developed. *The continuous factors of multi-criteria decision making are thus fuzzy set membership functions, whereas Boolean constraints are crisp set membership functions.* But it should be recognized that the terms factor and constraint imply more than fuzzy or crisp membership functions. Rather, these terms give some meaning also to the manner in which they are aggregated with other information.

Decision Rule Uncertainty and Indirect Evidence: Bayes versus Dempster Shafer

Not all evidence can be directly related to the decision set. In some instances we only have an indirect relationship between the two. In this case, we may set up what can be called a *belief function* of the degree to which evidence implies the membership in the decision set. Two important tools for accomplishing this are Bayesian Probability Theory and Dempster-Shafer Theory of Evidence. These will be dealt with at more length later in this chapter in Part B on Uncertainty Management.

Decision Risk

Decision Risk may be understood as the likelihood that the decision made will be wrong.⁵⁶ Risk arises as a result of uncertainty, and its assessment thus requires a combination of uncertainty estimates from the various sources involved (database and decision rule uncertainty) and procedures, such as Bayesian Probability theory, through which it can be determined. Again, this topic will be discussed more thoroughly in Part B of this chapter.

56. Note that different fields of science define risk in different ways. For example, some disciplines modify the definition given here to include a measure of the cost or consequences of a wrong decision (thus allowing for a direct relationship to cost/benefit analysis). The procedures developed in IDRISI do not preclude such an extension. We have tried here to present a fairly simple perspective that can be used as a building block for more specific interpretations.

A Typology of Decisions

Given these definitions, it is possible to set out a very broad typology of decisions as illustrated in Figure 12-1.

	Single Criterion	Multi-Criteria
Single Objective		
Multi-Objective		

Figure 12-1

Decisions may be characterized as *single-* or *multi-objective* in nature, based on either a *single criterion* or *multiple criteria*. While one is occasionally concerned with single criterion problems, most problems approached with a GIS are multi-criteria in nature. For example, we might wish to identify areas of concern for soil erosion on the basis of slope, land use, soil type and the like. In these instances, our concern lies with how to combine these criteria to arrive at a composite decision. As a consequence, the first major area of concern in GIS with regard to Decision Theory is Multi-Criteria Evaluation.

Most commonly, we deal with decision problems of this nature from a single perspective. However, in many instances, the problem is actually multi-objective in nature (Diamond and Wright, 1988). Multi-objective problems arise whenever the same resources belong to more than one candidate set. Thus, for example, a paper company might include all forest areas in its candidate set for consideration of logging areas, while a conservation group may include forest areas in a larger candidate set of natural areas to be protected. Any attempt, therefore, to reconcile their potential claims to this common set of resources presents a multi-objective decision problem.

Despite the prevalence of multi-objective problems, current GIS software is severely lacking in techniques to deal with this kind of decision. To date, most examples of multi-objective decision procedures in the literature have dealt with the problem through the use of linear programming optimization (e.g., Janssen and Rietveld 1990; Carver, 1991; Campbell et al., 1992; Wright et. al., 1983). However, in most cases, these have been treated as choice problems between a limited number (e.g., less than 20) of candidate sites previously isolated in a vector system. The volume of data associated with raster applications (where each pixel is a choice alternative) clearly overwhelms the computational capabilities of today's computing environment. In addition, the terminology and procedures of linear programming are unknown to most decision makers and are complex and unintuitive by nature. As a consequence, the second major area of Decision Theory of importance to GIS is Multi-Objective Land Allocation. Here, the focus will be on a simple decision heuristic appropriate to the special needs of raster GIS.

Multi-Criteria Decision Making in GIS

As indicated earlier, the primary issue in Multi-Criteria Evaluation is concerned with how to combine the information from several criteria to form a single index of evaluation. In the case of Boolean criteria (constraints), the solution usually lies in the union (logical OR) or intersection (logical AND) of conditions. However, for continuous factors, a weighted linear combination (Voogd, 1983: 120) is most commonly used. With a weighted linear combination, factors are combined by applying a weight to each followed by a summation of the results to yield a suitability map, i.e.:

$$S = \sum w_i x_i \quad \text{where} \quad S = \text{suitability}$$
$$w_i = \text{weight of factor } i$$
$$x_i = \text{criterion score of factor } i$$

This procedure is not unfamiliar in GIS and has a form very similar to the nature of a regression equation. In cases where

Boolean constraints also apply, the procedure can be modified by multiplying the suitability calculated from the factors by the product of the constraints, i.e.:

$$S = \sum w_i x_i * \prod c_j \quad \text{where} \quad c_j = \text{criterion score of constraint } j \\ \prod = \text{product}$$

All GIS software systems provide the basic tools for evaluating such a model. In addition, in IDRISI, a special module named MCE has been developed to facilitate this process. However, the MCE module also offers a special procedure called an Ordered Weighted Average that greatly extends the decision strategy options available. The procedure will be discussed more fully in the section on Evaluation below. For now, however, the primary issues relate to the standardization of criterion scores and the development of the weights.

Criterion Scores

Because of the different scales upon which criteria are measured, it is necessary that factors be standardized⁵⁷ before combination using the formulas above, and that they be transformed, if necessary, such that all factors maps are positively correlated with suitability.⁵⁸ Voogd (1983: 77-84) reviews a variety of procedures for standardization, typically using the minimum and maximum values as scaling points. The simplest is a linear scaling such as:

$$x_i = (R_i - R_{\min}) / (R_{\max} - R_{\min}) * \text{standardized_range} \quad \text{where } R = \text{raw score}$$

However, if we recognize that continuous factors are really fuzzy sets, we easily recognize this as just one of many possible set membership functions. In IDRISI, the module named FUZZY is provided for the standardization of factors using a whole range of fuzzy set membership functions. The module is quick and easy to use, and provides the option of standardizing factors to either a 0-1 real number scale or a 0-255 byte scale. This latter option is recommended because the MCE module has been optimized for speed using a 0-255 level standardization. Importantly, the higher value of the standardized scale must represent the case of being *more* likely to belong to the decision set.

A critical issue in the standardization of factors is the choice of the end points at which set membership reaches either 0.0 or 1.0 (or 0 and 255). Our own research has suggested that blindly using a linear scaling (or indeed any other scaling) between the minimum and maximum values of the image is ill advised. In setting these critical points for the set membership function, it is important to consider their inherent meaning. Thus, for example, if we feel that industrial development should be placed as far away from a nature reserve as possible, it would be dangerous to implement this without careful consideration. Taken literally, if the map were to cover a range of perhaps 100 km from the reserve, then the farthest point away from the reserve would be given a value of 1.0 (or 255 for a byte scaling). Using a linear function, then, a location 5 km from the reserve would have a standardized value of only 0.05 (13 for a byte scaling). And yet it may be that the primary issue was noise and minor disturbance from local citizens, for which a distance of only 5 kilometers would have been equally as good as being 100 km away. Thus the standardized score should really have been 1.0 (255). If an MCE were undertaken using the blind linear scaling, locations in the range of a few 10s of km would have been severely devalued when in fact they might have been quite good. In this case, the recommended critical points for the scaling should have been 0 and 5 km. In developing standardized factors using FUZZY, then, careful consideration should be given to the inherent meaning of the end points chosen.

Criterion Weights

A wide variety of techniques exist for the development of weights. In very simple cases, assigning criteria weights may be accomplished by dividing 1.0 among the criteria. (It is sometimes useful for people to think about "spending" one dollar, for example, among the criteria). However, when the number of criteria is more than a few, and the considerations are

57. In using the term *standardization*, we have adopted the terminology of Voogd (1983), even though this process should more properly be called *normalization*.

58. Thus, for example, if locations near to a road were more advantageous for industrial siting than those far away, a distance map would need to be transformed into one expressing proximity.

many, it becomes quite difficult to make weight evaluations on the set as a whole. Breaking the information down into simple pairwise comparisons in which only two criteria need be considered at a time can greatly facilitate the weighting process, and will likely produce a more robust set of criteria weights. A pairwise comparison method has the added advantages of providing an organized structure for group discussions, and helping the decision making group hone in on areas of agreement and disagreement in setting criterion weights.

The technique described here and implemented in IDRISI is that of pairwise comparisons developed by Saaty (1977) in the context of a decision making process known as the Analytical Hierarchy Process (AHP). The first introduction of this technique to a GIS application was that of Rao et. al. (1991), although the procedure was developed outside the GIS software using a variety of analytical resources.

In the procedure for Multi-Criteria Evaluation using a weighted linear combination outlined above, it is necessary that the weights sum to one. In Saaty's technique, weights of this nature can be derived by taking the principal eigenvector of a square reciprocal matrix of pairwise comparisons between the criteria. The comparisons concern the relative importance of the two criteria involved in determining suitability for the stated objective. Ratings are provided on a 9-point continuous scale (Figure 12-2). For example, if one felt that proximity to roads was very strongly more important than slope gradient in determining suitability for industrial siting, one would enter a 7 on this scale. If the inverse were the case (slope gradient was very strongly more important than proximity to roads), one would enter 1/7.

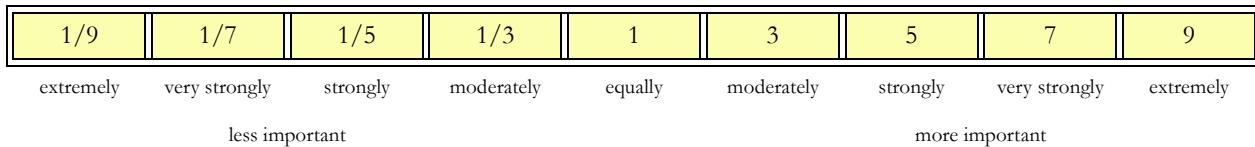


Figure 12-2 The Continuous Rating Scale

In developing the weights, an individual or group compares every possible pairing and enters the ratings into a pairwise comparison matrix (Figure 12-3). Since the matrix is symmetrical, only the lower triangular half actually needs to be filled in. The remaining cells are then simply the reciprocals of the lower triangular half (for example, since the rating of slope gradient relative to town proximity is 4, the rating of town proximity relative to slope gradient will be 1/4). Note that where empirical evidence exists about the relative efficacy of a pair of factors, this evidence can also be used.

Rating of the Row Factor Relative to the Column Factor					
	Road Proximity	Town Proximity	Slope Gradient	Small Holder Settlement	Distance from Park
Road Proximity	1				
Town Proximity	1/3	1			
Slope Gradient	1	4	1		
Small Holder Set.	1/7	2	1/7	1	
Distance from Park	1/2	2	1/2	4	1

Figure 12-3 An example of a pairwise comparison matrix for assessing the comparative importance of five factors to industrial development suitability.

The procedure then requires that the principal eigenvector of the pairwise comparison matrix be computed to produce a *best fit* set of weights (Figure 12-4). If no procedure is available to do this, a good approximation to this result can be achieved by calculating the weights with each column and then averaging over all columns. For example, if we take the first column of figures, they sum to 2.98. Dividing each of the entries in the first column by 2.98 yields weights of 0.34, 0.11, 0.34, 0.05, and 0.17 (compare to the values in Figure 12-4). Repeating this for each column and averaging the weights over the columns usually gives a good approximation to the values calculated by the principal eigenvector. In the case of IDRISI, however, a special module named WEIGHT has been developed to calculate the principal eigenvector directly. Note that these weights will sum to one, as is required by the weighted linear combination procedure.

RoadProx	0.33
TownProx	0.08
Slope	0.34
SmalHold	0.07
ParkDist	0.18
Consistency Ratio 0.06	

Figure 12-4 Weights derived by calculating the principal eigenvector of the pairwise comparison matrix.

Since the complete pairwise comparison matrix contains multiple paths by which the relative importance of criteria can be assessed, it is also possible to determine the degree of consistency that has been used in developing the ratings. Saaty (1977) indicates the procedure by which an index of consistency, known as a *consistency ratio*, can be produced (Figure 12-4). The consistency ratio (CR) indicates the probability that the matrix ratings were randomly generated. Saaty indicates that matrices with CR ratings greater than 0.10 should be re-evaluated. In addition to the overall consistency ratio, it is also possible to analyze the matrix to determine where the inconsistencies arise. This has also been developed as part of the WEIGHT module in IDRISI.

Evaluation

Once the criteria maps (factors and constraints) have been developed, an evaluation (or aggregation) stage is undertaken to combine the information from the various factors and constraints. The MCE module offers three logics for the evaluation/aggregation of multiple criteria: boolean intersection, weighted linear combination (WLC), and the ordered weighted average (OWA).

MCE and Boolean Intersection

The most simplistic type of aggregation is the boolean intersection or logical AND. This method is used only when factor maps have been strictly classified into boolean suitable/unsuitable images with values 1 and 0. The evaluation is simply the multiplication of all the images.

MCE and Weighted Linear Combination

The derivation of criterion (or factor) weights is described above. The weighted linear combination (WLC) aggregation method multiplies each standardized factor map (i.e., each raster cell within each map) by its factor weight and then sums the results. Since the set of factor weights for an evaluation must sum to one, the resulting suitability map will have the same range of values as the standardized factor maps that were used. This result is then multiplied by each of the constraints in turn to "mask out" unsuitable areas. All these steps could be done using either a combination of SCALAR and OVERLAY, or by using the Image Calculator. However, the module MCE is designed to facilitate the process.

The WLC option in the MCE module requires that you specify the number of criteria (both constraints and factors), their names, and the weights to be applied to the factors. All factors must be standardized to a byte (0-255) range. (If you have factors in real format, then use one of the options other than MCE mentioned above.) The output is a *suitability* map masked by the specified constraints.

MCE and the Ordered Weighted Average

In its use and implementation, the ordered weighted average approach is not unlike WLC. The dialog box for the OWA option is almost identical to that of WLC, with the exception that a second set of weights appears. This second set of weights, the *order weights*, controls the manner in which the weighted factors are aggregated (Eastman and Jiang, 1996; Yager, 1988). Indeed, WLC turns out to be just one variant of the OWA technique. To introduce the OWA technique, let's first review WLC in terms of two new concepts: tradeoff and risk.

Tradeoff

Factor weights are weights that apply to specific factors, i.e., all the pixels of a particular factor image receive the same factor weight. They indicate the relative degree of importance each factor plays in determining the suitability for an objective. In the case of WLC the weight given to each factor also determines how it will tradeoff relative to other factors. For example, a factor with a high factor weight can tradeoff or compensate for poor scores on other factors, even if the unweighted suitability score for that highly-weighted factor is not particularly good. In contrast, a factor with a high suitability score but a small factor weight can only weakly compensate for poor scores on other factors. The factor weights determine how factors tradeoff but, as described below, order weights determine the *overall* level of tradeoff allowed.

Risk

Boolean approaches are extreme functions that result either in very risk-averse solutions when the AND operator is used or in risk-taking solutions when the OR operator is used.⁵⁹ In the former, a high aggregate suitability score for a given location (pixel) is only possible if *all* factors have high scores. In the latter, a high score in any factor will yield a high aggregate score, even if all the other factors have very low scores. The AND operation may be usefully described as the *minimum*, since the minimum score for any pixel determines the final aggregate score. Similarly, the OR operation may be called the *maximum*, since the maximum score for any pixel determines the final aggregate score. The AND solution is *risk-averse* because we can be sure that the score for every factor is at least as good as the final aggregate score. The OR solution is *risk-taking* because the final aggregate score only tells us about the suitability score for the single most suitable factor.

The WLC approach is an averaging technique that softens the hard decisions of the boolean approach and avoids the extremes. In fact, given a continuum of risk from minimum to maximum, WLC falls exactly in the middle; it is neither risk-averse nor risk-taking.

Order Weights, Tradeoff and Risk

The use of order weights allows for aggregation solutions that fall anywhere along the risk continuum between AND and OR. Order weights are quite different from factor weights. They do not apply to any specific factor. Rather, they are applied on a pixel-by-pixel basis to factor scores as determined by their rank ordering across factors at each location (pixel). Order weight 1 is assigned to the lowest-ranked factor for that pixel (i.e., the factor with the lowest score), order weight 2 to the next higher-ranked factor for that pixel, and so forth. Thus, it is possible that a single order weight could be applied to pixels from any of the various factors depending upon their relative rank order.

To examine how order weights alter MCE results by controlling levels of tradeoff and risk, let us consider the case where factor weights are equal for three factors A, B, and C. (Holding factor weights equal will make clearer the effect of the

59. The logic of the Boolean AND and OR is implemented with fuzzy sets as the minimum and maximum. Thus, as we are considering continuous factor scores rather than boolean 0-1 images in this discussion, the logical AND is evaluated as the minimum value for a pixel across all factors and the logical OR is evaluated as the maximum value for a pixel across all factors.

order weights.) Consider a single pixel with factor scores A= 187, B=174, and C=201. The factor weights for each of the factors is 0.33. When ranked from minimum value to maximum value, the order of these factors for this pixel is [B,A,C]. For this pixel, factor B will be assigned order weight 1, A order weight 2 and C order weight 3.

Below is a table with thirteen sets of order weights that have been applied to this set of factor scores [174,187,201]. Each set yields a different MCE result even though the factor scores and the factor weights are the same in each case.

Order Weights			Result
Min (1)	(2)	Max (3)	
1.00	0.00	0.00	174
0.90	0.10	0.00	175
0.80	0.20	0.00	177
0.70	0.20	0.10	179
0.50	0.30	0.20	183
0.40	0.30	0.30	186
0.33	0.33	0.33	187
0.30	0.30	0.40	189
0.20	0.30	0.50	191
0.10	0.20	0.70	196
0.00	0.20	0.80	198
0.00	0.10	0.90	200
0.00	0.00	1.00	201

The first set of order weights in the table is [1, 0, 0]. The weight of factor B (the factor with the minimum value in the set [B, A, C]) will receive all possible weight while factors A and C will be given no weight at all. Such a set of order weights make irrelevant the factor weights. Indeed, the order weights have altered the evaluation such that no tradeoff is possible. As can be seen in the table, this has the effect of applying a minimum operator to the factors, thus producing the traditional intersection operator (AND) of fuzzy sets.

Similarly, the last set of order weights [0, 0, 1] has the effect of a maximum operator, the traditional union operator (OR) of fuzzy sets. Again, there is no tradeoff and the factor weights are not employed.

Another important example from the table is where the order weights are equal, [.33, .33, .33]. Here all ranked positions get the same weight; this makes tradeoff fully possible and locates the analysis exactly midway between AND and OR. Equal order weights produce the same result as WLC.

In all three cases, the order weights have determined not only the level of tradeoff but have situated the analysis on a continuum from (risk-averse, minimum, AND) to (risk-taking, maximum, OR).

As seen in the table, the order weights in the OWA option of MCE are not restricted to these three possibilities, but instead can be assigned any combination of values that sum to 1.0. Any assignment of order weights results in a decision rule that falls somewhere in a triangular decision strategy space that is defined by the dimensions of risk and tradeoff as shown in Figure 12-5.

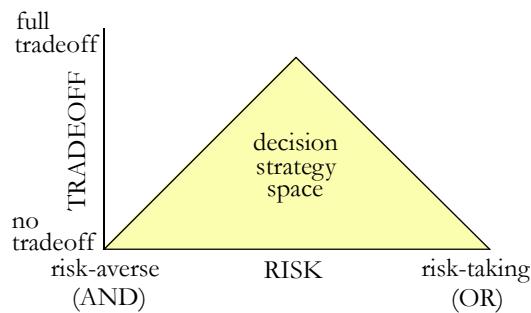


Figure 12-5

Whether most of the order weight is assigned to the left, right or center of the order weights determines the position in the risk dimension. The logical AND operator is the most risk-averse combination and the logical OR is the most risk-taking combination. When order weights are predominantly assigned to the lower-ranked factors, there is greater risk aversion (more of an AND approach). When order weights are more dominant for the higher-ranked factors, there is greater risk taking (more of an OR approach). As discussed above, equal order weights yield a solution at the middle of the risk axis.

The degree of tradeoff is governed by the relative distribution of order weights between the ranked factors. Thus, if the sum of the order weights is evenly spread between the factors, there is strong tradeoff, whereas if all the weight is assigned to a single factor rank, there is no tradeoff. (It may be helpful to think of this in terms of a graph of the order weights, with rank order on the x axis and the order weight value on the y axis. If the graph has a sharp peak, there is little tradeoff. If the graph is relatively flat, there is strong tradeoff.)

Thus, as seen from the table, the order weights of [0.5 0.3 0.2] would indicate a strong (but not perfect) degree of risk aversion (because weights are skewed to the risk-averse side of the risk axis) and some degree of tradeoff (because the weights are spread out over all three ranks). Weights of [0 1 0], however, would imply neither risk aversion nor acceptance (exactly in the middle of the risk axis), and no tradeoff (because all the weight is assigned to a single rank).

The OWA method is particularly interesting because it provides this continuum of aggregation procedures. At one extreme (the logical AND), each criterion is considered necessary (but not sufficient on its own) for inclusion in the decision set. At the other extreme (the logical OR), each criterion is sufficient on its own to support inclusion in the decision set without modification by other factors. The position of the weighted linear combination operator halfway between these extremes is therefore not surprising. This operator considers criteria as neither necessary nor sufficient—strong support for inclusion in the decision set by one criterion can be equally balanced by correspondingly low support by another. It thus offers full tradeoff.

Using OWA

Given this introduction, it is worth considering how one would use the OWA option of MCE. Some guidelines are as follows:

1. Divide your criteria into three groups: hard constraints, factors that should not tradeoff, and factors that should tradeoff. For example, factors with monetary implications typically tradeoff, while those associated with some safety concern typically do not.
2. If you find that you have factors that both tradeoff and do not tradeoff, separate their consideration into two stages of analysis. In the first, aggregate the factors that tradeoff using the OWA option. You can govern the degree of tradeoff by

manipulating the order weights. Then use the result of the first stage as a new factor that is included in the analysis of those that do not tradeoff.

3. If you run an analysis with absolutely no tradeoff, the factor weights have no real meaning and can be set to any value.

Completing the Evaluation

Once a suitability map has been prepared, it is common to decide, as a final step, which cells should belong to the set that meets a particular land allocation area target (the decision set). For example, having developed a map of suitability for industrial development, we may then wish to determine which areas constitute the *best* 5000 hectares that may be allocated. Oddly, this is an area where most raster systems have difficulty achieving an exact solution. One solution would be to use a choice function where that set of cells is chosen which maximizes the sum of suitabilities. However, the number of combinations that would need to be evaluated is prohibitive in a raster GIS. As a result, we chose to use a simple choice heuristic—to rank order the cells and choose as many of the highest ranks as will be required to meet the area target. In IDRISI, a module named RANK is available that allows a rapid ranking of cells within an image. In addition, it allows the use of a second image to resolve the ranks of ties. The ranked map can then be reclassified to extract the highest ranks to meet the area goal.

Multi-Objective Decision Making in GIS

Multi-objective decisions are so common in environmental management that it is surprising that specific tools to address them have not yet been further developed within GIS. The few examples one finds in the literature tend to concentrate on the use of mathematical programming tools outside the GIS, or are restricted to cases of complementary objectives.

Complementary Objectives

As indicated earlier, the case of complementary objectives can be dealt with quite simply by means of a hierarchical extension of the multi-criteria evaluation process (e.g., Carver, 1991). Here a set of suitability maps, each derived in the context of a specific objective, serve as the factors for a new evaluation in which the objectives are themselves weighted and combined by linear summation. Since the logic which underlies this is multiple use, it also makes sense to multiply the result by all constraints associated with the component objectives.

Conflicting Objectives

With conflicting objectives, land can be allocated to one objective but not more than one (although hybrid models might combine complementary and conflicting objectives). As was indicated earlier, one possible solution lies with a prioritization of objectives (Rosenthal, 1985). After the objectives have been ordered according to priority, the needs of higher priority objectives are satisfied (through rank ordering of cells and reclassification to meet areal goals) before those of lower priority ones. This is done by successively satisfying the needs of higher priority objectives and then removing (as a new constraint) areas taken by that objective from consideration by all remaining objectives. A prioritized solution is easily achieved with the use of the RANK, RECLASS and OVERLAY modules in IDRISI. However, instances are rare where a *prioritized* solution makes sense. More often a *compromise* solution is required.

As noted earlier, compromise solutions to the multi-objective problem have most commonly been approached through the use of mathematical programming tools outside GIS (e.g., Diamond and Wright, 1988; Janssen and Rietveld, 1990; Campbell, et. al., 1992). Mathematical programming solutions (such as linear or integer programming) can work quite well in instances where only a small number of alternatives are being addressed. However, in the case of raster GIS, the massive data sets involved will typically exceed present-day computing power. In addition, the concepts and methodology of linear and integer programming are not particularly approachable to a broad range of decision makers. As a result, we have sought a solution to the problem of multi-objective land allocation under conditions of conflicting objectives such that large raster datasets may be handled using procedures that have an immediate intuitive appeal.

The procedure we have developed is an extension of the decision heuristic used for the allocation of land with single objective problems. This is best illustrated by the diagram in Figure 12-6a. Each of the suitability maps may be thought of as an axis in a multi-dimensional space. Here we consider only two objectives for purposes of simple explanation. However, any number of objectives can be used.

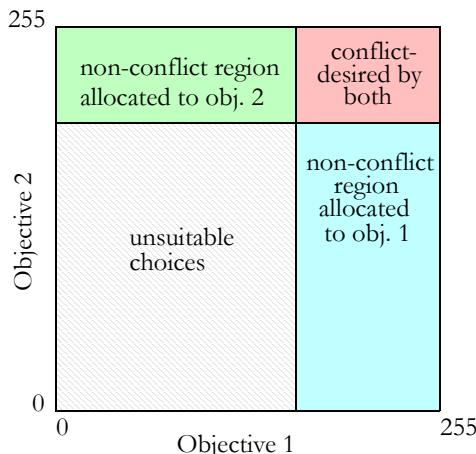


Figure 12-6a

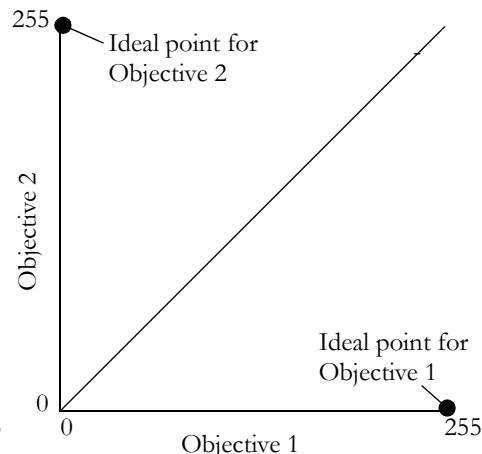


Figure 12-6b

Every raster cell in the image can be located within this decision space according to its suitability level on each of the objectives. To find the best x hectares of land for Objective 1, we simply need to move a decision line down from the top (i.e., far right) of the Objective 1 suitability axis until enough of the best raster cells are captured to meet our area target. We can do the same with the Objective 2 suitability axis to capture the best y hectares of land for it. As can be seen in Figure 12-6a, this partitions the decision space into four regions—areas best for Objective 1 and not suitable for Objective 2, areas best for Objective 2 and not suitable for Objective 1, areas not suitable for either, and areas judged best for both. The latter represents areas of conflict.

To resolve these areas of conflict, a simple partitioning of the affected cells is used. As can be seen in Figure 12-6b, the decision space can also be partitioned into two further regions: those closer to the *ideal point* for Objective 1 and those closer to that for Objective 2. The ideal point represents the best possible case—a cell that is maximally suited for one objective and minimally suited for anything else. To resolve the conflict zone, the line that divides these two regions is overlaid onto it and cells are then allocated to their closest ideal point. Since the conflict region will be divided between the objectives, both objectives will be short on achieving their area goals. As a result, the process will be repeated with the decision lines being lowered for both objectives to gain more territory. The process of resolving conflicts and lowering the decision lines is iteratively repeated until the exact area targets are achieved.

It should be noted that a 45-degree line between a pair of objectives assumes that they are given equal weight in the resolution of conflicts. However, unequal weighting can be given. Unequal weighting has the effect of changing the angle of this dividing line. In fact, the tangent of that angle is equal to the ratio of the weights assigned to those objectives.

It should also be noted that just as it was necessary to standardize criteria for multi-criteria evaluation, it is also required for multi-objective evaluation. The process involves a matching of the histograms for the two suitability maps. In cases where the distributions are normal, conversion to standard scores (using the module named STANDARD) would seem appropriate. However, in many cases, the distributions are not normal. In these cases, the matching of histograms is most easily achieved by a non-parametric technique known as histogram equalization. This is a standard option in many image processing systems such as IDRISI. However, it is also the case that the ranked suitability maps produced by the RANK module are also histogram equalized (i.e., a histogram of a rank map is uniform). This is fortuitous since the logic outlined in Figure 12-6a is best achieved by reclassification of ranked suitability maps.

As a result of the above considerations, the module named MOLA (Multi-Objective Land Allocation) was developed to undertake the compromise solution to the multi-objective problem. MOLA requires the names of the objectives and their relative weights, the names of the ranked suitability maps for each, and the areas that should be allocated to each. It then iteratively reclassifies the ranked suitability maps to perform a first stage allocation, checks for conflicts, and then allocates conflicts based on a minimum-distance-to-ideal-point rule using the weighted ranks.

A Worked Example

To illustrate these multi-criteria/multi-objective procedures, we will consider the following example of developing a zoning map to regulate expansion of the carpet industry (one of the largest and most rapidly growing industries in Nepal) within agricultural areas of the Kathmandu Valley of Nepal. The problem is to zone 1500 hectares of current agricultural land outside the ring road of Kathmandu for further expansion of the carpet industry. In addition, 6000 hectares will be zoned for special protection of agriculture. The problem clearly falls into the realm of multi-objective/multi-criteria decision problems. In this case, we have two objectives: to protect lands that are best for agriculture, and at the same time find other lands that are best suited for the carpet industry. Since land can be allocated to only one of these uses at any one time, the objectives must be viewed as conflicting (i.e., they may potentially compete for the same lands). Furthermore, the evaluation of each of these objectives can be seen to require multiple criteria.

In the illustration that follows, a solution to the multi-objective/multi-criteria problem is presented as developed with a group of Nepalese government officials as part of an advanced seminar in GIS.⁶⁰ While the scenario was developed purely for the purpose of demonstrating the techniques used, and while the result does not represent an actual policy decision, it is one that incorporates substantial field work and the perspectives of knowledgeable decision makers. The procedure follows a logic in which each of the two objectives is first dealt with as a separate multi-criteria evaluation problem. The result consists of two separate suitability maps (one for each objective) which are then compared to arrive at a single solution that balances the needs of the two competing objectives.

1. Solving the Single Objective Multi-Criteria Evaluations

1.1 Establishing the Criteria: Factors and Constraints

The decision making group identified five factors as being relevant to the siting of the carpet industry: proximity to water (for use in dyeing and the washing of carpets), proximity to roads (to minimize road construction costs), proximity to power, proximity to the market, and slope gradient. For agriculture they identified three of the same factors: proximity to water (for irrigation), proximity to market, and slope gradient, as well as a fourth factor, soil capability. In both cases, they identified the same constraints: the allocation would be limited to areas outside the ring road surrounding Kathmandu, land currently in some form of agricultural use, and slope gradients less than 100%. For factor images, distance to water, road and power lines was calculated based on the physical distance, and the proximity to market was developed as a cost distance surface (accounting for variable road class frictions).

1.2 Standardizing the Factors

Each of the constraints was developed as a Boolean map while the factors were standardized using the module FUZZY so that the results represent fuzzy membership in the decision set. For example, for the carpet industry allocation, the proximity to water factor map was standardized using a *sigmoidal monotonically decreasing* fuzzy membership function with control points at 10 and 700 meters. Thus, areas less than 10 meters were assigned a set membership of 255 (on a scale from 0-255), those between 10 and 700 meters were assigned a value which progressively decreased from 255 to 0 in the

60. The seminar was hosted by UNITAR at the International Center for Integrated Mountain Development (ICIMOD) in Nepal, September 28-October 2, 1992.

manner of an s-shaped curve, and those beyond 700 meters to a river were considered to be too far away (i.e., they were assigned a value of 0). Figure 12-7 illustrates the standardized results of all five factors and the constraints for the carpet industry allocation.

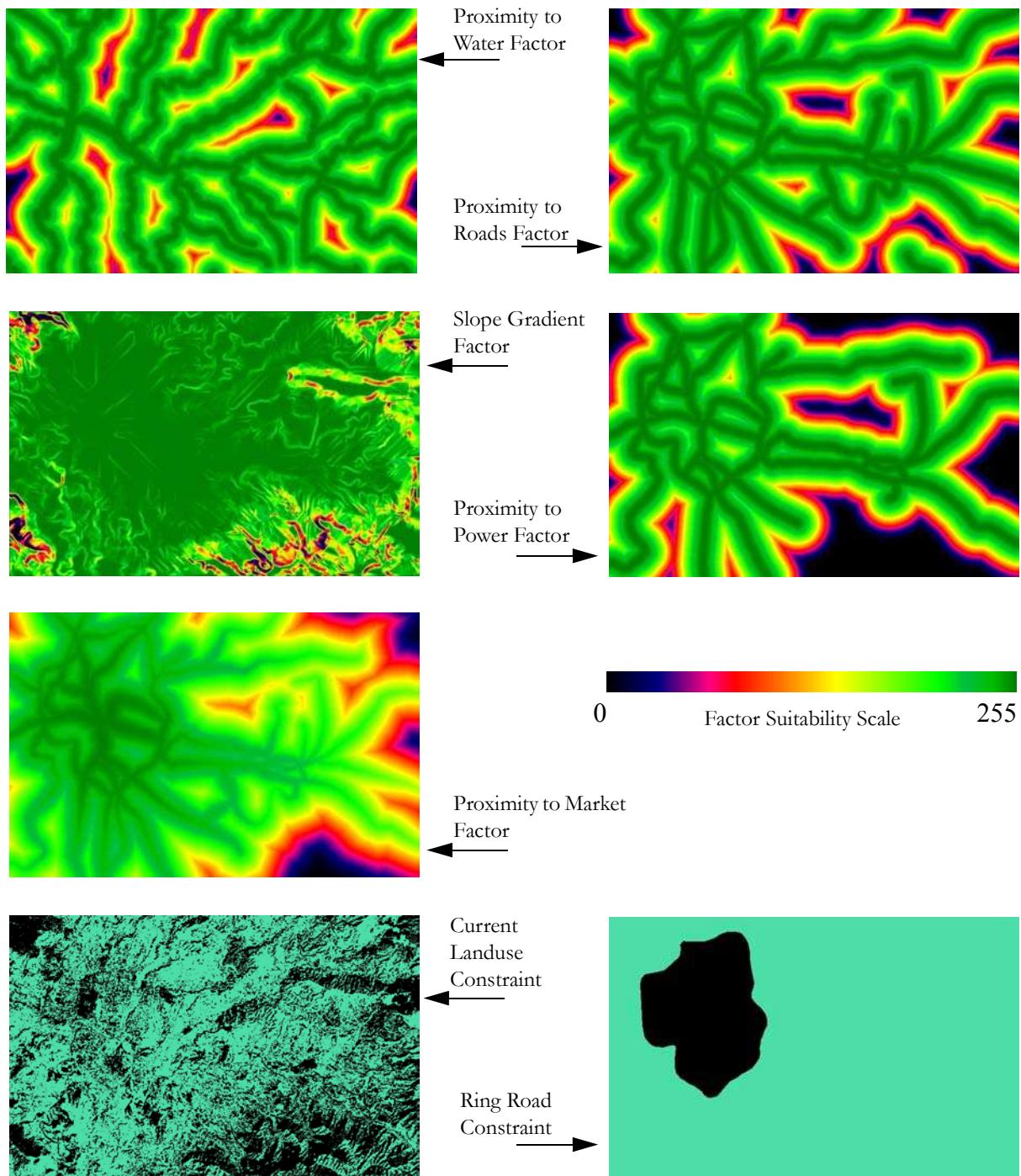


Figure 12-7 Carpet Industry Factors and Constraints.

1.3 Establishing the Factor Weights

The next stage was to establish a set of weights for each of the factors. In the nature of a focus group, the GIS analyst worked with the decision makers as a group to fill out a pairwise comparison matrix. Each decision maker was asked in turn to estimate a rating and then to indicate why he or she assigned the rating. The group would then be asked if they agreed. Further discussion would ensue, often with suggestions for different ratings. Ultimately, if another person made a strong case for a different rating that seemed to have broad support, the original person who provided the rating would be asked if he/she were willing to change (the final decision would in fact rest with the original rater). Consensus was not difficult to achieve using this procedure. It has been found through repeated experimentation with this technique that the only cases where strong disagreement arose were cases in which a new variable was eventually identified as needing to be incorporated. This is perhaps the greatest value of the pairwise comparison technique—it is very effective in uncovering overlooked criteria and reaching a consensus on weights through direct participation by decision makers.

Once the pairwise comparison matrices were filled, the WEIGHT module was used to identify inconsistencies and develop the best fit weights. Figure 12-8 shows the factor weights evaluated for the suitability for carpet industry development.

Factor	Factor Weight
Proximity to Water	0.51
Proximity to Roads	0.05
Proximity to Power	0.25
Accessibility to Market	0.16
Low Slopes	0.03

Figure 12-8

1.4 Undertaking the Multi-Criteria Evaluation

Once the weights were established, the module MCE (for Multi-Criteria Evaluation) was used to combine the factors and constraints in the form of a weighted linear combination (WLC option). The procedure is optimized for speed and has the effect of multiplying each factor by its weight, adding the results and then successively multiplying the result by each of the constraints. Since the weights sum to 1.0, the resulting suitability maps have a range from 0-255. Figure 12-9 shows the result of separate multi-criteria evaluations to derive suitability maps for the carpet and agricultural industries.

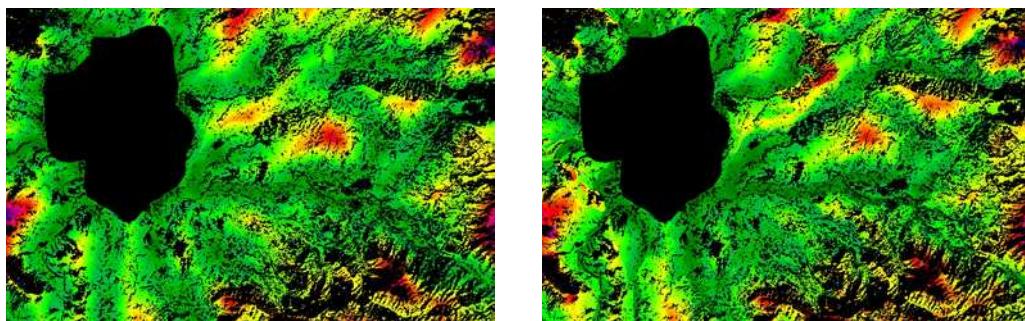


Figure 12-9 Composite Suitability images for Carpet Industry (left) and Agriculture (right). Suitability scale corresponds to that in Figure 12-7.

2. Solving the Multi-Objective Land Allocation Problem

Once the multi-criteria suitability maps have been created for each objective, the multi-objective decision problem can be approached.

2.1 Standardizing the Single-Objective Suitability Maps

The first step was to use the RANK module to rank order the cells in each of the two suitability maps. This prepares the data for use with the MOLA procedure and has the additional effect of standardizing the suitability maps using a non-parametric histogram equalization technique. Ranks were developed in descending order (i.e., the best rank was 1). In both cases tied ranks were resolved by examining the other suitability map and ranking in reverse order to the suitability on that map. This preserves the basic logic of the uncorrelated ideal points for conflicting objectives that is used in the resolution of conflicts.

2.2 Solving the Multi-Objective Problem

The second step was to submit the ranked suitability maps to the MOLA procedure. MOLA requires the names of the objectives, the relative weight to assign to each, and the area to be allocated to each. The module then undertakes the iterative procedure of allocating the best ranked cells to each objective according to the areal goals, looking for conflicts, and resolving conflicts based on the weighed minimum-distance-to-ideal-point logic. Figure 12-10 shows the final result, achieved after 6 iterations.

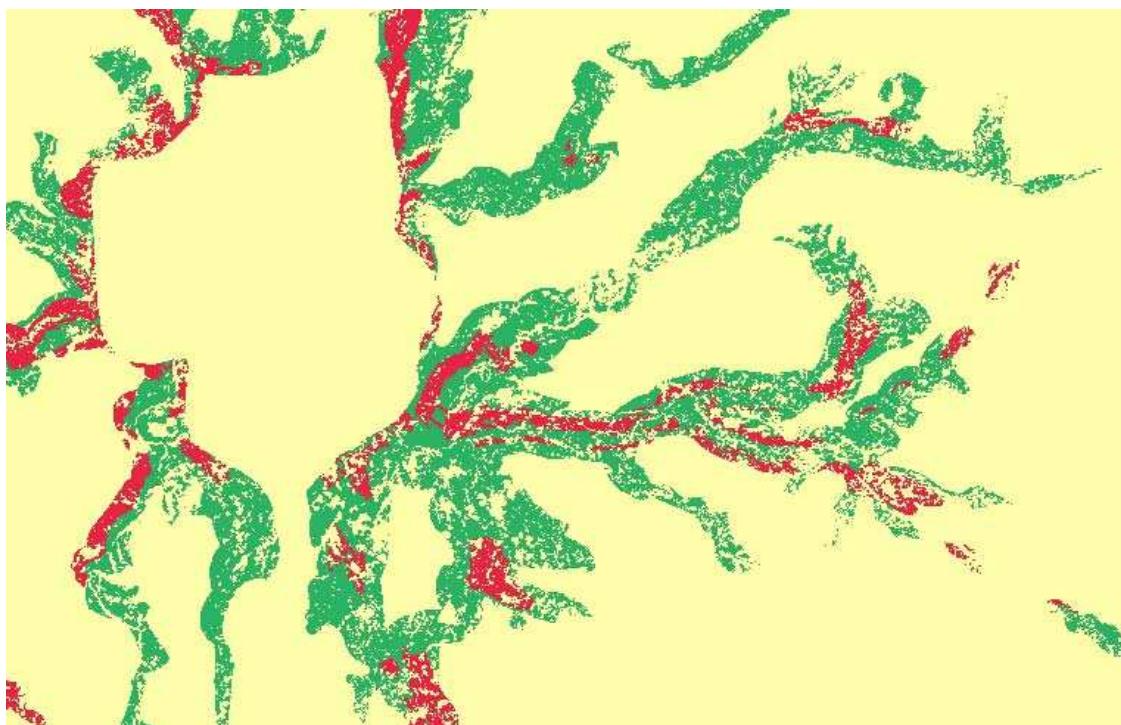


Figure 12-10 Final allocation to the carpet industry (red) and agriculture (green) objectives.

The Multi-Criteria/Multi-Objective Decision Support Wizard

The Decision Support Wizard (i.e., a set of linked dialogs) helps guide users through multi-criteria/multi-objective resource allocation procedures like those illustrated above. The wizard steps the user through each phase of building the full model and records the decision rules in a file that can be saved and later modified. A special section of the Help System provides additional information for each wizard screen. Novice users will find the wizard helpful in organizing their progress through the sequence of steps, while advanced users will appreciate the ability to save a full MCE/MOLA model that can be altered and run repeatedly to produce alternative final allocations. The wizard is launched from the Analysis/Decision Support menu.

A Closing Comment

The decision support tools provided in IDRISI are still under active development. We therefore welcome written comments and observations to further improve the modules and enhance their application in real-world situations.

References / Further Reading

- Alonso, W., 1968. Predicting Best with Imperfect Data, *Journal of the American Institute of Planners*, 34: 248-255.
- Carver, S.J., 1991. Integrating Multi-Criteria Evaluation with Geographical Information Systems, *International Journal of Geographical Information Systems* 5(3): 321-339.
- Campbell, J.C., Radke, J., Gless, J.T. and Wirtshafter, R.M., 1992. An Application of Linear Programming and Geographic Information Systems: Cropland Allocation in Antigua, *Environment and Planning A*, 24: 535-549.
- Diamond, J.T. and Wright, J.R., 1988. Design of an Integrated Spatial Information System for Multiobjective Land-Use Planning, *Environment and Planning B: Planning and Design*, 15: 205-214.
- Diamond, J.T. and Wright, J.R., 1989. Efficient Land Allocation, *Journal of Urban Planning and Development*, 115(2): 81-96.
- Eastman, J.R., 1996. Uncertainty and Decision Risk in Multi-Criteria Evaluation: Implications for GIS Software Design, *Proceedings, UN University International Institute for Software Technology Expert Group Workshop on Software Technology for Agenda'21: Decision Support Systems*, February 26-March 8.
- Eastman, J.R., and Jiang, H., 1996. Fuzzy Measures in Multi-Criteria Evaluation, *Proceedings, Second International Symposium on Spatial Accuracy Assessment in Natural Resources and Environmental Studies*, May 21-23, Fort Collins, Colorado, 527-534.
- Eastman, J.R., Jin, W., Kyem, P.A.K., and Toledano, J., 1995. Raster Procedures for Multi-Criteria/Multi-Objective Decisions, *Photogrammetric Engineering and Remote Sensing*, 61(5): 539-547.
- Eastman, J.R., Kyem, P.A.K., and Toledano, J., 1993. A Procedure for Multi-Objective Decision Making in GIS Under Conditions of Competing Objectives, *Proceedings, EGIS'93*, 438-447.
- Eastman, J.R., Kyem, P.A.K., Toledano, J. and Jin, W., 1993. *GIS and Decision Making*, Explorations in Geographic Information System Technology, 4, UNITAR, Geneva.
- FAO, 1976. *A Framework for Land Evaluation*, Soils Bulletin 32. Food and Agricultural Organization of the United Nations, Rome.

- Feiring, B.R., 1986. Linear Programming: An Introduction, *Quantitative Applications in the Social Sciences*, Vol. 60, Sage Publications, London.
- Honea, R.B., Hake, K.A., and Durfee, R.C., 1991. Incorporating GISs into Decision Support Systems: Where Have We Come From and Where Do We Need to Go? In: M. Heit abd A. Shortreid (eds.), *GIS Applications in Natural Resources*. GIS World, Inc., Fort Collins, Colorado.
- Ignizio, J.P., 1985. Introduction to Linear Goal Programming, *Quantitative Applications in the Social Sciences*, Vol. 56, Sage Publications, London.
- Janssen, R. and Rietveld, P., 1990. Multicriteria Analysis and Geographical Information Systems: An Application to Agricultural Land Use in the Netherlands. In: H.J. Scholten and J.C.H. Stillwell, (eds.), *Geographical Information Systems for Urban and Regional Planning*: 129-139. Kluwer Academic Publishers, Dordrecht, The Netherlands.
- Rao, M., Sastry, S.V.C., Yadav, P.D., Kharod, K., Pathan, S.K., Dhinwa, P.S., Majumdar, K.L., Sampat Kumar, D., Patkar, V.N. and Phatak, V.K., 1991. *A Weighted Index Model for Urban Suitability Assessment—A GIS Approach*. Bombay Metropolitan Regional Development Authority, Bombay, India.
- Rosenthal, R.E., 1985. Concepts, Theory and Techniques: Principles of Multiobjective Optimization. *Decision Sciences*, 16(2): 133-152.
- Saaty, T.L., 1977. A Scaling Method for Priorities in Hierarchical Structures. *J. Math. Psychology*, 15: 234-281.
- Voogd, H., 1983. *Multicriteria Evaluation for Urban and Regional Planning*. Pion, Ltd., London.
- Wright, J., ReVelle, C. and Cohon, J., 1983. A Multiobjective Integer Programming Model for the Land Acquisition Problem. *Regional Science and Urban Economics*, 13: 31-53.
- Zadeh, L.A., 1965. Fuzzy Sets. *Information and Control*, 8: 338-353.

Decision Support: Uncertainty Management

Uncertainty is inevitable in the decision making process. In the GIS community, the issue of uncertainty has received a considerable amount of interest (see Goodchild and Gopal, 1989), however, attention has focused particularly on measurement error: the expression of error (Burrough, 1986; Lee et. al., 1987; Maling, 1989; Stoms, 1987), error assessment (Congalton, 1991), error propagation (Burrough, 1986), and the reporting of data quality (Moellering et. al., 1988; Sloanecker and Tosta, 1992). There has also been considerable interest in other forms of uncertainty such as that expressed by fuzzy sets (e.g., Fisher, 1991). However, there has been less attention paid to how these uncertainties combine to affect the decision process and decision risk.

As the field becomes more conversant in the understanding and handling of uncertainty and its relationship to decision risk, it is inevitable that we will see a movement of GIS away from the *hard* decisions of traditional GIS (where it is assumed that the database and models are perfect) to procedures dominated by *soft* decisions. Given a knowledge of uncertainties in the database and uncertainties in the decision rule, it is possible to change the hard Boolean results of traditional GIS decisions into soft probabilistic results—to talk not of whether an area does or does not have a problem with soil erosion, but of the *likelihood* that it has a problem with soil erosion; not of whether an area is suitable or not for land allocation, but of the degree to which it is suitable. This would then allow a final hard decision to be developed based on the level of risk one is willing to assume. Thus, for example, one might decide to send an agricultural extension team to visit only those farms where the likelihood (or possibility) of a soil erosion problem exceeds 70%.

The movement to soft decision rules will require, in part, the development of uncertainty management capabilities in GIS. It requires data structures to carry uncertainty information and a revision of existing routines to assess and propagate error information. It also requires new procedures for analyzing different kinds of uncertainty and their effects on decision making. In IDRISI, a variety of procedures are available for this task.

A Typology of Uncertainty

Uncertainty includes any known or unknown error, ambiguity or variation in both the database and the decision rule. Thus, uncertainty may arise from such elements as measurement error, inherent variability, instability, conceptual ambiguity, over-abstraction, or simple ignorance of important model parameters.

Considering the decision making process as a set membership problem is a useful perspective from which to understand the source and role of uncertainty in decision making. As previously defined, a *decision frame* contains all the alternatives (or hypotheses) under consideration, and *evidence* is that information through which set membership of a location in the *decision set* (the set of chosen alternatives) can be evaluated. Thus, the decision making process contains three basic elements within which uncertainty can occur—the evidence, the decision set, and the relation that associates the two.

Uncertainty in the Evidence

In examining evidence to decide which elements of the candidate set belong to the set of alternatives to be chosen (the decision set), one evaluates the qualities and characteristics of those entities as represented in the database. However, there is a significant concern here with measurement error and how it propagates through a decision rule. This kind of uncertainty is usually represented by an RMS (root mean square) error in the case of quantitative data, or proportional error in the case of qualitative data, and relies upon classical probability theory and statistical inference for its assessment and propagation.

Uncertainty in the Relation

The second basic element of a decision is the specification of the relationship between the evidence and the decision set. Uncertainty arises here from at least three sources.

1. The first is in cases where the *definition* of a criterion (as opposed to its measurement) is subject to uncertainty. Sets with clearly defined attributes are known as *crisp* sets and are subject to the logic of classical sets. Thus, for example, the set of areas that would be inundated by a rise in sea level is clearly defined. Disregarding measurement error, if an area is lower than the projected level of the sea, it is unambiguously a member of the set. However, not all sets are so clearly defined. Consider, for example, the set of areas with steep slopes. What constitutes a steep slope? If we specify that a slope is steep if it has a gradient of 10% or more, does this mean that a slope of 9.99999% is not steep? Clearly there is no sharp boundary here. Such sets are called *fuzzy* sets (Zadeh, 1965) and are typically defined by a set membership function, as will be discussed further below. Although recognition of the concept of fuzzy sets is somewhat new in GIS, it is increasingly clear that such sets are prevalent (if not dominant) in land allocation decisions.
2. The second case where uncertainty arises is in cases where the evidence does not directly and perfectly imply the decision set under consideration. In the examples of inundated lands or steep slopes, there is a direct relationship between the evidence and the set under consideration. However, there are also cases where only indirect and imperfect evidence can be cited. For example, we may have knowledge that water bodies absorb infrared radiation. Thus we might use the evidence of low infrared reflectance in a remotely sensed image as a statement of the belief that the area is occupied by deep open water. However, this is only a belief since other materials also absorb infrared radiation.

Statements of belief in the degree to which evidence implies set membership are very similar in character to fuzzy set membership functions. However, they are not definitions of the set itself, but simply statements of the degree to which the evidence suggests the presence of the set (however defined). Thus the logic of fuzzy sets is not appropriate here, but rather, that of *Bayes* and *Dempster-Shafer* theory.

3. The third area where uncertainty can occur in specifying the relation between the evidence and the decision set is most often called *model specification error* (Alonso, 1968). In some instances, decisions may be based on a single criterion, but commonly several criteria are required to define the decision set. Thus, for example, one might define areas suitable for development as being those on shallow slopes and near to roads. Two issues here would be of concern: are these criteria adequate to define suitable areas, and have we properly aggregated the evidence from these criteria? If set membership indicated by slopes is 0.6 and proximity to roads is 0.7, what is the membership in the decision set? Is it the 0.42 of probabilities, the 0.6 of fuzzy sets, the 0.78 of Bayes, the 0.88 of Dempster-Shafer, or the 0.65 of linear combination? Further, how well does this aggregated value truly predict the degree to which the alternative under consideration truly belongs to the decision set? Clearly the construction of the decision rule can have an enormous impact on the set membership value deduced.

Uncertainty in the Decision Set

The final area of concern with respect to uncertainty in the decision process concerns the final set deduced. As outlined above, the process of developing the decision set consists of converting the evidence for each criterion into an elementary set statement, and then aggregating those statements into a single outcome that incorporates all of the criteria considered. Clearly, uncertainty here is some aggregate of the uncertainties which arose in acquiring the evidence and in specifying the relationship between that evidence and the decision set. However, in the presence of uncertainty about the degree to which any candidate belongs to the final set (as a result of the evidence gathered or its implications about set membership), some further action is required in order to develop the final set—a threshold of uncertainty will need to be established to determine which alternatives will be judged to belong to the decision set. To do so thus logically implies some likelihood that the decision made will be wrong—a concept that can best be described as *decision risk*. For example, given a group of locations for which the likelihood of being below a projected new sea level has been assessed, the final decision about which locations will be assumed to ultimately flood will be solved by establishing a threshold of likelihood. Clearly this threshold is best set in the context of decision risk.

In the remainder of this chapter, a set of tools in IDRISI will be explored for the management of uncertainty that arises in the evidence (database uncertainty) and in specifying the relation between that evidence and the decision set (decision rule uncertainty). In addition, in each of these two sections, consideration will be given to the problem of making a definitive judgment in the context of uncertainty, and thus the accommodation of decision risk.

Database Uncertainty and Decision Risk

An assessment of measurement error and an analysis of its propagation through data models combining different data layers is an essential aspect of uncertainty management. In this section, we examine procedures available in IDRISI for error assessment and propagation, and very importantly, procedures for evaluating the effects of this error on the decision process through a consideration of decision risk.

Error Assessment

The assessment of measurement error is normally achieved by selecting a sample of sites to visit on the ground, remeasuring the attribute at those locations using some more accurate instrument, and then comparing the new measurements to those in the data layer. To assist this procedure, IDRISI provides the SAMPLE and ERRMAT modules.

SAMPLE has the ability to lay out a sample of points (in vector format) according to a random, systematic or stratified random scheme. The latter is usually preferred since it combines the best qualities of the other two—the unbiased character of the random sampling scheme with the even geographic coverage of the systematic scheme.

The size of the sample (n) to be used is determined by multiplying an estimate of the standard error of the evaluation statistic being calculated by the square of the standard score (ζ) required for the desired level of confidence (e.g., 1.96 for 95% confidence), and dividing the result by the square of the desired confidence interval (ϵ) (e.g., 0.01 for $\pm 10\%$). For estimates of the sample size required for estimating an RMS error, this formula simplifies to:

$$n = \zeta^2 s^2 / 2\epsilon^2 \quad \text{where } s \text{ is the estimated RMS.}$$

For estimates of the proportional error in categorical data, the formula becomes:

$$n = \zeta^2 pq / \epsilon^2 \quad \text{where } p \text{ is the estimated proportional error and } q = (1-p).$$

Note that the term *stratified* in stratified random means that it is spatially stratified according to a systematic division of the area into rectangular regions. In cases where some other stratification is desired, and/or where the region to be sampled is not rectangular in shape, the following procedure can be used:

1. Determine the area of the stratum or irregular region using AREA and divide it by the area of the total image. This will indicate the proportional area of the stratum or irregular region.
2. Divide the desired sample size by the proportional area. This will indicate a new (and larger) sample size that will be required to ensure that the desired number of sample points will fall within the area of interest.
3. Run SAMPLE with the new sample size and use only those points that fall within the area of interest.

Once the ground truth has been undertaken at the sample points, the characteristic error can be assessed. In the case of assessing quantitative data using RMS, the standard formula for RMS derived from a sample can be used:

$$\text{RMS} = \sqrt{\frac{\sum(x_i - t)}{n - 1}}$$

where x_i = a measurement
 t = true value

However, in the case of qualitative data, an error matrix should be used to assess the relationship between mapped categories and true values. To facilitate this process, the ERRMAT module can be used. ERRMAT requires two input files: the original categorical image (e.g., a landuse map) and a second image containing the true categories. This *truth* map is typically in the form of a map dominated by zeros (the background) with isolated cells indicating the positions of sample points with their true values. Using these data, ERRMAT outputs an error matrix and summary statistics.

The error matrix produced by ERRMAT contains a tabulation of the number of sample points found in each possible combination of true and mapped categories. Figure 13-1 illustrates the basic error matrix output. As can be seen, tabulations along the diagonal represent cases where the mapped category matched the true value. Off-diagonal tabulations represent errors and are tabulated as totals in the margins. The error marginals represent the proportional error by category, with the total proportional error appearing in the bottom-right corner of the table. Proportional errors along the bottom of the graph are called *errors of omission* while those along the right-hand edge are called *errors of commission*. The former represents cases where sample points of a particular category were found to be mapped as something different, while the latter includes cases where locations mapped as a particular category were found to be truly something else. Careful analysis of these data allows not only an assessment of the amount of error, but also of where it occurs and how it might be remedied. For example, it is typical to look at errors of omission as a basis for judging the adequacy of the mapping, and the errors of commission as a means of determining how to fix the map to increase the accuracy.

		True						
		Conifers	Mixed	Deciduous	Water	Total	error	
Mapped	Conifers	24	0	0	3	27	0.11	errors of commission
	Mixed	3	36	16	0	55	0.35	
	Deciduous	0	0	28	0	28	0.00	
	Water	2	0	0	14	16	0.12	
	Total	29	36	44	17	126		
	error	0.17	0.00	0.36	0.18		0.19	

errors of omission

Figure 13-1 An Error Matrix

In addition to the basic error matrix, ERRMAT also reports the overall and per category Kappa Index of Agreement (KIA) values. The Kappa Index of Agreement is similar to a proportional accuracy figure (and thus the complement of proportional error), except that it adjusts for chance agreement.

Error Propagation

When uncertainty exists in data layers, that error will propagate through any analysis and combine with the error from other sources. Specific formulas do exist for the expected error propagation arising from typical GIS mathematical operations (such as those involved with SCALAR and OVERLAY). **Appendix 1** contains a representative set of such formulae. In addition, IDRISI contains two modules that under certain circumstances will propagate error information automatically using such procedures. The first is the MCE module described earlier in this chapter while the second is SURFACE.

If all of the input factors presented to MCE contain error (RMS) information in the *value error* field of their documentation

files, MCE will determine the propagated output error and place it in the documentation file of the result. However, bear in mind that it makes two *large* assumptions—first, that there is no correlation between the factors, and second, that there is no uncertainty in the weights since that uncertainty has been resolved through deriving a consensus. If these assumptions are not valid, a new assessment should be derived using a Monte Carlo procedure as described further below.

In the case of SURFACE, error information will also be propagated when deriving slopes from a digital elevation model where the RMS error has been entered in the *value error* field of its documentation file.

Despite the availability of propagation formulas, it is generally difficult to apply this approach to error propagation because:

1. propagation is strongly affected by intercorrelation between variables and the correlation may not always be known at the outset;
2. only a limited number of formulas are currently available, and many GIS operations have unknown propagation characteristics.

As a result, we have provided in IDRISI the tools for a more general approach called *Monte Carlo Simulation*.

Monte Carlo Simulation

In the analysis of propagation error through Monte Carlo Simulation, we simulate the effects of error in each of the data layers to assess how it propagates through the analysis. In practice, the analysis is run twice—first in the normal fashion, and then a second time using data layers containing the simulated error. By comparing the two results, the effects of the error can be gauged—the only reason they differ is because of the error introduced. Typically, HISTO would be used to examine the distribution of these errors as portrayed in a difference image produced with OVERLAY. With a normally distributed result, the standard deviation of this difference image can be used as a good indicator of the final RMS.⁶¹

The tool that is used to introduce the simulated error is RANDOM. RANDOM creates images with random values according to any of a rectilinear, normal or lognormal model. For normal and lognormal distribution, the RMS error can be either one uniform value for the entire image, or be defined by an image that has spatially varied values. For categorical data, the rectilinear model outputs integer values that can be used as category codes. For quantitative data, all models can generate real numbers. For example, to add simulated error for a digital elevation model with an RMS error of 3 meters, RANDOM would be used to generate a surface using a normal model with a mean of 0 and a standard deviation of 3. This image would then be *added* to the digital elevation model. Note that the result is not meant to have any specific claim to reality—just that it contains error of the same nature as that believed to exist in the original.

Database Uncertainty and Decision Risk

Given an estimate of measurement error and an analysis of how it has propagated through the decision rule, the PCLASS module can be used to determine a final decision in full recognition of the decision risk that these uncertainties present. PCLASS evaluates the likelihood that the data value in any raster cell exceeds or is exceeded by a specified threshold. PCLASS assumes a random model of measurement error, characterized by a Root Mean Square (RMS) error statement. In the IDRISI system, the metadata for each raster image contains a field where error in the attribute values can be stated, either as an RMS for quantitative data, or as a proportional error for quantitative data. PCLASS uses the RMS recorded for a quantitative image to evaluate the probability that each value in the image lies either above or below a specified threshold. It does so by measuring the area delineated by that threshold under a normal curve with a standard deviation

61. Monte Carlo Simulation relies upon the use of a very large set of simulations to derive its characterizations. In cases such as this where each cell provides a new simulation, the total composite of cells can provide such a large sample. Results are improved by repeated runs of such an analysis and an averaging of results.

equal to the RMS (Figure 13-2). The result is a probability map as is illustrated in Figure 13-3, expressing the likelihood that each area belongs to the decision set.

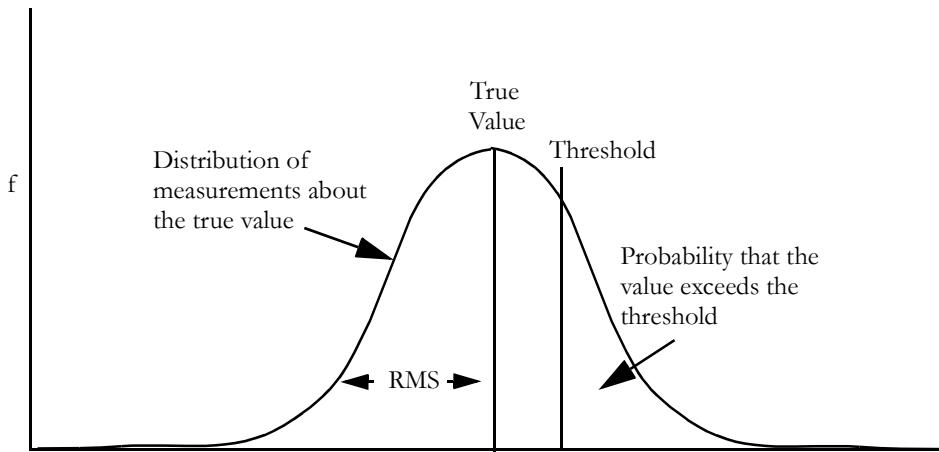


Figure 13-2

With PCLASS we have the *soft* equivalent of a *hard* RECLASS operation. For example, consider the case of finding areas that will be inundated by a rise in sea level as a result of global warming. Traditionally, this would be evaluated by reclassifying the heights in a digital elevation model into two groups—those below the projected sea level and those above. With PCLASS, however, recognition is made of the inherent error in the measurement of heights so that the output map is not a *hard* Boolean map of zeros and ones, but a *soft* probability map that ranges continuously from zero to one. Figure 13-3, for example, illustrates the output from PCLASS after evaluating the probability that heights are less than a new projected sea level of 1.9 meters above the current level in Boston Harbor in the USA. Given this continuous probability map, a final decision can be made by reclassifying the probability map according to the level of decision risk one is willing to assume. Figures 13-4 and 13-5, for example, show the difference between the original coastline and that associated with the new sea level while accepting only a 5% chance of being wrong compared to that of accepting a 25% chance. Clearly, the Digital Elevation Model (DEM) used in this assessment is not very precise. However, this illustrates the fact that even poor data can be used effectively if we know how poor they are.

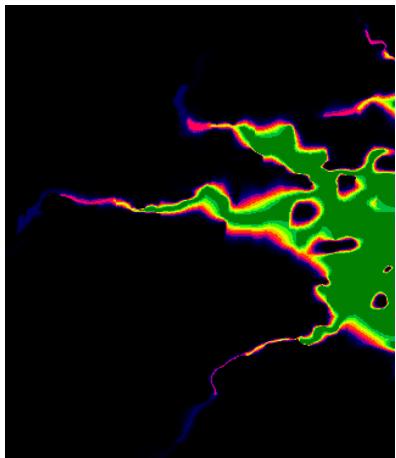


Figure 13-3 Probability of being flooded.

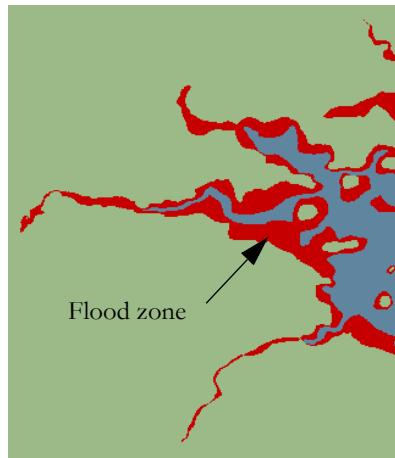


Figure 13-4 Flood zone at 5% risk.

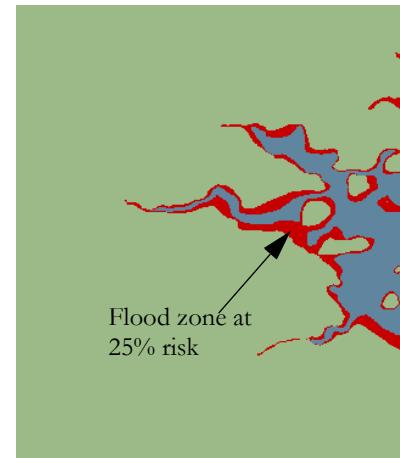


Figure 13-5 Flood zone at 25% risk.

Decision Rule Uncertainty

In the *Typology of Uncertainty* presented earlier, the second major element of uncertainty that was identified (after measurement error) was that in specifying the relationship between the evidence and the final decision set—an aspect that can broadly be termed *decision rule uncertainty*. This is an area where much further research is required. However, the IDRISI system does include an extensive set of tools to facilitate the assessment and propagation (or aggregation in this context) of this form of uncertainty.

All of these tools are concerned with the uncertainty inherent in establishing whether an entity belongs in the final decision set, and thus fall into a general category of uncertain set membership expression, known as a *fuzzy measure*. The term *fuzzy measure* (not to be confused with the more specific instance of a Fuzzy Set) refers to any set function which is monotonic with respect to set membership (Dubois and Prade, 1982). Notable examples of fuzzy measures include Bayesian probabilities, the *beliefs* and *plausibilities* of Dempster-Shafer theory, and the *possibilities* of Fuzzy Sets.

A common trait of fuzzy measures is that they follow DeMorgan's Law in the construction of the intersection and union operators (Bonissone and Decker, 1986), and thereby, the basic rules of uncertainty propagation in the aggregation of evidence. DeMorgan's Law establishes a triangular relationship between the intersection, union and negation operators such that:

$$T(a, b) = \sim S(\sim a, \sim b)$$

where T = Intersection (AND) = T-Norm

and S = union (OR) = T-CoNorm

and \sim = Negation (NOT)

The intersection operators in this context are known as *triangular norms*, or simply *T-Norms*, while the union operators are known as *triangular co-norms*, or *T-CoNorms*.

A T-Norm can be defined as (Yager, 1988):

a mapping $T: [0,1] * [0,1] \rightarrow [0,1]$ such that :

$$T(a,b) = T(b,a) \quad (\text{commutative})$$

$$T(a,b) \geq T(c,d) \text{ if } a \geq c \text{ and } b \geq d \quad (\text{monotonic})$$

$$T(a,T(b,c)) = T(T(a,b),c) \quad (\text{associative})$$

$$T(1,a) = a$$

Some examples of T-Norms include:

$$\min(a,b) \quad (\text{the intersection operator of fuzzy sets})$$

$$a * b \quad (\text{the intersection operator of probabilities})$$

$$1 - \min(1,((1-a)^p + (1-b)^p)^{(1/p)}) \quad (\text{for } p \geq 1)$$

$$\max(0,a+b-1)$$

Conversely, a T-CoNorm is defined as:

a mapping $S: [0,1] * [0,1] \rightarrow [0,1]$ such that :

$$S(a,b) = S(b,a) \quad (\text{commutative})$$

$$S(a,b) \leq S(c,d) \text{ if } a \leq c \text{ and } b \leq d \quad (\text{monotonic})$$

$$S(a,S(b,c)) = S(S(a,b),c) \quad (\text{associative})$$

$$S(0,a) = a$$

Some examples of T-CoNorms include:

$$\max(a,b) \quad (\text{the union operator of fuzzy sets})$$

$$a + b - a*b \quad (\text{the union operator of probabilities})$$

$$\min(1,(a^p + b^p)^{(1/p)}) \quad (\text{for } p \geq 1)$$

$$\min(1,a+b)$$

These examples show that a very wide range of operations are available for fuzzy measure aggregation, and therefore, criteria aggregation in decision making processes. Among the different operators, the most extreme (in the sense that they yield the most extreme numeric results upon aggregation) are the *minimum* T-Norm operator and the *maximum* T-CoNorm operator. These operators also have special significance as they are the most commonly used aggregation operators for fuzzy sets. Furthermore, they have been shown by Yager (1988) to represent the extreme ends of a continuum of related aggregation operators that can be produced through the operation of an Ordered Weighted Average. As was indicated in the chapter **Decision Support: Decision Strategy Analysis**, this continuum also includes the traditional Weighted Linear Combination operator that is commonly encountered in GIS. However, the important issue here is not that a particular family of aggregation operators is correct or better than another, but simply that different expressions of decision rule uncertainty require different aggregation procedures.

Currently, three major logics are in use for the expression of decision rule uncertainty, all of which are represented in the IDRISI module set: Fuzzy Set theory, Bayesian statistics, and Dempster-Shafer theory. Each is distinct, and has its own very different set of T-Norm/T-CoNorm operators. However, the context in which one uses one as opposed to another is not always clear. In part, this results from the fact that decision rules may involve more than one form of uncertainty. However, this also results from a lack of research within the GIS field on the context in which each should be used. That said, here are some general guidelines that can be used:

- Decision problems that can be cast in the framework of suitability mapping can effectively be handled by the logic of Fuzzy Sets. This procedure has been covered in detail under the section on Multi-Criteria Evaluation in the chapter **Decision Support: Decision Strategy Analysis**. For example, if we define suitability in terms of a set of continuous factors (distance from roads, slope, etc.), the expression of suitability is continuous. There is no clear separation between areas that are suitable and those that are not. Many (if not most) GIS resource allocation problems fall into this category, and thus belong in the realm of Fuzzy Sets.

- The presence of *fuzziness*, in the sense of ambiguity, does not always imply that the problem lies in the realm of Fuzzy Sets. For example, measurement uncertainty associated with a crisp set can lead to a set membership function that is essentially identical in character to that of a Fuzzy Set. Rather, the distinguishing characteristic of a fuzzy set is that the set is itself inherently ambiguous. For example, if one considers the case of deciding on whether an area will be flooded as the result of the construction of a dam, some uncertainty will exist because of error in the elevation model. If one assumes a random error model, and spatial independence of errors, then a graph of the probability of being inundated against reported height in the database will assume an s-shaped cumulative normal curve, much like the typical membership function of a fuzzy set. However, the set itself is not ambiguous—it is crisp. It is the measure of elevation that is in doubt.

- The presence of *fuzziness*, in the sense of inconclusiveness, generally falls into the realm of Bayesian probability theory or its variant known as Dempster-Shafer theory. The problem here is that of indirect evidence—that the evidence at hand does not allow one to directly assess set membership, but rather to infer it with some degree of uncertainty. In their prototypical form, however, both logics are concerned with the substantiation of crisp sets—it is the strength of the relationship between the evidence and the decision set that is in doubt. A classic example here is the case of the supervised classification procedure in the analysis of remotely sensed imagery. Using training site data, a Bayesian classifier (i.e., decision engine) establishes a statistical relationship between evidence and the decision set (in the form of a conditional probability density function). It is this established, but uncertain, relationship that allows one to infer the degree of membership of a pixel in the decision set.

- Despite their common heritage, the aggregation of evidence using Bayes and Dempster-Shafer can yield remarkably different results. The primary difference between the two is characterized by the role of the absence of evidence. Bayes considers the absence of evidence in support of a particular hypothesis to therefore constitute evidence in support of alternative hypotheses, whereas Dempster-Shafer does not. Thus, despite the fact that both consider the hypotheses in the decision frame to be exhaustive, Dempster-Shafer recognizes the concept of ignorance while Bayes does not. A further difference is that the Bayesian approach combines evidence that is conditioned upon the hypothesis in the decision set (i.e., it is based on training data), while Dempster-Shafer theory aggregates evidence derived from independent sources.

Despite these broad guidelines, the complete implementation of these logics is often difficult because their theoretical development has been restricted to prototypical contexts. For example, Fuzzy Set theory expresses ambiguity in set membership in the form of a membership function. However, it does not address the issue of uncertainty in the form of the membership function itself. How, for example, does one aggregate evidence in the context of indirect evidence *and* an ambiguous decision set? Clearly there is much to be learned here. As a start, the following section begins to address the issues for each of these major forms for the expression of uncertainty.

Fuzzy Sets

Fuzzy Sets are sets (or classes) without sharp boundaries; that is, the transition between membership and nonmembership of a location in the set is gradual (Zadeh, 1965; Schmucker, 1982). A Fuzzy Set is characterized by a fuzzy membership grade (also called a *possibility*) that ranges from 0.0 to 1.0, indicating a continuous increase from nonmembership to complete membership. For example, in evaluating whether a slope is steep, we may define a fuzzy membership function such that a slope of 10% has a membership of 0, and a slope of 25% has a membership of 1.0. Between 10% and 25%, the fuzzy membership of a slope gradually increases on the scale from 0 to 1 (Figure 13-6). This contrasts with the classic *crisp* set which has distinct boundaries. However, a crisp set can also be seen as a special case of fuzzy set where fuzzy membership changes instantaneously from 0 or 1.

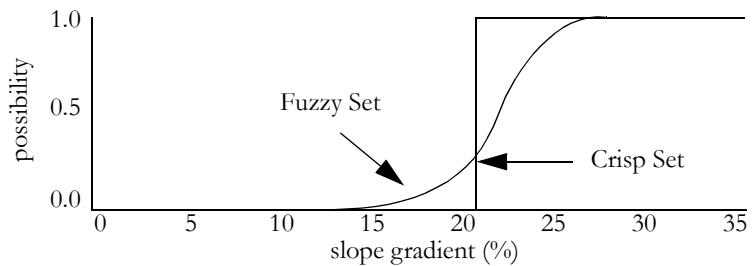


Figure 13-6 Fuzzy vs. Crisp Set Membership Functions

Fuzzy Set theory provides a rich mathematical basis for understanding decision problems and for constructing decision rules in criteria evaluation and combination. In use, the FUZZY module in IDRISI is designed for the construction of Fuzzy Set membership functions, while the OWA option of the MCE module offers a range of appropriate aggregation operators. FUZZY offers four types of membership function:

1. Sigmoidal: The *sigmoidal* ("s-shaped") membership function is perhaps the most commonly used function in Fuzzy Set theory. It is produced here using a cosine function as described in the on-line Help System. In use, FUZZY requires the positions (along the X axis) of 4 *inflection points* governing the shape of the curve. These are indicated in Figure 13-7 as points *a*, *b*, *c* and *d*, and represent the inflection points as the membership function rises above 0, approaches 1, falls below 1 again, and finally approaches 0. The right-most function of Figure 13-7 shows all four inflection points as distinct. However, this same function can take different forms. Figure 13-7 shows all possibilities. Beginning at the left, the *monotonically increasing* function shape rises from 0 to 1 then never falls. The previously mentioned concept of steep slopes is a good example here where the first inflection point *a* would be 10%, and the second *b* would be 25%. Since it never falls again, inflection points *c* and *d* would be given the same value as *b* (FUZZY understands this convention). However, the FUZZY interface facilitates data input in this case by requesting values only for inflection points *a* and *b*. The second curve of Figure 13-7 shows a *monotonically decreasing* function that begins at 1 then falls and stays at 0. In this case where the membership function starts at 1 and falls to 0 but never rises, *a* and *b* would be given identical values to *c* (the point at which it begins to fall), and *d* would be given the value of the point at which it reaches 0. The FUZZY interface only requires inflection points *c* and *d* for this type of function. The last two functions shown are termed *symmetric* as they rise then fall again. In the case where the function rises and then immediately falls (the third curve in Figure 13-7), points *b* and *c* take on the same value. Finally, where it rises, stays at 1 for a while, and then falls, all four values are distinct. In both cases, the FUZZY interface requires input of all four inflection points. Note that there is no requirement of geometric symmetry for symmetric functions, only that the curve rise then fall again. It is quite likely that the shape of the curve between *a* and *b* and that between *c* and *d* would be different, as illustrated in the right-most curve of Figure 13-7.

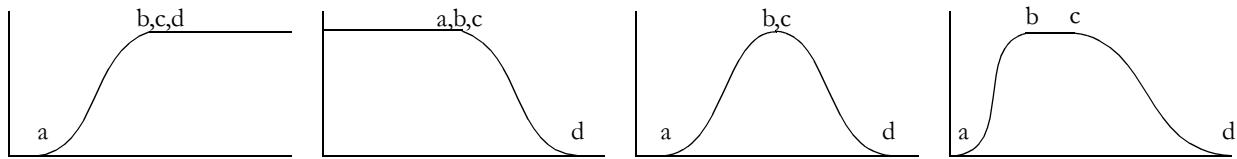


Figure 13-7 Sigmoidal Membership Function

2. J-Shaped: The J-Shaped function is also quite common, although in most cases it would seem that a sigmoidal function would be better. Figure 13-8 shows the different possibilities of J-shaped functions and the positions of the inflection points. *It should be pointed out that with the J-shaped function, the function approaches 0 but only reaches it at infinity. Thus the inflection*

points a and d indicate the points at which the function reaches 0.5 rather than 0.

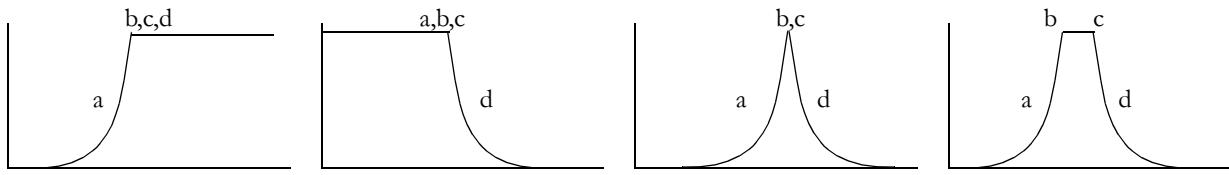


Figure 13-8 J-Shaped Membership Function

3. **Linear:** Figure 13-9 shows the Linear function and its variants, along with the position of the inflection points. This function is used extensively in electronic devices advertising fuzzy set logic, in part because of its simplicity, but also in part because of the need to monitor output from essentially linear sensors.

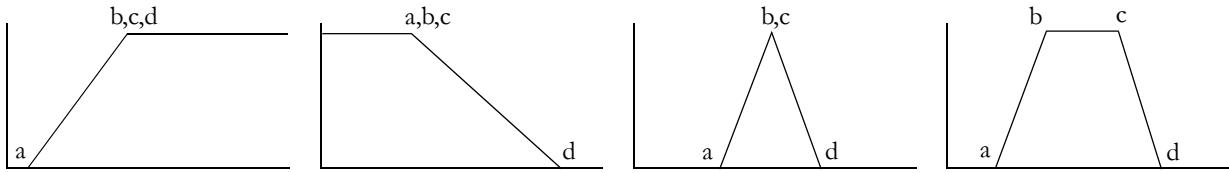


Figure 13-9 Linear Membership Function

4. **User-defined:** When the relationship between the value and fuzzy membership does not follow any of the above three functions, the user-defined function is most applicable. An unlimited number of control points may be used in this function to define the fuzzy membership curve. The fuzzy membership between any two control points is linearly interpolated, as in Figure 13-10.

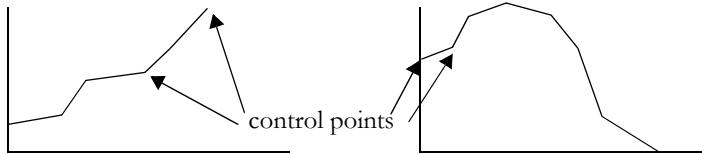


Figure 13-10 User-Defined Membership Function

In Multi-Criteria Evaluation, fuzzy set membership is used in the standardization of criteria. Exactly which function should be used will depend on the understanding of the relationship between the criterion and the decision set, and on the availability of information to infer fuzzy membership. In most cases, either the sigmoidal or linear functions will be sufficient.

Bayesian Probability Theory

When complete information is available or assumed, the primary tool for the evaluation of the relationship between the indirect evidence and the decision set is Bayesian Probability theory. Bayesian Probability theory is an extension of Classical Probability theory which allows us to combine new *evidence* about an hypothesis along with prior knowledge to arrive at an estimate of the likelihood that the hypothesis is true. The basis for this is Bayes' Theorem which states that (in the notation of probability theory):

$$\text{where: } p(h|e) = \frac{p(e|h) \cdot p(h)}{\sum_i p(e|h_i) \cdot p(h_i)}$$

$p(h|e)$ = the probability of the hypothesis being true given the evidence (posterior probability)

$p(e|h)$ = the probability of finding that evidence given the hypothesis being true

$p(h)$ = the probability of the hypothesis being true regardless of the evidence (prior probability)

For those unfamiliar with probability theory, this formula may seem intimidating. However, it is actually quite simple. The simplest case is when we have only two hypotheses to choose from—an hypothesis h and its complement $\sim h$ (that h is **not** true), the probabilities of which are represented by $p(h)$ and $p(\sim h)$, respectively. For example, is an area going to be flooded or is it not? The first question to consider is whether we have any prior knowledge that leads us to the probability that one or the other is true. This is called an *a priori probability*. If we do not, then the hypotheses are assumed to be equally probable.

The term $p(e|h)$ expresses the probability that we would find the evidence we have if the hypothesis being evaluated were true. It is known as a *conditional probability*, and is assessed on the basis of finding areas in which we know the hypothesis to be true and gathering data to evaluate the probability that the evidence we have is consistent with this hypothesis. We will refer to this as ground truth data even though it may be assessed on theoretical grounds or by means of a simulation.

The term $p(h|e)$ is a *posterior probability* created after prior knowledge and evidence for the hypothesis are combined. By incorporating extra information about the hypotheses, the probability for each hypothesis is modified to reflect the new information. It is the assumption of Bayes' Theorem that complete information is achievable, and thus the only reason that we do not have an accurate probability assessment is a lack of evidence. By adding more evidence to the prior knowledge, theoretically one could reach a true probability assessment for all the hypotheses.

Dempster-Shafer Theory

Dempster-Shafer theory, an extension of Bayesian probability theory, allows for the expression of ignorance in uncertainty management (Gordon and Shortliffe, 1985; Lee et al., 1987). The basic assumptions of Dempster-Shafer theory are that ignorance exists in the body of knowledge, and that belief for a hypothesis is not necessarily the complement of belief for its negation.

First, Dempster-Shafer theory defines hypotheses in a hierarchical structure (Figure 13-11) developed from a basic set of hypotheses that form the *frame of discernment*.⁶² For example, if the frame of discernment includes three basic hypotheses: {A, B, C}, the structure of hypotheses for which Dempster-Shafer will accept evidence includes all possible combinations, [A], [B], [C], [A, B], [A, C], [B, C], and [A, B, C]. The first three are called singleton hypotheses as each contains only one basic element. The rest are non-singleton hypotheses containing more than one basic element. Dempster-Shafer recognizes these hierarchical combinations because it often happens that the evidence we have supports some combinations of hypotheses without the ability to further distinguish the subsets. For example, we may wish to include classes of [deciduous] and [conifer] in a land cover classification, and find that evidence from a black and white aerial photograph can dis-

62. The *frame of discernment* in Dempster-Shafer theory has essentially the same meaning as the term *decision frame* as used in this paper—i.e., the set of alternative hypotheses or classes that can be substantiated or assigned to entities. Dempster-Shafer considers these hypotheses to be exhaustive. Thus, statements of support for any hierarchical combination of classes represents a degree of inability to commit to one of the singleton hypotheses in the frame of discernment. However, in practice, Dempster-Shafer does treat these hierarchical combinations as additional hypotheses. In addition, in a GIS and Remote Sensing context, there may be good reason to treat some unresolvable commitment to one of these hierarchical combinations as truly evidence of an independent class/hypothesis to which entities might be assigned. For example, with a frame of discernment that includes [forest] and [wetland], the presence of commitment to a [forest wetland] combination may in fact represent the presence of a "forested wetland" class that cannot be resolved by attaining better evidence. As a result, we recognize here that the analyst may wish to consider the decision frame as containing all of the hierarchical combinations, and not just the more limited set of singletons that forms the Dempster-Shafer frame of discernment. This does not violate the logic of Dempster-Shafer, since we are simply making the post-analysis judgement that certain combinations represent new classes and thus may form a decision set.

tinguish forest from non-forested areas, but not the type of forest. In this case we may use this evidence as support for the hierarchical combination [deciduous, coniferous]. Clearly this represents a statement of uncertainty. However, it also provides valuable information that will be used to advantage by the Dempster-Shafer procedure in any statement of belief about these hypotheses.

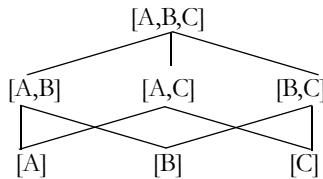


Figure 13-11 Hierarchical Structure of the Subsets in the Whole Set [A,B,C]

In expressing commitment to any of these hypotheses, Dempster-Shafer theory recognizes six important concepts: *basic probability assignment (BPA)*, *ignorance*, *belief*, *disbelief*, *plausibility*, and *belief interval*.

A basic probability assignment (BPA) represents the support that a piece of evidence provides for one of these hypotheses and not its proper subsets. Thus a BPA for [A, B] represents that mass of support for [A,B], but not [A] or [B]—i.e., that degree of support for some indistinguishable combination of [A] and [B]. This is usually symbolized with the letter "m" (for *mass*), e.g.:

$$m(A,B) = \text{basic probability assignment to } [A, B]$$

The basic probability assignment for a given hypothesis may be derived from subjective judgment or empirical data. Since a BPA is a fuzzy measure, the FUZZY module can also be used in IDRISI to develop a BPA from a given data set.

The sum of all BPAs will equal 1.0 at all times. Thus, the BPA for the ultimate superset ([A, B, C] in this example) will equal the complement of the sum of all other BPAs. This quantity thus represents *ignorance*—the inability to commit to any degree of differentiation between the elements in the frame of discernment.

Belief represents the total support for an hypothesis, and will be drawn from the BPAs for all subsets of that hypothesis, i.e.:

$$\text{BEL}(X) = \sum m(Y) \quad \text{when } Y \subseteq X$$

Thus the belief in [A, B] will be calculated as the sum of the BPAs for [A, B], [A], and [B]. In this example, belief represents the probability that an entity is A or B. Note that in the case of singleton hypotheses, the basic probability assignment and belief are identical.

In contrast to belief, *plausibility* represents the degree to which an hypothesis cannot be disbelieved. Unlike the case in Bayesian probability theory, *disbelief* is not automatically the complement of belief, but rather, represents the degree of support for all hypotheses that do not intersect with that hypothesis. Thus:

$$\text{PL}(X) = 1 - \text{BEL}(\bar{X}) \quad \text{where } \bar{X} = \text{not } X$$

$$\text{thus } \text{PL}(X) = \sum m(Y) \text{ when } Y \cap X \neq \emptyset$$

Interpreting these constructs, we can say that while belief represents the degree of hard evidence in support of an hypothesis, plausibility indicates the degree to which the conditions *appear to be right* for that hypothesis, even though hard evidence is lacking. For each hypothesis, then, belief is the lower boundary of our commitment to that hypothesis, and plausibility represents the upper boundary. The range between the two is called the *belief interval*, and represents the degree of uncertainty in establishing the presence or absence of that hypothesis. As a result, areas with a high belief interval are those in which new evidence will supply the greatest degree of information. Dempster-Shafer is thus very useful in establishing the value of information and in designing a data gathering strategy that is most effective in reducing uncertainty.

Compared with Bayesian probability theory, it is apparent that Dempster-Shafer theory is better able to handle uncertainty that involves ignorance. In Bayesian probability theory only singleton hypotheses are recognized and are assumed to be exhaustive (i.e., they must sum to 1.0). Thus, ignorance is not recognized, and a lack of evidence for a hypothesis therefore constitutes evidence against that hypothesis. These requirements and assumptions are often not warranted in real-world decision situations. For example, in establishing the habitat range for a particular bird species, evidence in the form of reported sightings might be used. However, the absence of a sighting at a location does not necessarily imply that the species was not present. It may simply indicate that there was no observer present, or that the observer failed to see a bird that was present. In cases such as this, Dempster-Shafer theory is appropriate (Gordon and Shortliffe, 1985; Srinivasan and Richards, 1990).

Dempster-Shafer Aggregation Operators

The full hierarchy of hypotheses and the BPAs associated with each represent a state of knowledge that can be added to at any time. In aggregating probability statements from different sources of evidence, Dempster-Shafer employs the following rule of combination:

$$m(Z) = \frac{\sum m_1(X) \bullet m_2(Y)}{1 - \sum m_1(X) \bullet m_2(Y)} \begin{cases} \text{when } (X \cap Y) = Z \\ \text{when } (X \cap Y) = \emptyset \end{cases}$$

If $\sum m_1(X) \bullet m_2(Y) = 0$ for $X \cap Y = \emptyset$, then the equation becomes

$$m(Z) = \sum m_1(X) \bullet m_2(Y) \text{ for } X \cap Y = Z.$$

The final belief, plausibility, and belief interval for each of the hypotheses can then be calculated based on the basic probability assignment calculated using the above equations. Ignorance for the whole set can also be derived. In most cases, after adding new evidence, the ignorance is reduced.

Working with Dempster-Shafer Theory: Belief

In IDRISI, the Belief module can be used to implement the Dempster-Shafer logic. Belief constructs and stores the current state of knowledge for the full hierarchy of hypotheses formed from a frame of discernment. In addition, it has the ability to aggregate new evidence with that knowledge to create a new state of knowledge, that may be queried in the form of map output for the belief, plausibility or belief interval associated with any hypothesis.

Belief first requires that the basic elements in the frame of discernment be defined. As soon as the basic elements are entered, all hypotheses in the hierarchical structure will be created in the hypothesis list. For each line of evidence entered, basic probability assignment images (in the form of real number images with a 0 - 1 range) are required with an indication of their supported hypothesis. The BUILD KNOWLEDGE BASE item in the ANALYSIS menu then incorporates this new evidence by recalculating the state of knowledge using the Dempster-Shafer rule of combination, from which summary images in the form of belief, plausibility or belief interval statements for each hypothesis can be selected. All the information entered can be saved in a knowledge base file for later use when more evidence is obtained.

The Dempster-Shafer rule of combination provides an important approach to aggregating indirect evidence and incomplete information. Consider, for example, the problem of estimating where an archaeological site of a particular culture might be found. The decision frame includes two basic elements, [site] and [non-site].⁶³ Four pieces of evidence are used:

63. The total number of hypotheses that Dempster-Shafer generates in the full hierarchy is $2^n - 1$. Implicitly, there is an extra hypothesis that is the null set, which is assumed by Dempster-Shafer to be automatically false. Thus in this example, the [non-site] hypothesis is not the null set, nor is it automatically assumed by Dempster-Shafer. In this example it was entered as a positive hypothesis, and member of the frame of discernment.

the locations of known sites, the frequency of surface artifacts (such as pottery shards), proximity to permanent water, and slopes. The first may be seen as direct evidence (at the exact positions of the sites themselves) for areas that have known archaeological sites. However, what we are concerned about are the areas that do not have a site, for which the known sites do not provide direct information. Therefore, the evidence is largely indirect. For areas that are close to the existing sites, one could believe the likelihood for the presence of another site would be higher. Thus the FUZZY module is used to transform a map of distance from known sites into an image of probability (a basic probability assignment image in support of the [site] hypothesis). The frequency of surface artifacts is also used as evidence in support of the [site] hypothesis. The distance from permanent water and slope images, however, have been used as disbelief images (see note 1 under "Using Belief" below). They therefore have both been scaled to a 0-1 range using FUZZY to provide support for the [non-site] hypothesis. Figure 13-12 shows these basic probability assignment images.

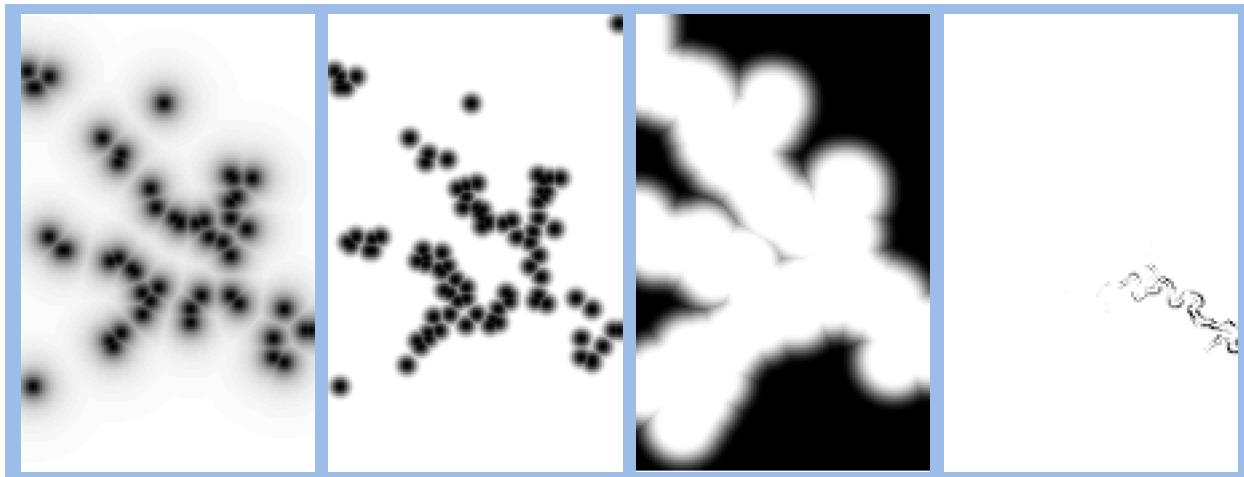


Figure 13-12 Basic Probability Assignment Images used in Aggregating Evidence for Archaeological Sites. From left to right, the BPA's support the hypothesis [Site] based on distance from known sites, [Site] based on frequency of surface artifacts, [Non-Site] based on distance from permanent water, and [Non-Site] based on slope. In all cases, darker areas represent a higher BPA.

The module Belief combines information from all four sources and has been used to produce belief, plausibility and belief interval images for the [site] hypothesis as illustrated in Figure 13-13. The belief interval image is particularly interesting in that it shows us where we have substantial uncertainty. Further sampling of evidence in these areas might prove profitable since the conditions support the plausibility of a site, even though concrete evidence is poor.

Using Belief

1. You may find it difficult to decide whether a particular piece of evidence should be used to support the belief of an hypothesis or, alternatively, the complement of that image should be used to support its disbelief. The latter is actually a statement in support of the plausibility of an hypothesis, but not its belief, and is very common in GIS. For example, in the case above, proximity to permanent water was treated as a distance image in support of disbelief in the possibility of a site. The reason for this is that if one were near to water there is no reason to believe that a site would or would not be present, but if one were far from water, there is excellent reason to assume that a site could not have existed. In deciding how to treat lines of evidence, consider carefully whether the data provide true evidence in support of an hypothesis, or simply support for its plausibility (i.e., the inability to deny its possibility).
2. To enter a disbelief, indicate that the evidence supports the collection of all hypotheses that do not include the one of concern. In the archaeology example, distance from water was entered as evidence for [non-site]. In a case with three

hypotheses [A, B, C], to indicate that a particular line of evidence supports the disbelief in A, you would indicate that it provides support for [B, C].

3. For each line of evidence that is incorporated using Belief, make sure that you enter all of the hypotheses that a particular piece of evidence supports in one run. The reason for this is that Belief needs to undertake some internal calculations related to ignorance, and thus it needs to know also about the hypotheses for which that evidence does not add support. You only need to enter a basic probability assignment image if the evidence supports the hypothesis to some degree larger than zero. For the hypotheses that the evidence does not support, the module assumes 0 probability.
4. For each line of evidence, the basic probability assignment images must be real number images with a range that does not exceed 0-1.

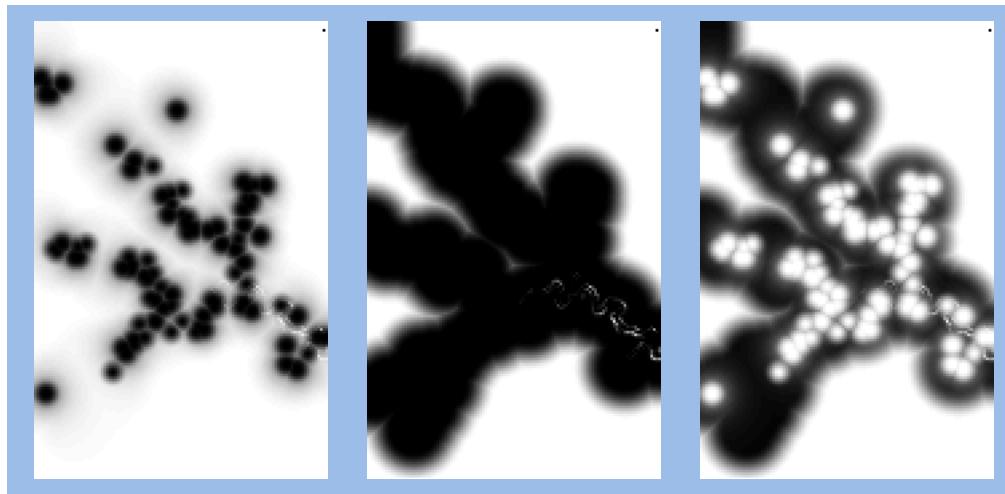


Figure 13-13 Belief (left), Plausibility (middle) and Belief Interval (right) images for the presence of Archaeological Sites after Dempster-Shafer combination of evidence.

Decision Rule Uncertainty and Decision Risk

In the context of measurement error, it is a fairly straightforward matter to relate uncertainty to decision risk. In IDRISI, the PCLASS module achieves this based on the logic of classical sets (as was discussed earlier). However, as we move from the strong frequentist interpretation of probability associated with measurement error, to the more indirect relationship of Bayesian and Dempster-Shafer beliefs, to the quite independently established concept of Fuzzy Sets, we move further and further away from the ability to establish risk in any absolute sense (Eastman, 1996). Indeed, with a decision based on Fuzzy Sets, we can establish that the inclusion of an alternative is less risky than another, but not what the actual risk is. Thus, instead of calculating *absolute risk*, we need to be able to establish *relative risk*.

The concept of relative risk is one that is quite familiar. For example, in evaluating a group of candidates for employment, we might examine a number of quantifiable criteria—grades, rating charts, years of experience, etc.,—that can permit the candidates to be ranked. We then attempt to hire the best ranked individuals on the assumption that they will perform well. However, there is no absolute scale by which to understand the likelihood that they will achieve the goals we set. In a similar manner, the RANK module in IDRISI can be used to rank the suitabilities achieved through a multi-criteria aggregation procedure. This result can then be divided by the maximum rank to produce an image of relative risk. This result can then be thresholded to extract a specific percentage of the best (i.e., least risky) solutions available. The importance of this solution is that it can be applied to any decision surface regardless of the nature of the uncertainties involved.

A Closing Comment

The decision support tools provided in IDRISI are still under active development. We therefore welcome written comments and observations to further improve the modules and enhance their application in real-world situations.

References / Further Reading

- Alonso, W., 1968. Predicting Best with Imperfect Data, *Journal of the American Institute of Planners*, 34: 248-255.
- Bonham-Carter, G.F., Agterberg, F.P. and Wright, D.F., 1988. Integration of Geological Datasets for Gold Exploration in Nova Scotia, *Photogrammetric Engineering and Remote Sensing*, 54(11): 1585-1592.
- Bonissone, P.P. and Decker, K., 1986. Selecting Uncertainty Calculi and Granularity: An Experiment in Trading-Off Precision and Complexity. In L.N. Kanal and J.F. Lemmer eds., *Uncertainty in Artificial Intelligence*, Elsevier Science, Holland.
- Burrough, P.A., 1986. *Principles of Geographical Information Systems for Land Resources Assessment*, Clarendon Press, Oxford.
- Congalton, R.G., 1991. A Review of Assessing the Accuracy of Classifications of Remotely Sensed Data, *Remote Sensing and the Environment*, 37: 35-46.
- Eastman, J.R., 1996. Uncertainty and Decision Risk in Multi-Criteria Evaluation: Implications for GIS Software Design, *Proceedings, UN University International Institute for Software Technology Expert Group Workshop on Software Technology for Agenda'21: Decision Support Systems*, February 26-March 8.
- Eastman, J.R., Kyem, P.A.K., Toledano, J. and Jin, W., 1993. *GIS and Decision Making*, Explorations in Geographic Information System Technology, 4, UNITAR, Geneva.
- Fisher, P.F., 1991. First Experiments in Viewshed Uncertainty: The Accuracy of the Viewshed Area, *Photogrammetric Engineering & Remote Sensing* 57(10): 1321-1327.
- Goodchild, M.F., and Gopal, S., eds., 1989. *Accuracy of Spatial Databases*. Taylor and Francis, London.
- Gordon, J., and Shortliffe, E.H., 1985. A Method for Managing Evidential Reasoning in a Hierarchical Hypothesis Space, *Artificial Intelligence*, 26: 323-357.
- Honea, R.B., Hake, K.A., and Durfee, R.C., 1991. Incorporating GISs into Decision Support Systems: Where Have We Come From and Where Do We Need to Go? In: M. Heit and A. Shortreed (eds.), *GIS Applications in Natural Resources*. GIS World, Inc., Fort Collins, Colorado.
- Klir, George J., 1989. Is There More to Uncertainty Than Some Probability Theorists Might Have Us Believe? *International Journal of General Systems*, 15: 347-378.
- Lee, N.S., Grize, Y.L. and Dehnad, K., 1987. Quantitative Models for Reasoning Under Uncertainty in Knowledge-Based Expert Systems, *International Journal of Intelligent Systems*, 2: 15-38.
- Maling, D.H., 1989. *Measurement from Maps: Principles and Methods of Cartography*, Pergamon Press, Oxford.
- Moellering, H., 1988. Digital Cartographic Data Quality. *The American Cartographer* 15(1).
- Schmucker, K.J., 1982. *Fuzzy Sets, Natural Language Computations and Risk Analysis*. Computer Science Press.
- Slonecker, E.T. and Tosta, N., 1992. National Map Accuracy: Out of Sync, Out of Time. *Geoinfo Systems*, 2(1): 23-26.
- Stoms, D., 1987. Reasoning with Uncertainty in Intelligent Geographic Information Systems. *Proceedings, GIS '87*, 692-700.

Srinivasan, A. and Richards, J.A., 1990. Knowledge-Based Techniques for Multi-Source Classification. *International Journal of Remote Sensing*, 11(3): 505-525.

Yager, R. 1988. On Ordered Weighted Averaging Aggregation Operators in Multicriteria Decision Making, *IEEE Transactions on Systems, Man, and Cybernetics*. 8(1): 183-190.

Zadeh, L.A., 1965. Fuzzy Sets. *Information and Control*, 8: 338-353.

Image Restoration

In the **Introduction to Remote Sensing and Image Processing** chapter, image restoration is broken down into two broad sub-areas: radiometric restoration and geometric restoration. Radiometric restoration is concerned with the fidelity of the readings of electromagnetic energy by the sensor while geometric restoration is concerned with the spatial fidelity of images. However, while the word restoration suggests a return to conditions that once existed, the reality is that image restoration is concerned with establishing measurement conditions that probably never existed—the measurement of radiometric characteristics under perfect and invariant conditions on an abstract geographic reference ellipsoid. Radiometric restoration is thus concerned with issues such as atmospheric haze, sensor calibration, topographic influences on illumination, system noise, and so on. Geometric restorations are less of a burden to the general data user because most geometric restorations are already completed by the imagery distributor. The most significant task the data user must complete is georeferencing. All of these rectifications can be achieved using existing IDRISI modules.

Radiometric Restoration

Sensor Calibration

The detectors on sensors can vary between instruments (such as on successive satellites in a series, such as the NOAA TIROS-N weather satellites) and within an instrument over time or over the face of the image if multiple detectors are used (as is commonly the case). Sensor calibration is thus concerned with ensuring uniformity of output across the face of the image, and across time.

Radiance Calibration

Pixel values in satellite imagery typically express the amount of radiant energy received at the sensor in the form of uncalibrated relative values simply called Digital Numbers (DN). Sometimes these DN are referred to as the brightness values. For many (perhaps most) applications in remote sensing (such as classification of a single-date image using supervised classification), it is not necessary to convert these values. However, conversion of DN to absolute radiance values is a necessary procedure for comparative analysis of several images taken by different sensors (for example, Landsat-2 versus Landsat-5). Since each sensor has its own calibration parameters used in recording the DN values, the same DN values in two images taken by two different sensors may actually represent two different radiance values.

Usually, detectors are calibrated so that there is a linear relationship between DN and spectral radiance. This linear function is typically described by three parameters: the range of DN values in the image, and the lowest (L_{min}) and highest (L_{max}) radiances measured by a detector over the spectral bandwidth of the channel. Most commonly, the data are distributed in 8-bit format corresponding to 256 DN levels. L_{min} is the spectral radiance corresponding to the minimum DN value (usually 0). L_{max} is the radiance corresponding to the maximum DN (usually 255). Not only each sensor, but each band within the same sensor, has its own L_{min} and L_{max} . The information about sensor calibration parameters (L_{min} and L_{max}) is usually supplied with the data or is available elsewhere.⁶⁴ The equation⁶⁵ relating DN in remotely sensed data to radiance is:

64. The Landsat satellites calibration parameters can be found in: EOSAT Landsat Technical Notes No. 1, August 1986, or the Landsat Data User's Handbook.

65. For an explanation of radiance computation from DN, you may wish to consult: Lillesand, T.M. and R.W. Kiefer, 1994. *Remote Sensing and Image Interpretation*. Third Edition. John Wiley and Sons.

$$L = \left(\frac{L_{max} - L_{min}}{255} \right) DN + L_{min}$$

where L is the radiance expressed in $\text{Wm}^{-2} \text{sr}^{-1}$.

Alternatively, the calibration of the sensor may be expressed in the form of an offset and gain. In this case, radiance can be calculated as:

$$L = \text{Offset} + (\text{Gain} * DN)$$

Note that it is also possible to convert between an Offset/Gain specification and L_{min}/L_{max} as follows:

$$\text{Offset} = L_{min}$$

and

$$\text{Gain} = \frac{L_{max} - L_{min}}{255}$$

or alternatively:

$$L_{min} = \text{Offset}$$

$$L_{max} = (\text{Gain} * 255) + L_{min}$$

Either the CALIBRATE or RADIANCE modules in IDRISI can be used to convert raw DN values to calibrated radiances. The RADIANCE module has the most extensive options. It contains a lookup table of L_{min} and L_{max} for both the MSS and TM sensors on Landsat 1-5. For other satellite systems, it permits the user to enter system-specific L_{min}/L_{max} , or Offset/Gain values. CALIBRATE is more specifically geared towards brightness level matching, but does allow for adjustment to a specific offset and gain. In either case, special care must be taken that the calibration coefficients are correctly matched to the output units desired. The most common expression of radiance is in $\text{mWcm}^{-2}\text{sr}^{-1} \text{mm}^{-1}$ (i.e., milliWatts per square centimeter per steradian per micron). However, it is also common to encounter $\text{Wm}^{-2}\text{sr}^{-1} \text{mm}^{-1}$ (i.e., Watts per square meter per steradian per micron).

Band Striping

Striping or banding is systematic noise in an image that results from variation in the response of the individual detectors used for a particular band. This usually happens when a detector goes out of adjustment and produces readings that are consistently much higher or lower than the other detectors for the same band. In the case of MSS data, there are 6 detectors per band which scan in a horizontal direction. If one of the detectors is miscalibrated, then horizontal banding occurs repetitively on each 6th line. Similarly, in the case of TM data with 16 detectors per band, each 16th line in the image will be affected. Multispectral SPOT data have a pushbroom scanner with 3000 detectors for each band, one detector for each pixel in a row. Since detectors in a pushbroom scanner are arranged in a line perpendicular to the satellite orbit track, miscalibration in SPOT detectors produces vertical banding. Since the SPOT satellite has an individual detector for each column of data, there is no repetitive striping pattern in the image.

The procedure that corrects the values in the bad scan lines is called *desstriping*. It involves the calculation of the mean (or median) and standard deviation for the entire image and then for each detector separately. Some software packages offer an option for applying a mask to the image to exclude certain areas from these calculations (for example, clouds and cloud shadows should be excluded). Also, sometimes only a portion of the image, usually a homogeneous area such as a body of water, is used for these calculations. Then, depending on the algorithm employed by a software system, one of the following adjustments is usually made:

1. The output from each detector is scaled to match the mean and standard deviation of the entire image. In this case, the value of each pixel in the image is altered.
2. The output from the problem detector is scaled to resemble the mean and standard deviation of the other detectors. In this case, the values of the pixels in normal data lines are not altered.

The IDRISI module DESTRIPE employs the first method. Results of this transformation are shown in Figures 14-1 and 14-2.

If satellite data are acquired from a distributor already fully georeferenced, then radiometric correction via DESTRIPE is no longer possible. In this case, we suggest a different methodology. One can run an Unstandardized Principal Components Analysis on the collection of input bands. The last few components usually represent less than 1 percent of the total information available and tend to hold information relevant to striping. If these components are removed completely and the rest of the components re-assembled, the improvement can be dramatic as the striping effect can even disappear. To re-assemble component images, it is necessary to save the table information reporting the eigenvectors for each component. In this table, the rows are ordered according to the band number and the column eigenvectors reading from left to right represent the set of transformation coefficients required to linearly transform the bands to produce the components. Similarly, each row represents the coefficients of the reverse transformation from the components back to the original bands. Multiplying each component image by its corresponding eigenvector element for a particular band and summing the weighted components together reproduces the original band of information. If the noise components are simply dropped from the equation, it is possible to compute the new bands, free of these effects. This can be achieved very quickly using the Image Calculator in IDRISI.

The chapter on **Fourier Analysis** details how that technique may also be used to remove striping or noise from satellite imagery.

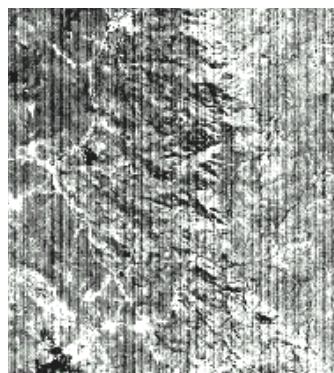


Figure 14-1



Figure 14-2

Mosaicking

Mosaicking refers to the process of matching the radiometric characteristics of a set of images that fit together to produce a larger composite. In IDRISI, the MOSAIC module facilitates this process. The basic logic is to equalize the means and variances of recorded values across the set of images, based on an analysis of comparative values in overlap areas. The first image specified acts as the master, to which all other images are adjusted.

Atmospheric Correction

The atmosphere can affect the nature of remotely sensed images in a number of ways. At the molecular level, atmospheric gases cause Rayleigh scattering that progressively affects shorter wavelengths (causing, for example, the sky to appear blue). Further, major atmospheric components such as oxygen, carbon dioxide, ozone and water vapor (particularly these latter two) cause absorption of energy at selected wavelengths. Aerosol particulates (an aerosol is a gaseous suspension of fine solid or liquid particles) are the primary determinant of haze, and introduce a largely non-selective (i.e., affecting all wavelength equally) Mie scattering. Atmospheric effects can be substantial (see Figure 14-3). Thus remote sensing specialists have worked towards the modeling and correction of these effects. IDRISI offers several approaches to atmospheric correction, with the most sophisticated being the module ATMOSC.

Dark Object Subtraction Model

The effect of haze is usually a relatively uniform elevation in spectral values in the visible bands of energy. One means of reducing haze in imagery is to look for values in areas of known zero reflectance, such as deep water. Any value above zero in these areas is likely to represent an overall increase in values across the image and can be subtracted easily from all values in the individual band using SCALAR. However, ATMOSC also offers a Dark Object Subtraction model with the added benefit that it compensates for variations in solar output according to the time of year and the solar elevation angle. To do this, it requires the same estimate of the Dn of haze (e.g., the Dn of deep clear lakes), the date and time of the image, the central wavelength of the image band, the sun elevation, and radiance conversion parameters. These additional parameters are normally included with the documentation for remotely sensed images.

Cos(t) Model

One of the difficulties with atmospheric correction is that the data necessary for a full accommodation are often not available. The Cos(t) model was developed by Chavez (1996) as a technique for approximation that works well in these instances. It is also available in the ATMOSC module and incorporates all of the elements of the Dark Object Subtraction model (for haze removal) plus a procedure for estimating the effects of absorption by atmospheric gases and Rayleigh scattering. It requires no additional parameters over the Dark Object Subtraction model and estimates these additional elements based on the cosine of the solar zenith angle (90 - solar elevation).

Full Correction Model

The full model is the most demanding in terms of data requirements. In addition to the parameters required for the Dark Object Subtraction and Cos(t) models, it requires an estimate of the optical thickness of the atmosphere (the Help System for ATMOSC gives guidelines for this) and the spectral diffuse sky irradiance (the downwelling diffuse sky irradiance at the wavelength in question arising from scattering—see Forster (1984) and Turner and Spencer (1972). In cases where this is unknown, the default value of 0 can be used.

Apparent Reflectance Model

ATMOSC offers a fourth model known as the Apparent Reflectance Model. It is rarely used since it does very little accommodation to atmospheric effect (it only accommodates the sun elevation, and thus the effective thickness of the atmosphere). It is included, however, as a means of converting Dn into approximate reflectance values.

An Alternative Haze Removal Strategy

Another effective method for reducing haze involves the application of Principal Components Analysis. The PCA module in IDRISI separates a collection of bands into statistically separate components. The method of removing a component is described in the *Noise Effects* section (this chapter) for the removal of image striping. We suggest using this method to reduce any other atmospheric effects as well. Figures 14-3 and 14-4 are Landsat TM Band 1 images before restoration and after. The area is in north-central Vietnam at a time with such heavy amounts of haze relative to ground reflectance as to make sensor banding also very apparent. Principal Components Analysis was applied on the seven original bands. The 6th and 7th components, which comprised less than 0.002 percent of the total information carried across the bands, were dropped when the components were reverse-transformed into the new band. (The reverse-transformation process is detailed above in the section on Band Striping.) Notice that not only do the haze and banding disappear in the second

image, but also what appears to be clouds are greatly reduced.



Figure 14-3



Figure 14-4

Topographic Effects

*Topographic effect*⁶⁶ is defined simply as the difference in radiance values from inclined surfaces compared to horizontal ones. The interaction of the angle and azimuth of the sun's rays with slopes and aspects produce topographic effects resulting in variable illumination. Images are often taken in the early morning hours or late afternoon, when the effect of sun angle on slope illumination can be extreme. In mountainous environments, reflectances of slopes facing away from the sun are considerably lower than the overall reflectance or mean of an image area. In extreme terrain conditions, some areas may be shadowed to the extent that meaningful information is lost altogether.

Shadowing and scattering effects exaggerate the difference in reflectance information coming from similar earth materials. The signature of the same land cover type on opposite facing slopes may not only have a different mean and variance, but may even have non-overlapping reflectance ranges. In the classification process, the highly variable relationship between slope, land cover, and sun angle can lead to a highly exaggerated number of reflectance groups that make final interpretation of data layers more costly, difficult, and time consuming. Even when land covers are not the same on opposite sides of the mountain (which is often the case since slope and aspect help determine the cover type present), variable illumination nonetheless makes it difficult to derive biomass indices or perform other comparisons between land cover classes.

Several techniques for mitigating topographic effect have evolved in recent years. However, many tend to be only appropriate for the specific environment in which they were developed, or they require high-detail ancillary data that is often unavailable. The three most accessible techniques used are band ratioing, partitioning an image into separate areas for classification, and illumination modeling based on a DEM. More sophisticated techniques (which are not discussed here) involve the modeling of such illumination effects as backscattering and indirect diffusion effects.

Band Ratioing

In band ratioing, one band image is divided by another.

$$\frac{\text{BandA}}{\text{BandB}}$$

The resulting output image is then linearly stretched back to the 0 to 255 value range, and used for image classification. Band ratioing is based on the principle that terrain variations (in slope and aspect) cause variations in illumination that are consistent across different wavelengths. Thus in an area with a uniform land cover, the relative reflectance of one band to

66. The Topographic Effects section is condensed from the "Mitigating Topographic Effects in Satellite Imagery" exercise in Schneider and Robbins, 1995. UNITAR Explorations in GIS Technology, Volume 5, *GIS and Mountain Environments*, UNITAR, Geneva, also available from the Clark Labs..

another will be the same regardless of slope and aspect variations. Band ratioing is the simplest technique to implement to mitigate topographic effect. It will not be very effective, however, when the variance in signature ranges is highly compressed. This commonly occurs in extreme shadowing conditions.

Image Partitioning

Image partitioning works from the simple assumption that since different areas within an image are affected differently by illumination effects resulting from slope and aspect, these distinct areas should be classified separately. Using a digital elevation model to produce mask images, the bands are subdivided according to different elevation, slope, and aspect categories. These sub-scenes are then classified separately and the results recombined after classification. Like band ratioing, the technique is simple and intuitive, but is effective only under the right conditions. Such partitioning works best where land cover conditions are stratified environmentally. Otherwise there is the potential to create hundreds of meaningless clusters when using an unsupervised classification or to misclassify pixels when applying supervised techniques.

The thresholds set for slope, aspect, and elevation are dependent upon the known sun angle and azimuth. Without solar information, the significant thresholds of topographic effect may be imprecisely determined by analyzing the shadow effect visually. The registration of the DEM to the satellite data must be as precise as possible, otherwise the inexact nature of thresholds will further increase the possibility of less meaningful classifications.

Illumination Modeling

The analytical tools associated with most raster GIS software systems offer a very effective technique for modeling illumination effects. The steps, using IDRISI, are as follows:

1. Use HISTO to calculate the mean of the image band to be corrected.
2. Using a Digital Elevation Model (DEM) for the image area, use the HILLSHADE (SURFACE) module to create a map of analytical hillshading. This will be the model of illumination effects for all bands and simply needs to be calibrated for use.
3. Use REGRESS to calculate the linear relationship between the hillshading map and the image to be corrected. Use the image as the dependent variable and the hillshading as the independent variable.
4. Use CALIBRATE to apply the offset (the intercept of the regression equation) and gain (the slope of the regression equation) to the hillshading map. The result is a model of the terrain-induced illumination component.
5. Use Image Calculator to subtract the result of the previous step from the original image and then add the mean calculated in the first step. The result is a reasonable estimate of what the image would have looked like if the ground had been flat.

Noise

Noise in images occurs because of any number of mechanical or electronic interferences in the sensing system that lead to transmission errors. Noise either degrades the recorded signal or it virtually eliminates all radiometric information. Noise can be systematic, such as the periodic malfunctioning of a detector, resulting in striping or banding in the imagery. Or it can be more random in character, causing radiometric variations described as having a "salt-and-pepper" appearance. In RADAR imagery, "speckle" occurs because the signal's interaction with certain object edges or buildings produces a highly elevated recording, which when frequent, has a similar effect as "salt-and-pepper" noise.

Scan Line Drop Out

Scan line drop out occurs when temporary signal loss from specific detectors causes a complete loss of data for affected lines. In IDRISI this problem can be solved by a sequence of steps. First, reclassify the affected band in RECLASS to create a boolean mask image in which the pixels where drop-lines are present have a value of 1 and the rest have a value of zero. Then, for horizontal drop lines, run a user-defined 3 x 3 FILTER across the original band containing the following

kernel values:

0	0.5	0
0	0	0
0	0.5	0

This will have the effect of assigning to each pixel the average of the values in the scanlines above and below. If the drop line is vertical, simply rotate the filter kernel values by 90 degrees. Then, use OVERLAY to multiply the mask and the filtered image together. This creates a result in which filtered values only appear in those locations where scan lines were lost. OVERLAY this result on the original band using the COVER operation, which will cause these values to be placed only where the data were lost.

"Salt-and-Pepper" Noise

Random noise often produces values that are abnormally high or low relative to surrounding pixel values. Given the assumption that noisy reflectance values show abrupt changes in reflectance from pixel to pixel, it is possible to use filtering operations to replace these values with another value generated from the interpretation of surrounding pixels. FILTER in IDRISI provides several options for this purpose. A 3 x 3 or 5 x 5 median filter commonly are applied. The noisy pixels are replaced by the median value selected from the neighbors of the specified window. Because all pixels are processed by median filtering, some detail and edges may be lost. This is especially problematic with RADAR imagery because of the particularly high level of speckle that can occur. Therefore, we have included an Adaptive Box filter that is an extension of the common Lee filter. The Adaptive Box filter determines locally within a specified window (3 x 3, 5 x 5, or 7 x 7) the mean and the min/max value range based on a user-specified standard deviation. If the center window value is outside the user-specified range, then it is assumed to be noise and the value is replaced by an average of the surrounding neighbors. You may choose the option of replacing the value with a zero. The filter also allows the user to specify a minimum threshold variance in order to protect pixels in areas of very low variation. See the on-line Help System of the FILTER module to learn more about this highly flexible approach.

Geometric Restoration

As stated in the chapter **Introduction to Remote Sensing and Image Processing**, most elements of geometric restoration associated with image capture are corrected by the distributors of the imagery, most importantly *skew correction* and *scanner distortion correction*. Distributors also sell imagery already *georeferenced*. Georeferencing is not only a restoration technique but a method of reorienting the data to satisfy the specific desires and project requirements of the data user. As such, it is particularly important that georeferenced imagery meets the data user's standards and registers well with other data in the same projection and referencing system.

It is our experience that even if one's standards are not very stringent for the particular imaging task at hand, it is well worth the time one takes to georeference the imagery oneself rather than having the distributor do so. This is true for a number of reasons. First, certain radiometric corrections become more difficult (if not impossible) to perform if the data are already georeferenced. Of particular concern is the ability to reduce the effects of banding, scan line drop, and topographic effects on illumination. If the geometric orientation of the effects is altered, then standard restoration techniques are rendered useless. Given that the severity of these effects is not usually known prior to receiving the data, georeferencing the data oneself is important in maintaining control over the image restoration process.

Another reason to georeference imagery oneself is to gain more control over the spatial uncertainties produced by the georeferencing process. Only then is it possible to know how many control points are used, where they are located, what the quality of each is individually, and what the most satisfying combination of control points is to select. The RESAMPLE module in IDRISI provides the user with significant control over this process. The user may freely drop and add points and evaluate the effects on the overall RMS error and the individual residuals of points as they are fitted to a new

equation. This interactive evaluation is especially important if significant rubbersheeting is required to warp the data to fit the projection needs of one's area.

See the chapter on **Georeferencing** for a broader discussion of this issue. See also the exercise on Georeferencing in the **Tutorial** for a worked example of how the RESAMPLE module is used in IDRISI to georegister a satellite image.

References

Chavez, P.S., (1996) "Image-Based Atmospheric Corrections - Revisited and Improved", Photogrammetric Engineering and Remote Sensing, 62, 9, 1025-1036.

Forster, B.C., (1984) "Derivation of atmospheric correction procedures for Landsat MSS with particular reference to urban data", International Journal of Remote Sensing, 5, 5, 799-817.

Lillesand, T.M. and R.W. Kiefer, (1994) *Remote Sensing and Image Interpretation*. Third Edition. John Wiley and Sons.

Turner, R.E., and Spencer, M.M., (1972) "Atmospheric Model for Correction of Spacecraft Data", Proceedings, Eighth International Symposium on Remote Sensing of the Environment, Vol. II, 895-934.

Fourier Analysis

Fourier Analysis is a signal/image decomposition technique that attempts to describe the underlying structure of an image as a combination of simpler elements. Specifically, it attempts to describe the signal/image as a composite of simple sine waves. Thus the intent of Fourier Analysis is somewhat similar to that of Principal Components Analysis—to break the image down into its structural components for the purpose of analyzing and modifying those components before eventual reconstruction into an enhanced form. While Fourier Analysis has application in a number of fields ranging from optics to electronics, in the context of image processing, it is most often used for noise removal.

The Logic of Fourier Analysis

It is unfortunate that the mathematical treatment of Fourier Analysis makes it conceptually inaccessible to many. Since there are ample treatments of Fourier Analysis from a mathematical perspective, the description offered here is intended as a more conceptual treatment.

Any image can be conceptually understood as a complex wave form. For example, if one were to graph the grey levels along any row or column, they would form the character of a complex wave. For a two dimensional image, the logical extension of this would be a surface, like the surface of an ocean or lake. Imagine dropping a stone into a pool of water—a simple sine wave pattern would be formed with a wave length dependent upon the size of the stone. Now imagine dropping a whole group of stones of varying size and at varying locations. At some locations the waves would cancel each other out while at other locations they would reinforce each other, leading to even higher amplitude waves. The surface would thus exhibit a complex wave pattern that was ultimately created by a set of very simple wave forms. Figures 15-1 and 15-2 illustrate this effect. Figure 15-1 shows a series of sine waves of varying frequency, amplitude, and phase (these terms will be explained below). Figure 15-2 shows the complex wave form that would result from the combination of these waves.

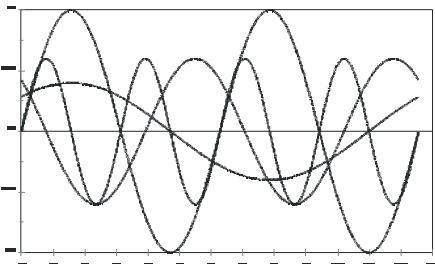


Figure 15-1

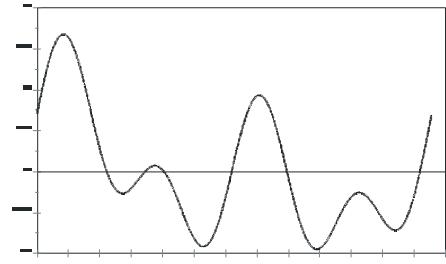


Figure 15-2

Fourier Analysis uses this logic in reverse. It starts with a complex wave form and assumes that this is the result of the additive effects of simple sine waves of varying *frequency*, *amplitude* and *phase*. Frequency refers to the number of complete wavelengths that occur over a specified distance. Figure 15-3 shows a series of sine waves that differ only in their frequency. Amplitude refers to the height or strength of the wave. Figure 15-4 shows a series of sine waves that vary only in amplitude. Finally, phase refers to the offset of the first wave crest from origin. Figure 15-5 shows a series of waves that vary only in phase. In the decomposition of digital images, a finite set of waves are assumed, ranging from the lowest frequency wave which completes a single wavelength over the extent of the image in X and Y, to one that completes 2 wavelengths over that extent, to 3 wavelengths, and so on, up to the highest frequency wave with a wavelength equal to twice the pixel resolution (known as the *Nyquist frequency*). The task of the Fourier Analysis process is then simply one of esti-

mating the phase and amplitudes of these waves.

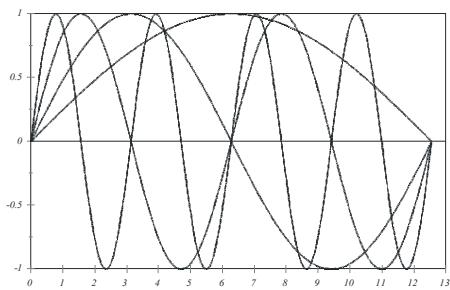


Figure 15-3

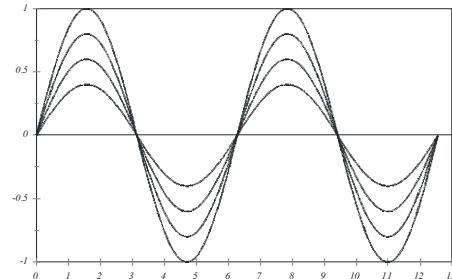


Figure 15-4

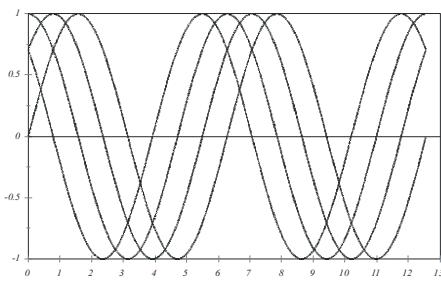


Figure 15-5

How Fourier Analysis Works

The easiest way to understand how Fourier Analysis works is to use an analogy. In the presence of a sound (a complex wave form), the string of a guitar (or any other stringed instrument) will vibrate, or resonate, if the sound contains the same note (frequency). The strength of that sympathetic vibration will be a function of the amplitude of that note within the original sound. In essence, this is how Fourier Analysis works—it "listens" for the degree of resonance of a set of specific frequencies with the sound (image) being analyzed. It is like placing a harp into the presence of a sound, and gauging the degree to which each string resonates.

The process of testing for the presence of varying frequencies is achieved by multiplying the complex wave by the corresponding amplitude of the sine wave being evaluated, and summing the results. The resulting sum is the resonance at that frequency. The only problem, however, is phase. What if the wave in question is present, but our test wave is out of phase? In the worst case they are exactly out of phase, so that the peaks in the test frequency are balanced by troughs in the complex wave—the net effect is that they will cancel each other out, leading one to believe that the wave is not present at all.

The answer to the problem of phase is to test the complex wave against two versions of the same frequency wave, exactly out of phase with each other. This can easily be done by testing both a sine wave and a cosine wave of the same frequency (since sines and cosines are identical except for being exactly out of phase). In this way, if there is little resonance with the sine wave because of a problem of phase, it is guaranteed to resonate with the cosine wave.⁶⁷

67. The use of a sine/cosine pair is identical in concept to describing locations using a pair of coordinates—X and Y. In both cases, the reference pair are known as *basis vectors*. In plane two-dimensional space, any location can be defined by its X and Y coordinates. Similarly, in the frequency domain, any wave can be defined by its sine and cosine coordinates.

Interpreting the Mathematical Expression

Given the discussion above, the formula for the Fourier Series is not so difficult to understand. Considering the one-dimensional case, the complex function over x can be described as the sum of sine and cosine components as follows:

$$f(x) = a_0 + \sum_{k=1}^{\infty} (a_k \cos(k\omega x) + b_k \sin(k\omega x))$$

where $f(x)$ is the value of the function at position x , ω is equal to $2\pi/T$ where T is the period (the length of the series), and k is the harmonic.⁶⁸ The coefficients a and b are determined by means of the Fourier Transform.

The Fourier Transform itself makes use of Euler's Formula:

$$e^{-i2\pi ux} = \cos 2\pi ux - i \sin 2\pi ux$$

where i^2 is -1, leading to the following formulation for the case of discrete data:

$$F(u) = \frac{1}{N} \sum f(x) e^{-ik\omega x}$$

and an inverse formula of:

$$f(x) = \sum F(u) e^{ik\omega x}$$

It is not critical that these mathematical expressions be fully understood in order to make productive use of the Fourier Transform. However, it can be appreciated from the above that:

1. these formulas express the forward and inverse transforms for one-dimensional data. Simple extensions make these applicable to two-dimensional data.
2. the implementation of the Fourier Transform uses complex-valued inputs and outputs. A complex number is one with both real and imaginary parts of the form $a+bi$, where $i^2 = -1$. In the case considered here, for *input* into the FOURIER module, the image grey-level values make up the real part, while the imaginary part is set to zero (this is done automatically by IDRISI) since it doesn't exist. Thus the input to the Fourier Transform is a single image.
3. the resulting *output* of the forward transform thus consists of two images—a real and an imaginary part. The real part expresses the cosine component of the Fourier series while the imaginary part expresses the sine component. These are the amplitudes a and b of the cosine and sine components expressed in the first formula in this section. From these, the amplitude and phase of the wave can be determined as follows:

$$\text{Amplitude} = \sqrt{a^2 + b^2}$$

$$\text{Phase} = \tan^{-1}(b/a)$$

Together, the real and imaginary parts express the frequency spectrum of an image. While both the amplitude and phase can be readily calculated from these parts (Image Calculator can be used in IDRISI to do this), neither is commonly used. More commonly, the power spectrum is calculated and the phase is ignored. The power spectrum is simply the square of the amplitude. However, for purposes of visual examination, it is commonly expressed as a power spectrum image as follows:

$$\text{PowerSpectrum} = \ln(1 + \text{amplitude}^2)$$

68. The term "harmonic" used here refers to the relation of quantities whose reciprocals are in arithmetic progression (e.g., 1, 1/2, 1/3, 1/4, etc.); or to points, lines, functions, etc. involving such a relation.

This is the formula used in IDRISI. Thus the forward transform produced by the FOURIER module yields three outputs—a real part image, an imaginary part image, and a power image. These are commonly referred to as frequency domain images (as opposed to the original image which expresses the spatial domain).

The primary intent of the Fourier Transform is to examine the spectrum and modify its characteristics before reconstructing the original by means of the inverse transform. Examination of the spectrum is done with the power image. However, modifications are implemented on the real and imaginary parts. The FILTERFQ, DRAWFILT and FREQDIST modules can be used to create a variety of filters to be applied to the real and imaginary parts of the frequency domain.

The FOURIER module can also be used to compute the inverse transform. In this case, it is necessary to supply both the real and imaginary parts of the frequency domain. The result is a real image—the imaginary part is assumed to be zero, and is discarded.

The actual implementation of the Fourier Transform in IDRISI is by means of the *Fast Fourier Transform* (FFT) procedure. This algorithm is comparatively very fast, but requires that the image size (in X and Y) be a power of 2 (e.g., 2, 4, 8, 16, 32, 64, 128, etc.). In cases where the image is some other size, the edges can be padded with zeros to make the image conform to the proper dimensions. The ZEROPAD module facilitates this process in IDRISI.

Fourier Analysis assumes that the image itself is periodic to infinity. Thus it is assumed that the image repeats itself endlessly in X and Y. The effect is similar to bending the image in both X and Y such that the last column touches the first and the last row touches the first. If the grey values at these extremes are quite different, their juxtaposition will lead to spurious high frequency components in the transform. This can be mitigated by zero padding the edges. Zero padding is therefore often quite desirable.

Using Fourier Analysis in IDRISI

Gathering together and extending the information above, the following procedures are typical of Fourier Analysis in IDRISI.

1. Prepare the image for analysis. Both the rows and columns must be powers of 2. However, it is not necessary that they be the same. Thus, for example, an original image of 200 columns and 500 rows would need to be padded out to be an image of 256 columns and 512 rows. Use the ZEROPAD module in IDRISI to do this. If the original image already has rows and columns that are a power of 2, this step can be omitted.
2. Run FOURIER with the forward transform using the image prepared in Step 1 as the input. The image can contain byte, integer or real data—the data type is not important. FOURIER will produce three outputs—a real part image, an imaginary part image, and a power spectrum image. The last of these is intended for visual analysis while the first two are used for numeric analysis.
3. Examine the power spectrum image and design a filter (a topic covered at greater length in the next section). To create the filter, use either the FREQDIST module (followed by either RECLASS or FUZZY), the FILTERFQ module, or the DRAWFILT module.
4. Apply the filter to the real and imaginary part images created in Step 2. This is typically done through multiplication using the OVERLAY module.
5. Run FOURIER again and use the modified real and imaginary parts as input to the inverse transform. The result is the reconstructed image. Note that this output is always a real number image which may be converted to either byte or integer form if desired (and the range of values permits). The original image that was submitted to the forward transform must also be supplied. This image provides reference system information. If zero padding was added prior to the forward transform, then that image (with the zero padding) should be given here as the original image.
6. If zero padding was added, use WINDOW to extract the image to a new file.

Interpreting Frequency Domain Images

When first encountered, frequency domain images appear very strange. Indeed it is hard to believe that they contain all the information required to completely restore the full spatial domain image. However, the pixels in the real and imaginary part images contain a complete record of the sine waves that will form the image when combined. In these images as well as the power spectrum image, the position of each pixel in relation to the center cell indicates the frequency of the wave, while the pixel value indicates the amplitude.

For purposes of visual analysis, the power spectrum image is always used since it contains a visually enhanced record of the amplitude of the component waves.⁶⁹ Within this image, each pixel represents a different wave frequency, with the sole exception of the central pixel located at (columns/2) and ((rows/2) - 1). This pixel represents a frequency of 0—an expression of the average grey level of the image, similar in concept to an intercept in regression analysis.

The pixels to the immediate right ((columns/2) + 1) and above (rows/2) represent the lowest frequency (longest wavelength) waves in the decomposition, with a frequency of one complete wave over the entire extent of the image, i.e., a frequency of $(1/(nd))$ where n=number of rows or columns, and d=pixel resolution. Thus with an image of 512 columns by 512 rows, and 30 meter cells, the pixel to the immediate right or above the center represents a frequency of $(1/(nd)) = (1/(512*30)) = 1/15,360$ meters. Similarly, the second-most pixel to the right or above the center represents a frequency of $(2/(nd)) = (2/(512*30)) = 1/7,680$ meters. Likewise, the third-most pixel to the right or above the center represents a frequency of $(3/(nd)) = (3/(512*30)) = 1/5,120$ meters. This logic continues to the right-most and top-most edges, which would have a frequency of $(255/(nd)) = (255/(512*30)) = 1/60.24$ meters. This latter value is only one short of the limiting frequency of $(256/(nd)) = (256/(512*30)) = 1/60$ meters. This limiting frequency is the *Nyquist frequency*, and represents the shortest wavelength that can be described by pixels at a given resolution. In this example, the shortest wave repeats every 60 meters.

The upper-right quadrant describes waves with positive frequencies in X and Y. All other quadrants contain at least one dimension that describes negative waves. For example, the lower-left quadrant describes frequencies that are negative in both X and Y. Negative frequencies are a consequence of the fact that Fourier Analysis assumes that the information analyzed is infinitely periodic. This is achieved by imagining that the original image could be bent into a cylinder shape in both X and Y so that the first and last columns adjoin one another and similarly that the top and last rows adjoin one another. Note also that the upper-right and lower-left quadrants are mirror images of each other as are the upper-left and lower-right quadrants. This symmetry arises from the fact that the input data are real number and not complex number images.

Finally, note that the first column and the last row (and the last column and the first row) represent the amplitude and power of waves at the Nyquist frequency for both positive and negative frequencies—i.e., using the example above, both $(256/(nd)) = (256/(512*30)) = 1/60$ meters and $(-256/(nd)) = (-256/(512*30)) = -1/60$ meters. The reason why these represent both the positive and negative frequencies relates to the cylindrical folding indicated above that is necessary to create an infinitely periodic form.

Given the above logic to the structure of the amplitude and power spectrum images, several key points can be appreciated:

1. Amplitude and power spectrum images have a character that is radial about the central point representing zero frequency.
2. Noise elements are readily apparent as aberrant point or line features in the power spectrum. Linear noise elements will appear at a 90 degree angle in the power spectrum image to their spatial domain direction, e.g., vertical striping in the original image will produce horizontal elements in the power spectrum image.
3. These noise elements can be filtered out by reducing their amplitudes to 0 in the frequency domain and then doing the

69. The phase information is not important for visual analysis, and is lost in the production of the power spectrum image. However, all mathematical manipulations are undertaken on the real and imaginary part images, which together contain complete amplitude and phase information.

inverse transform.

Frequency Domain Filter Design

Figure 15-6a shows an example of an image with severe horizontal banding, while Figure 15-6b shows its power spectrum created with FOURIER. Figure 15-6c shows a notch filter created with FILTERFQ. Both the real and imaginary parts of the frequency domain transform were then multiplied by this filter in order to block out these wavelengths (reduce their amplitudes to 0). Finally, Figure 15-6d shows the result of applying the reverse transform on the modified real and imaginary part images.

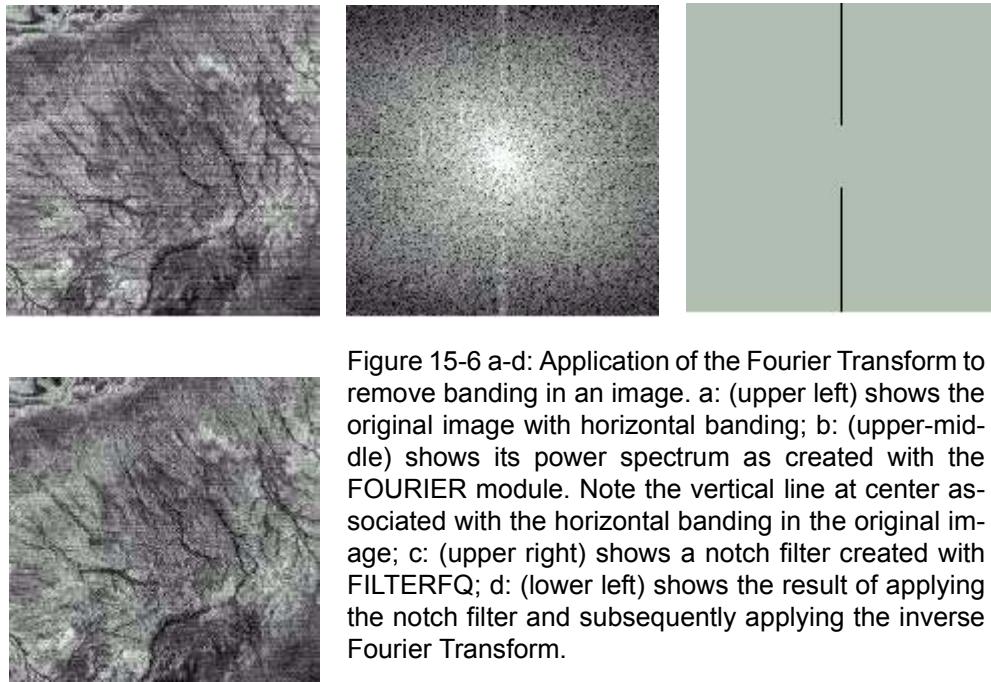


Figure 15-6 a-d: Application of the Fourier Transform to remove banding in an image. a: (upper left) shows the original image with horizontal banding; b: (upper-middle) shows its power spectrum as created with the FOURIER module. Note the vertical line at center associated with the horizontal banding in the original image; c: (upper right) shows a notch filter created with FILTERFQ; d: (lower left) shows the result of applying the notch filter and subsequently applying the inverse Fourier Transform.

IDRISI supplies a variety of facilities for developing frequency domain filters. One would most commonly use FILTERFQ, which offers 26 filters, each of which can be controlled for the specific characteristics of the data being manipulated as well as for the specific purpose of the filter. The next most commonly used module is DRAWFILT, an interactive filter development tool in which one literally draws the areas (i.e., frequencies) to be removed. Finally, for even further flexibility, IDRISI offers a module named FREQDIST. As the name suggests, this module creates a frequency distance image (as measured from the center point of the power spectrum). This can then be used as the basic input to a variety of filter shaping tools such as RECLASS or FUZZY. The frequency distance image can also be submitted to SURFACE to create an aspect image which can then be shaped by RECLASS or FUZZY to create directional filters.

Regardless of how the filter is created, however, application of that filter is achieved the same way in all cases—by simple multiplication using OVERLAY with both the real and imaginary part images. As it turns out, multiplication in the frequency domain is the equivalent of convolution in the spatial domain.

References

Good references for frequency domain filtering include:

Gonzalez, R.C., and Woods, R.E., 1992. *Digital Image Processing*, Addison-Wesley, Reading, Massachusetts.

Mather, P., 1987. *Computer Processing of Remotely Sensed Images*, John Wiley and Sons, New York.

Jensen, J.R., 1996. *Introductory Digital Image Processing: A Remote Sensing Perspective*, Prentice Hall, Upper Saddle River, NJ.

Classification of Remotely Sensed Imagery

Introduction

Classification is the process of developing interpreted maps from remotely sensed images. As a consequence, classification is perhaps the most important aspect of image processing to GIS. Traditionally, classification was achieved by visual interpretation of features and the manual delineation of their boundaries. However, with the advent of computers and digital imagery, attention has focused on the use of computer-assisted interpretation. Although the human eye still brings a superior set of capabilities to the classification process, the speed and consistency of digital procedures make them very attractive. As a consequence, the majority of classification projects today make use of digital classification procedures, guided by human interpretation.

Supervised versus Unsupervised Classification

As indicated in the **Introduction to Remote Sensing and Image Processing** chapter, there are two basic approaches to the classification process: supervised and unsupervised classification. With supervised classification, one provides a statistical description of the manner in which expected land cover classes should appear in the imagery, and then a procedure (known as a *classifier*) is used to evaluate the likelihood that each pixel belongs to one of these classes. With unsupervised classification, a very different approach is used. Here another type of classifier is used to uncover commonly occurring and distinctive reflectance patterns in the imagery, on the assumption that these represent major land cover classes. The analyst then determines the identity of each class by a combination of experience and ground truth (i.e., visiting the study area and observing the actual cover types).

In both of these cases, the process of classification can be seen as one of determining the set to which each pixel belongs. In the case of supervised classification, the sets are known (or assumed to be known) before the process is begun. Classification is thus a decision making process based upon available information. With unsupervised classification, however, the classes are unknown at the outset. Thus the process is really one of segmentation rather than decision making per se.

Spectral Response Patterns versus Signatures

As explained in the **Introduction to Remote Sensing and Image Processing** chapter, each material interacts with electromagnetic energy by either reflecting, absorbing or transmitting it, with the exact nature of that interaction varying from one wavelength to the next—a pattern known as a *Spectral Response Pattern* (SRP). The basis for classification is thus to find some area of the electromagnetic spectrum in which the nature of that interaction is distinctively different from that of other materials that occur in the image. Many refer to this as a *signature*—a spectral response pattern that is characteristic of that material. However, in practice, the determination of consistently distinctive signatures is difficult to achieve for the following reasons:

- most vegetation types do not have consistent spectral response patterns—phenological changes throughout the growing season can lead to highly variable signatures.
- changes in illumination (because of slope or the time of year) and moisture variations can also lead to significantly different spectral response patterns.
- most land covers consist of mixtures of elementary features that are sensed as single pixels. For example, a row crop such as maize actually contains a mixture of plant and soil as sensed by a satellite. Likewise, a pixel may contain a mixture of conifers and deciduous species in a forest area.
- for a given sensor, there is no guarantee that the wavelengths in which it senses will be the same as those in which a material is most distinctive. Currently, multispectral sensors examine several very important areas of the spectrum, partic-

ularly for the differentiation of vegetation. However, the usable areas not examined far outnumber those that are, and many of the wavelengths that could potentially distinguish many rock types, for example, are not typically examined.

As a result of these problems, there has been a strong orientation within the remote sensing community to develop signatures with reference to specific examples within the image to be classified rather than relying on the use of more general libraries of characteristic spectral response patterns. These very specific examples are called *training sites*, named thus because they are used to train the classifier on what to look for. By choosing examples from within the image itself (usually confirmed by a ground truth visit), one develops signatures that are specific to the wavelengths available. One also avoids the problems of variations in both solar zenith angle and stage of the growing season. One can also choose examples that are characteristic of the various cover class mixtures that exist.

Despite this very pragmatic approach to the classification process, it remains very much a decision problem. We ask the process to create a definitive classification in the presence of considerable variation. For example, despite differences in growth stage, soil background and the presence of intercropping, we ask the process to distill all variations of maize cropping into a single maize class.

Recently, however, interest has focused on relaxing this traditional approach in two areas, both strongly represented in IDRISI. The first is the development of *soft classifiers*, while the second extends the logic of multispectral sensing to *hyperspectral sensing*.

Hard versus Soft Classifiers

Traditional classifiers can be called *hard classifiers* since they yield a hard decision about the identity of each pixel. In contrast, *soft classifiers* express the degree to which a pixel belongs to each of the classes being considered. Thus, for example, rather than deciding that a pixel is either deciduous or coniferous forest, it might indicate that its membership grade in the deciduous class is 0.43 and coniferous is 0.57 (which a hard classifier would conclude is coniferous). One of the motivations for using a soft classifier is to determine the mixture of land cover classes present. If we could assume that these two classes were the only ones present, it might be reasonable to conclude that the pixel contains 43% deciduous cover and 57% coniferous. Such a conclusion is known as *sub-pixel classification*.

A second motivation for the use of a soft classifier is to measure and report the strength of evidence in support of the best conclusion that can be made. IDRISI introduces special soft classifiers that allow us to determine, for example, that evidence for deciduous is present to a level of 0.26, for coniferous to 0.19 and some unknown type to 0.55. This would immediately suggest that while the pixel has some similarities to our training sites for these two classes, it really belongs to some type that we have not yet identified.

A third motivation for the use of soft classifiers concerns the use of GIS data layers and models to supplement the information used to reach a final decision. For example, one might extract a mapping of the probability that each pixel belongs to a residential land cover class from the spectral data. Then a GIS data layer of roads might be used to develop a mapping of distance from roads, from which the probability of *not* being residential might be deduced (areas away from roads are unlikely to be residential). These two lines of evidence can then be combined to produce a stronger statement of the probability that this class exists. A final hard decision can subsequently be achieved by submitting the individual class membership statements to an appropriate *hardener*—a decision procedure that chooses the most likely alternative.

Multispectral versus Hyperspectral Classifiers

The second major new development in classifiers is the use of *hyperspectral* data. Most sensors today are termed *multispectral* in that they sense electromagnetic energy in more than one area of the spectrum at once. Each of these areas is called a band and is represented by a single monochrome image. For example, the Landsat Thematic Mapper (TM) sensor system images seven simultaneous bands in the blue (Band 1), green (Band 2), red (Band 3), near infrared (Band 4), middle infrared (Bands 5 and 7) and thermal infrared (Band 6) wavelength areas. Hyperspectral sensors are really no different in concept except that they image in many narrowly defined bands. For example, the AVIRIS experimental system developed by the Jet Propulsion Laboratory (JPL) images in 224 bands over a somewhat similar wavelength range as the TM sensor.

Similarly, the EOS-MODIS system, launched in December 1999, spreads 36 bands over essentially the same range as that covered by the five bands on the corresponding AVHRR system of the NOAA series satellites.

It is tempting to think that more is better—i.e., that the greater number and higher spectral resolution of hyperspectral bands would naturally lead to better classifications. However, this is not necessarily the case. Hyperspectral images are most often highly correlated with other bands of similar wavelength. Thus one must process a substantially increased amount of data (which does affect the level of sophistication of the classifier algorithm) without a corresponding gain of information. The real benefit of hyperspectral imagery is gained from the ability to prospect for very narrow absorption features (spectral regions exhibiting strong absorption from specific materials) at high resolution. This has achieved most notable success in the context of geological applications. For example, recent extraterrestrial missions such as the NASA Mars Surveyor, Galileo, and Cassini missions all carry hyperspectral sensors for the purpose of mineral mapping. The high spectral resolution of these systems provides the ability to measure mineral absorption patterns with high precision, leading to the ability to map both the presence and abundance of surficial materials. Hyperspectral classification is still quite new and effectively experimental in character. IDRISI includes a range of procedures for working with these data.

Overview of the Approach in this Chapter

In the sections that follow, we will cover the general logic and strategy to be used in working with the classification modules in the IDRISI system. Detailed notes on the use of each module can be found in the on-line Help System. Furthermore, examples in the form of exercises can be found in the **Tutorial** manual.

Supervised Classification

General Logic

There is a consistent logic to all of the supervised classification routines in IDRISI, regardless of whether they are hard or soft classifiers. In addition, there is a basic sequence of operations that must be followed no matter which of the supervised classifiers is used. This sequence is described here. The **Tutorial** manual also contains worked examples of this process.

1. Define Training Sites

The first step in undertaking a supervised classification is to define the areas that will be used as training sites for each land cover class. This is usually done by using the on-screen digitizing feature as outlined in the **Using IDRISI** chapter. You should choose a band with strong contrast (such as a near-infrared band) or a color composite for use in digitizing. Then display that image on the screen (use autoscaling if necessary to gain good contrast) and use the on-screen digitizing feature to create one or more vector files of training site polygons—vector outlines of the training site areas.⁷⁰

Good training site locations are generally those with as pure a sample of the information class as possible. For example, if you were choosing to define a site of deciduous forest, it would be important to choose an area that was not mixed with conifers and that had little soil background or understory vegetation visible. When digitizing training sites you should also

70. IDRISI offers two procedures for the digitizing of training site polygons. With the default procedure, one digitizes a set of points that form the boundary of the training site polygon. The second procedure creates the polygon by aggregating together all contiguous pixels surrounding a designated point that fall within a specified tolerance of the spectral characteristics of the central pixel. This is called a flood polygon since it is analogous to the concept of water flowing outward from the designated point. You will also note that with this option a maximum distance can be specified to limit how far this growth procedure will spread. Note that the system also allows you to define training sites by means of a raster image. In some instances, it may make sense to define these locations by direct reference to ground locations (such as by means of point locations gathered with a GPS). However, this requires very exact prior georeferencing of the image, and a confidence that positional errors will not include unwanted pixels in the training sites. Some georeferencing procedures also alter image characteristics which may be undesirable. It is for these reasons that classification is commonly undertaken on ungeoreferenced imagery using on-screen digitizing of training sites. The final classified image is then georeferenced at a later stage.

avoid including any pixels belonging to adjacent land covers. This will be easiest to achieve if you zoom in on the area before digitizing that site.

In general, you should aim to digitize enough pixels so that there are *at least* 10 times as many pixels for each training class as there are bands in the image to classify. Thus, for a Landsat TM image with seven bands, you should aim to have at least 70 pixels per training class (more than that is not difficult to achieve and is recommended—the more the better). If this is difficult to achieve with a single site, simply digitize more than one training site for that class. The on-screen digitizing facility requires that you give an integer identifier for each feature. Thus to digitize more than one training site for a land cover class, simply assign the same identifier to each example. It may be helpful to make a list of ID's and their corresponding information classes.

Finally, note that there is no requirement that all training sites be included in a single vector file created with the on-screen digitizing feature. You can create a single vector file for each information class if you wish. This will simply require that you undertake the signature development stage for each of these files. Alternatively, you can join these vector files into a single vector file with CONCAT or rasterize all the vector files into a single raster image and develop the signatures from that single vector file or raster image.

2. Extract Signatures

After the training site areas have been digitized, the next step will be to create statistical characterizations of each informational class. These are called *signatures* in IDRISI. This is usually achieved with the MAKESIG module.⁷¹ MAKESIG will ask for the name of the vector or raster file that contains the training sites for one or more informational classes, and the bands to be used in the development of signatures. It will then ask for a name for each of the included classes. These names should be suitable as IDRISI file names since they will be used to create signatures files (.sig extension) for each informational class. If you used more than one vector file to store your training site polygons, run MAKESIG for each of these files. Your goal is to create a SIG file for every informational class.

SIG files contain a variety of information about the land cover classes they describe.⁷² These include the names of the image bands from which the statistical characterization was taken, the minimum, maximum and mean values on each band, and the full variance/covariance matrix associated with that multispectral image band set for that class. To examine the contents of this file in detail, use SIGCOMP. Note also that the SEPSIG module can be used to assess the distinctiveness of your set of signatures.

3. Classify the Image

The third (and sometimes final) step is to classify the image. This can be done with any of the hard or soft classifiers described below. Clearly there are many choices here. However, here are some tips:

- the parallelepiped procedure (the PIPED module) is included for pedagogic reasons only. Generally it should not be used.
- when training sites are known to be strong (i.e., well-defined with a large sample size), the MAXLIKE procedure should be used. However, if there are concerns about the quality of the training sites (particularly their uniformity), the MINDIST procedure with standardized distances should be used. The MINDIST module with the standardized distances option is a very strong classifier and one that is less susceptible to training site problems than MAXLIKE.
- the Fisher Classifier can perform exceptionally well when there are not substantial areas of unknown classes and when the training sites are strongly representative of their informational classes.

71. The FUZSIG module offers an interesting, but quite different logic for extracting signatures from impure training sites. This is discussed further in the section on fuzzy signatures below. The ENDSIG module also creates signatures for use with the UNMIX classifier.

72. Each SIG file also has a corresponding SPF file that contains the actual pixel values used to create the SIG file. It is used only by HISTO in displaying histograms of signatures.

- for sub-pixel classification (i.e., analysis of mixture components), use one of the the UNMIX soft classifiers. The other soft classifiers are primarily used as part of an In-Process Classification Assessment (IPCA) process (see the next stage). They are also used in cases where GIS modeling is envisioned as a significant part of the classification process.
- when a new area is first being considered for classification, consider using CLUSTER as a precursor to the selection of training sites.

4. In-Process Classification Assessment (IPCA)

The key concern that an analyst faces in classification is the accuracy of the classification process. Traditionally this is addressed through an accuracy assessment, as described in a later section below. However, with the soft classifiers in IDRISI, an *In-Process Classification Assessment* (IPCA) procedure is feasible. As the name implies, this is an assessment that is undertaken as part of an iterative process of classification improvement, and typically involves the comparison of a group of classified results.

IPCA is very much an experimental procedure at this time, and requires knowledge of the soft classification procedures that are discussed later in this chapter. However, the concept is quite simple. The process involves a comparison of the results of a hard classifier and its corresponding soft classifier. For example, for the MINDIST hard classifier, the FUZCLASS soft classifier (un-normalized option) would be used, whereas for the MAXLIKE hard classifier, the BELCLASS soft classifier would be used. Each of these soft classifiers outputs a *classification uncertainty* image which expresses the degree of difficulty the classifier has in determining a single class to assign to a pixel. Areas of high uncertainty are clearly those that need work in terms of refining the classification.

There are two basic reasons for high uncertainty on the part of a classifier. The first is that the pixel contains a mixture of more basic categories and thus cannot easily be assigned to just one interpretation. The other is that the pixel doesn't look like any of the signatures provided.⁷³

In the case where uncertainty is high because of the presence of a mixture of classes, two possibilities exist. Either the training data are poor, and thus not adequately distinctive to separate these two classes, or the mixture class truly exists at the resolution of the analysis. Significant mixtures should be examined carefully, preferably with a ground visit to resolve the problem. Then consider either developing new training sites for the confused classes, or consider adding a new class, with an appropriate training site, to represent the indistinguishable mixture.

Note that in cases where the ultimate classifier is MAXLIKE, the MAXSET classifier (as described in the section on Unsupervised Classification below) can be used as a very efficient means of identifying mixtures in combination with the classification uncertainty image from BELCLASS.

In cases where uncertainty is high because there is no strong match to any of the training sites provided, a ground truth visit should be considered to determine the identity of the missed class. This class should then be added with an appropriate training site.

Clearly the purpose of IPCA is to identify where problems in the classification process are occurring and to rectify them through an iterative process. Progressive refining or redefining of training sites and subsequent reclassification of the image would be followed by further assessment.

5. Generalization

The fifth stage is optional and frequently omitted. After classification, there may be many cases of isolated pixels that belong to a class that differs from the majority that surround them. This may be an accurate description of reality, but for mapping purposes, a very common post-processing operation is to generalize the image and remove these isolated pixels.

73. This second possibility does not exist if one uses the BAYCLASS module or FUZCLASS with normalized output. Both of these soft classification procedures assume that the classes considered are the only ones possible. It is for this reason that the BELCLASS procedure is recommended for comparison to MAXLIKE and the FUZCLASS with un-normalized output is recommended for comparison to MINDIST.

This is done by passing a mode filter over the result (using the FILTER module in IDRISI). The mode filter replaces each pixel with the most frequently occurring class within a 3x3 window around each pixel. This effectively removes class patches of one or a few pixels and replaces them with the most common neighboring class. Use this operation with care—it is a generalization that truly alters the classified result.

6. Accuracy Assessment

The final stage of the classification process usually involves an accuracy assessment. Traditionally this is done by generating a random set of locations (using the stratified random option of the SAMPLE module in IDRISI) to visit on the ground for verification of the true land cover type. A simple values file is then made to record the true land cover class (by its integer index number) for each of these locations. This values file is then used with the vector file of point locations to create a raster image of the true classes found at the locations examined. This raster image is then compared to the classified map using ERRMAT. ERRMAT tabulates the relationship between true land cover classes and the classes as mapped. It also tabulates *errors of omission* and *errors of commission* as well as the overall proportional error.

The size of the sample (n) to be used in accuracy assessment can be estimated using the following formula:

$$n = \zeta^2 pq / e^2$$

where

ζ is the standard score required for the desired level of confidence (e.g., 1.96 for 95% confidence, 2.58 for 99%, etc.) in the assessment

e is the desired confidence interval (e.g., 0.01 for $\pm 10\%$)

p is the *a priori* estimated proportional error, and

$$q = 1 - p$$

Hard Classifiers

The hard classifiers are so named because they all reach a hard (i.e., unequivocal) decision about the class to which each pixel belongs. They are all based on a logic that describes the expected position of a class (based on training site data) in what is known as *band space*, and then gauging the position of each pixel to be classified in the same band space relative to these class positions. From this perspective, the easiest classifier to understand is the MINDIST procedure.

Minimum-Distance-to-Means

The MINDIST module implements a *Minimum-Distance-to-Means* classifier. Based on training site data, MINDIST characterizes each class by its mean position on each band. For example, if only two bands were to be used, Figure 16-1 might characterize the positions of a set of known classes as determined from the training site data.

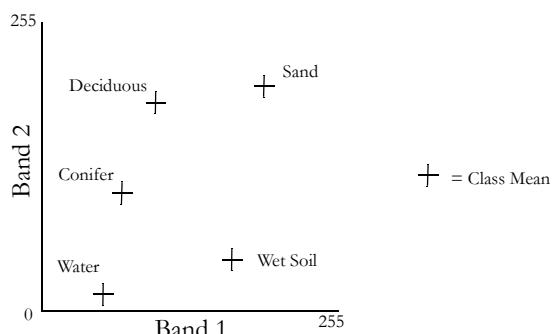


Figure 16-1

Here each axis indicates reflectance on one of the bands. Thus, using the mean reflectance on these bands as X,Y coordinates, the position of the mean can be placed in this band space. Similarly, the position of any unclassified pixel can also be placed in this space by using its reflectance on the two bands as its coordinates.

To classify an unknown pixel, MINDIST then examines the distance from that pixel to each class and assigns it the identity of the nearest class. For example, the unclassified pixel shown in Figure 16-2 would be assigned the "sand" class since this is the class mean to which it is closest.

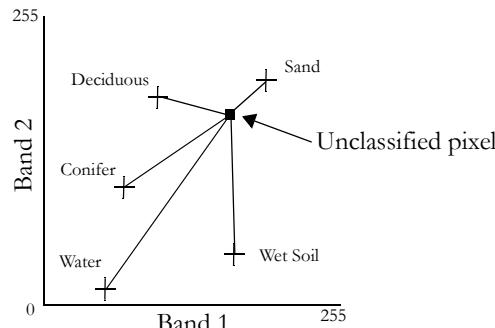


Figure 16-2

Despite the simplicity of this approach, it actually performs quite well. It is reasonably fast and can employ a maximum distance threshold which allows for any pixels that are unlike any of the given classes to be left unclassified. However, the approach does suffer from problems related to signature variability. By characterizing each class by its mean band reflectances only, it has no knowledge of the fact that some classes are inherently more variable than others. This, in turn, can lead to misclassification. For example, consider the case of a highly variable deciduous class and a very consistent sand class in classifying the unclassified pixel in Figure 16-3.

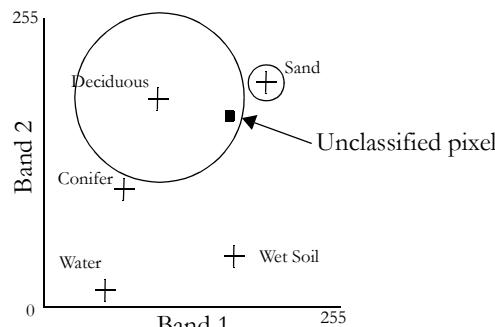


Figure 16-3

The circles in this figure illustrate the variability of each of these two classes. If we assume that these circles represent a distance of two standard deviations from the mean, we can see that the pixel lies within the variability range of the deciduous category, and outside that of sand. However, we can also see that it is closer to the mean for sand. In this case, the classifier would misclassify the pixel as sand when it should really be considered to be deciduous forest.

This problem of variability can be overcome if the concept of distance is changed to that of standard scores. This transformation can be accomplished with the following equation:

$$\text{standardized distance} = (\text{original distance} - \text{mean}) / \text{standard deviation}$$

The MINDIST procedure in IDRISI offers this option of using standardized distances, which is highly recommended. In

in the example above, the pixel would be correctly classified as deciduous since its standardized distance from the mean for deciduous would be less than 2 (perhaps 1.95 in this illustration), while that for sand would be greater than 2 (probably close to 4 in this illustration).

Our experience with MINDIST has been that it can perform very well when standardized distances are used. Indeed, it often outperforms a maximum likelihood procedure whenever training sites have high variability.

Parallelepiped

The PIPED module implements the parallelepiped procedure for image classification. The parallelepiped procedure characterizes each class by the range of expected values on each band. This range may be defined by the minimum and maximum values found in the training site data for that class, or (more typically) by some standardized range of deviations from the mean (e.g., ± 2 standard deviations). With multispectral image data, these ranges form an enclosed box-like polygon of expected values known as a *parallelepiped*. Unclassified pixels are then given the class of any parallelepiped box they fall within. If a pixel does not fall within any box, it is left unassigned. Figure 16-4 illustrates this effect.

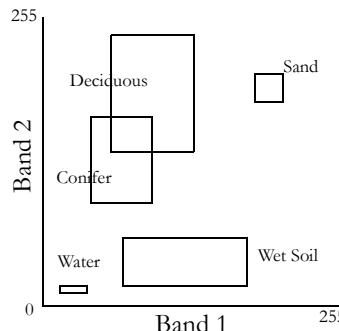


Figure 16-4

This classifier has the advantage of speed and the ability to take into account the differing variability of classes. In addition, the rectangular shape accommodates the fact that variability may be different along different bands. However, the classifier generally performs rather poorly because of the potential for overlap of the parallelepipeds. For example, the conifer and deciduous parallelepipeds overlap in this illustration, leaving a zone of ambiguity in the overlap area. Clearly, any choice of a class for pixels falling within the overlap is arbitrary.

It may seem that the problem of overlapping parallelepipeds would be unlikely. However, they are extremely common because of the fact that image data are often highly correlated between bands. This leads to a cigar-shaped distribution of likely values for a given class that is very poorly approximated by a parallelepiped as shown in Figure 16-5.

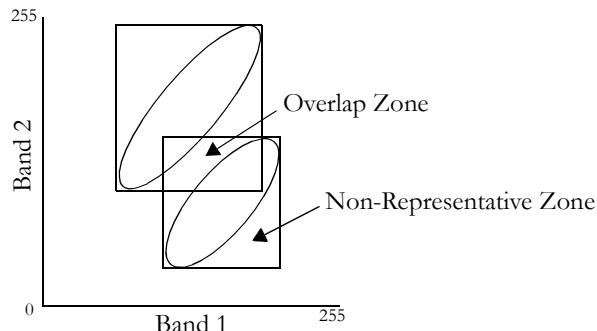


Figure 16-5

Clearly the MINDIST procedure would not encounter this problem, since the line of separation between these classes would fall in between these two distributions. However, in this context of correlation between bands (which is virtually guaranteed), the parallelepiped procedure produces both zones of overlap and highly non-representative areas that really should not be included in the class. In general, then, the parallelepiped procedure should be avoided, despite the fact that it is the fastest of the supervised classifiers.⁷⁴

Maximum Likelihood

To compensate for the main deficiencies of both the Parallelepiped and Minimim-Distance-to-Means procedures, the Maximum Likelihood procedure, provided by the MAXLIKE module in IDRISI, is used. The Maximum Likelihood procedure is based on Bayesian probability theory. Using the information from a set of training sites, MAXLIKE uses the mean and variance/covariance data of the signatures to estimate the posterior probability that a pixel belongs to each class.

In many ways, the MAXLIKE procedure is similar to MINDIST with the standardized distance option. The difference is that MAXLIKE accounts for intercorrelation between bands. By incorporating information about the covariance between bands as well as their inherent variance, MAXLIKE produces what can be conceptualized as an elliptical zone of characterization of the signature. In actuality, it calculates the posterior probability of belonging to each class, where the probability is highest at the mean position of the class, and falls off in an elliptical pattern away from the mean, as shown in Figure 16-6.

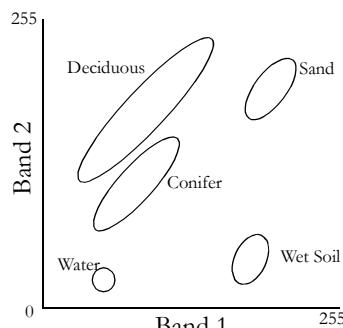


Figure 16-6

74. In the early days of image processing when computing resources were poor, this classifier was commonly used as a *quick look* classifier because of its speed.

Linear Discriminant Analysis (Fisher Classifier)

The final classifier to be discussed in this section is more difficult to describe graphically. The FISHER classifier conducts a linear discriminant analysis of the training site data to form a set of linear functions that express the degree of support for each class. The assigned class for each pixel is then that class which receives the highest support after evaluation of all functions. These functions have a form similar to that of a multivariate linear regression equation, where the independent variables are the image bands, and the dependent variable is the measure of support. In fact, the equations are calculated such that they maximize the variance between classes and minimize the variance within classes. The number of equations will be equal to the number of bands, each describing a hyperplane of support. The intersections of these planes then form the boundaries between classes in band space.

Of the four hard supervised classifiers, MAXLIKE and FISHER are clearly the most powerful. They are also, not surprisingly, the slowest to calculate. However, with high-quality (i.e., homogenous) training sites, they are both capable of producing excellent results.

Soft Classifiers

Unlike hard classifiers, soft classifiers defer making a definitive judgment about the class membership of any pixel in favor of a group of statements about the degree of membership of that pixel in each of the possible classes. Like traditional supervised classification procedures, each uses training site information for the purpose of classifying each image pixel. However, unlike traditional hard classifiers, the output is not a single classified land cover map, but rather a set of images (one per class) that express for each pixel the degree of membership in the class in question. In fact, each expresses the degree to which each pixel belongs to the set identified by a signature according to one of the following set membership metrics:

- BAYCLASS based on Bayesian probability theory,
- BELCLASS based on Dempster-Shafer theory,
- MAHALCLASS based on Mahalanobis distance,
- FUZCLASS based on Fuzzy Set theory, and
- UNMIX based on the Linear Mixture model.

It is important to recognize that each of these falls into a general category of what are known as *Fuzzy Measures* (Dubois and Prade, 1982) of which Fuzzy Sets is only one instance. Fuzziness can arise for many reasons and not just because a set is itself fuzzy. For example, measurement error can lead to uncertainty about the class membership of a pixel even when the classes (sets) are crisply defined. It is for this reason that we have adopted the term *soft*—it simply recognizes that the class membership of a pixel is frequently uncertain for reasons that are varied in origin.

Image Group Files

Since the output of each of these soft classifiers is a set of images, each also outputs a raster image group file (.rgf). This can be used with cursor query to examine the set membership values for a pixel in each class simultaneously in either numeric or graph form (see the on-line Help System section on Display). Note that the classification uncertainty image described below is also included in each group file produced.

Classification Uncertainty

In addition to these set membership images, each of these soft classifiers outputs an image that expresses the degree of *classification uncertainty* it has about the class membership of any pixel. Classification uncertainty measures the degree to which no class clearly stands out above the others in the assessment of class membership of a pixel. In the case of BAYCLASS, BELCLASS and FUZCLASS, it is calculated as follows:

$$\text{ClassificationUncertainty} = 1 - \frac{\max - \frac{\text{sum}}{n}}{1 - \frac{1}{n}}$$

where

- \max = the maximum set membership value for that pixel
- sum = the sum of the set membership values for that pixel
- n = the number of classes (signatures) considered

The logic of this measure is as follows:

- The numerator of the second term expresses the difference between the maximum set membership value and the total dispersion of the set membership values over all classes.
- The denominator of the second term expresses the extreme case of the difference between a maximum set membership value of 1 (and thus total commitment to a single class) and the total dispersion of that commitment over all classes.
- By taking the ratio of these two quantities, one develops a measure that expresses the degree of commitment to a specific class relative to the largest possible commitment that can be made. Classification uncertainty is thus the complement of this ratio.

In spirit, the measure of uncertainty developed here is similar to the entropy measure used in Information Theory. However, it differs in that it is concerned not only with the degree of dispersion of set membership values between classes, but also the total amount of commitment present. Following are some examples that can clarify this concept.

Examples

Assuming a case where three classes are being evaluated, consider those with the following allocations of set membership:

(0.0 0.0 0.0)	Classification Uncertainty = 1.00
(0.0 0.0 0.1)	Classification Uncertainty = 0.90
(0.1 0.1 0.1)	Classification Uncertainty = 1.00
(0.3 0.3 0.3)	Classification Uncertainty = 1.00
(0.6 0.3 0.0)	Classification Uncertainty = 0.55
(0.6 0.3 0.1)	Classification Uncertainty = 0.60
(0.9 0.1 0.0)	Classification Uncertainty = 0.15
(0.9 0.05 0.05)	Classification Uncertainty = 0.15
(1.0 0.0 0.0)	Classification Uncertainty = 0.00

With UNMIX, however, classification uncertainty is measured as the residual error after calculation of the fractions of constituent members. This will be discussed further below.

BAYCLASS and Bayesian Probability Theory

BAYCLASS is a direct extension of the MAXLIKE module. It outputs a separate image to express the posterior probability of belonging to each considered class according to Bayes' Theorem:

$$p(h|e) = \frac{p(e|h) \cdot p(h)}{\sum_i p(e|h_i) \cdot p(h_i)}$$

where :

- $p(h|e)$ = the probability of the hypothesis being true given the evidence (posterior probability)
- $p(e|h)$ = the probability of finding that evidence given the hypothesis being true

$p(h)$ = the probability of the hypothesis being true regardless of the evidence (prior probability)

In this context, the variance/covariance matrix derived from training site data is that which allows one to assess the multivariate conditional probability $p(e|h)$. This quantity is then modified by the prior probability of the hypothesis being true and then normalized by the sum of such considerations over all classes. This latter step is important in that it makes the assumption that the classes considered are the only classes that are possible as interpretations for the pixel under consideration. Thus even weak support for a specific interpretation may appear to be strong if it is the strongest of the possible choices given.

This posterior probability $p(h|e)$ is the same quantity that MAXLIKE evaluates to determine the most likely class, and indeed, if the output images of BAYCLASS were to be submitted directly to HARDEN, the result would be identical to that of MAXLIKE. In essence, BAYCLASS is a *confident* classifier. It assumes that the only possible interpretation of a pixel is one of those classes for which training site data have been provided. It therefore admits to no ignorance. As a result, lack of evidence for an alternative hypothesis constitutes support for the hypotheses that remain. In this context, a pixel for which reflectance data only very weakly support a particular class is treated as unequivocally belonging to that class ($p = 1.0$) if no support exists for any other interpretation.

The prime motivation for the use of BAYCLASS is sub-pixel classification—i.e., to determine the extent to which mixed pixels exist in the image and their relative proportions. It is also of interest to observe the underlying basis of the MAXLIKE procedure. However, for *In-Process Classification Assessment* (IPCA), the BELCLASS procedure is generally preferred because of its explicit recognition that some degree of ignorance may surround the classification process.

In the context of mixture analysis, the probabilities of BAYCLASS are interpreted directly as statements of proportional representation. Thus if a pixel has posterior probabilities of belonging to deciduous and conifer of 0.68 and 0.32 respectively, this would be interpreted as evidence that the pixel contains 68% deciduous species and 32% conifers. Note, however, that this requires several important assumptions to be true. First, it requires that the classes for which training site data have been provided are exhaustive (i.e., that there are no other possible interpretations for that pixel). Second, it assumes that the conditional probability distributions $p(e|h)$ do not overlap in the case of pure pixels. In practice, these conditions may be difficult to meet.

In testing at Clark Labs, we have found that while BAYCLASS is effective in determining the constituent members of mixed pixels, it is often not so effective in determining the correct proportions. Rather, we have found that procedures based on the Linear Mixture model (UNMIX) perform considerably better in this respect. However, Linear Spectral Unmixing has its own special limitations. Thus, we favor a hybrid approach using the better qualities of Bayesian decomposition and Linear Spectral Unmixing, as will be discussed below.

BELCLASS and Dempster-Shafer Theory

BELCLASS is probably the most complex of the soft classifier group in its underlying theory. It is based on Dempster-Shafer theory—a variant of Bayesian probability theory that explicitly recognizes the possibility of ignorance. Dempster-Shafer theory is explained more fully in the **Decision Support: Uncertainty Management** chapter. However, a good introduction can be provided by considering the output from BAYCLASS.

Consider a classification where training sites have been developed for the classes [conifer], [deciduous], [grass], [urban], and [water]. If a pixel shows some degree of similarity to [conifer] and to no others, BAYCLASS will assign a value of 1.0 to that class and 0.0 to all others. In fact, it will do so even if the actual support for the [conifer] class is low, because Bayesian probability theory does not recognize the concept of ignorance. It assumes that lack of evidence *for* a hypothesis constitutes evidence *against* that hypothesis. Thus, in this example, the absence of evidence for any combination of [deciduous grass urban water] is therefore interpreted as evidence that it must be [conifer], no matter how weak the direct evidence for [conifer] actually is (so long as it is greater than 0).

In contrast to this, Dempster-Shafer theory does not assume that it has full information, but accepts that the state of one's knowledge may be incomplete. The absence of evidence about a hypothesis is treated as just that—lack of evidence. Unlike Bayesian probability theory, it is not assumed that this, therefore, constitutes evidence against that hypothesis. As

a consequence, there can be a difference between one's belief in an hypothesis and one's attendant disbelief in that same hypothesis!

In the language of Dempster-Shafer theory, the degree to which evidence provides concrete support for an hypothesis is known as *belief*, and the degree to which the evidence does not refute that hypothesis is known as *plausibility*. The difference between these two is then known as a *belief interval*, which acts as a measure of uncertainty about a specific hypothesis.

Returning to the previous example, if the evidence supports [conifer] to the degree 0.3 and all other classes to the degree 0.0, Bayesian probability would assign a posterior probability of 1.0 to [conifer]. However, Dempster-Shafer theory would assign a belief of 0.3 to [conifer] and a plausibility of 1.0, yielding a belief interval of 0.7. Furthermore, it would assign a belief of 0.0 to all other classes and a plausibility of 0.7.

If a second piece of evidence were then to be considered, and it was found that this evidence supported [urban] to a degree of 0.6 and gave no support to any other hypothesis, this would affect the hypothesis [conifer] by lowering its plausibility to 0.4. At this point, then, belief in [conifer] is 0.3 and plausibility is 0.4. Thus our uncertainty about this class is very low (0.1).

The combination of evidence is generally somewhat more complex than these contrived examples would suggest and uses a logic known as *Dempster's Rule*. The Belief module in IDRISI implements this rule. However, BELCLASS is less concerned with the combination of evidence than it is with decomposing the evidence to determine the degree of support (expressed as belief or plausibility) for each of the classes for which training data have been supplied.⁷⁵

In addition to the concepts of belief and plausibility, the logic of Dempster-Shafer theory can also express the degree to which the state of one's knowledge does not distinguish between the hypotheses. This is known as *ignorance*.

Ignorance expresses the incompleteness of one's knowledge as a measure of the degree to which we cannot distinguish between any of the hypotheses. Using the example above, ignorance thus expresses one's commitment to the indistinguishable set of all classes [conifer deciduous grass urban water]—the inability to tell to which class the pixel belongs.

In normal use, Dempster-Shafer theory requires that the hypotheses (classes) under consideration be mutually exclusive and exhaustive. However, in developing BELCLASS, we felt that there was a strong case to be made for non-exhaustive categories—that the pixel may indeed belong to some unknown class, for which a training site has not been provided. In order to do this we need to add an additional category to every analysis called [other], and assign any incompleteness in one's knowledge to the indistinguishable set of all possible classes (including this added one)—e.g., [conifer deciduous grass urban water other]. This yields a result which is consistent with Dempster-Shafer theory, but which recognizes the possibility that there may be classes present about which we have no knowledge.

Defining ignorance as a commitment to the indistinguishable set of all classes suggests that it may have some relationship to the classification uncertainty image produced by BAYCLASS. The classification uncertainty image of BAYCLASS expresses the uncertainty the classifier has in assigning class membership to a pixel. Uncertainty is highest whenever there is no class that clearly stands out above the others in the assessment of class membership for a pixel. We have found that in the context of BELCLASS, this measure of uncertainty is almost identical to that of Dempster-Shafer ignorance. As a result, we have modified the output of the classification uncertainty image of BELCLASS slightly so that it outputs true Dempster-Shafer ignorance.⁷⁶

The operation of BELCLASS is essentially identical to that of BAYCLASS and thus also MAXLIKE. Two choices of output are given: beliefs or plausibilities. In either case, a separate image of belief or plausibility is produced for each class. In addition, a classification uncertainty image is produced which can be interpreted in the same manner as the classifica-

75. The logic of the decomposition process is detailed in the module description for BELCLASS in the on-line Help System.

76. The file name for the classification uncertainty image in BELCLASS is composed by concatenating the prefix supplied by the user and the letter string "clu".

tion uncertainty image produced by all of the soft classifiers (but which is truly a measure of Dempster-Shafer ignorance).

The prime motivation for the use of BELCLASS is to check for the quality of one's training site data and the possible presence of unknown classes during *In-Process Classification Assessment*. In cases where one believes that one or more unknown classes exist (and thus that some portion of total ignorance arises because of this presence of an unknown class), the BELCLASS routine should be used. BELCLASS does this by implicitly adding an [other] class to the set of classes being considered. This is a theoretical concession to the mechanics of the BELCLASS process and will not be directly encountered by the user.

Comparing the output of BELCLASS with BAYCLASS, you will notice a major difference. Looking at the images produced by BAYCLASS, it will appear as if your training sites are strong. BAYCLASS is a very confident classifier (perhaps overly confident) since it assumes no ignorance. BELCLASS, however, appears to be a very reserved classifier. Here we see a result in which all of the uncertainties in our information become apparent. It does not presume to have full information, but explicitly recognizes the possibility that one or more unknown classes may exist.

MIXCALC and MAXSET

In BELCLASS, concern is directed to the degree of membership that each pixel exhibits for each of the classes for which training data have been provided. However, the logic of Dempster-Shafer theory recognizes a whole hierarchy of classes, made up of the indistinguishable combinations of these basic classes. For example, given basic classes (called *singletons*) of [conifer] [deciduous] [grass], Dempster-Shafer theory recognizes the existence of all of the following classes:⁷⁷

- [conifer]
- [deciduous]
- [grass]
- [conifer deciduous]
- [conifer grass]
- [deciduous grass]
- [conifer deciduous grass]

Dempster-Shafer theory also allows one to make two different kinds of assessments about each of these classes. The first is clearly belief. The second is known as a *Basic Probability Assignment* (BPA). Both require further explanation.

When evidence provides some degree of commitment to one of these non-singleton classes and not to any of its constituents separately, that expression of commitment is known as a Basic Probability Assignment (BPA). The BPA of a non-singleton class thus represents the degree of support for the presence of one or more of its constituents, but without the ability to tell which.

Given this understanding of a BPA, belief in a non-singleton class is then calculated as the sum of BPAs for that class and all sub-classes. For example to calculate the belief in the class [conifer deciduous], you would add the BPAs for [conifer deciduous], [conifer] and [deciduous]. Belief is thus a broader concept than a BPA. It represents the total commitment to all members of a set combined.

In the context of remote sensing, these non-singleton classes are of interest in that they represent mixtures, and thus might be used for a more detailed examination of sub-pixel classification. However, the sheer number of such classes makes it impractical to have a software module that outputs all possible classes. For a set of n singleton classes, the total number of classes in the entire hierarchy is $(2^n - 1)$. Thus in a case where 16 land cover classes are under consideration, 65,535 classes are included in the full hierarchy—over two terabytes of output for a full Landsat scene!

As it turns out, however, only a small number of these non-singleton classes contain any significant information in a typical application. These can very effectively be determined by running the MAXSET module. MAXSET is a hard classifier

77. Clearly these are sets of classes. However, since evidence may support one of these sets without further distinction about which members of the set are supported, the set itself can be thought of as a class.

that assigns to each pixel the class with the greatest degree of commitment from the full Dempster-Shafer class hierarchy. The significance of these mixtures can further be determined by running the AREA module on the MAXSET result. Then MIXCALC can be used to calculate the mixture BPA that underlies the MAXSET result.

Belief

Both BELCLASS and MIXCALC deconstruct the evidence to infer belief and plausibility for each class. One of the motivations for doing so is that it allows the user to combine ancillary information with that determined from the reflectance data. New evidence can be combined with existing knowledge with the Belief module. Belief is described more fully in the chapter on **Decision Support: Uncertainty Management**, and is used in exactly the same manner for the data described here.

FUZCLASS and Fuzzy Set Theory

The third soft classifier in IDRISI is FUZCLASS. As the name suggests, this classifier is based on the underlying logic of Fuzzy Sets. Just as BAYCLASS and BELCLASS are based on the fundamental logic of MAXLIKE, FUZCLASS is based on the underlying logic of MINDIST—i.e., fuzzy set membership is determined from the distance of pixels from signature means as determined by MAKESIG.

There are two important parameters that need to be set when using FUZCLASS. The first is the z-score distance where fuzzy membership becomes zero. The logic of this is as follows.

It is assumed that any pixel at the same location in band space as the class mean (as determined by running MAKESIG) has a membership grade of 1.0. Then as we move away from this position, the fuzzy set membership grade progressively decreases until it eventually reaches zero at the distance specified. This distance is specified as a standard score (z-score) to facilitate its interpretation. Thus, specifying a distance of 1.96 would force 5% of the data cells to have a fuzzy membership of 0, while 2.58 would force 1% to have a value of 0.

The second required parameter setting is whether or not the membership values should be normalized. Normalization makes the assumption (like BAYCLASS) that the classes are exhaustive, and thus that the membership values for all classes for a single pixel must sum to 1.0. This is strictly required to generate true fuzzy set membership grades. However, as a counterpart to BELCLASS, the option is provided for the calculation of un-normalized values. As was suggested earlier, this is particularly important in the context of In-Process Classification Assessment for evaluation of a MINDIST-based classification.

UNMIX and the Linear Mixture Model

The Linear Mixture Model assumes that the mixture of materials within a pixel will lead to an aggregate signature that is an area-weighted average of the signatures of the constituent classes. Thus if two parent materials (called end members in the language of Linear Spectral Unmixing) had signatures of 24, 132, 86 and 56, 144, 98 on three bands, a 50/50 mixture of the two should yield a signature of 40, 138, 92. Using this simple model, it is possible to estimate the proportions of end member constituents within each pixel by solving a set of simultaneous equations. For example, if we were to encounter a pixel with the signature 32, 135, 89, and assumed that the pixel contained a mixture of the two end members mentioned we could set up the following set of equations to solve:

$$f1(24) + f2(56) = 32$$

$$f1(132) + f2(144) = 135$$

$$f1(86) + f2(98) = 89$$

where f1 and f2 represent the fractions (proportions) of the two end members. Such a system of simultaneous equations can be solved using matrix algebra to yield a best fit estimate of f1 and f2 (0.75 and 0.25 in this example). In addition, the

sum of squared residuals between the fitted signature values and the actual values can be used as a measure of the uncertainty in the fit.

The primary limitation of this approach is that the number of end members cannot exceed the number of bands. This can be a severe limitation in the case of SPOT imagery, but of little consequence with hyperspectral imagery. IDRISI thus offers three approaches (in UNMIX) to Linear Spectral Unmixing:

1. The standard linear spectral unmixing approach (as indicated above) for cases where sufficient bands are available.
2. A probability guided option for cases where insufficient bands exist. Although the total number of possible end members may be large, the number that coexist within a single pixel is typically small (e.g., 2-3). This approach thus uses a first stage based on the BAYCLASS module to determine the most likely constituents (up to the number of bands), with a second stage linear spectral unmixing to determine their fractions. Experiments at Clark Labs have shown this to produce excellent results.
3. An exhaustive search option for cases where insufficient bands exist. In this instance, one specifies the number of constituents to consider (up to the total number of bands). It then tests all possible combinations of that many end members and reports the fractions of that combination with the lowest sum of squared residuals. This approach is considerably slower than the other options. In addition, experiments at Clark Labs have shown that this approach yields inferior results to the probability guided procedure in cases where the end member signatures are drawn from training sites. Pure end member signatures, created with ENDSIG, should be used.

End member signatures are specified using standard signature (.sig) files, created using either MAKESIG or ENDSIG. The former is used in the case where end members are derived from training sites, while the latter is used in cases where pure end member values are known (such as from a spectral library). Note that only the mean value on each band is used from these signature files—variance/covariance data are ignored.

Accommodating Ambiguous (Fuzzy) Signatures in Supervised Classification

All of the discussions to this point have assumed that the signature data were gathered from pure examples of each class. However, it sometimes happens that this is impossible. For example, it might be difficult to find a pure and uniform stand of white pine forest, because differences in tree spacing permit differing levels of the understory material (perhaps a mixture of soil, dead needles and ferns) to show through the canopy. From the perspective of a single pixel, therefore, there are different grades of membership in the white pine class, ranging from 0.0, where no white pine trees are present within the pixel to 1.0 where a dense closed canopy of white pine exists (such as a plantation). Thus, even though the white pine set (class) is itself inherently crisp, our gathering of data by pixels that span several to many meters across forces our detection of that set to be necessarily fuzzy in character.

Wang (1990) has cited examples of the above problem to postulate that the logic of class membership decision problems in remote sensing is that of Fuzzy Sets. However, as indicated earlier in this chapter, while this problem truly belongs in the realm of Fuzzy Measures, it is not strictly one of Fuzzy Sets in most cases. For example, the white pine class mentioned earlier is not a fuzzy set—it is unambiguous. It is only our difficulty in detecting it free of other cover types that leads to ambiguity. The problem is thus one of imprecision that can best be handled by a Bayesian or Dempster-Shafer procedure (which is, in fact, ultimately the procedure used by Wang, 1990).

As pointed out earlier, ambiguity can arise from a variety of sources, and not just fuzzy sets. However, the special case of resolution and mixed pixels is one that is commonly encountered in the classification process. As a result, it is not unusual that even the best signatures have some degree of inter-class mixing. In such cases, it is desirable to use a procedure for signature development that can recognize this ambiguity. Wang (1990) has proposed an interesting procedure for signature development in this context that we have implemented in a module named FUZSIG. As the name suggests, the module is a variant of MAKESIG for the special case of ambiguous (i.e., fuzzy) training site data. However, we use the association with fuzzy in the more general sense of fuzzy measures. The logic of the procedure is built around the concept of mixtures and should be restricted to instances where it is that which is the source of the fuzziness accommodated.

The use of FUZSIG requires that a specific sequence of operations be followed.

1. Define Training Sites

This stage proceeds much the same as usual: training sites are digitized using the on-screen digitizing facility. However, there is no requirement that training sites be as homogeneous as possible—only that the relative proportions of cover types within each training site pixel can be estimated.

2. Rasterize the Training Sites

Although this stage is not strictly necessary, it can make the next stage much easier if the training sites are collected into a single raster image. This is done by running INITIAL to create a blank byte binary image (i.e., one initialized with zeros), and then rasterizing the digitized polygons with RASTERVECTOR.

3. Create Fuzzy Partition Matrix in Database Workshop

The next step is to create a *fuzzy partition matrix* in Database Workshop. A fuzzy partition matrix indicates the membership grades of each training site in each class. To do this, first set up a database with an integer identifier field and one field (column) per information class in 4-byte real number format. Then put numeric identifiers in the ID field corresponding to each of the training sites (or training site groups if more than one polygon is used per class). For N classes and M training sites, then, an $N \times M$ matrix is formed.

The next step is to fill out the fuzzy partition matrix with values to indicate the membership grades of each training site (or training site group) in the candidate classes. This is best filled out by working across the columns of each row in turn. Since each row represents a training site (or training site group), estimate the proportions of each class that occur within the training site and enter that value (as a real number from 0.0 to 1.0). In this context, the numbers should add to 1.0 along each row, but typically will not along each column.

Once the fuzzy partition matrix is complete, use the Export as Values File option from the Database Workshop File menu to create a series of values files, one for each class. Next create a series of raster images expressing the membership grades for each class. To do so, in IDRISI use ASSIGN to assign each values file to the training site raster image created in the previous step (this is the feature definition image for ASSIGN). Name the output image for each class using a name that is a concatenation of the letters "fz" plus the name of the class. For example, if the class is named "conifer", the output produced with the ASSIGN operation would be named "fzconifer." This "fz" prefix is a requirement for the next stage.

4. Extract Fuzzy Signatures

The next stage is to create the fuzzy signatures. This is done with the FUZSIG module. The operation of FUZSIG is identical to MAKESIG. You will need to specify the name of the file that defines your signatures (if you followed the steps above, this will be the image file you created in Step 2), the number of bands to use in the signature development, the names of the bands and the names of the signatures to be created. It is very important, however, that the signature names you specify coordinate with the names of the fuzzy membership grade images you created in Step 3. Continuing with the previous example, if you specify a signature name of "conifer", it will expect to find an image named "fzconifer" in your working directory. Similarly, a signature named "urban" would be associated with a fuzzy membership grade image named "fzurban."

The output from FUZSIG is a set of signature files (.sig) of identical format to those output from MAKESIG. FUZSIG gives each pixel a weight proportional to its membership grade in the determination of the mean, variance and covariance of each band for each class (see Wang, 1990). Thus a pixel that is predominantly composed of conifers will have a large weight in the determination of the conifer signature, but only a low weight in determining the signature for other constituents.

Because these signature files are of identical format to those produced by MAKESIG, they can be used with any of the

classifiers supported by IDRISI, both hard and soft. However, there are some important points to note about these files:

- The minimum and maximum reflectances on each band cannot meaningfully be evaluated using the weighting procedure of FUZSIG. As a consequence, the minimum and maximum value recorded are derived only from those pixels where the membership grade for the signature of concern is greater than 0.5. This will typically produce a result that is identical to the output of MAKESIG. However, it is recommended that if the PIPED classifier is to be used with these signatures, the parallelepipeds should be defined by standard deviation units rather than the minimum and maximum data values.
- Unlike MAKESIG, FUZSIG does not output a corresponding set of signature pixel files (.spf) to the signature files produced. As a consequence, EDITSIG cannot be used to examine or modify the signature files produced. However, since the signature files are simple ASCII text files, they can be examined in Edit. Their simple structure is explained in the on-line Help System.

Hardeners

Once a soft classifier has been applied to a multispectral image set, the soft results can be re-evaluated to produce a hard classification by using one of the following hardeners from the module HARDEN;

BAYCLASS

Using the results from BAYCLASS, this option determines the class possessing the maximum posterior probability for each cell, given a set of probability images. Up to four levels of abstraction can be produced. The first is the most likely class, just described. The second outputs the class of the second highest posterior probability, and so on, up to the fourth highest probability.

BELCLASS

Using the results from BELCLASS, this option is essentially identical to the BAYCLASS hardener, except that it is designed for use with Dempster-Shafer beliefs.

FUZCLASS

Using the results from FUZCLASS, this option is essentially identical to the BAYCLASS hardener, except that it is designed for use with Fuzzy Sets.

UNMIX

Using the results from UNMIX, this option is essentially identical to the BAYCLASS hardener, except that it is designed for use with the mixture fractions produced by UNMIX.

MAHALCLASS

Using the results from MAHALCLASS, this option is essentially identical to the BAYCLASS hardener, except that it is designed for use with the typicalities produced by MAHALCLASS.

Unsupervised Classification

General Logic

Unsupervised classification techniques share a common intent to uncover the major land cover classes that exist in the image without prior knowledge of what they might be. Generically, such procedures fall into the realm of *cluster analysis*, since they search for clusters of pixels with similar reflectance characteristics in a multi-band image. They are also all gen-

eralizations of land cover occurrence since they are concerned with uncovering the *major* land cover classes, and thus tend to ignore those that have very low frequencies of occurrence. However, given these broad commonalities, there is little else that they share in common. There are almost as many approaches to clustering as there are image processing systems on the market. IDRISI is no exception. The primary unsupervised procedure it offers is unique (CLUSTER). However, IDRISI also offers a variant on one of the most common procedures to be found (ISOCLUST). As implemented here, this procedure is really an iterative combination of unsupervised and supervised procedures, as is also the case with the third procedure offered, MAXSET.

CLUSTER

The CLUSTER module in IDRISI implements a special variant of a Histogram Peak cluster analysis technique (Richards, 1993). The procedure can best be understood from the perspective of a single band. If one had a single band of data, a histogram of the reflectance values on that band would show a number of peaks and valleys. The peaks represent clusters of more frequent values associated with commonly occurring cover types.

The CLUSTER procedure thus searches for peaks by looking for cases where the frequency is higher than that of its immediate neighbors on either side. In the case of two bands, these peaks would be hills, while for three bands they would be spheres, and so on. The concept can thus be extended to any number of bands. Once the peaks have been located, each pixel in the image can then be assigned to its closest peak, with each such class being labeled as a cluster. It is the analyst's task to then identify the land cover class of each cluster by looking at the cluster image and comparing it to ground features.

CLUSTER offers two levels of generalization. With the broad level of generalization, clusters must occur as distinct peaks in the multi-dimensional histogram as outlined above. However, with the fine level of generalization, CLUSTER also recognizes shoulders in the curve as cluster peaks. Shoulders occur when two adjacent clusters overlap to a significant extent. Peaks and shoulders are identified in the histogram shown in Figure 16-7.

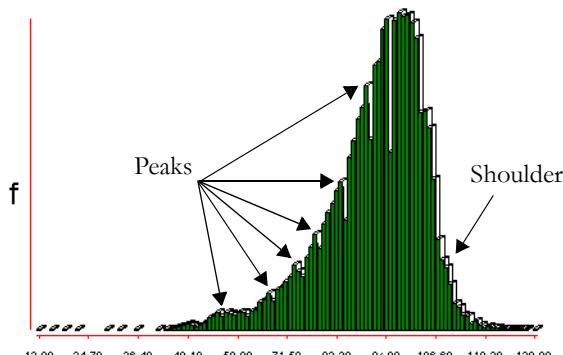


Figure 16-7

The CLUSTER procedure in IDRISI has been modified and tailored to work with the special case of three bands as described by an 8-bit color composite image created with the COMPOSIT module. The reason for doing so is based largely on the fact that the procedure involved in creating an 8-bit color composite image is essentially the same as the first stage of multi-dimensional histogram generation in the clustering algorithm. Since it is not uncommon to experiment with various clusterings of a single multi-band image, speed is greatly enhanced by not repeating this histogram generation step. While it may seem that the restriction of working with a three-band composite is limiting, bear in mind that the underlying "bandness" of a multi-spectral image in the visible-through-middle infrared is rarely more than 2 or 3 (to confirm this, try running a Principal Components Analysis on a higher spectral resolution multi-band image set). In most environments, creating composite images using the red and near-infrared bands, along with a middle-infrared band (such as Landsat Band 5) will essentially capture all of the information in the image.

Experience in using the CLUSTER routine has shown that it is fast and is capable of producing excellent results. However, we have learned that the following sequence of operations is particularly useful.

1. Run CLUSTER using the most informative bands available (generally, these include the red visible band, the near-infrared band, and a middle infrared band—e.g., Landsat TM bands 3, 4 and 5 respectively). Use the linear stretch with saturation option with 1% saturation the "fine generalization level" and "retain all clusters" options.
2. Display a histogram of this image. This histogram shows the frequency of pixels associated with each of the clusters that can be located in the image. Many of these clusters have very small frequencies, and thus are somewhat insignificant. Figure 16-8 presents an example of just such a histogram.

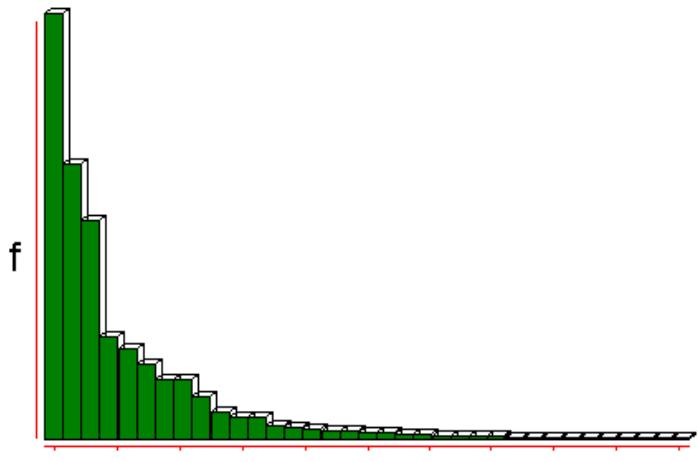


Figure 16-8

As can be seen, there are three clusters that dominate the image. Then there is a sharp break with a second group of strong clusters through to Cluster 12. Then there is a third group that follows until Cluster 25, followed by a small group of very insignificant clusters. Experience suggests then that a good generalization of the data would be to extract the first 12 clusters, with a more detailed analysis focusing on the first 25.

3. Once the number of clusters to be examined has been determined, run CLUSTER again, but this time choose the "fine generalization" and set the "maximum number of clusters" to the number you determined (e.g., 12).
4. Display the resulting image with a qualitative color palette and then try to identify each of the clusters in turn. You can use the interactive legend editing option to change the legend caption to record your interpretation. You may also wish to change the color of that category to match a logical color scheme. Also, remember that you may highlight all pixels belonging to a category by holding down the mouse button over a legend category color box.
5. At the end of the identification process, you may need to combine several categories. For example, the cluster analysis may have uncovered several pavement categories, such as asphalt and concrete, that you may wish to merge into a single category. The simplest way of doing this is to use Edit to create a values file containing the integer reassessments. This file has two columns. The left column should record the original cluster number, while the right column should contain the new category number to which it should be reassigned. After this file has been created, run ASSIGN to assign these new category indices to the original cluster data.

The CLUSTER procedure in IDRISI is fast and remarkably effective in uncovering the basic land cover structure of the image. It can also be used as a preliminary stage to a hybrid unsupervised/supervised process whereby the clusters are used as the training sites to a second classification stage using the MAXLIKE classifier.⁷⁸ This has the advantage of allowing the use of a larger number of raw data bands, as well as providing a stronger classification stage of pixels to their most

similar cluster. In fact, it is this basic logic that underlies the ISOCLUST procedure described below.

ISOCLUST

The ISOCLUST module is an iterative self-organizing unsupervised classifier based on a concept similar to the well-known ISODATA routine of Ball and Hall (1965) and cluster routines such as the H-means and K-means procedures. The typical logic is as follows:

1. The user decides on the number of clusters to be uncovered. One is clearly blind in determining this. As a consequence, a common approach is to ask for a large number and then aggregate clusters after interpretation. A more efficient approach to this problem will be offered below, based on the specific implementation in IDRISI.
2. A set of N clusters is then arbitrarily located in band space. In some systems, these locations are randomly assigned. In most, they are systematically placed within the region of high frequency reflectances.
3. Pixels are then assigned to their nearest cluster location.
4. After all pixels have been assigned, a new mean location is computed.
5. Steps 3 and 4 are iteratively repeated until no significant change in output is produced.

The implementation of this general logic in IDRISI is different in several respects.

- After entering the raw image bands to be used, you will be presented with a histogram of clusters that expresses the frequency with which they occur in the image. You should examine this graph and look for significant breaks in the curve. These represent major changes in the generality of the clusters. Specify the number of clusters to be created based on one of these major breaks.
- The cluster seeding process is actually done with the CLUSTER module in IDRISI. CLUSTER is truly a clustering algorithm (as opposed to a segmentation operation as is true of many so-called clustering routines). This leads to a far more efficient and accurate placement of clusters than either random or systematic placement.
- The iterative process makes use of a full Maximum Likelihood procedure. In fact, you will notice it make iterative calls to MAKESIG and MAXLIKE. This provides a very strong cluster assignment procedure.
- Because of the efficiency of the seeding step, very few iterations are required to produce a stable result. The default of 3 iterations works well in most instances.

The ISOCLUST procedure is relatively new to IDRISI. Therefore we invite your comments and suggestions based on experiences you have with this module.

MAXSET

As previously described, MAXSET is a hard classifier that assigns to each pixel the class with the greatest degree of commitment from the full Dempster-Shafer class hierarchy that describes all classes and their hierarchical combination. Although it is run as if it were a supervised classifier (it requires training site data), ultimately it behaves as if it were an unsupervised classifier in that it can assign a pixel to a class for which no exclusive training data have been supplied.

MAXSET is very similar in concept to the PIPED, MINDIST, and MAXLIKE classifiers in that it makes a hard determination of the most likely class to which each pixel belongs according to its own internal logic of operation. MAXSET is different, however, in that it recognizes that the best assignment of a pixel might be to a class that is mixed rather than unique. For example, it might determine that a pixel more likely belongs to a class of mixed conifers and deciduous forest than it does to either conifers or deciduous exclusively. The logic that it uses in doing so is derived from Dempster-Shafer

78. This is possible because MAKESIG can create signatures based on training sites defined by either a vector file or an image. In this case, the image option is used.

theory, a special variant of Bayesian probability theory that is described more fully in the section on the BELCLASS soft classifier above. Dempster-Shafer theory provides a logic for expressing one's belief in the degree to which an item belongs to a particular set. MAXSET evaluates the degree of support for the membership of every pixel in the hierarchy of sets which includes each of the basic classes plus all possible combinations of classes. Thus, for example, in a case with basic land cover classes A, B and C, MAXSET would evaluate the degree of membership in each of the following classes:

[A]
[B]
[C]
[A,B]
[A,C]
[B,C]
[A,B,C]

The importance of the supersets (sets with combinations of classes) is that they represent indistinguishable combinations of the basic classes. Thus when evidence supports the combination [A,B], it can be interpreted as support for A, or B, or A and B, but that it is unable to determine which. The reasons for this are basically twofold. Either the evidence is inconclusive, or the indistinguishable superset really exists. Thus if MAXSET concludes that a pixel belongs to the indistinguishable superset [conifer, deciduous], it may be because the pixel truly belongs to a mixed forest class, or it may simply mean that the training sites chosen for these two classes have not yielded unambiguous signatures.

MAXSET is an excellent starting point for In-Process Classification Assessment (as described above).

Hyperspectral Remote Sensing

As indicated in the introduction to this chapter, IDRISI includes a set of routines for working with hyperspectral data. The basic procedures are similar to those used with supervised classification of multispectral data. Signatures are developed for each land cover of interest, then the entire image is classified using information from those signatures. Details of the process are given below.

Importing Hyperspectral Data

IDRISI works with hyperspectral data as a series—i.e., as a collection of independent images that are associated through either a raster image group file (.rgf) or a sensor band file (.sbf). Many hyperspectral images available today are distributed in Band-Interleaved-by-Line (BIL) format. Thus you may need to use GENERICRASTER in the import/export module group to convert these data to IDRISI format. In addition, if the data has come from a UNIX system in 16-bit format, make sure you indicate this within the GENERICRASTER dialog box.

Hyperspectral Signature Development

The signature development stage uses the module HYPERSIG, which creates and displays hyperspectral signature files. Two sources of information may be used with HYPERSIG for developing hyperspectral signatures: training sites (as described in the supervised classification section above) or library spectral curve files.

Image-based Signature Development

IDRISI offers both a supervised and unsupervised procedure for image-based signature development. The former approach is to use a procedure similar to that outlined in earlier sections for supervised classification where training sites are delineated in the image and signatures are developed from their statistical characteristics. Because of the large number of bands involved, both the signature development and classification stages make use of different procedures from those used with multispectral data. In addition, there is a small variation to the delineation of training sites that needs to be

introduced. The steps are as follows:

1. Create a color composite using three of the hyperspectral bands and digitize the training sites.
2. Rasterize the vector file of training sites by using INITIAL to create a blank image and then using RASTERVECTOR to rasterize the data.
3. Run HYPERSIG to create a hyperspectral signature file⁷⁹ for each land cover class. Hyperspectral signature files have an .hsg extension, and are similar in intent (but different in structure) to a multispectral signature file (.sig).
4. Run any of the HYPERSAM, HYPERMIN, HYPERUSP, HYPEROSP, HYPERUNMIX, or HYPERABSORB modules to classify the image.

The unsupervised procedure is somewhat experimental. HYPERAUTOSIG discovers signatures based on the concept of signature power. Users wishing to experiment with this should consult the on-line Help System for specific details.

Library-based Signature Development

The second approach to classifying hyperspectral data relies upon the use of a library of spectral curves associated with specific earth surface materials. These spectral curves⁸⁰ are measured with very high precision in a lab setting. These curves typically contain over a thousand readings spaced as finely as 0.01 micrometers over the visible, near- and middle-infrared ranges. Clearly there is a great deal of data here. However, there are a number of important issues to consider in using these library curves.

Since the curves are developed in a lab setting, the measurements are taken without an intervening atmosphere. As a consequence, measurements exist for areas in the spectrum where remote sensing has difficulty in obtaining useable imagery. You may therefore find it necessary to remove those bands in which atmospheric attenuation is strong. A simple way to gauge which bands have significant atmospheric attenuation is to run PROFILE and examine the standard deviation of selected features over the set of hyperspectral images. Atmospheric absorption tends to cause a dramatic increase in the variability of feature signatures. To eliminate these bands, edit the sensor band file, described below, to delete their entries and adjust the number of bands at the top of the file accordingly.

Even in bands without severe attenuation, atmospheric effects can cause substantial discrepancies between the spectral curves measured in the lab and those determined from hyperspectral imagery. As a consequence, IDRISI assumes that if you are working with spectral libraries, the imagery has already been atmospherically corrected. The SCREEN module can be used to screen out bands in which atmospheric scattering has caused significant image degradation. The ATMOSC module can then be used on the remaining bands to correct for atmospheric absorption and haze.

Library spectral curves are available from a number of research sites on the web, such as the United States Geological Survey (USGS) Spectroscopy Lab (<http://speclab.cr.usgs.gov>). These curve files will need to be edited to meet IDRISI specifications. The format used in IDRISI is virtually identical to that used by the USGS. It is an ASCII text file with an .isc extension. File structure details may be found in the file formats section of the on-line Help System.

Spectral curve files are stored in a subdirectory of the IDRISI program directory named "waves" (e.g., c:\idrisi32\waves) and a sample of library files has been included.

Spectral curve files do not apply to any specific sensor system. Rather, they are intended as a general reference from which the expected reflectance for any sensor system can be derived in order to create a hyperspectral signature file. This is done using the HYPERSIG module.

In order for HYPERSIG to create a hyperspectral signature from a library spectral curve file, it will need to access a file

79. The contents of these files are also known as *image spectra*.

80. The contents of these files are also known as *library spectra*.

that describes the sensor system being used. This is a *sensor band file* with an .sbf extension that is also located in the "waves" subdirectory under the IDRISI program directory. The file must be in ASCII format and file structure details may be found in the file formats section of the on-line Help System.

For comparison, a file for the AVIRIS system, June 1992, has been supplied as a sample (aviris92.sbf) in the "waves" subdirectory of the IDRISI program directory (e.g., c:\idrisi32\waves\aviris92.sbf).

Once the hyperspectral signatures have been created for each of the land cover classes, classification of the imagery can proceed with either the HYPERSAM or HYPERMIN modules.

PROFILE

The PROFILE module offers an additional tool for exploring hyperspectral data. A profile generated over a hyperspectral series will graphically (or numerically) show how the reflectance at a location changes from one band to the next across the whole series. Thus the result is a spectral response pattern for the particular locations identified.

PROFILE requires a raster image of the sample spots to be profiled. Up to 15 profiles can be generated simultaneously,⁸¹ corresponding to sample sites with index values 1-15. A sample site can consist of one or many pixels located in either a contiguous grouping or in several disconnected groups.

The second piece of information that PROFILE will require is the name of a file that contains the names of the IDRISI image files that comprise the hyperspectral series. You have two choices here. You may use either an image group file (.rgf) or a sensor band file (.sbf), since both contain this information. In either case, you will need to have created one of these before running PROFILE. You may wish to choose the sensor band file option, since it can be used with other operations as well (such as HYPERSIG).

Hyperspectral Image Classification

IDRISI offers a range of procedures for the classification of hyperspectral imagery. All but one (HYPERABSORB) work best with signatures developed from training sites, and can be divided into hard and soft classifier types.

Hard Hyperspectral Classifiers

HYPERSAM

HYPERSAM is an implementation of the Spectral Angle Mapper algorithm for hyperspectral image classification (Kruse et al., 1993). The Spectral Angle Mapper algorithm is a minimum-angle procedure that is specifically designed for use with spectral curve library data (although it can also be used with image-based signatures).

The reflectance information recorded in a spectral curve file (.isc) is measured in a laboratory under constant viewing conditions. However, the data in a hyperspectral image contains additional variations that exist because of variations in illumination. For example, solar elevation varies with the time of year and topographic variations lead to variations in aspect relative to the sun. As a consequence, there can be significant differences between the spectral response patterns as recorded by the sensor system and those measured in the lab. The Spectral Angle Mapper algorithm is based on the assumption that variations in illumination conditions will lead to a set of signatures that fall along a line connected to the

81. The display can become quite difficult to read whenever more than just a few profiles are generated at the same time. Under normal use, you may wish to limit the profiling to no more than 5 sites.

origin of the band space as illustrated in Figure 16-9.

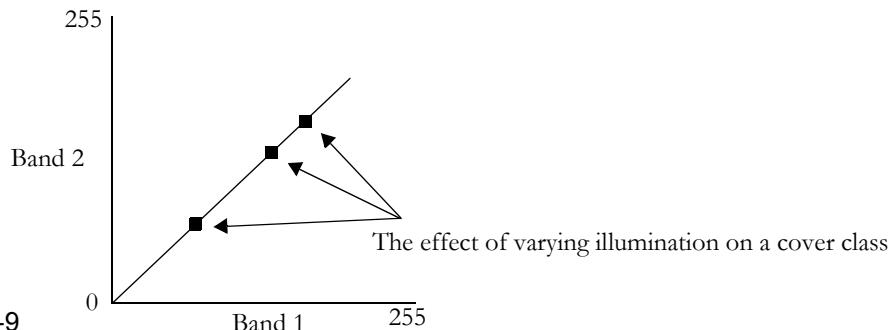


Figure 16-9

Thus, in the presence of significant illumination variations, it would be anticipated that a traditional distance-based classifier would have some difficulty in identifying the feature in all cases. The Spectral Angle Mapper thus uses a minimum-angle approach. In essence, it treats each signature as a vector. Then by comparing the angle formed by an unknown pixel, the origin, and a class mean, and comparing that to all other classes, the class that will be assigned to the unknown pixel is that with the minimum angle, as illustrated in Figure 16-10.

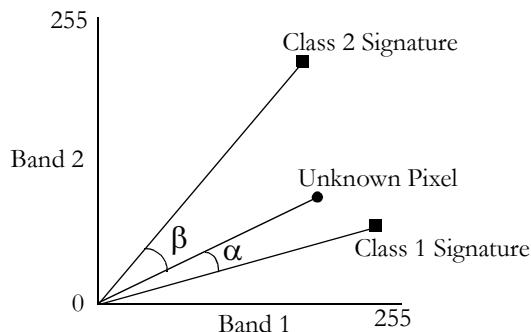


Figure 16-10

In the above figure, the unknown pixel would be assigned to Class 1 since the angle it subtends with the unknown pixel (α) is smaller than that with Class 2 (β).

HYPERMIN

HYPERMIN is a minimum-distance classifier for hyperspectral data that is specifically intended for use with image-based signatures developed from training sites. It uses a logic that is identical to that of the multispectral hard classifier MIND-IST using standardized distances.

Soft Hyperspectral Classifiers

HYPERUNMIX

HYPERUNMIX uses the same approach as the Linear Spectral Unmixing option of UNMIX, except that it uses hyperspectral signature files.

HYPEROSP

HYPEROSP uses a procedure known as Orthogonal Subspace Projection. It is closely related to HYPERUNMIX in that it is based on the logic of Linear Spectral Unmixing. However, it attempts to improve the signal-to-noise ratio for a specific cover type by explicitly removing the contaminating effects of mixture elements. The result is an image that expresses the degree of support for the presence of the signature in question. Note that this measure is not a fraction per

se, but simply a measure of support.

HYPERUSP

HYPERUSP is an unsupervised soft classifier based on the logic of HYPERAUTOSIG for the development of signatures (clusters), and HYPERUNMIX for the soft classification.

Hyperspectral Classifiers for use with Library Spectra

IDRISI offers a single classifier specifically designed for use with library spectra (HYPERABSORB). It is similar in basic operation to the TRICORDER (now renamed TETRACORDER) algorithm developed by the USGS.

HYPERABSORB

HYPERABSORB specifically looks for the presence of absorption features associated with specific materials. It follows a logic that has proven to be particularly useful in mineral mapping in arid and extraterrestrial environments. The basic principle is as follows. Particular materials cause distinctive absorption patterns in library spectra. Familiar examples include the massive absorption in the red and blue wavelengths due to the presence of chlorophyll, and the distinctive water absorption features in the middle infrared. However, many minerals exhibit very specific and narrow absorption features related to the movement of electrons in crystal lattices and vibrational effects of molecules. HYPERABSORB measures the degree of absorption evident in pixels as compared to a library spectrum through a process of continuum removal and depth analysis. The on-line Help System gives specific details of the process. However, the basic concept is to co-register the pixel spectrum and the library spectrum by calculating the convex hull over each. This is a polyline drawn over the top of the curve such that no points in the original spectrum lie above the polyline, and in which no concave sections (valleys) occur within the polyline. Using segments of this polyline as a datum, absorption depth is measured for intermediate points. The correlation between absorption depths in the pixel spectrum and the library spectrum then gives a measure of fit, while the volume of the absorption area relative to that in the library spectrum gives a measure of abundance.

The analysis of hyperspectral images is still very much in a developmental stage. We welcome comments on experiences in using any of these procedures and suggestions for improvement.

References and Further Reading

Ball, G.H., and Hall, D.J., 1965. *A Novel Method of Data Analysis and Pattern Classification*, Stanford Research Institute, Menlo Park, California.

Clark, R.N., A.J. Gallagher, and G.A. Swayze, 1990, Material absorption band depth mapping of imaging spectrometer data using a complete band shape least-squares fit with library reference spectra, Proceedings of the Second Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) Workshop, JPL Publication 90-54, 176-186.

Clark, R.N., G.A. Swayze, A. Gallagher, N. Gorelick, and F. Kruse, 1991, Mapping with imaging spectrometer data using the complete band shape least-squares algorithm simultaneously fit to multiple spectral features from multiple materials, Proceedings of the Third Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) Workshop, JPL Publication 91-28, 2-3.

Eastman, J.R., Kyem, P.A.K., Toledano, J., and Jin, W., 1993. *GIS and Decision Making*, UNITAR, Geneva.

Kruse, F.A., Lefkoff, A.B., Boardman, J.W., Heidebrecht, K.B., Shapiro, A.T., Barloon, P.J., and Goetz, A.F.H., 1993. The Spectral Image Processing System (SIPS)—Interactive Visualization and Analysis of Imaging Spectrometer Data, *Remote Sensing of the Environment*, 44: 145-163.

Harsanyi, J. C. and Chang, C.-I., 1994. Hyperspectral image classification and dimensionality reduction: An orthogonal

- subspace projection approach. *IEEE Transactions on Geoscience and Remote Sensing*, 32(4), pp.779-785.
- Richards, J.A., 1993. *Remote Sensing Digital Image Analysis: An Introduction*, Second Edition, Springer-Verlag, Berlin.
- Settle J.J. and N.A. Drake, 1993, Linear mixing and the estimation of ground proportions, *International Journal of Remote Sensing*, 14(6), pp. 1159-1177.
- Shimabukuro Y.E. and J.A. Smith, 1991, The least-squares mixing models to generate fraction images derived from remote sensing multispectral data, *IEEE Transactions on Geoscience and Remote Sensing*, 29(1), pp.16-20.
- Sohn, Y. and R.M. McCoy, 1997, Mapping desert shrub rangeland using spectra unmixing and modeling spectral mixtures with TM data, *Photogrammetric Engineering and Remote Sensing*, 63(6) pp.707-716.
- Wang, F., 1990. Fuzzy Supervised Classification of Remote Sensing Images, *IEEE Transactions on Geoscience and Remote Sensing*, 28(2): 194-201.

RADAR Imaging and Analysis

The term RADAR is an acronym for Radio Detection And Ranging. RADAR imaging uses radio waves to detect surface characteristics. There are a number of RADAR Sensing instruments that provide a range of data products to the user community. These include: ERS (European Remote Sensing Satellite), JERS (Japanese Earth Resource Satellite) and the Canadian RADARSAT. RADARSAT is a commercial system (unlike the other systems which are largely scientific missions) intended to provide data for a wide range of uses.

Optical multispectral remote sensing systems like Landsat and SPOT gather data as reflected electromagnetic energy. The source of the energy for these sensors is the sun. Unlike these passive optical systems, RADAR systems such as RADARSAT are active sensing systems, meaning that they provide the energy that is then measured as it returns to the sensor. RADAR systems use energy transmitted at microwave frequencies that are not detectable by the human eye. Most RADAR systems operate at one single frequency. RADARSAT, for example, operates at what is known as C-band frequency⁸² (5.3 Ghz frequency or 5.6 cm wavelength) and thus acquires single band data. The sensing instrument used on RADARSAT is known as Synthetic Aperture Radar (SAR). This instrument uses the motion of the satellite and doppler frequency shift to electronically synthesize the large antennae required for the acquisition of high resolution RADAR imagery. The sensor sends a microwave energy pulse to the earth and measures the time it takes for that pulse to return after interaction with the earth's surface.

The Nature of RADAR Data: Advantages and Disadvantages

Since RADARSAT-SAR uses microwave energy, it is able to penetrate atmospheric barriers that often hinder optical imaging. SAR can "see" through clouds, rain, haze and dust and can operate in darkness, making data capture possible in any atmospheric conditions. Because microwave energy can penetrate the land surface to appreciable depths, it is useful in arid environments. Additionally, microwave energy is most appropriate for studying subsurface conditions and properties which are relevant to resource prospecting and archaeological explorations. The data gathered by this sensor is of the form known as HH (horizontal transmit, horizontal receive)⁸³ polarized data. The signal sent to the earth surface has a horizontal orientation (or polarization) and is captured using the same polarization. Variations in the return beam, termed *backscatter*, result from varying surface roughness, topography and surface moisture conditions. These characteristics of the RADAR signal can be exploited to infer the structural and textural characteristics of the target objects, unlike the simple reflection signal of optical sensing. RADARSAT-SAR collects data in a variety of beam modes allowing a choice of incidence angles and resolutions (from 20m up to 100m). Since the instrument can be navigated to view the same location from different beam positions, it can acquire data in stereo pairs that can be used for a large number of terrain and topographic analysis needs.

There are, however, some problems that are unique to RADAR imaging. Most, if not all, RADAR images have a speckled or grainy appearance. This is the result of a combination of multiple scattering within a given pixel. In RADAR terms, a large number of surface materials exhibit *diffuse* reflectance patterns. Farmlands, wet soils and forest canopies, for example, show high signal returns, while surfaces like roads, pavements and smooth water surfaces show *specular* reflectance patterns with low signal returns. A mixture of different cover types generates a complex image surface. The data are thus inherently "noisy" and require substantial preprocessing before use in a given analysis task. In mountainous terrain, shad-

82. Other RADAR frequencies include X - band 12.5-8.0 Ghz (2.4 - 3.75 cm), L - band : 2.0-1.0 Ghz (15-30 cm) and P-band 1.0-0.3 Ghz (30-100 cm). The frequency of the C-band is 8.0-4.0 Ghz (3.75-7.5 cm).

83. Other options are vertical transmit, vertical receive (VV), horizontal transmit, vertical receive (HV) and vertical transmit, horizontal receive (VH).

owing or relief displacement occurs because the reflected RADAR pulse from the top of a mountain ridge reaches the antenna before the pulse from the base or lee side of the feature does. This is known as the *layover effect* and is significant in areas of steep slopes. If the slopes facing the antenna are less steep, the pulse reaches the base before the top. This leads to slopes appearing compressed on the imagery—an effect known as *foreshortening*. In urban areas, double reflectance from *corner reflectors* like buildings causes imagery to have a bright, sparkled appearance. Mixed with reflectance from trees and road surfaces, the urban scene makes filtering operations rather challenging. If a bright target dominates a given area, it may skew the distribution of the data values. It must be noted, however, that the RADAR's advantages, including its ability for all-weather imaging, far outweigh its disadvantages.

Using RADAR Data in IDRISI

The module named RADARSAT can be used to import RADARSAT data into IDRISI. Furthermore, the generic data import tools of IDRISI can be used to import most other formats. Since each scene is a single band, it can be best viewed using a grey scale palette. The RADAR palette in IDRISI can be used to view RADAR imagery including those scenes that have data distributions that exemplify negatively skewed histograms. An alternative method is to generate a histogram, estimate the range within which a majority of the data values fall, and STRETCH the image to that level. This works very well for visual analysis of the data. (See the Image Exploration exercise in the **Tutorial** manual.)

Currently, the use of RADAR data in environmental analysis and resource management is still in an early stage and is quite limited compared to the widespread use of more traditional image data (e.g., MSS, TM, aerial photography). More sophisticated tools for interpretation of RADAR data are likely to evolve in coming years. Some useful techniques for working with RADAR data (e.g., spatial filter kernels) have been reported in the literature.

In order to facilitate processing and interpretation of RADAR data, IDRISI offers a texture analysis module (TEXTURE) and several options in the FILTER module, including the Adaptive Box Filter. These filtering procedures can be used to minimize some of the speckled "noise" effects in RADAR imagery (although they cannot be completely eliminated) and also provide a quantitative interpretation of RADAR imaged surfaces. The Adaptive Box filter, which is an adaptation of the Lee filter, is highly recommended by Eliason and McEwen (1990) for reducing speckled effects (Figure 17-1). One must experiment with various filter kernel sizes and threshold options in order to achieve good results. Different filters may work better for different scenes, depending on the mixtures of land surface cover types in the imagery.

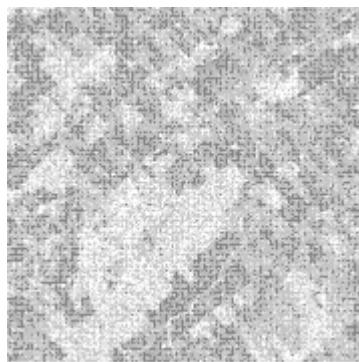


Figure 17-1

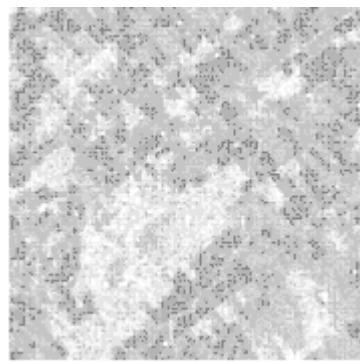


Figure 17-2

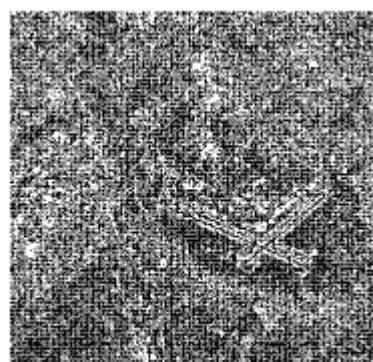


Figure 17-3

One other method suggested for further reducing speckled effects is *multi-look processing* (Lillesand and Kiefer, 1987). This simply involves averaging (with OVERLAY or Image Calculator) scenes from the same area, acquired at different incidence angles, to produce a smoother image.

Quick-look images can be generated using the MEAN filter in IDRISI (Figure 17-2) to enhance some of the features in image scenes in order to select sites for detailed analysis. The Sobel Edge Detector, the High Pass and the Laplacian Edge Enhancement filters are useful in detecting edges and linear features in imagery. Fault lines, surface drainage patterns, folds, roads and land/water boundaries are useful in various geological, resource management and a variety of environmental and planning applications. Note also that user-defined filters may be tailored to fit specific needs using both the 3 x 3 kernel and the variable size kernel in IDRISI's FILTER module. See the module description of FILTER in the on-line Help System for a detailed explanation of these filtering techniques.

As already mentioned, the RADAR backscatter signal is composed of the various diffuse and specular responses of scene elements to the sent RADAR pulse. The compound pattern of the varying surface responses can be exploited in order to determine the textural characteristics of the land surface. These characteristics can be derived using the TEXTURE module in IDRISI. TEXTURE includes three categories of analysis. The first uses variability in a moving window to assess several different measures, including entropy. The second estimates the fractal dimension of the image surface. The third provides directional edge enhancement filters to enhance edge patterns in different directions. For example, Figure 17-3 shows a fractal surface derived using the TEXTURE module. Each of the surfaces derived from TEXTURE can be used as input in a classification scheme for RADAR imagery. Thus, instead of using spectral responses, one utilizes scene textural characteristics in classification.

The 24-day repeat cycle of RADARSAT can be exploited in the monitoring of surface phenomena that exhibit temporal variability, as each time step has a different RADAR backscatter signal. Using the COMPOSIT routine in IDRISI, multi-date RADAR imagery can be combined to produce a false color composite (very much like multispectral single scene data) that shows the temporal transitions in given earth surface components like crops or different vegetation types or surface cover types.

It is anticipated that as RADAR data becomes more widely used, there will be accompanying developments in software to exploit the unique character of RADAR imagery.

References

- Eliason, E. M., and McEwen, A. S., 1990. Adaptive Box Filters for Removal of Random Noise from Digital Images, *Photogrammetric Engineering and Remote Sensing*, 56(4): 453-458.
- Lillesand, T., and Kiefer, R. W., 1987. *Remote Sensing and Image Interpretation*, John Wiley and Sons, New York, 472-524.
- RADARSAT International, 1995. *RADARSAT Guide to Products and Services*, RADARSAT International, Richmond, B.C., Canada.

Vegetation Indices

by Amadou Thiam and J. Ronald Eastman

Introduction

Analysis of vegetation and detection of changes in vegetation patterns are keys to natural resource assessment and monitoring. Thus it comes as no surprise that the detection and quantitative assessment of green vegetation is one of the major applications of remote sensing for environmental resource management and decision making.

Healthy canopies of green vegetation have a very distinctive interaction with energy in the visible and near-infrared regions of the electromagnetic spectrum. In the visible regions, plant pigments (most notably chlorophyll) cause strong absorption of energy, primarily for the purpose of photosynthesis. This absorption peaks in the red and blue areas of the visible spectrum, thus leading to the characteristic green appearance of most leaves. In the near-infrared, however, a very different interaction occurs. Energy in this region is not used in photosynthesis, and it is strongly scattered by the internal structure of most leaves, leading to a very high apparent reflectance in the near-infrared. It is this strong contrast, then, most particularly between the amount of reflected energy in the red and near-infrared regions of the electromagnetic spectrum, that has been the focus of a large variety of attempts to develop quantitative indices of vegetation condition using remotely sensed imagery.

The aim of this chapter is to present a set of vegetation index (VI) models designed to provide a quantitative assessment of green vegetation biomass. The proposed VIs are applicable to both low and high spatial resolution satellite images, such as NOAA AVHRR, Landsat TM and MSS, SPOT HRV/XS, and any others similar to these that sense in the red and near-infrared regions. They have been used in a variety of contexts to assess green biomass and have also been used as a proxy to overall environmental change, especially in the context of drought (Kogan, 1990; Tripathy et al., 1996; Liu and Kogan, 1996) and land degradation risk assessment. As a consequence, special interest has been focused on the assessment of green biomass in arid environments where soil background becomes a significant component of the signal detected.

This chapter reviews the character of over 20 VIs that are provided by the TASSCAP and VEGINDEX modules in the IDRISI system software. They are provided to facilitate the use of these procedures and to further the debate concerning this very important environmental index. We welcome both your comments on the VIs currently included in IDISI as well as your suggestions for future additions to the set.

Classification of Vegetation Indices

Jackson and Huete (1991) classify VIs into two groups: *slope-based* and *distance-based* VIs. To appreciate this distinction, it is necessary to consider the position of vegetation pixels in a two-dimensional graph (or *bi-spectral plot*) of red versus infrared reflectance. The slope-based VIs are simple arithmetic combinations that focus on the contrast between the spectral response patterns of vegetation in the red and near-infrared portions of the electromagnetic spectrum. They are so named because any particular value of the index can be produced by a set of red/infrared reflectance values that form a line emanating from the origin of a bi-spectral plot. Thus different levels of the index can be envisioned as producing a spectrum of such lines that differ in their slope. Figure 18-1a, for example, shows a spectrum of Normalized Difference Vegetation Index (the most commonly used of this group) lines ranging from -0.75 fanning clockwise to +0.75 (assuming infrared as the X axis and red as the Y axis), with NDVI values of 0 forming the diagonal line.

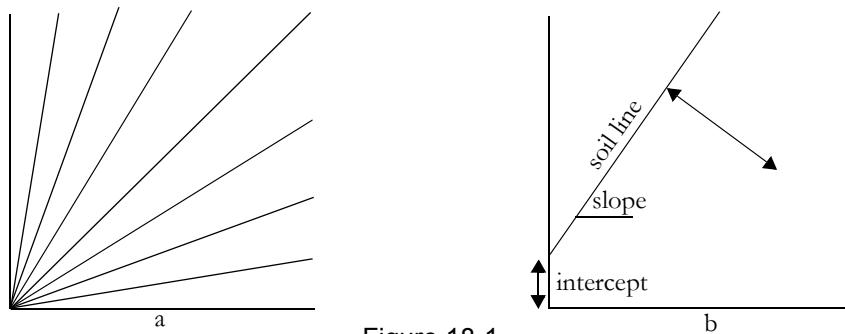


Figure 18-1

In contrast to the slope-based group, the distance-based group measures the degree of vegetation present by gauging the difference of any pixel's reflectance from the reflectance of bare soil. A key concept here is that a plot of the positions of bare soil pixels of varying moisture levels in a bi-spectral plot will tend to form a line (known as a *soil line*). As vegetation canopy cover increases, this soil background will become progressively obscured, with vegetated pixels showing a tendency towards increasing perpendicular distance from this soil line (Figure 18-1b). All of the members of this group (such as the Perpendicular Vegetation Index—PVI) thus require that the slope and intercept of the soil line be defined for the image being analyzed.

To these two groups of vegetation indices, a third group can be added called *orthogonal transformation* VIs. Orthogonal indices undertake a transformation of the available spectral bands to form a new set of uncorrelated bands within which a green vegetation index band can be defined. The Tasseled Cap transformation is perhaps the most well-known of this group.

A Special Note About Measurement Scales: IDRISI differs from most other GIS and image processing software in that it supports real number images. Thus the descriptions that follow describe these vegetation indices without rescaling to suit more limited data types. However, in most implementations, a subsequent rescaling is required to make the index suitable for expression in an integer form (e.g., a rescaling of values from a -1.0 to +1.0 real number range to a 0-255 8-bit integer range). In IDRISI, this is not required, and thus the indices are produced and described in their purest form.

The Slope-Based VIs

Slope-based VIs are combinations of the visible red and the near infrared bands and are widely used to generate vegetation indices. The values indicate both the status and abundance of green vegetation cover and biomass. The slope-based VIs include the RATIO, NDVI, RVI, NRVI, TVI, CTVI, and TTVI. The module VEGINDEX in IDRISI may be used to generate an image for each of these VIs.

The Ratio Vegetation Index (RATIO) was proposed by Rouse, et al. (1974) to separate green vegetation from soil background using Landsat MSS imagery. The RATIO VI is produced by simply dividing the reflectance values contained in the near infrared band by those contained in the red band, i.e.:

$$\text{RATIO} = \frac{\text{NIR}}{\text{RED}}$$

The result clearly captures the contrast between the red and infrared bands for vegetated pixels, with high index values being produced by combinations of low red (because of absorption by chlorophyll) and high infrared (as a result of leaf structure) reflectance. In addition, because the index is constructed as a ratio, problems of variable illumination as a result of topography are minimized. However, the index is susceptible to division by zero errors and the resulting measurement scale is not linear. As a result, RATIO VI images do not have normal distributions (Figure 18-2), making it difficult to

apply some statistical procedures.

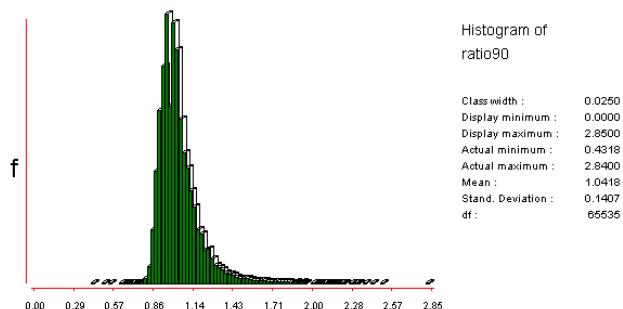


Figure 18-2 Histogram of a RATIO VI Image

The Normalized Difference Vegetation Index (NDVI) was also introduced by Rouse et al. (1974) in order to produce a spectral VI that separates green vegetation from its background soil brightness using Landsat MSS digital data. It is expressed as the difference between the near infrared and red bands normalized by the sum of those bands, i.e.:

$$\text{NDVI} = \frac{\text{NIR} - \text{RED}}{\text{NIR} + \text{RED}}$$

This is the most commonly used VI as it retains the ability to minimize topographic effects while producing a linear measurement scale. In addition, division by zero errors are significantly reduced. Furthermore, the measurement scale has the desirable property of ranging from -1 to 1 with 0 representing the approximate value of no vegetation. Thus negative values represent non-vegetated surfaces.

The Transformed Vegetation Index (TVI) (Deering et al., 1975) modifies the NDVI by adding a constant of 0.50 to all its values and taking the square root of the results. The constant 0.50 is introduced in order to avoid operating with negative NDVI values. The calculation of the square root is intended to correct NDVI values that approximate a Poisson distribution and introduce a normal distribution. With these two elements, the TVI takes the form:

$$\text{TVI} = \sqrt{\left(\frac{\text{NIR} - \text{RED}}{\text{NIR} + \text{RED}}\right)} + 0.5$$

However, the use of TVI requires that the minimum input NDVI values be greater than -0.5 to avoid aborting the operation. Negative values still will remain if values less than -0.5 are found in the NDVI. Moreover, there is no technical difference between NDVI and TVI in terms of image output or active vegetation detection.

The Corrected Transformed Vegetation Index (CTVI) proposed by Perry and Lautenschlager (1984) aims at correcting the TVI. Clearly adding a constant of 0.50 to all NDVI values does not always eliminate all negative values as NDVI values may have the range -1 to +1. Values that are lower than -0.50 will leave small negative values after the addition operation. Thus, the CTVI is intended to resolve this situation by dividing $(\text{NDVI} + 0.50)$ by its absolute value $\text{ABS}(\text{NDVI} + 0.50)$ and multiplying the result by the square root of the absolute value $(\text{SQRT}[\text{ABS}(\text{NDVI} + 0.50)])$. This suppresses the negative NDVI. The equation is written:

$$\text{CTVI} = \frac{\text{NDVI} + 0.5}{\text{ABS}(\text{NDVI} + 0.5)} \times \sqrt{\text{ABS}(\text{NDVI} + 0.5)}$$

Given that the correction is applied in a uniform manner, the output image using CTVI should have no difference with the initial NDVI image or the TVI whenever TVI properly carries out the square root operation. The correction is intended to eliminate negative values and generate a VI image that is similar to, if not better than, the NDVI. However, Thiam (1997) indicates that the resulting image of the CTVI can be very "noisy" due to an overestimation of the greenness. He suggests ignoring the first term of the CTVI equation in order to obtain better results. This is done by simply taking the square root of the absolute values of the NDVI in the original TVI expression to have a new VI called **Thiam's Transformed Vegetation Index (TTVI)**.

$$\text{TTVI} = \sqrt{\text{ABS}(\text{NDVI} + 0.5)}$$

The simple **Ratio Vegetation Index (RVI)** was suggested by Richardson and Wiegand (1977) as graphically having the same strengths and weaknesses as the TVI (see above) while being computationally simpler. RVI is clearly the reverse of the standard simple ratio (RATIO) as shown by its expression:

$$RVI = \frac{RED}{NIR}$$

The **Normalized Ratio Vegetation Index (NRVI)** is a modification of the RVI by Baret and Guyot (1991) whereby the result of $RVI - 1$ is normalized over $RVI + 1$.

$$NRVI = \frac{RVI - 1}{RVI + 1}$$

This normalization is similar in effect to that of the NDVI, i.e., it reduces topographic, illumination and atmospheric effects and it creates a statistically desirable normal distribution.

The Distance-Based VIs

This group of vegetation indices is derived from the Perpendicular Vegetation Index (PVI) discussed in detail below. The main objective of these VIs is to cancel the effect of soil brightness in cases where vegetation is sparse and pixels contain a mixture of green vegetation and soil background. This is particularly important in arid and semi-arid environments.

The procedure is based on the soil line concept as outlined earlier. The soil line represents a description of the typical signatures of soils in a red/near-infrared bi-spectral plot. It is obtained through linear regression of the near-infrared band against the red band for a sample of bare soil pixels. Pixels falling near the soil line are assumed to be soils while those far away are assumed to be vegetation. Distance-based VIs using the soil line require the slope (b) and intercept (a) of the line as inputs to the calculation. Unfortunately, there has been a remarkable inconsistency in the logic with which this soil line has been developed for specific VIs. One group requires the red band as the independent variable and the other requires the near-infrared band as the independent variable for the regression. The on-line Help System for VEGINDEX should be consulted for each VI in the Distance-based group to indicate which of these two approaches should be used.

Figure 18-3 shows the soil line and its parameters as calculated for a set of soil pixels using the REGRESS module in IDRISI. The procedure requires that you identify a set of bare soil pixels as a boolean mask (1=soil / 0=other). REGRESS is then used to regress the red band against the near-infrared band (or vice versa, depending upon the index), using this mask to define the pixels from which the slope and intercept should be defined. A worked example of this procedure can be found in the **Tutorial** in the Vegetation Analysis in Arid Environments exercise.

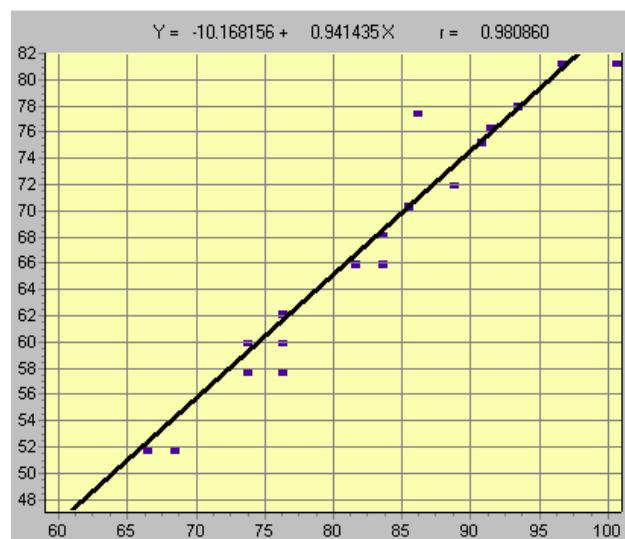


Figure 18-3

The Perpendicular Vegetation Index (PVI) suggested by Richardson and Wiegand (1977) is the parent index from which this entire group is derived. The PVI uses the perpendicular distance from each pixel coordinate (e.g., Rp5, Rp7) to the soil line as shown in Figure 18-4.

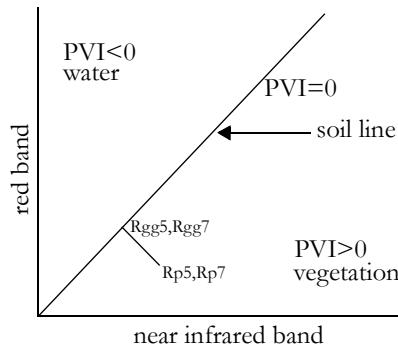


Figure 18-4 The Perpendicular Vegetation Index (from Richardson and Wiegand, 1977)

To derive this perpendicular distance, four steps are required:

- 1) Determine the equation of the soil line by regressing bare soil reflectance values for red (dependent variable) versus infrared (independent variable).⁸⁴ This equation will be in the form:

$$Rg5 = a_0 + a_1 Rg7$$

where $Rg5$ is a Y position on the soil line

$Rg7$ is the corresponding X coordinate

a_1 is the slope of the soil line

a_0 is the Y-intercept of the soil line

- 2) Determine the equation of the line that is perpendicular to the soil line. This equation will have the form:

$$Rp5 = b_0 + b_1 Rp7$$

where $b_0 = Rp5 - b_1 Rp7$

where

$Rp5$ = red reflectance

$Rp7$ = infrared reflectance

and

$$b_1 = -1/a_1$$

where

a_1 = the slope of the soil line

84. Check the on-line Help System for each VI to determine which band should be used as the dependent and independent variables. In this example, for the PVI, red is dependent and infrared is independent.

3) Find the intersection of these two lines (i.e., the coordinate Rgg5,Rgg7).

$$Rgg5 = \frac{b_1 a_0 - b_0 a_1}{b_1 - a_1}$$

$$Rgg7 = \frac{a_0 - b_0}{b_1 - a_1}$$

4) Find the distance between the intersection (Rgg5,Rgg7) and the pixel coordinate (Rp5,Rp7) using the Pythagorean Theorem.

$$PVI = \sqrt{(Rgg5 - Rp5)^2 + (Rgg7 - Rp7)^2}$$

Attempts to improve the performance of the PVI have yielded three others suggested by Perry and Lautenschlager (1984), Walther and Shabaani (1991), and Qi, et al. (1994). In order to avoid confusion, the derived PVIs are indexed 1 to 3 (PVI₁, PVI₂, PVI₃).

PVI₁ was developed by Perry and Lautenschlager (1984) who argued that the original PVI equation is computationally intensive and does not discriminate between pixels that fall to the right or left side of the soil line (i.e., water from vegetation). Given the spectral response pattern of vegetation in which the infrared reflectance is higher than the red reflectance, all vegetation pixels will fall to the right of the soil line (e.g., pixel 2 in Figure 18-5). In some cases, a pixel representing non-vegetation (e.g., water) may be equally far from the soil line, but lies to the left of that line (e.g., pixel 1 in Figure 18-5). In the case of PVI, that water pixel will be assigned a high vegetation index value. PVI₁ assigns negative values to those pixels lying to the left of the soil line.

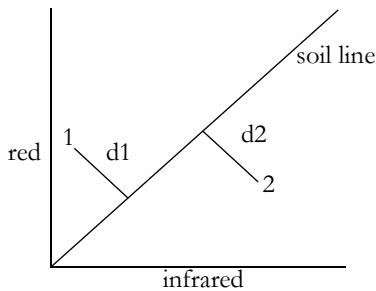


Figure 18-5 Distance from the Soil Line

The equation is written:

$$PVI_1 = \frac{(bNIR - RED + a)}{\sqrt{b^2 + 1}}$$

where

- | | | |
|-----|---|---------------------------------------|
| NIR | = | reflectance in the near infrared band |
| RED | = | reflectance in the visible red band |
| a | = | intercept of the soil line |
| b | = | slope of the soil line |

PVI₂ (Walther and Shabaani, 1991; Bannari, et al., 1996) weights the red band with the intercept of the soil line and is

written⁸⁵:

$$PVI_2 = \frac{(NIR - a) \cdot (RED + b)}{\sqrt{1 + a^2}}$$

where

NIR = reflectance in the near infrared band

RED = reflectance in the visible red band

a = intercept of the soil line

b = slope of the soil line

PVI₃, presented by Qi, et al (1994), is written:

$$PVI_3 = a pNIR - b pRED$$

where

$pNIR$ = reflectance in the near infrared band

$pRED$ = reflectance in the visible red band

a = intercept of the soil line

b = slope of the soil line

Difference Vegetation Index (DVI) is also suggested by Richardson and Wiegand (1977) as an easier vegetation index calculation algorithm. The particularity of the DVI is that it weights the near-infrared band by the slope of the soil line. It is written:

$$DVI = g MSS7 - MSS5$$

where

g = the slope of the soil line

$MSS7$ = reflectance in the near infrared 2 band

$MSS5$ = reflectance in the visible red band

Similar to the PVI₁, with the DVI, a value of zero indicates bare soil, values less than zero indicate water, and those greater than zero indicate vegetation.

The Ashburn Vegetation Index (AVI) (Ashburn, 1978) is presented as a measure of green growing vegetation. The values in MSS7 are multiplied by 2 in order to scale the 6-bit data values of this channel to match with the 8-bit values of MSS5. The equation is written:

$$AVI = 2.0MSS7 - MSS5$$

This scaling factor would not apply wherever both bands are 7-bit or 8-bit and the equation is rewritten as a simple subtraction.

The Soil-Adjusted Vegetation Index (SAVI) is proposed by Huete (1988). It is intended to minimize the effects of soil background on the vegetation signal by incorporating a constant soil adjustment factor L into the denominator of the NDVI equation. L varies with the reflectance characteristics of the soil (e.g., color and brightness). Huete (1988) provides

85. In Bannari, et al. (1996), **a** is used to designate the slope and **b** is used to designate the intercept. More commonly in linear regression, **a** is the intercept and **b** the slope of the fitted line. This has been corrected here.

a graph from which the values of L can be extracted (Figure 18-6). The L factor chosen depends on the density of the vegetation one wishes to analyze. For very low vegetation, Huete et al., (1988) suggest using an L factor of 1.0, for intermediate 0.5 and for high densities 0.25. Walther and Shabaani (1991) suggest that the best L value to select is where the difference between SAVI values for dark and light soil is minimal. For $L = 0$, SAVI equals NDVI. For $L = 100$, SAVI approximates PVI.

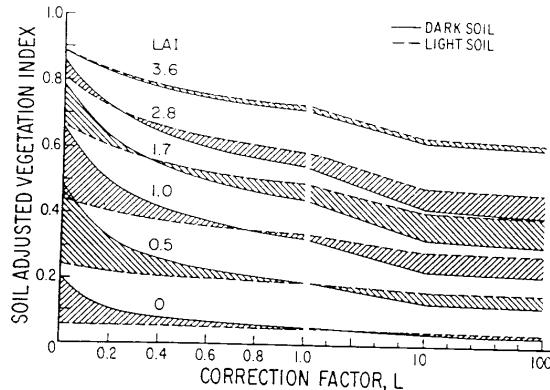


Figure 18-6 Influence of light and dark soil on the SAVI values of cotton as a function of the shifted correction factor L (from Huete, 1988).

The equation is written:

$$\text{SAVI} = \frac{\rho_{\text{nir}} - \rho_{\text{red}}}{(\rho_{\text{nir}} + \rho_{\text{red}} + L)} \cdot (1 + L)$$

where

- ρ_{nir} = near-infrared band
- ρ_{red} = visible red band
- L = soil adjustment factor

The Transformed Soil-Adjusted Vegetation Index (TSAVI₁) was defined by Baret, et al. (1989) who argued that the SAVI concept is exact **only** if the constants of the soil line are $a=1$ and $b=0$ (note the reversal of these common symbols). Because this is not generally the case, they transformed SAVI. By taking into consideration the PVI concept, they proposed a first modification of TSAVI designated as TSAVI₁. The transformed expression is written:

$$\text{TSAVI}_1 = \frac{a((\text{NIR} - a)(\text{RED} - b))}{\text{RED} + a \cdot \text{NIR} - a \cdot b}$$

where

- NIR = reflectance in the near infrared band (expressed as reflectances)
- RED = reflectance in the visible red band (expressed as reflectances)
- a = slope of the soil line
- b = intercept of the soil line

With some resistance to high soil moisture, TSAVI₁ could be a very good candidate for use in semi-arid regions. TSAVI₁ was specifically designed for semi-arid regions and does not work well in areas with heavy vegetation.

TSAVI was readjusted a second time by Baret, et al (1991) with an additive correction factor of 0.08 to minimize the effects of the background soil brightness. The new version is named **TSAVI₂** and is given by:

$$\text{TSAVI}_2 = \frac{a(\text{NIR} - a\text{RED} - b)}{\text{RED} + a\text{NIR} - ab + 0.08(1 + a^2)}$$

The Modified Soil-Adjusted Vegetation Indices (MSAVI₁ and MSAVI₂) suggested by Qi, et al. (1994) are based on a modification of the L factor of the SAVI. Both are intended to better correct the soil background brightness in different vegetation cover conditions.

With MSAVI₁, L is selected as an empirical function due to the fact that L decreases with decreasing vegetation cover as is the case in semi-arid lands (Qi, et al., 1994). In order to cancel or minimize the effect of the soil brightness, L is set to be the product of NDVI and WDVI (described below). Therefore, it uses the opposite trends of NDVI and WDVI. The full expression of MSAVI₁ is written:

$$\text{MSAVI}_1 = \frac{\text{NIR} - \text{RED}}{\text{NIR} + \text{RED} + L} \cdot (1 + L)$$

where

NIR = reflectance in the near infrared band

RED = reflectance in the visible red band

L = $1 - 2 \gamma \text{NDVI} * \text{WDVI}$

where

NDVI = Normalized Difference Vegetation Index

WDVI = Weighted Difference Vegetation Index

γ = slope of the background soil line

2 = used to increase the L dynamic range

range of L = 0 to 1

The second modified SAVI, MSAVI₂, uses an inductive L factor to:

1. remove the soil "noise" that was not canceled out by the product of NDVI by WDVI, and
2. correct values greater than 1 that MSAVI₁ may have due to the low negative value of NDVI*WDVI. Thus, its use is limited for high vegetation density areas.

The general expression of MSAVI₂ is:

$$\text{MSAVI}_2 = \frac{2p\text{NIR} + 1 - \sqrt{(2p\text{NIR} + 1)^2 - 8(p\text{NIR} - p\text{RED})}}{2}$$

where

pNIR = reflectance of the near infrared band

pRED = reflectance of the red band

The Weighted Difference Vegetation Index (WDVI) has been attributed to Richardson and Wiegand (1977), and Clevers (1978) by Kerr and Pichon (1996), writing the expression as:

$$\text{WDVI} = \rho_n - \gamma \rho_r$$

where

- ρ_n = reflectance of near infrared band
 ρ_r = reflectance of visible red band
 γ = slope of the soil line

Although simple, WDVI is as efficient as most of the slope-based VIs. The effect of weighting the red band with the slope of the soil line is the maximization of the vegetation signal in the near-infrared band and the minimization of the effect of soil brightness.

The Orthogonal Transformations

The derivation of vegetation indices has also been approached through orthogonal transformation techniques such as the PCA, the GVI of the Kauth-Thomas Tasseled Cap Transformation and the MGVI of the Wheeler-Misra orthogonal transformation. The link between these three techniques is that they all express green vegetation through the development of their second component.

Principal Components Analysis (PCA) is an orthogonal transformation of n -dimensional image data that produces a new set of images (components) that are uncorrelated with one another and ordered with respect to the amount of variation (information) they represent from the original image set. PCA is typically used to uncover the underlying dimensionality of multi-variate data by removing redundancy (evident in inter-correlation of image pixel values), with specific applications in GIS and image processing ranging from data compression to time series analysis. In the context of remotely sensed images, the first component typically represents albedo (in which the soil background is represented) while the second component most often represents variation in vegetative cover. For example, component 2 generally has positive loadings on the near-infrared bands and negative loadings on the visible bands. As a result, the green vegetation pattern is highlighted in this component (Singh and Harrison, 1985; Fung and LeDrew, 1987; Thiam, 1997). This is illustrated in Table 18-1 corresponding to the factor loadings of a 1990 MSS image of southern Mauritania.

Table 18-1 Factor loadings of the 1990 PCA

	Comp1	Comp2	Comp3	Comp4
MSS1	0.86	-0.46	-0.22	0.01
MSS2	0.91	-0.36	0.19	-0.08
MSS3	0.95	0.25	0.09	0.14
MSS4	0.75	0.65	-0.08	-0.09

The Green Vegetation Index (GVI) of the Tasseled Cap is the second of the four new bands that Kauth and Thomas (1976) extracted from raw MSS images. The GVI provides global coefficients that are used to weight the original MSS digital counts to generate the new transformed bands. The TASSCAP module in IDRISI is specifically provided to calculate the Tasseled Cap bands from Landsat MSS or TM images. The output from TASSCAP corresponding to GVI is xxgreen (xx = the two character prefix entered by the user) by default. The expression of the green vegetation index band, GVI, is written as follows for MSS or TM data:

$$GVI = [(-0.386MSS4) + (-0.562MSS5) + (0.600MSS6) + (0.491MSS7)]$$

$$GVI = [(-0.2848TM1) + (-0.2435TM2) + (-0.5436TM3) + (0.7243TM4) + (0.0840TM5) + (-0.1800TM7)]$$

The negative weights of the GVI on the visible bands tend to minimize the effects of the background soil, while its positive weights on the near infrared bands emphasize the green vegetation signal.

Misra's Green Vegetation Index (MGVI) is the equivalent of the Tasseled Cap GVI and is proposed by Wheeler et al. (1976) and Misra, et al. (1977) as a spectral vegetation index. It is the second of the four new bands produced from an application of the Principal Components Analysis to MSS digital counts. The algebraic expression of the MGVI is:

$$\text{MGVI} = -0.386\text{MSS4} - 0.530\text{MSS5} + 0.535\text{MSS6} + 0.532\text{MSS7}$$

The principle of the MGVI is to weight the original digital counts by some global coefficients provided by Wheeler and Misra in order to generate a second Principal Component. However, the use of these global coefficients may not yield the same result as a directly calculated second Principal Component, as they may be site specific. The coefficients correspond to the eigenvectors that are produced with a Principal Components Analysis. The eigenvectors indicate the direction of the principal axes (Mather, 1987). They are combined with the original spectral values to regenerate Principal Components. For example PCA1 is produced by combining the original reflectances with the eigenvectors (column values) associated with component 1. Likewise, component 2 (MGVI) is produced by combining the original digital counts with the eigenvectors associated with component 2 as highlighted in Table 18-2.

Table 18-2 Eigenvectors of the 1990 PCA

	Comp1	Comp2	Comp3	Comp4
eigvec.1	0.49	-0.51	-0.70	0.08
eigvec.2	0.52	-0.40	0.60	-0.45
eigvec.3	0.55	0.28	0.27	0.74
eigvec.4	0.43	0.71	-0.27	-0.49

The PCA module in IDRISI generates eigenvectors as well as factor loadings with the component images. A site-specific MGVI image can then be produced with Image Calculator by using the appropriate eigenvector values. The following equation would be used to produce the MGVI image for the example shown in Table 18-2:

$$\text{MGVI90} = (-0.507\text{MSS4}) + (-0.400\text{MSS5}) + (0.275\text{MSS6}) + (0.712\text{MSS7})$$

Summary

The use of any of these transformations depends on the objective of the investigation and the general geographic characteristics of the application area. In theory, any of them can be applied to any geographic area, regardless of their sensitivity to various environmental components that might limit their effectiveness. In this respect, one might consider applying the slope-based indices as they are simple to use and yield numerical results that are easy to interpret. However, including the well known NDVI, they all have the major weakness of not being able to minimize the effects of the soil background. This means that a certain proportion of their values, negative or positive, represents the background soil brightness. The effect of the background soil is a major limiting factor to certain statistical analyses geared towards the quantitative assessment of above-ground green biomass.

Although they produce indices whose extremes may be much lower and greater than those of the more familiar NDVI, the distance-based VIs have the advantage of minimizing the effects of the background soil brightness. This minimization is performed by combining the input bands with the slope and intercept of the soil line obtained through a linear regression between bare soil sample reflectance values extracted from the red and near-infrared bands. This represents an

important quantitative and qualitative improvement of the significance of the indices for all types of applications, particularly for those dealing with arid and semi-arid environments. To take advantage of these, however, you do need to be able to identify bare soil pixels in the image.

The orthogonal VIs, namely the Tasseled Cap, Principal Components Analysis and the Wheeler-Misra transformation (MGVI), proceed by a decorrelation of the original bands through orthogonalization in order to extract new bands. By this process, they produce a green band that is somehow free of soil background effects, since almost all soil characteristics are ascribed to another new band called brightness.

Despite the large number of vegetation indices currently in use, it is clear that much needs to be learned about the application of these procedures in different environments. It is in this spirit that the VEGINDEX and TASSCAP modules have been created. However, it has also become clear that remote sensing offers a significant opportunity for studying and monitoring vegetation and vegetation dynamics.

References

- Ashburn, P., 1978. The vegetative index number and crop identification, *The LACIE Symposium Proceedings of the Technical Session*, 843-850.
- Bannari, A., Huete, A. R., Morin, D., and Zagolski, 1996. Effets de la Couleur et de la Brillance du Sol Sur les Indices de Végétation, *International Journal of Remote Sensing*, 17(10): 1885-1906.
- Baret, F., Guyot, G., and Major, D., 1989. TSAVI: A Vegetation Index Which Minimizes Soil Brightness Effects on LAI and APAR Estimation, *12th Canadian Symposium on Remote Sensing and IGARSS'90*, Vancouver, Canada, 4.
- Baret, F., and Guyot, G., 1991. Potentials and Limits of Vegetation Indices for LAI and APAR Assessment, *Remote Sensing and the Environment*, 35: 161-173.
- Deering, D. W., Rouse, J. W., Haas, R. H., and Schell, J. A., 1975. Measuring "Forage Production" of Grazing Units From Landsat MSS Data, *Proceedings of the 10th International Symposium on Remote Sensing of Environment, II*, 1169-1178.
- Fung, T., and LeDrew, E., 1988. The Determination of Optimal Threshold Levels for Change Detection Using Various Accuracy Indices, *Photogrammetric Engineering and Remote Sensing*, 54(10): 1449-1454.
- Huete, A. R., 1988. A Soil-Adjusted Vegetation Index (SAVI), *Remote Sensing and the Environment*, 25: 53-70.
- Jackson, R. D., 1983. Spectral Indices in *n*-Space, *Remote Sensing and the Environment*, 13: 409-421.
- Kauth, R. J., and Thomas, G. S., 1976. The Tasseled Cap - A Graphic Description of the Spectral Temporal Development of Agricultural Crops As Seen By Landsat. *Proceedings of the Symposium on Machine Processing of Remotely Sensed Data*, Purdue University, West Lafayette, Indiana, 41-51.
- Kogan, F. N., 1990. Remote Sensing of Weather Impacts on Vegetation in Nonhomogeneous Areas, *International Journal of Remote Sensing*, 11(8): 1405-1419.
- Liu, W. T., and Kogan, F. N., 1996. Monitoring Regional Drought Using the Vegetation Condition Index, *International Journal of Remote Sensing* 17(14): 2761-2782.
- Misra, P. N., and Wheeler, S.G., 1977. Landsat Data From Agricultural Sites - Crop Signature Analysis, *Proceedings of the 11th International Symposium on Remote Sensing of the Environment*, ERIM.
- Misra, P. N., Wheeler, S. G., and Oliver, R. E., 1977. Kauth-Thomas Brightness and Greenness Axes, IBM personal communication, Contract NAS-9-14350, RES 23-46.
- Perry, C. Jr., and Lautenschlager, L. F., 1984. Functional Equivalence of Spectral Vegetation Indices, *Remote Sensing and the Environment*, 15: 111-124.

Environment 14: 169-182.

Qi, J., Chehbouni A., Huete, A. R., Kerr, Y. H., and Sorooshian, S., 1994. A Modified Soil Adjusted Vegetation Index. *Remote Sensing and the Environment*, 48: 119-126.

Richardson, A. J., and Wiegand, C. L., 1977. Distinguishing Vegetation From Soil Background Information, *Photogrammetric Engineering and Remote Sensing*, 43(12): 1541-1552.

Rouse, J. W. Jr., Haas, R., H., Schell, J. A., and Deering, D.W., 1973. Monitoring Vegetation Systems in the Great Plains with ERTS, *Earth Resources Technology Satellite-1 Symposium*, Goddard Space Flight Center, Washington D.C., 309-317.

Rouse, J. W. Jr., Haas, R., H., Deering, D. W., Schell, J. A., and Harlan, J. C., 1974. Monitoring the Vernal Advancement and Retrogradation (Green Wave Effect)of Natural Vegetation. *NASA/GSFC Type III Final Report*, Greenbelt, MD., 371.

Singh, A., and Harrison, A., 1985. Standardized Principal Components, *International Journal of Remote Sensing*, 6(6): 883-896.

Thiam, A.K. 1997. *Geographic Information Systems and Remote Sensing Methods for Assessing and Monitoring Land Degradation in the Sahel: The Case of Southern Mauritania*. Doctoral Dissertation, Clark University, Worcester Massachusetts.

Tripathy, G. K., Ghosh, T. K., and Shah, S. D., 1996. Monitoring of Desertification Process in Karnataka State of India Using Multi-Temporal Remote Sensing and Ancillary Information Using GIS, *International Journal of Remote Sensing*, 17(12): 2243-2257.

Time Series/Change Analysis

This chapter gives a brief overview⁸⁶ of the special procedures available for change and time series analysis in IDRISI.

The techniques for the analysis of change are broken down into three broad categories for this chapter. The first of these contains techniques that are designed for comparisons between pairs of images; the second is made up of techniques that are concerned with the analysis of trends and anomalies across multiple images (i.e., a time series), and the third consists of methods for predictive modeling and assessment of models.

Pairwise Comparisons

With pairwise comparisons we can further break down the techniques according to whether they are suitable for quantitative or qualitative data. Quantitative data has values that indicate an amount or measurement, such as NDVI, rainfall or reflectance. Qualitative data has values that indicate different categories, such as census tract ID's or landuse classes.

Quantitative Data

Image Differencing

With quantitative data, the simplest form of change analysis is *image differencing*. In IDRISI, this can be achieved with the OVERLAY module through a simple subtraction of one image from the other. However, a second stage of analysis is often required since the difference image will typically contain a wide range of values. Both steps are included in the module IMAGEDIFF, which produces several common image difference products: a simple difference image (later - earlier), a percentage change image (later-earlier/earlier), a standardized difference image (z-scores), or a classified standardized difference image (z-scores divided into 6 classes). Mask images that limit the study area may also be specified.

Care must be taken in choosing a threshold to distinguish true change from natural variability in any of these difference images. There are no firm guidelines for this operation. A commonly used value for the threshold is 1 standard deviation (STD) (i.e., all areas within 1 STD are considered non-change areas and those beyond 1 STD in either the positive or negative direction are considered change areas), but this should be used with caution. Higher values may be more appropriate and in some cases natural breaks in a histogram of the simple difference or percentage change images may be more sensible as a basis for choosing the threshold values.

Image Ratioing

While image differencing looks at the absolute difference between images, *image ratioing* looks at the relative difference. Again, this could be achieved with OVERLAY using the ratio option. However, because the resulting scale of relative change is not symmetric about 1 (the no change value), it is recommended that a logarithmic transformation be undertaken before thresholding the image. The module IMAGERATIO offers both a simple ratio and a log ratio result.

Regression Differencing

A third form of differencing is called *regression differencing*. This technique should be used whenever it is suspected that the measuring instrument (e.g., a satellite sensor) has changed its output characteristics between the two dates being compared. Here the earlier image is used as the independent variable and the later image as the dependent variable in a linear

86. For those wishing a more in-depth discussion with tutorial exercises, please refer to Eastman, J., McKendry, J., and Fulk, M., 1994. UNITAR Explorations in GIS Technology, Vol. 1, *Change and Time Series Analysis*, 2nd ed., UNITAR, Geneva, also available from Clark Labs.

regression. The intercept and slope of this regression expresses the offset and gain required to adjust the earlier image to have comparable measurement characteristics to the later. In effect, we create a predicted later image in which the values are what we would expect if there were no change other than the offset and gain caused by the changes in the sensor. The equation is:

$$\text{predicted later image} = (\text{earlier image} * \text{gain}) + \text{offset}$$

With the sensor differences accounted for, the predicted later image and the actual later image may then be analyzed for change. Note that this technique requires that the overall numeric characteristics of the two images be equal except for sensor changes. The technique may not be valid if the two images represent conditions that are overall very different between the two dates.

The module CALIBRATE automates the image adjustment process. The input image (the one to calibrate) is used as the independent variable and the reference image is used as the dependent variable in the regression. The output image is adjusted to the characteristics of the reference image and thus can be used in a standard comparison operation (such as IMAGEDIFF or IMAGERATIO) with any image also based on this reference, including the reference image itself.

Note that CALIBRATE also offers options to adjust an image by entering offset and gain values or by entering mean and standard deviation values.

Change Vector Analysis

Occasionally, one needs to undertake pairwise comparisons on multi-dimensional images. For example, one might wish to undertake a change analysis between two dates of satellite imagery where each is represented by several spectral bands. To do so, *change vector analysis* can be used. With change vector analysis, difference images are created for each of the corresponding bands. These difference images are then squared and added. The square root of the result represents the magnitude of the change vector. All these operations can be carried out with the Image Calculator, or a combination of TRANSFORM and OVERLAY. The resulting image values are in the same units as the input images (e.g., dn).

When only two bands (for each of the two dates) are involved, it is also possible to create a direction image (indicating the direction of change in band space). The module CVA calculates both magnitude and direction images for 2-band image pairs. Figure 19-1 illustrates these calculations. The magnitude image is in the same units as the input bands and is the distance between the Date 1 and Date 2 positions. The direction image is in azimuths measured clockwise from a vertical line extending up from the Date 2 position.

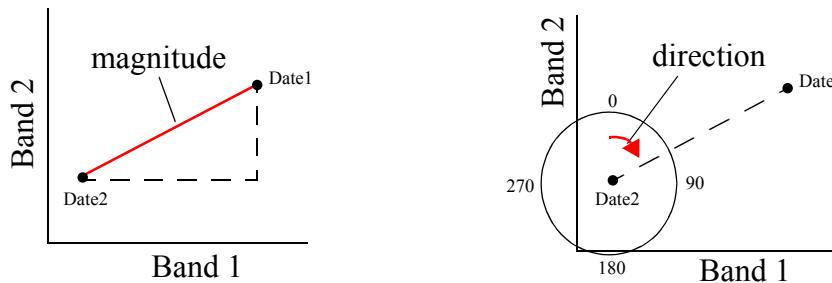


Figure 19-1

Qualitative Data

Crosstabulation / Crossclassification

With qualitative data, CROSSTAB should be used for change analysis between image pairs and there are several types of output that can be useful. The crosstabulation table shows the frequencies with which classes have remained the same

(frequencies along the diagonal) or have changed (off-diagonal frequencies). The Kappa Index of Agreement (KIA) indicates the degree of agreement between the two maps, both in an overall sense and on a per-category basis. Finally, the crossclassification image can readily be reclassified into either a change image or an agreement image. Note that the numeric values of data classes must be identical on both maps for the output from CROSSTAB to be meaningful.

Multiple Image Comparisons

With multiple images, a variety of techniques can be used. For analysis, the most important of the routines offered in IDRISI is Time Series Analysis (TSA).

Time Series Analysis (TSA)

TSA can produce an analysis of up to 256 input images, providing both spatial and temporal outputs. It analyzes the series as a whole based on a Standardized Principal Components Analysis. An ordered set of uncorrelated component images are produced, each expressing underlying themes (trends, shifts, periodicities, etc.) in the data of successively lower magnitude (in terms of the total variability explained in the original image set).

As a prelude to running TSA (and many of the other routines in this section), you will need to create a time series (.ts) file using Collection Editor. This file lists in order the names of the files of the time series.

TSA requires the name of the time series file and, among other things, the number of component images to be produced and the type of output to be used for the temporal data, the *loadings*. Assuming that the series contains a large number of images, you may wish to limit your output to no more than 10-12 components. For many studies, this proves to be sufficient. Also, for most purposes, integer components provide adequate precision while saving valuable disk space.

For temporal output, saving the component loadings as a DIF file provides the greatest flexibility. The output can then be brought into a spreadsheet for the preparation of loadings graphs. For quick analyses, output of IDRISI profiles is convenient.

Analysis of the results of TSA is done by examining the component images (the spatial output) in combination with the component loadings (the temporal output). The first component image describes the pattern that can account for the greatest degree of variation among the images. The loadings indicate the degree to which the original images correlate with this component image. Quite typically, most images will correlate strongly and roughly evenly (i.e., with correlations in the 0.93-0.98 range) with the first component image. Since graphing programs tend to autoscale loadings graphs, your graph of the first component may tend to look overly irregular. We therefore suggest that you combine components 1 and 2 onto the same graph. This will cure the problem for reasons that will become apparent once you try it.

Unless the region analyzed is very small, component 1 will most likely represent the typical or characteristic value over the series. Since each component is uncorrelated with the others, all successive components represent change. Indeed, the process can be thought of as one of successive residuals analysis. High positive values on the image can be thought of as areas that correlate strongly with the temporal pattern in the loadings graph while those with high negative values correlate strongly with the inverse of the graph. (To get a sense of what the inverse is, imagine that you could hold the X axis of the graph at the ends and rotate it until the Y axis is inverted.⁸⁷ This can also be done by looking at the graph through the back of the paper, upside down). We sometimes find it useful to invert the graph in this way and to invert the palette⁸⁸ as well. Areas that correlate strongly with the inverted graph then have the same color as those areas that correlate

87. Inverting the signs associated with the loadings and component values constitutes a permissible rotation in Principal Components Analysis and does not affect the interpretation so long as both elements are rotated in concert.

88. To invert the palette, open the palette in Symbol Workshop. Use the Reverse function, then save the palette under a new name. Then change the palette to be used for the image display to the one just created.

strongly with the non-inverted graph.

Just as strong anomalies on the image can be thought to represent areas that are strongly associated (positively and negatively) with the temporal loadings, time periods with high positive loadings can be thought of as those with spatial patterns very similar to those in the component image, while those with strong negative loadings are associated with the inverse pattern (that you would get by inverting the palette).

Interpretation of the output from TSA requires care and patience. However, the results can be enormously illuminating.⁸⁹ Since each successive component represents a further residual, you will find yourself peering deeper and deeper into the data in a manner that would otherwise be impossible. Here you may find *latent images* of trends that would be almost impossible to detect through direct examination of the data.

A Note About Masking and Component Rotation

This version of TSA offers the option of specifying a mask image to exclude portions of the images from the analysis. Generally, we do not recommend that you do this. Rather, experiments at Clark Labs have shown that it is generally preferable to mask the input images first (e.g., to 0) and include these as data in the analysis (i.e., run the analysis with a mask). If the masking is applied identically to all images (e.g., through the use of a dynagroup in Macro Modeler), it adds no source of variability to the series. Thus its effect will be completely described by the first component. All other components will be free of its effects. However, it does contribute to the intercorrelation between images. The effect is thus equivalent to rotating the axes such that the first component is focused on the commonalities between the images. We have found this generally to be highly desirable, leading to a highly consistent logic to the interpretation of the initial components, with the first component showing the characteristic pattern over time, and the second component showing the main seasonal effects.

Time Series Correlation

TSA is an inductive procedure that attempts to isolate the major components of variation over time. In contexts where you are looking for evidence of a particular temporal phenomenon, however, you may find it more effective to use the CORRELATE module. CORRELATE compares each pixel, over time, with the values of an index in a non-spatial time series. For example, one might compare a time series of vegetation index images with a measure of central Pacific ocean temperature on a monthly basis. The result would be an image where each pixel contains a Pearson Product-Moment correlation expressing the correlation between that vegetation index and sea surface temperature over the series.

Time Profiling

To examine changes in values at specific locations over time, PROFILE may be used. The module requires two inputs—a raster image defining up to 15 sites or classes to monitor, and a time series file listing the images to be analyzed. A variety of statistics can be produced (e.g., mean, minimum, maximum, range, standard deviation). Output is in the form of a graph, although you also have the option to save the profiles as values files.

Image Deviation

As an extension to simple differencing with pairwise comparisons, *image deviation* can be used with time series data. The logic of image deviation is to produce a characteristic image for the series as a whole, from which any of the individual images can be subtracted to examine how it differs from the sequence at large. The most common procedure is to create a mean image, then difference each image from that mean. This can be accomplished with Image Calculator or a combination of OVERLAY and SCALAR. If the input images are all in byte binary form, it is also possible to use the Multi-Criteria Evaluation (MCE) routine to create the mean image. To do so, simply treat each input image as a factor and apply a weight equal to $(1/n)$ where n represents the number of images in the sequence. The resulting difference images must

89. For an example, please refer to Eastman, J.R., and Fulk, M., 1993. Long Sequence Time Series Evaluation using Standardized Principal Components, *Photogrammetric Engineering and Remote Sensing*, 59(8): 1307-1312.

then be thresholded as discussed above in the section on pairwise image differencing.

Change Vector Analysis II

Occasionally one needs to examine the difference between two series. For example, one might have 12 months of image data for one year and 12 months of image data for another. To compare the two years, *change vector analysis* can be used. Use CVA to produce a change magnitude image for each of the monthly pairs, then sum the results using OVERLAY or Image Calculator. Please note that change vector analysis of this type can suffer from significant problems of temporal misregistration. The assumption behind the technique is that human-defined slices of time are environmentally meaningful—however, they rarely are. For example, if the image sets represented vegetation index data for an arid area with a pronounced rainy season, a small difference in the onset of rains between the years would be considered by the technique as substantial change, when there might be little difference in agricultural output and timing of the harvest.

Time Series Correlation

It is a common need in change and time series analysis to identify key temporal patterns occurring through the image set. For example, you may have developed one or more typical drought scenarios that tend to produce particular patterns in monthly rainfall or NDVI data. You might then wish to examine a monthly time series of images to see where that pattern has most closely occurred. The module CORRELATE is designed to calculate the degree to which each pixel location corresponds to a given pattern as recorded in an attribute values file. The measure used is the Pearson Product-Moment coefficient of correlation.

Predictive Change Modeling

In some cases, knowing the changes that have occurred in the past may help predict future changes. A suite of modules in IDRISI has been developed to provide the basic tools for predictive landcover change modeling. These tools are primarily based on Markov Chain Analysis and Cellular Automata.

Markov Chain Analysis

A Markovian process is one in which the state of a system at time 2 can be predicted by the state of the system at time 1 given a matrix of transition probabilities from each cover class to every other cover class. The MARKOV module can be used to create such a transition probability matrix. As input, it takes two landcover maps. It then produces the following outputs:

- A transition probability matrix. This is automatically displayed, as well as saved. Transition probabilities express the likelihood that a pixel of a given class will change to any other class (or stay the same) in the next time period.
- A transition areas matrix. This expresses the total area (in cells) expected to change in the next time period.
- A set of conditional probability images—one for each landcover class. These maps express the probability that each pixel will belong to the designated class in the next time period. They are called conditional probability maps since this probability is conditional on their current state.

STCHOICE is a stochastic choice decision module. Given the set of conditional probability images produced by MARKOV, STCHOICE can be used to produce any number of potential realizations of the projected changes embodied in the conditional probability maps. If you try this, however, you will find the results to be disappointing. The output from MARKOV has only very limited spatial knowledge. To improve the spatial sense of these conditional probability images (or in fact, any statistic), use DISAGGREGATE. Given an image of the likely internal spatial pattern of an areal statistic, DISAGGREGATE redistributes the statistic such that it follows the suggested pattern, but maintains the overall area total. NORMALIZE can then be used to ensure that probabilities add to 1.0 at each pixel (this may need to be

applied iteratively with DISAGGREGATE).

Cellular Automata

One of the basic spatial elements that underlies the dynamics of many change events is proximity: areas will have a higher tendency to change to a class when they are near existing areas of the same class (i.e., an expansion phenomenon). These can be very effectively modeled using cellular automata. A cellular automaton is a cellular entity that independently varies its state based on its previous state and that of its immediate neighbors according to a specific rule. Clearly there is a similarity here to a Markovian process. The only difference is application of a transition rule that depends not only upon the previous state, but also upon the state of the local neighborhood.

Many cellular automata transition rules can be implemented through a combination of FILTER and RECLASS. Take, for example, the case of Conway's *Game of Life*. In this hypothetical illustration, the automata live or die according to the following criteria:

- An empty cell becomes alive if there are three living automata in the 3x3 neighborhood (known as the Moore neighborhood) surrounding the cell.
- The cell will stay alive so long as there are 2 or 3 living neighbors. Fewer than that, it dies from loneliness; more than that it does from competition for resources.

This can be implemented using the following kernel with the FILTER module:

1	1	1
1	10	1
1	1	1

followed by the following RECLASS rule:

0 - 2	=	0
3 - 4	=	1
4 - 11	=	0
12 - 13	=	1
14 - 18	=	0

The critical element of this rule is the use of the 10 multiplier in the central cell. As a result of the filter step, you know that the central cell is occupied if the result is 10 or greater. The CELLATOM module can be used to implement this kind of Cellular Automaton rule. However, a cellular automaton procedure very specific to the context of predictive landcover change modeling is implemented with the CA_MARKOV module.

CA_MARKOV takes as input the name of the landcover map from which changes should be projected, the transition areas file produced by MARKOV from analysis of that image and an earlier one, and a collection (.rgf) of suitability images that express the suitability of a pixel for each of the landcover types under consideration. It then begins an iterative process of reallocating land cover until it meets the area totals predicted by the MARKOV module. The logic it uses is this:

- The total number of iterations is based on the number of time steps set by the user. For example, if the projection is for 10 years into the future, the user might choose to complete the model in 10 steps.

- Within each iteration, every landcover class will typically lose some of its land to one or more of the other classes (and it may also gain land from others). Thus within the consideration of each host within each iteration, claimant classes select land from the host based on the suitability map for the claimant class. Since there will commonly be competition for specific land parcels, this process of land allocation is undertaken using a multi-objective allocation procedure (the MOLA module).

- The Cellular Automaton component arises in part from the iterative process of land allocation, and in part from a filtering stage with each iteration that reduces the suitability of land away from existing areas of that type. By default, the module uses a 5x5 mean filter to achieve this contiguity constraint. By filtering a Boolean mask of the class being considered, the mean filter yields a value of 1 when it is entirely within the existing class and 0 when it is entirely outside it. However, when it crosses the boundary, it will yield values that quickly transition from 1 to 0. This result is then multiplied by the suitability image for that class, thereby progressively downweighting the suitabilities as one moves away from existing instances of that class. Note that it is possible to apply a different filter by specifying an alternative filter file (.fil). Also note that class masks are defined at each step to incorporate new areas of growth.

The net result of this iterative process is that landcover changes develop as a growth process in areas of high suitability proximate to existing areas. CA_MARKOV is computationally intensive—a typical run might involve several thousand GIS operations. Thus you should start the run when you can leave your computer for 15-30 minutes.

Model Validation

An important stage in the development of any predictive change model is validation. Typically, one gauges one's understanding of the process, and the power of the model, by using it to predict some period of time when the landcover conditions are known. This is then used as a test for validation. IDRISI supplies a pair of modules to assist in the validation process.

The first is called VALIDATE, and provides a comparative analysis on the basis of the Kappa Index of Agreement. Kappa is essentially a statement of proportional accuracy, adjusted for chance agreement. However, unlike the traditional Kappa statistic, VALIDATE breaks the validation down into several components, each with a special form of Kappa or associated statistic (based on the work of Pontius (2000)):

- Kappa for no information = K_{no}
- Kappa for location = $K_{location}$
- Kappa for quantity = $K_{quantity}$
- Kappa standard = $K_{standard}$
- Value of Perfect Information of Location = VPI_L
- Value of Perfect Information of Quantity = VPI_Q

With such a breakdown, for example, it is possible to assess the success with which one is able to specify the location of change versus the quantity of change.

The other validation procedure is the ROC (Relative Operating Characteristic). It is used to compare any statement about the probability of an occurrence against a Boolean map which shows the actual occurrences. It can be useful, for example, in validating modifications to the conditional probability maps output from MARKOV. Note that LOGISTICREG incorporates ROC directly in its output.

References:

Pontius Jr., R.G., 2000. Quantification error versus location error in comparison of categorical maps. Photogrammetric Engineering and Remote Sensing. 66(8) pp. 1011-1016.

Anisotropic Cost Analysis

Cost surface modeling is now a familiar feature of many raster geographic information systems. In developing a cost surface, one accounts for the cost of moving through space, where costs are a function of both the standard (or *base*) costs associated with movement, and also of frictions and forces that impede or facilitate that movement.

Isotropic Costs

Isotropic cost surface modeling is accomplished in IDRISI with the COST module. Given input images of a set of features from which cost distances should be calculated and the frictions that affect movement, COST outputs a cost surface that expresses costs of movement in terms of distance equivalents. Thus, for example, if a cell contains a value of 100, it simply expresses that the cost of moving from the nearest starting feature (target) to that point is the equivalent of moving over 100 cells at the base cost. It could equally arise from traveling over 100 cells with a relative friction (i.e., relative to the friction associated with the base cost) of 1, or 50 cells with frictions of 2, or 1 cell with a relative friction of 100.

Anisotropic Costs

With the COST module, frictions have identical effect in any direction. It doesn't matter how you move through a cell—its friction will be the same. We can call such a friction *isotropic* since it is equal in all directions. However, it is not very difficult to imagine *anisotropic* frictions—frictional elements that have different effects in different directions. Take, for example, the case of slopes. If we imagine the costs of walking (perhaps in calories per hour at normal walking speed), then slopes will affect that cost differently in different directions. Traveling upslope will cause that friction to act full-force; traveling perpendicularly across the slope will have no effect at all; and traveling downslope will act as a force that reduces the cost. Traditional cost analysis cannot accommodate such an effect.

Anisotropic Cost Modules in IDRISI

In IDRISI, four modules are supplied for the modeling of anisotropic costs. Anisotropic cost analysis is still a very new area of analysis, and we therefore encourage users to send information, in writing, on their applications and experiences using these modules.

At the core of the set are two different modules for the analysis of anisotropic costs, VARCOST and DISPERSE, and two support modules for the modeling of forces and frictions that affect those costs, RESULTANT and DECOMP.

VARCOST models the effects of anisotropic frictions on the movement of phenomena that have their own motive force. The example just given of walking in the presence of slopes is an excellent example, and one that is perfectly modeled by VARCOST. DISPERSE, on the other hand, models the movement of phenomena that have no motive force of their own, but which are acted upon by anisotropic forces to disperse them over time. A good example of this would be a point-source pollution problem such as a chemical spill on land. Upon absorption into the soil, the contaminant would move preferentially with ground water under the force of gravity according to the hydraulic gradient. The resulting pattern of movement would look plume-like because of the decreasing probability of movement as one moved in a direction away from the maximum gradient (slope). DISPERSE and VARCOST are thus quite similar in concept, except in the nature of how forces and frictions change in response to changes in the direction of movement. This we call the anisotropy.

pic function, as will be discussed below. However, to understand such functions, it is useful to review the distinction between forces and frictions in the modeling of costs.

Forces and Frictions

In cost modeling, forces and frictions are not inherently different. In all of the cost modeling procedures—COST, VARCOST and DISPERSE—frictions are expressed as relative frictions using the base cost as a reference. Thus, for example, if it takes 350 calories to walk along flat ground, and 700 calories to walk across more rugged terrain at equal speed, we would indicate that rugged terrain has a friction of 2. However, if we were to walk down a slope such that our energy expended was only 175 calories, then we would express that as a friction of 0.5. But what are frictions less than 1? They are, in fact, forces. To retain consistency, all relative frictions in IDRISI are expressed as values greater than 1 and relative forces are expressed as values less than 1. Thus, if we were concerned with wind forces and we had a base force of 10 km/hour, a wind of 30 km/hour would be specified as a relative force of 0.33.

With anisotropic cost modeling, a single image cannot describe the nature of forces and frictions acting differently in different directions. Rather, a pair of images is required—one describing the magnitude of forces and frictions, expressed as relative quantities exactly as indicated above, and the other describing the direction of those forces and frictions, expressed as azimuths.⁹⁰ These magnitude/direction image pairs thus describe a field of force/friction vectors which, along with the anisotropic function discussed below, can be used to determine the force or friction in any direction at any point. The term *force/friction image pair* refers to a magnitude image and its corresponding direction image for either forces (used with DISPERSE) or frictions (used with VARCOST).

It is important to understand the nature of the direction images required for both VARCOST and DISPERSE. With VARCOST, the *friction* direction image must represent the direction of movement that would incur the greatest cost to movement. For example, if you are modeling the movement of a person walking across a landscape and the frictions encountered are due to slopes (going uphill is difficult, going downhill is easy), then the values in the friction direction image should be azimuths from north that point uphill.

With DISPERSE, the *force* direction image must represent the direction in which the force acts most strongly. For example, if you are modeling the dispersion of a liquid spill over a landscape (flowing easily downhill, flowing with great difficulty uphill), then the values in the force direction image should be azimuths from north that point downhill.

In the use of VARCOST and DISPERSE, a single anisotropic force/friction vector image pair is specified. Since analyses may involve a number of different forces acting simultaneously, a pair of modules has been supplied to allow the combination of forces or frictions. The first of these is RESULTANT. RESULTANT takes the information from two force/friction image pairs to produce a new force/friction image pair expressing the resultant vector produced by their combined action. Thus, RESULTANT can be used to successively combine forces and frictions to produce a single magnitude/direction image pair to be used as input to VARCOST or DISPERSE.

The second module that can be used to manipulate force/friction image pairs is DECOMP. DECOMP can decompose a force/friction image pair into its X and Y component images (i.e., the force/friction in X and the force/friction in Y). It can also recompose X and Y force/friction components into magnitude and direction image pairs. Thus DECOMP could be used to duplicate the action of RESULTANT.⁹¹ However, a quite different and important use of DECOMP is with the interpolation of force/friction vectors. If one takes the example of winds, it is not possible to interpolate the data at point locations to produce an image, since routines such as TREND and INTERPOL cannot tell that the difference between 355 and 0 degrees is the same as between 0 and 5. However, if a raster image pair of the point force/friction data

90. Azimuths express directions in degrees, clockwise from north. In IDRISI, it is also permissible to express an azimuth with the value of -1 to indicate that no direction is defined.

91. To undertake a process similar to RESULTANT, DECOMP is used to decompose all force/friction image pairs acting upon an area into their X and Y components. These X and Y component images are then added to yield a resulting X and Y pair. The recombination option of DECOMP is then used with these to produce a resultant magnitude/direction image pair.

is constructed and then decomposed into X and Y components (using DECOMP), these component images can be interpolated (e.g., with TREND) and then recomposed into a force/friction pair using DECOMP.

Anisotropic Functions

With force/friction image pairs, one has an indication of both the magnitude and direction with which forces and frictions act. However, what is the interpretation of direction? If a force is said to act at 45° (northeast), does this mean it acts fully at 45° and not at all at 44° ? The answer to this is not easily determined and it ultimately depends upon the application. If one takes the earlier example of walking against slopes of varying degrees, the force/friction image describes only the direction and magnitude of the steepest descending slope. If one faced directly into the slope one would feel the full force of the friction (i.e., effective friction = stated friction). Facing directly away from the slope (i.e., pointing down-slope), the friction would be transformed into a force to the fullest possible extent (i.e., effective friction = $1/($ stated friction)). Between the two, intermediate values would occur. Moving progressively in a direction farther away from the maximum friction, the friction would progressively decrease until one reached 90° . At 90° , the effect of the slope would be neutralized (effective friction = 1). Then as one moves past 90° towards the opposite direction, frictions would become forces progressively increasing to the extreme at 180° .

This variation in the effective friction/force as a function of direction is here called the anisotropic function. With VARCOST, the following default function is used:

$$\text{effective_friction} = \text{stated_friction}^f$$

where $f = \cos^k \alpha$

and $k = \text{a user-defined coefficient}$

and $\alpha = \text{difference angle.}$

The difference angle in this formula measures the angle between the direction being considered and the direction *from which* frictions are acting (or equivalently, the direction *to which* forces are acting). Figure 20-1 indicates the nature of this function for various exponents (k) for difference angles from 0 to 90° .

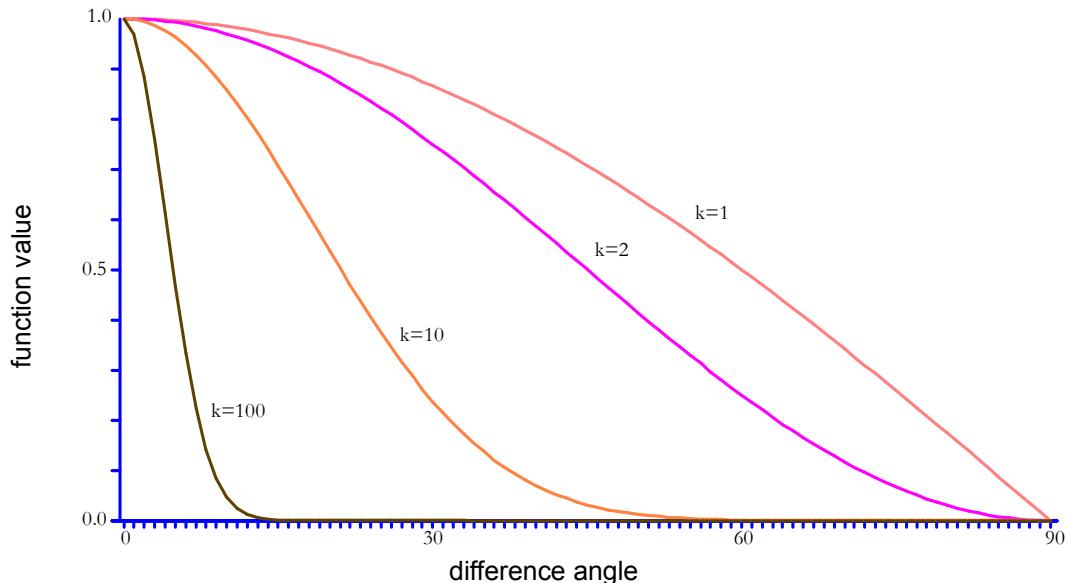


Figure 20-1

You will note in Figure 20-1 that the exponent k makes the function increasingly direction-specific. At its limit, an extremely high exponent would have the effect of causing the friction to act fully at 0° , to become a fully acting force at

180° , and to be neutralized at all other angles. The default anisotropic function returns negative values for all difference angles from 90° to 270° regardless of the exponent used (i.e., negative cosine values, when raised to odd or even exponents, return negative values for the function). Hence, these angles always yield effective friction values that are less than one (i.e., act as forces).

We have not presumed that this function will be appropriate in all circumstances. As a result, we have provided the option of entering a user-defined function. The procedure for doing so is quite simple—VARCOST has the ability to read a data file of function values for difference angles from $0-360^\circ$ in increments of 0.05° . The format for this file is indicated in the VARCOST module description in the on-line Help System. The important thing to remember, however, is that with VARCOST, the values of that function represent an exponent as follows:

$$\text{effective_friction} = \text{stated_friction}^f$$

where f = a user-defined function.

With DISPERSE, the same general logic applies to its operation except that the anisotropic function is different:

$$\text{effective_friction} = \text{stated_friction} * f$$

where $f = 1/\cos^k\alpha$

and k = a user-defined coefficient

and α = difference angle.

The effect of this function is to modify frictions such that they have full effect at an angle of 0° with progressive increases in friction until they reach infinity at 90° . The function is designed so that effective frictions remain at infinity for all difference angles greater than 90° . Figure 20-2 shows the values returned by the default functions of f , illustrating this difference between the functions of VARCOST and DISPERSE.

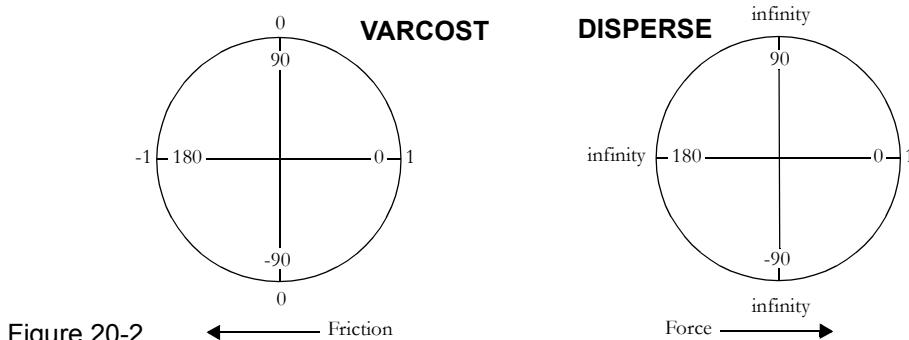


Figure 20-2

Like VARCOST, DISPERSE also allows the entry of a user-defined function. The procedure is identical, allowing for the reading of a data file containing function values for difference angles from $0-360^\circ$ in increments of 0.05° . The format for this file is indicated in the DISPERSE module description in the on-line Help System. Unlike VARCOST, however, the values of that function represent a multiplier (rather than an exponent) as follows:

$$\text{effective_friction} = \text{stated_friction} * f$$

where f = a user-defined function.

Applications of VARCOST and DISPERSE

VARCOST and DISPERSE have proven useful in a variety of circumstances. VARCOST is a direct extension of the logic of the COST module (i.e., as a means of gauging the effects of frictions and forces on the costs of movement through space, with the special additional capability to moderate frictional effects with varying directions of movement through cells). One might use VARCOST, for example, along with ALLOCATE, to assign villages to rural health centers where the costs of travel on foot are accommodated given landuse types (an isotropic friction) and slopes (an anisotropic friction).

DISPERSE is useful in cases where the phenomenon under study has no motive force of its own, but moves due to forces that act upon it. Potential applications might include point source pollution studies, forest and rangeland fire modeling, and possibly oil spill monitoring and projection.

We encourage users to share with us their experiences using these modules and how they might be changed or augmented to facilitate such studies. We would also welcome the submission of user-defined anisotropic functions that meet the needs of special applications and might be useful to a broader user group.

Surface Interpolation

Introduction

In GIS, we often want to combine information from several layers in analyses. If we only know the values of a selection of points and these sample points do not coincide between the layers, then such analyses would be impossible. Even if the sample points do coincide, we often want to describe a process for *all* the locations within a study area, not just for selected points. In addition, we need full surfaces because many processes modeled in GIS act continuously over a surface, with the value at one location being dependent upon neighboring values.

Any GIS layer, whether raster or vector, that describes *all* locations in a study area might be called a surface. However, in surface analysis, we are particularly interested in those surfaces where the attributes are quantitative and vary continuously over space. A raster Digital Elevation Model (DEM), for instance, is such a surface. Other example surfaces might describe NDVI, population density, or temperature. In these types of surfaces, each pixel may have a different value than its neighbors.

A landcover map, however, would not be considered a surface by this definition. The values are qualitative, and they also do not vary continuously over the map. Another example of an image that does not fit this particular surface definition would be a population image where the population values are assigned uniformly to census units. In this case, the data are quantitative, yet they do not vary continuously over space. Indeed, change in values is present only at the borders of the census units.

No GIS surface layer can match reality at every scale. Thus the term *model* is often applied to surface images. The use of this term indicates a distinction between the surface as represented digitally and the actual surface it describes. It also indicates that different models may exist for the same phenomenon. The choice of which model to use depends upon many things, including the application, accuracy requirements, and availability of data.

It is normally impossible to measure the value of an attribute for every pixel in an image. (An exception is a satellite image, which measures average reflectance for every pixel.) More often, one needs to fill in the gaps between sample data points to create a full surface. This process is called interpolation. IDRISI offers several options for interpolation which are discussed in this chapter. Further technical information about these modules may be found in the on-line Help System.

Surface Interpolation

The choice of interpolation technique depends on what type of surface model you hope to produce and what data are available. In this section, the techniques available in IDRISI are organized according to input sample data type—points or lines. A description of the algorithm used and the general characteristics of the techniques are given. For a more theoretical treatment of the characteristics of surface models produced by particular interpolation techniques, consult the references provided at the end of this chapter.

Interpolation techniques may be described as global or local. A global interpolator derives the surface model by considering all the data points at once. The resulting surface gives a "best fit" for the entire sample data set, but may provide a very poor fit in particular locations. A local interpolator, on the other hand, calculates new values for unknown pixels by using the values of known pixels that are nearby. Interpolators may define "nearby" in various ways. Many allow the user to determine how large an area or how many of the nearest sample data points should be considered in deriving interpolated values.

Interpolation techniques are also classified as exact or inexact. An exact interpolation technique always retains the original

values of the sample data points in the resulting surface, while an inexact interpolator may assign new values to known data points.

Interpolation From Point Data

Trend Surface Analysis

Trend surfaces are typically used to determine whether spatial trends exist in a data set, rather than to create a surface model to be used in further analyses. Trend surfaces may also be used to describe and remove broad trends from data sets so more local influences may be better understood. Because the resulting surface is an ideal mathematical model, it is very smooth and is free from local detail.

In IDRISI, the module TREND is used to produce a trend surface image from sample data points. TREND is a global interpolator since it calculates a surface that gives the best fit, overall, to the entire set of known data points. TREND is also an inexact interpolator. The values at known data points may be modified to correspond to the best fit surface for the entire data set.

TREND fits one of three mathematically-defined ideal surface models, linear, quadratic, or cubic, to the input point data set. To visualize how TREND works, we will use an example of temperature data at several weather stations. The linear surface model is flat (i.e., a plane). Imagine the temperature data as points floating above a table top. The height of each point above the table top depends on its temperature. Now imagine a flat piece of paper positioned above the table. Without bending it at all, one adjusts the tilt and height of the paper in such a way that the sum of the distances between it and every point are minimized. Some points would fall above the plane of the paper and some below. Indeed, it is possible that no points would actually fall on the paper itself. However, the overall separation between the model (the plane) and the sample data points is minimized. Every pixel in the study area could then be assigned the temperature that corresponds to the height of the paper at that pixel location.

One could use the same example to visualize the quadratic and cubic trend surface models. However, in these cases, you would be allowed to bend the paper (but not crease it). The quadratic surface allows for broad bends in the paper while the cubic allows even more complex bending.

TREND operates much like this analogy except a polynomial formula describing the ideal surface model replaces the paper. This formula is used to derive values for all pixels in the image. In addition to the interpolated surface produced, TREND reports (as a percentage) how well the chosen model fits the input points. TREND also reports the F-ratio and degrees of freedom, which may be used to test if the modeled trend is significantly different from zero (i.e., no trend at all).

Thiessen or Voronoi Tessellation

The term *tessellation* means to break an area into pieces or tiles. With a Thiessen tessellation, the study area is divided into regions around the sample data points such that every pixel in the study area is assigned to (and takes on the value of) the data point to which it is closest.

Because it produces a tiled rather than a continuous surface, this interpolation technique is seldom used to produce a surface model. More commonly it is used to identify the *zones of influence* for a set of data points.

Suppose a set of new health centers were proposed for a rural area and its inhabitants needed to be assigned to their *closest* facility. If Euclidean distance was used as the definition of *closest*, then THIESSEN would provide the desired result. Zones of influence that are based on more complex variables than Euclidean distance may also be defined in IDRISI using the COST and ALLOCATE modules in sequence. In the same example, if shortest travel time rather than shortest euclidean distance defined *closest*, then COST would be used to develop a travel-time surface (incorporating information about road types, paths, etc.) and ALLOCATE would be used to assign each pixel to its nearest facility in terms of shortest travel time.

Distance-Weighted Average

The distance-weighted average preserves sample data values and is therefore an exact interpolation technique. In IDRISI, it is available in the module INTERPOL.

The user may choose to use this technique either as a global or a local interpolator. In the global case, all sample data points are used in calculating all the new interpolated values. In the local case, only the 4-8 sample points that are nearest to the pixel to be interpolated are used in the calculation. The local option is generally recommended, unless data points are very uniformly distributed and the user wants a smoother result.

With the local option, a circle defined by a search radius is drawn around each pixel to be interpolated. The search radius is set to yield, on average, 6 control points within the circle. This is calculated by dividing the total study area by the number of points and determining a radius that would enclose, on average, 6 points. This calculation assumes an even distribution of points, however, so some flexibility is built in. If less than 4 control points are found in the calculated search area, then the radius is expanded until at least 4 points are found. On the other hand, if more than 8 control points are found in the calculated search area, then the radius is decreased until at most 8 control points are found. At least 4 points must be available to interpolate any new value.

With either the global or local implementation, the user can define how the influence of a known point varies with distance to the unknown point. The idea is that the attribute of an interpolated pixel should be most similar to that of its closest known data point, a bit less similar to that of its next closest known data point, and so on. Most commonly, the function used is the inverse square of distance ($1/d^2$, where d is distance).

For every pixel to be interpolated, the distance to every sample point to be used is determined and the inverse square of the distance is computed. Each sample point attribute is multiplied by its respective inverse square distance term and all these values are summed. This sum is then divided by the sum of the inverse square distance terms to produce the interpolated value.

The user may choose to use an exponent other than 2 in the function. Using an exponent greater than 2 causes the influence of the closest sample data points to have relatively more weight in deriving the new attribute. Using an exponent of 1 would cause the data points to have more equal influence on the new attribute value.

The distance-weighted average will produce a smooth surface in which the minimum and maximum values occur at sample data points. In areas far from data points, the surface will tend toward the local average value, where *local* is determined by the search radius. The distribution of known data points greatly influences the utility of this interpolation technique. It works best when sample data are many and are fairly evenly distributed.

Potential Model

INTERPOL also offers a second technique called a potential model. It is similar in operation to the distance-weighted average. The difference is in the function that is employed. The calculation is the same as that described above except that the sum of weighted attribute values is not divided by the sum of weights. This causes the values at sample points to often be higher than the original value, especially when sample points are close together. The method is therefore an inexact interpolator. The surface appears to have spikes at sample points and tends to approach zero away from sample points.

This type of interpolation method is based on the *gravity model* concept and was developed to model potential interaction between masses measured at sample points. For example, the amount of interaction (e.g., in terms of commerce) between the people of two villages is related to the number of people in each village and how close these villages are to each other. More people who are closer together produce a greater total interaction. The interaction at a location far from any village would tend to be zero. The potential model method is applied for different purposes than the other methods discussed in this chapter. It would not be used to develop a surface model from elevation data, for example.

Triangulated Irregular Networks

A Triangulated Irregular Network, or TIN, is a vector data structure. The sample data points become the vertices of a set

of triangular facets that completely cover the study area. In IDRISI, the TIN is generated and then used to create a continuous raster surface model. The chapter **Triangulated Irregular Networks and Surface Generation** is devoted to this set of procedures.

Kriging and Simulation

Continuous surfaces can also be derived from point data using geostatistical techniques. Various kriging options are offered in IDRISI through three interfaces to the Gstat⁹² software package: Spatial Dependence Modeler, Model Fitting, and Kriging and Simulation. Like the techniques offered in INTERPOL, kriging methods may be used either as global or local interpolators. However, the local implementation is most often used. Kriging preserves sample data values and is therefore an exact interpolator. Simulation does not preserve sample data values, making it an inexact interpolator.

The main difference between kriging methods and a simple distance-weighted average is that they allow the user great flexibility in defining the model to be used in the interpolation for a particular data set. These customized models are better able to account for changes in spatial dependence across the study area. Spatial dependence is simply the idea that points that are closer together have more similar values than points that are further apart. Kriging recognizes that this tendency to be similar to nearby points is not restricted to a Euclidean distance relationship and may exhibit many different patterns.

The kriging procedure produces, in addition to the interpolated surface, a second image of variance. The variance image provides, for each pixel, information about how well the interpolated value fits the overall model that was defined by the user. The variance image may thereby be used as a diagnostic tool to refine the model. The goal is to develop a model with an even distribution of variance that is as close as possible to zero.

Kriging produces a smooth surface. Simulation, on the other hand, incorporates per-pixel variability into the interpolation and thereby produces a rough surface. Typically hundreds of such surfaces are generated and summarized for use in process modeling.

The geostatistical tools provided through IDRISI's interfaces to Gstat are discussed in greater detail in the chapter **Geostatistics**.

Interpolation From Isoline Data

Sometimes surfaces are created from isoline data. An isoline is a line of equal value. Elevation contours are one example of isolines. Isolines are rarely field measurements; they are more likely the result of digitizing paper maps. One must be aware that the methods involved in creating the isolines may have already included some sort of interpolation. Subsequent interpolation between isolines adds other types of error.

Linear Interpolation From Isolines

A linear interpolation between isolines is available in IDRISI through the INTERCON module. The isolines must first be rasterized, with the attributes of the pixels representing isolines equal to the isoline value. It is also possible to add points of known value prior to interpolation. It is perhaps more useful, however, to add in lines that define ridges, hill crests or other such break features that are not described by the original isoline data set.

In the interpolation, four lines are drawn through a pixel to be interpolated, as shown in Figure 21-1. The lines are extended until they intersect with a pixel of known value in each direction. The slope along each of the four lines is calculated by using the attributes of the intersected pixels and their X,Y coordinates. (Slope is simply the change in attribute from one end of the line to the other, divided by the length of the line.) The line with the greatest slope is chosen and is used to interpolate the unknown pixel

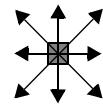


Figure 21-1

92. Gstat, © Edzer Pebesma, is licensed freeware available from GNU. See the on-line Help System for more details.

value.⁹³ The value at the location of the pixel to be interpolated is calculated based on the attribute values of the intersected pixels, the slope of the line, and the X,Y position of the pixel to be interpolated. This process is carried out for all unknown pixels.

Choice of resolution when the isolines are rasterized is crucial. If the resolution is too coarse, more than one line may rasterize into a single pixel. In this case, only the latter value is retained and a poor interpolation will result. It is recommended that one set the initial resolution to be equal or less than the distance between the closest isolines. A coarser resolution surface can be generated after the initial interpolation using RESAMPLE or CONTRACT. Note that one can easily produce a surface with more apparent detail than is actually present in the isoline data. Del Barrio et al (1992) present a quantitative method for determining a resolution that captures the optimum information level achievable given the characteristics of the input isoline data.

Linear interpolation from isolines may produce some obvious and undesirable artifacts in the resulting surface. A histogram of a surface produced by this interpolation technique tends to show a "scalloped" shape, with histogram peaks at the input isoline values. In addition, star-shaped artifacts may be present, particularly at peaks in the surface. These characteristics can be mitigated to some degree (but not removed) by applying a mean filter (with the FILTER module). Finally, hill tops and valley bottoms will be flat with the value of the enclosing contour. In many cases, if isoline data are available, the constrained and optimized TIN method described below will produce a better surface model.

INTERCON is an exact interpolator, since isolines retain their values. It could also be termed a local interpolator, though the isolines used to interpolate any particular pixel may be quite distant from that pixel.

Constrained Triangulated Irregular Networks

As discussed above, triangulated irregular networks may be generated from point data. In addition, the IDRISI TIN module allows for input of isoline data for TIN creation. In doing so, the TIN can be constrained so no triangular facet edge crosses an isoline. This forces the triangulation to preserve the character of the surface as defined by the isolines. A TIN developed from isolines can also be optimized to better model features such as hill tops and valley bottoms. Once the TIN is developed, it may be used to generate a raster surface model with the module TINSURF.

All the steps involved in this process are detailed in the chapter **Triangulated Irregular Networks and Surface Generation**.

Choosing a Surface Model

No single surface generation method is better than others in the abstract. The relative merit of any method depends upon the characteristics of the input sample data and the context in which the surface model will be used. The precision of sample point measurements, as well as the frequency and distribution of sample points relative to the needed scale of variation, influence the choice of interpolation technique to apply to those data. In addition, the scale of the processes to be modeled is key in guiding the creation of an interpolated surface model. Surface shape (e.g., convexity, concavity) and level of local variation are often key aspects of process models, where the value or events in one pixel influence those of the neighboring pixels. It is not unusual to develop several surface models and use each in turn to assess the sensitivity of an analysis to the type of surface generation techniques used.

References / Further Reading

Blaszcynski, J., 1997. Landform Characterization With Geographic Information Systems, *Photogrammetric Engineering and Remote Sensing*, 63(2): 183-191.

93. The line of greatest slope is used to avoid flat lines that result when a line intersects the same isoline on both ends. This is quite common with topographic maps and would lead to an abundance of flat areas in the interpolated surface.

Burrough, P., and McDonnell, R., 1998. *Principles of Geographical Information Systems*, 98-161, Oxford University Press, London.

del Barrio, G., Bernardo, A., and Diez, C., 1992. The Choice of Cell Size in Digital Terrain Models: An Objective Method, Conference on Methods of Hydrologic Comparison, Oxford, UK, September 29-October 20.

Desmet, J., 1997. Effects of Interpolation Errors on the Analysis of DEMs, *Earth Surface Processes and Landforms*, 22: 563-580.

Lam, N., 1983. Spatial Interpolation Methods: A Review, *The American Cartographer*, 10(2): 129-149.

Triangulated Irregular Networks and Surface Generation

Introduction

Triangulated Irregular Networks (TINs) are the most commonly-used structure for modeling continuous surfaces using a vector data model. They are also important to raster systems because they may be used to generate raster surface models, such as DEMs. With triangulation, data points with known attribute values (e.g., elevation) are used as the vertices (i.e., corner points) of a generated set of triangles. The result is a triangular tessellation of the entire area that falls within the outer boundary of the data points (known as the *convex hull*). Figure 22-1 illustrates a triangulation from a set of data points.

There are many different methods of triangulation. The Delaunay triangulation process is most commonly used in TIN modeling and is that which is used by IDRISI. A Delaunay triangulation is defined by three criteria: 1) a circle passing through the three points of any triangle (i.e., its circumcircle) does not contain any other data point in its interior, 2) no triangles overlap, and 3) there are no gaps in the triangulated surface. Figure 22-2 shows examples of Delaunay and non-Delaunay triangulations.

A natural result of the Delaunay triangulation process is that the minimum angle in any triangle is maximized. This property is used by the IDRISI algorithm in constructing the TIN. The number of triangles (N_t) that make up a Delaunay TIN is $N_t=2(N-1)-Nb$, and the number of edges (N_e) is $N_e=3(N-1)-Nb$, where N is the number of data points, and Nb is the number of points in the convex hull.

IDRISI includes options for using either true point data or vertex points extracted from isolines⁹⁴ as input for TIN generation. The TIN module also offers options to use non-constrained or constrained triangulation, to optimize the TIN by removing “tunnel” and “bridge” edges, and to generate a raster surface from the TIN by calling the module TINSURF. Modules are also available for preparing TIN input data. These are all discussed in detail below.

In IDRISI, the TIN file structure consists of a vector line file (containing the triangle edges) and an associated ASCII TIN file (containing information indicating which points make up each triangle). File structure details may be found in the on-line Help System.

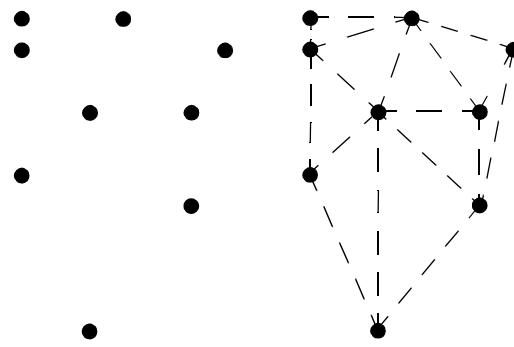


Figure 22-1 A set of data points (left) and a triangulation of those data points (right).

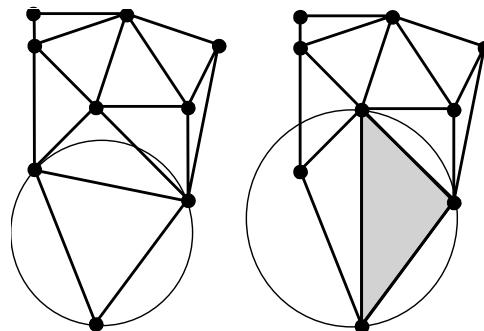


Figure 22-2 Delaunay triangulation (left) and non-Delaunay triangulation (right). The shaded triangle doesn't meet the empty circumcircle criterion.

94. In this chapter, the term isoline refers to any line representing a constant attribute value. Elevation contours are one example of isolines.

Preparing TIN Input Data

Points

Normally there will be little data preparation necessary when point data is used to create a TIN. In some cases it may be desirable to reduce the number of points to be used in the triangulation. For example, if the number and density of the points exceeds the required accuracy of the TIN, the user may choose to remove points since fewer points will lead to faster processing of the TIN. The module GENERALIZATION offers this point-thinning capability.

If point data are used as input to TIN generation, only the non-constrained triangulation option, described below, is available. If isoline data are used as input to TIN generation, both the non-constrained and constrained options are available and a better TIN result can be expected. If a raster surface is the desired final output of input point data, the INTERPOL module and the IDRISI interfaces to Gstat offer alternatives to TIN/TINSURF. (See the chapters **Surface Analysis** and **Geostatistics**.)

Lines

When an isoline file is used as input to TIN, only the vertices⁹⁵ that make up the lines are used in the triangulation. It may be useful to examine the density of the vertices in the isoline file prior to generating the TIN. The module GENERALIZATION may be used to extract the vertices of a line file to a vector point file for visualization.

It may be desirable to add points along the lines if points are so far apart they create long straight-line segments that result in large TIN facets. Point thinning along lines is also sometimes desirable, particularly with isoline data that was digitized in stream mode. In this case, the number of points in the lines may be much greater than that necessary for the desired resolution of the TIN, and thus will only serve to slow down the TIN generation process. The module TINPREP performs along-line point addition or thinning. Other line generalization options are also available in the GENERALIZATION module.

If line data are used to create a TIN, both the non-constrained and the constrained triangulation options are available. The differences between these options are described below. If a raster surface is the desired final output of input isoline data, the module INTERCON offers an alternative to TIN/TINSURF. However, the latter normally produces a superior result.

Command Summary

Following is a list and brief description of the modules mentioned in this section.

GENERALIZATION thins or "generalizes" point vector data, extracts the vertices (points) from a line vector to a point vector file, and generalizes line vector data..

INTERPOL interpolates a raster surface from point data.

IDRISI interfaces to Gstat provide geostatistical tools that can be used to create a raster surface from point data.

TINPREP adds or thins vertices along vector lines.

INTERCON interpolates a raster surface from rasterized isoline data.

TIN creates a TIN from point or line vector data. TINSURF may be automatically called from the TIN dialog if a raster surface output is desired.

TINSURF creates a raster surface from an existing TIN.

95. In this context, the term "vertices" refers to all the points that make up a line, including the beginning and ending points.

Non-Constrained and Constrained TINs

The non-constrained Delaunay triangulation is described in the Introduction section above and is implemented in the IDRISI TIN module using an algorithm designed for speed of processing. First, the set of input points (or isoline vertices) are divided into sections. Then each of the sections is triangulated. The resulting "mini-TINs" are then merged together. A local optimization procedure is always implemented during the merging process to maximize the minimum angles and thus satisfy Delaunay criteria for the triangulation.

A constrained Delaunay triangulation is an extension of the non-constrained triangulation described above, with additional conditions applied to the selection of triangle vertices. In IDRISI, the constrained Delaunay triangulation uses isolines as non-crossing break-line constraints to control the triangulation process. This process ensures that triangle edges do not cross isolines and that the resulting TIN model is consistent with the original isoline data. Not all triangles will necessarily meet the Delaunay criteria when the constrained triangulation is used.

In IDRISI, the constrained TIN is created in a two-step process. First, a non-constrained triangulation is completed. Then triangle edges are checked for isoline intersections. When such an intersection is encountered, a local optimization routine is again run until no isoline intersections remain.

Figure 22-3 shows constrained and unconstrained TINs created from the same set of isoline vertex data points.

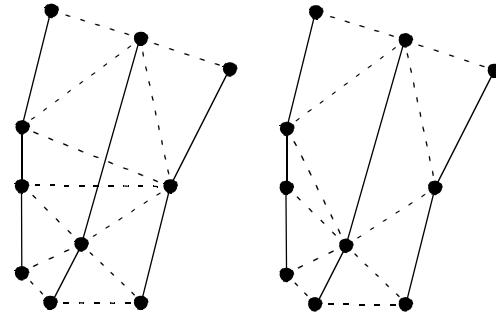


Figure 22-3 Unconstrained (left) and constrained (right) Delaunay triangulations. Solid lines represent isolines.

Removing TIN “Bridge” and “Tunnel” Edges

Contour lines at the top of a hill are shown in Figure 22-4a. In Figure 22-4b, the highest contour is shown along with the resulting triangles created within it when a constrained TIN is generated. Because all three of the points for all of the triangles have the same elevation, the top of the hill is perfectly flat in the TIN model. Our experience with actual terrain tells us that the true surface is probably not flat, but rather rises above the TIN facets. The edges of the TIN facets that lie below the true surface in this case are examples of what are called “tunnel edges”. These are identified in Figure 22-4b. A tunnel edge is any triangle edge that lies below the true surface. Similarly, if the contours of Figure 22-4a represented a valley bottom or depression, the TIN facets of 22-4b would describe a flat surface that is higher than the true surface. The edges of the TIN facets that lie above the true surface would then be termed “bridge edges”.

Bridge and tunnel (B/T) edges are not restricted to hill tops and depression bottoms. They can also occur along slopes, particularly where isolines are undulating, and along ridges or channels. Two such examples are shown in Figure 22-5.

To optimize a TIN, B/T edges may be removed. B/T edge removal could technically be performed on an unconstrained TIN, but this is not recommended and is not allowed in IDRISI. An optimal TIN will be generated if isolines are used as the original input for TIN generation, the constrained triangulation is used, and B/T edges are removed.

While many of the concepts of this section are illustrated with elevation data, the procedures are not limited to such data.

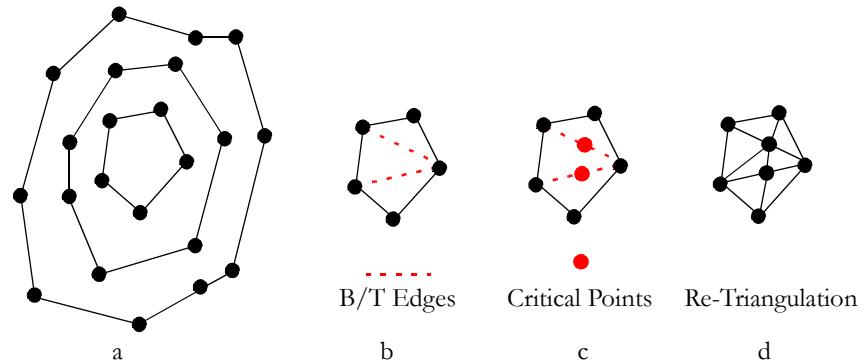


Figure 22-4 a: Contours at the top of a hill; b: triangulation of highest contour, with B/T edges identified; c: placement of critical points on B/T edges; d: re-triangulation.

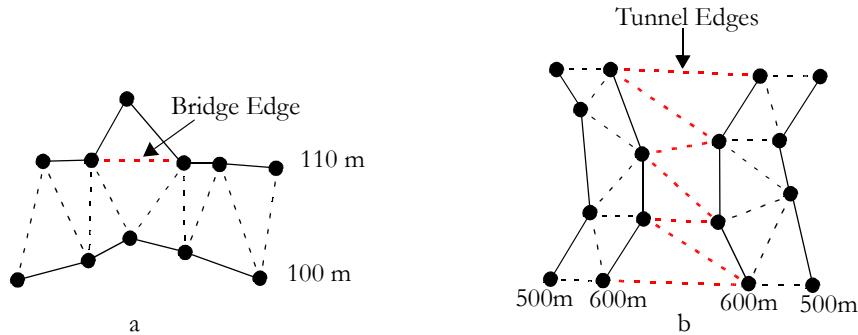


Figure 22-5 a: Contours at a stream; b: contours at a “saddle” feature. Contours are shown with solid lines, constrained triangle edges with dashed lines. B/T edges are shown in red.

Bridge and Tunnel Edge Removal and TIN Adjustment

The IDRISI TIN module includes an option to create a TIN with all B/T edges removed. This option is only available if isoline data and the constrained triangulation option are used. First, a normal TIN is created from the vector input data. Then, all of the B/T edges in the TIN are identified. In IDRISI, a B/T edge is defined as any triangle edge with endpoints of the same attribute, where these endpoints are not neighboring points on an isoline.

New points, termed *critical points*, are created at the midpoints of the B/T edges (Figure 22-4c). The areas around the critical points are then re-triangulated (Figure 22-4d). When a B/T edge is shared by two triangles, four new triangles result. When a B/T edge is part of the TIN boundary, and is thus used by only one triangle, two new triangles result.

Once the critical points have been placed and the triangulation has been adjusted, the next step is to assign appropriate attribute values (e.g., elevations) to these new points.

Attribute Interpolation for the Critical Points

In IDRISI, the recommended method for determining the attribute of a critical point uses a parabolic shape. The parabola, as a second-order non-linear polynomial method, was chosen because it combines computational simplicity and a

shape that is compatible with most topographic surfaces.⁹⁶ Before describing the mathematical details and the algorithm of the calculation of critical point values, we will use an illustration to think through the general logic.

General Logic

Let us assume that the contours of Figure 22.4a describe a hill and that the hilltop beyond the highest contour has a somewhat rounded peak. Given this, we could imagine fitting a parabolic surface (like an inverted, U-shaped bowl) to the top of the hill. The particular parabolic surface we would choose would depend on the shape of the nearby terrain. If slopes were gentle leading up to the highest contour, then we would choose a surface with gently sloping sides and a wide top. But if slopes were quite steep, we would choose a surface with more vertical sides and a narrower top. Once a particular surface was chosen, all critical points on the tunnel edges at the top of the hill could be projected onto the parabolic surface. They could then each be assigned the elevation of the surface at their location.

The actual implementation of the interpolation differs from the general logic described above in that two-dimensional parabolas are used rather than parabolic surfaces. Up to eight parabolas, corresponding to eight directions, are fit through each critical point location. An attribute for the critical point is derived for each parabola, and the final attribute value assigned to the point is their average. Details of the process are given below.

Calculating the Critical Point Attribute

A parabola is defined by the following equation:

$$(X-a)^2 = 2p(Y-b)$$

Where the point (a,b) defines the center (top or bottom) point of the parabola and the parameter p defines the steepness of the shape. When p is positive, the parabola is U-shaped. When p is negative, the parabola is inverted. The larger the absolute value of p , the wider the parabola.

Figure 22-6 shows several parabolas and their equations.

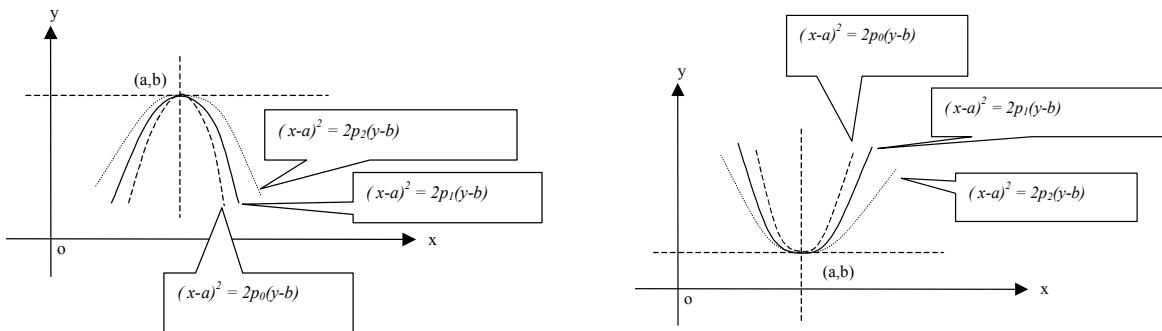


Figure 22-6 Example parabolas and their equations. On the left, p is negative. On the right, p is positive.

To translate the general parabolic equation to the critical point attribute interpolation problem, we re-label the axes of Figure 22-6 from X,Y to S,H where S represents distance from the origin (o) and H represents the attribute value (e.g., elevation) from the origin (o). (The origin is defined by the location and attribute of the *original point* as described below.)

96. Although the parabolic algorithm is recommended, linear and optimized linear options are also available as critical point interpolation methods in the module TIN. In the example of the hilltop, a sharp peak would be modeled by the linear method in contrast to the rounded peak of the parabolic method. The optimized linear method uses a linear interpolation unless slopes in all eight directions (see the discussion of the parabolic interpolation) are zero, in which case it uses the parabolic.

In the example of a critical point on a tunnel edge at the top of a hill, the plane of the parabola is a cross section of the hill.

To define a parabola for a critical point, three points with known coordinates and attributes that lie on that same parabola must be found.⁹⁷ Up to eight parabolas, each defined by three points, are developed for each critical point.

For each critical point, a search process is undertaken to find intersections with isolines in each of eight directions, as shown in Figure 22-7a. If two intersections are found in each direction, then eight parabolas can be defined. Each is defined by three points, with two points taken from one direction from the critical point and the other one taken from the opposite direction. In Figure 22-7b, the intersection points for one search direction, points P_0 , P_1 and P_2 , are used to define the parabola shown in Figure 22-7c. The point that lies between two intersections from the critical point is always termed the *original point* and is labeled P_0 . This point is set at $S=0$, so distances (S) to all other points are measured from this original point. P_1 lies between the critical point and the original point, and P_2 lies on the opposite side of the critical point.

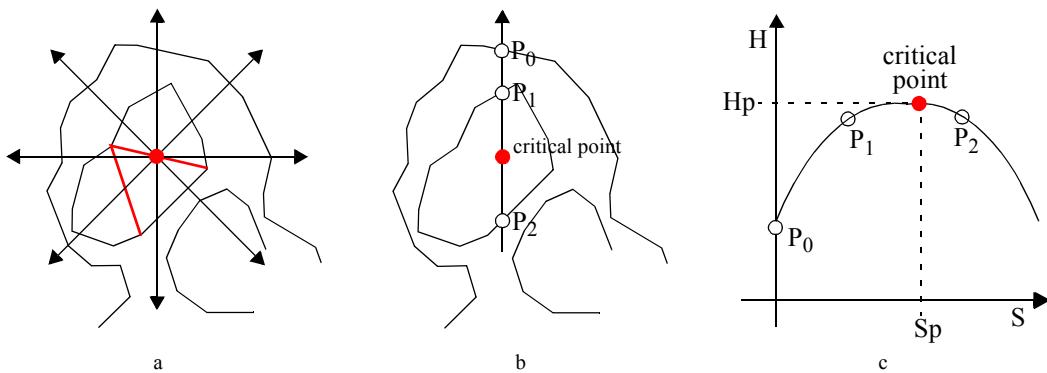


Figure 22-7 a: Eight-direction search for isoline intersections for one critical point; b: intersection points for one direction; c: parabola derived from intersection points. Attribute (h_p) for the critical point can be found, given the critical point's distance (S_{cp}) from P .

If three intersections are not found for a particular parabola (e.g., at the edge of a coverage), then it is undefined and the number of parabolas used to interpolate the attribute value for that critical point will be fewer than eight.

For each defined parabola, the attribute value of any point on the parabola can be found by entering its distance from the original point into the parabolic equation. The following equation can be used to calculate the attribute of a critical point for one of its parabolas:⁹⁸

$$H = \sum_{i=0}^2 h_i \cdot \prod_{j=0, j \neq i}^2 \left[\frac{(S_{point} - S_j)}{(S_i - S_j)} \right]$$

Where h_i , $i = 0, 1, 2$ are attribute values of the three intersection points, P_0 , P_1 and P_2 ; S_i , S_j , $i, j = 0, 1, 2$ represent the distances from the original point to the intersection points, and S_{point} represents the distance from the original point to the critical point. According to the above definitions of the intersection points (Figure 22-6b), we know $S_0 \equiv 0$, while $S_1 = P_1 P_0$,

97. Any parabola can be defined once three points on it are known. Three equations (one for each point) can be written as below. For each, the distance (S) from that point to the origin and the attribute (H) are known. The simultaneous equations can then be solved for a , b , and p .

$$(S_0 - a)^2 = 2p(H_0 - b) \quad (S_1 - a)^2 = 2p(H_1 - b) \quad (S_2 - a)^2 = 2p(H_2 - b)$$

98. The equation incorporates the derivation of the parabolic parameters a , b , and p .

and $S_2 = P_2 P_0$

For each parabola, the attribute value at the position of the critical point is calculated in this manner. The final attribute value that is assigned to the critical point is the average of all valid interpolated values (invalid cases are discussed below).

Figure 22-8 shows several examples of cases in which B/T edges would be identified and a new value for the critical points placed on their midpoints would be interpolated. In each figure, only one search direction is illustrated. Figures 22-8 a, b and c are examples of cases where critical points occur along slopes while figures 22-8 d, e and f are cases where critical points occur on hill tops. For cases in which the attribute value of the critical point is lower than those of the surrounding isolines, the curves would be inverted.

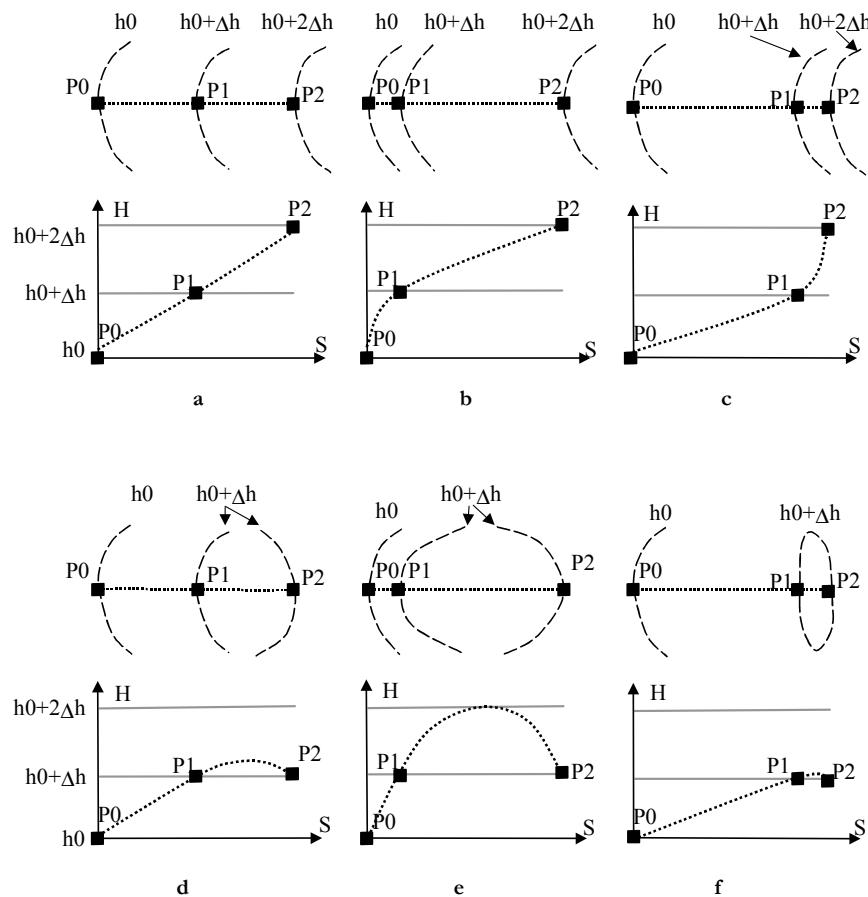


Figure 22-8 Six examples of parabolic critical point interpolation. The upper part of each example illustrates the map view of the isolines (dashed) and the intersection points (P_0 , P_1 and P_2) for one search direction (dotted line) for a critical point. The lower part shows the parabola for that set of intersection points. The attribute for the critical point (which is always between P_1 and P_2) can be found by plotting the critical point on the parabolic curve at its distance (S) from P_0 .

Invalid Cases

There are two extreme circumstances in which the parabolic interpolation procedure is invalid:

1. If all three intersection points have the same attribute value, the three points are not used for interpolation. An interpolated value for the critical point is therefore not calculated for this direction. The attribute value assigned would be an average of the other interpolated values.
2. If the interpolated value is greater than (in the case of tunnel edges) or less than (in the case of bridge edges) the value of the next expected contour, then the critical point is assigned the value of the next expected contour.⁹⁹ The nature of contour maps requires such a limitation.

Outputs of TIN

The outputs of the TIN module are a vector line file defining the triangle edges, an ASCII .TIN file containing the topological information for the triangulation and, if B/T edge removal was used, a point vector file of the critical points that were added. All these pieces except for the triangle edge vector file, in addition to the original vector data file, are used by the TINSURF module to create a raster surface from the TIN.

Generating a Raster Surface from a TIN

A raster surface may be generated from the TIN at the time the TIN is created or may be created from an existing TIN file later. The TINSURF module creates the raster surface. Its dialog asks only for the TIN file as input. However, the TIN file stores the name of the original vector file used to create the TIN as well as whether B/T edge removal was used. If the TIN is the result of B/T edge removal, then TINSURF also requires the critical point vector file. Therefore you should not delete, move or rename any of these files prior to creating the raster surface.

For each raster pixel in the output image, an attribute value is calculated. This calculation is based on the positions and attributes of the three vertex points of the triangular facet within which the pixel center falls and the position of the pixel center.¹⁰⁰ The logic is as follows:

1. Solve the following set of simultaneous equations for A, B and C:

$$H_1 = Ax_1 + By_1 + C$$

$$H_2 = Ax_2 + By_2 + C$$

$$H_3 = Ax_3 + By_3 + C$$

Where $H_{1,2,3}$ are the attribute values (e.g., elevations) of the three triangle facet vertices and $(x,y)_{1,2,3}$ are their reference system coordinates.

2. Given A, B and C, as derived above, solve the following for H_p :

$$H_p = Ax_p + By_p + C$$

Where H_p is the attribute of the pixel and $(x,y)_p$ is the reference system coordinate of the pixel center.

99. The algorithm uses the local contour interval for each critical point, so isoline data with variable contour intervals do not pose a problem.

100. Each pixel center will fall in only one TIN facet, but a single facet may contain several pixel center points.

3. Assign the pixel the attribute value H_p

The algorithm proceeds on a facet-by-facet basis, so the derivation of A, B, and C in step 1 is carried out only once for all the pixels that fall within a single facet.

Raster Surface Optimization

For optimal generation of a raster surface from a TIN model, care should be taken in preparing the data used to create the TIN. If isoline data is used, the isolines should not cross. The distribution of points in the input vector file should be evaluated visually and adjusted, if necessary, by thinning or adding points. If point attribute values are available at peaks and valleys in the study area, adding these to the input data will reduce bridge and tunnel edge effects and will enhance the quality of the resulting TIN and the subsequent raster surface.

A TIN will cover only the area inside the convex hull of the data points. This may present a problem if the original vector data does not cover the entire study area. The areas outside the convex hull will not be covered by triangles in the TIN and will be assigned a background value in the resulting raster surface. An option to add corner points is available on the TIN dialog to help mitigate this problem for the corners of the image. However, there may still be areas outside the convex hull even when corner points are added. If possible, it is recommended that the vector point or isoline data used to create the TIN extend beyond the limits of the desired raster study area. Then specify the final raster bounding coordinates in TINSURF. This will produce a TIN that covers the entire rectangular study area and a raster surface that contains no background values.

Further Reading

Lee J., 1991. Comparison of Existing Methods for Building Triangular Irregular Network Models of Terrain From Grid Digital Elevation Models, *International Journal of Geographic Information Systems*, 3: 267-285.

Tsai, V. J. D., 1993. Delaunay Triangulations in TIN Creation: an Overview and a Linear-time Algorithm, *International Journal of Geographic Information Systems*, 6: 501-512.

Zhu, H., Eastman, J. R., and Schneider, K., 1999. Constrained Delaunay Triangulation and TIN Optimization Using Contour Data, *Proceedings of the Thirteenth International Conference on Applied Geologic Remote Sensing*, 2: 373-380, Vancouver, British Columbia, Canada.

Geostatistics

Introduction

Geostatistics provides tools for the exploration and statistical characterization of sample point data. It also provides a number of techniques for interpolating surfaces from such data. *Ordinary kriging* is the most well-known of these. While the techniques originated with scientists working in the mining industry, a broader audience has been found in those fields in which both data values and their locations are considered analytically important.

Several interpolation techniques were introduced in the chapter **Surface Interpolation**. Geostatistical techniques are distinct from these in that they provide GIS analysts with the ability to incorporate information about patterns of spatial continuity into the interpolation model as well as to produce surfaces that include elements of local variation. The methods allow for a high degree of user flexibility in detecting and defining structures that describe the nature of a data set. Indeed, a set of structures can be nested, each describing a particular aspect of the data set.

With this flexibility, however, also comes some risk. From the same data set, it is possible to produce many surfaces—all very different, and all seemingly reasonable representations of reality. The new user is encouraged to enter into geostatistics deliberately and with some caution. An understanding of, and respect for, the underlying assumptions of these techniques is essential if the results are to provide meaningful information to any analysis.

This chapter presents a very brief overview of the geostatistical capabilities offered through IDRISI interfaces to Gstat.¹⁰¹ For more complete and theoretical treatments of geostatistics, consult the references listed at the end of this chapter. The **Tutorial** includes an extensive exercise illustrating the use of the geostatistical tools available in IDRISI.

Spatial Continuity

The underlying notion that fuels geostatistical methods is quite simple. For continuously varying phenomena (e.g., elevation, rainfall), locations that are close together in space are more likely to have similar values than those that are further apart. This tendency to be most similar to one's nearest neighbors is quantified in geography through measures of spatial autocorrelation and continuity. In geostatistics, the complement of continuity, variability, is more often the focus of analysis.

The first task in using geostatistical techniques to create surfaces is to describe as completely as possible the nature of the spatial variability present in the sample data. Spatial variability is assessed in terms of distance and direction. The analysis is carried out on *pairs* of sample data points. Every data point is paired with every other data point. Each pair may be characterized by its *separation distance* (the Euclidean distance between the two points) and its *separation direction* (the azimuth in degrees of the direction from one point to the other).¹⁰² The sample data point set shown in Figure 23-1 would produce pairs characterized as shown in Table 23-1.

101. IDRISI provides a graphical user interface to Gstat, a program for geostatistical modeling, prediction and simulation written by Edzer J. Pebesma (Department of Physical Geography, Utrecht University). Gstat is freely available under the GNU General Public License from <http://www.geog.uu.nl/gstat/>. Clark Labs' modifications of the Gstat code are available from the downloads section of the Clark Labs website at <http://www.clarklabs.org>.

102. The points in a pair are identified as the *from* point and the *to* point. No pair is repeated.

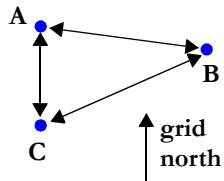


Figure 23-1

Pair	Separation Distance	Separation Direction
AB	80 m	100
AC	50 m	0
BC	85 m	235

Table 23-1

The distance measure is typically referred to in units of *lags*, where the length of a lag (i.e., the lag distance or lag interval) is set by the user. In specifying a particular lag during the analysis, the user is limiting the pairs under consideration to those that fall within the range of distances defined by the lag. If the lag were defined as 20 meters, for example, an analysis of data at the third lag would include only those data pairs with separation distances of 40 to 60 meters.

Direction is measured in degrees, clockwise from grid north. As with distance, direction is typically specified as a range rather than a single azimuth.

The h-scatterplot is used as a visualization technique for exploring the variability in the sample data pairs. In the h-scatterplot, the X axis represents the attribute at one point of the pair (the *from* point) and the Y axis represents that same attribute at the other point of the pair (the *to* point). The h-scatterplot may be used to plot all of the pairs, but is more often restricted to a selection of pairs based on a certain lag and/or direction. Figure 23-2 shows the spatial distribution of 250 rainfall sample points from a 1000 km^2 area. These points were paired and data pairs that are within 1 lag (0-1 km) and for all directions are plotted in the h-scatterplot shown in Figure 23-3.

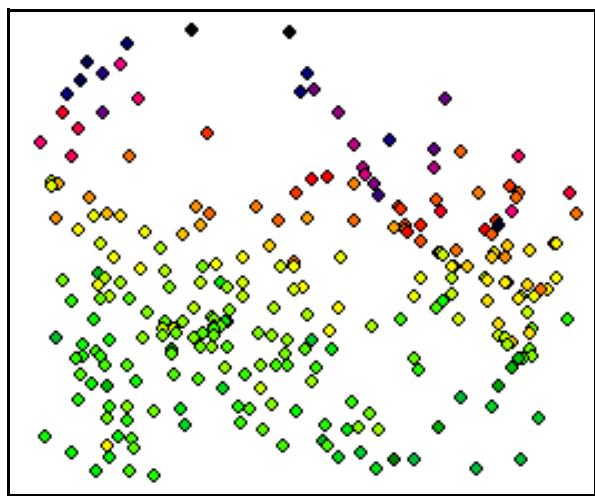


Figure 23-2

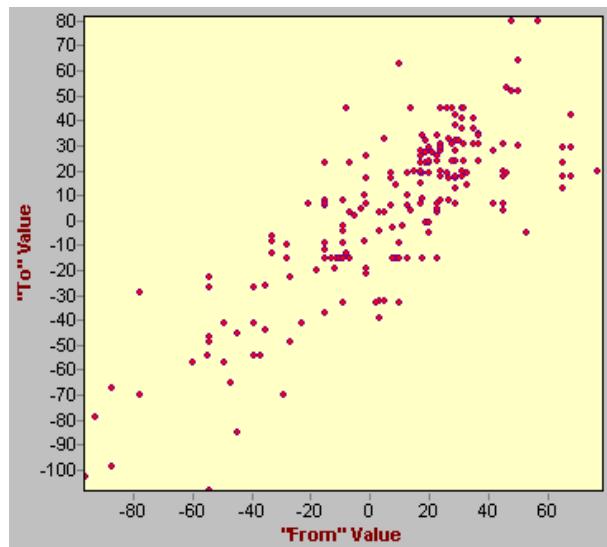


Figure 23-3

The h-scatterplot is typically used to get a sense of what aspects of the data pair distribution are influencing the summary of variability for a particular lag. H-scatterplots are interpreted by assessing the dispersion of the points. For example, if the pairs were perfectly linearly correlated (i.e., no variability at this separation and direction), then all the points would fall along a line. A very diffuse point pattern in the h-scatterplot indicates high variability for the given ranges of distance and direction. The h-scatterplot is available through the Spatial Dependence Modeler interface.

The semivariogram is another tool for exploring and describing spatial variability and is also available through the Spatial Dependence Modeler interface. The semivariogram summarizes the variability information of the h-scatterplots and may be presented both as a surface graph and a directional graph. The surface graph shows the average variability in all directions at different lags. The center position in the graph, called the *origin*, represents zero lags. The lags increase from the center toward the edges. The direction is represented in the surface graph with grid north directly up from the center pixel, 90 degrees directly to the right, and so on.¹⁰³ The magnitude of variability is represented by color using the default IDRISI palette. Low values are shown in darker colors and higher values in brighter colors. When one moves the cursor over the surface graph, its location, in terms of direction and distance from the origin, is shown at the bottom of the graph.

A surface graph semivariogram of the same sample rainfall points from Figure 23-2 is shown in Figure 23-4. The lag distance is set to 1 km. One can readily see that in the West-East direction, there is low variability among the pairs across all lags. It appears that the direction of minimum variability (i.e., maximum continuity) is approximately 95 (and 275) degrees. We would expect data points that are separated from each other in this direction to have attributes that are more similar than data points separated by the same distance but in a different direction.

The other graphic form of the semivariogram is the directional graph, as shown in Figure 23-5. It is used to develop the structures that describe the patterns of variability in the data. In the directional graph, a single summary point is plotted for each lag. The X-axis shows the separation distance, labeled in reference units (e.g., km), while the Y-axis shows the average variability for the sample data pairs that fall within each lag. All pairs may be considered regardless of direction (an omnidirectional plot), or the plot may be restricted to pairs from a particular range of directions.

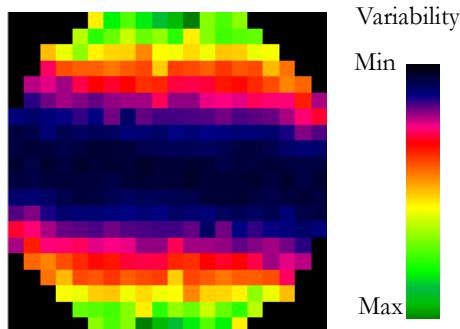


Figure 23-4

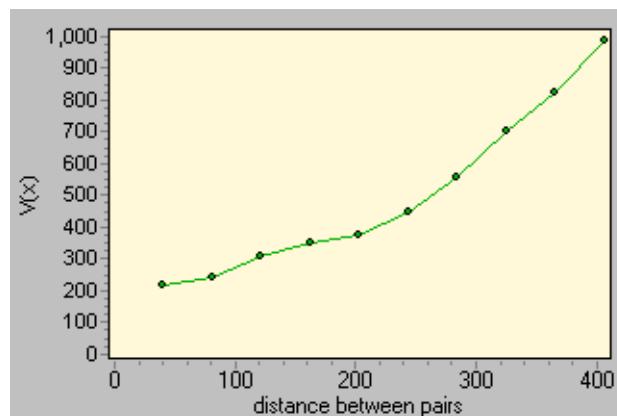


Figure 23-5

Usually one begins with plotting an omnidirectional semivariogram. From the omnidirectional graph, one may gain insight into the overall variability of the data. The user then may create several plots, using different directions and lag distances, to gain a better understanding of the structure of the data set.

103. Note that some geostatistical software plot zero degrees to the right rather than the top of the surface graph.

The structure of the data may be described by four parameters: the *sill*, the *range*, the *nugget* and *anisotropy*. The first three are labeled in Figure 23-6. In most cases involving environmental data, spatial variability between sample pairs increases as the separation distance increases. Eventually, the variability reaches a plateau where an increase in separation distance between pairs no longer increases the variability between them, i.e., there is no spatial dependence at this and larger distances. The variance value at which the curve reaches the plateau is called the *sill*. The total separation distance from the lowest variance to the sill is known as the *range*. The range signifies the distance beyond which sample data should not be considered in the interpolation process when selecting points that define a local neighborhood.

The *nugget* refers to the variance at a separation distance of zero, i.e., the y-intercept of the curve that is fit to the data. In theory, we would expect this to be zero.

However, noise or uncertainty in the sample data may produce variability that is not spatially dependent and this will result in a non-zero value, or a *nugget effect*. A nugget structure increases the variability uniformly across the entire graph because it is not related to distance or direction of separation.

The fourth parameter that defines the structure is the anisotropy of the data set. The transition of spatial continuity may be equal in all directions, i.e., variation is dependent on the separation distance only. This is known as an *isotropic model*. A model fit to any direction is good for all directions. In most environmental data sets, however, variability is not isotropic. The data used in Figure 23-2, for example, exhibits a minimum direction of variability in the West-East direction. In any other direction, variability increases more rapidly at the same separation distance. This type of data requires an *anisotropic model*. Anisotropy is described by directional axes of minimum and maximum continuity. To determine the parameters to be used, the user views directional semivariograms for multiple directions.

In kriging and simulation interpolation processes, structures that describe the pattern of spatial variability represented by directional semivariograms are used to determine the influence of spatial dependence on neighborhoods of sample points selected to predict unknown points. The structures influence how their attributes should be weighted when combined to produce an interpolated value. Semivariograms, however, because they are based on the inherent incompleteness of sample data, need smoother curves that define the shape of the spatial variability across all separation distances. Using ancillary information and the semivariograms, mathematical functions are combined to delineate a smooth curve of spatial variability. At this stage, a nugget structure, and sills, ranges, and anisotropies of additional structures are defined for the smooth curve. The Model Fitting interface offers several mathematical functions that may be used to design a curve for the spatial variability. Those functions that do not plateau at large separation distances, such as the linear and the power functions, are termed non-transitional. Those that do reach a plateau, such as the gaussian and exponential functions, are called transitional functions.

Together, the nugget structure, and the sills, ranges, and anisotropies of additional structures mathematically define a nested model of spatial variability. This is used when locally deriving weights for the attributes of sample data within the neighborhood of a location to be interpolated. Using the Spatial Dependence Modeler interface, one uncovers a pattern of spatial variability through the plotting of many variograms until a representative semivariogram can be determined. Through the Model Fitting interface, the user fits a mathematical curve described by sills, ranges, a nugget, anisotropy and selected functions to the detected spatial variability. This curve is used to derive the weights applied to locally selected samples during the interpolation by kriging or conditional simulation.

Semivariograms are statistical measures that assume the input sample data are normally distributed and that local neighborhood means and standard deviations show no trends. Each sample data set must be assessed for conformity to these assumptions. Transformations of the data, editing of the data set, and the selection of different statistical estimators of spatial variability are all used to cope with data sets that diverge from the assumptions.

The ability to identify true spatial variability in a data set depends to a great extent on ancillary knowledge of the underlying phenomenon measured. This detection process can also be improved with the inclusion of other attribute data. The

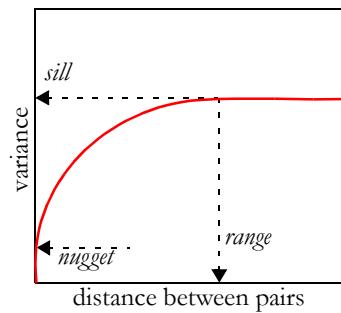


Figure 23-6

crossvariogram, like the semivariogram, plots variability along distances of joint datasets and uses one set of data to help explain and improve the description of variability in another. For example, when interpolating a rainfall surface from point rainfall data, incorporating a highly correlated variable such as elevation could help improve the estimation of rainfall. In such a case where the correlation is known, sampled elevation data could be used to help in the prediction of a rainfall surface, especially in those areas where rainfall sampling is sparse.

The semivariogram and another method, the robust estimator of the semivariogram, are the measures of variability that are used for the final fitting of a variability model to be used with the data set. They are also the only estimators of variability used by IDRISI for kriging and simulation. However, other methods for detecting spatial contiguity are available through the Spatial Dependence Modeler interface. These include the correlogram, the cross-correlogram, the covariogram, and the cross-covariogram.

Kriging and Conditional Simulation

The Kriging and Simulation interface utilizes the model developed in the Spatial Dependence Modeler and Model Fitting interfaces to interpolate a surface. The model is used to derive spatial continuity information that will define how sample data will be weighted when combined to produce values for unknown points. The weights associated with sample points are determined by direction and distance to other known points, as well as the number and character of data points in a user-defined local neighborhood.

With ordinary kriging, the variance of the errors of the fit of the model is minimized. Thus it is known as a Best Linear Unbiased Estimator (B.L.U.E.).

By fitting a smooth model of spatial variability to the sample data and by minimizing the error of the fit to the sample data, kriging tends to underestimate low values and overestimate large values. Kriging minimizes the error produced by the differences in the fit of the spatial continuity to each local neighborhood. In so doing, it produces a smooth surface.

The surface shown in Figure 23-7 was produced using kriging with the sample precipitation points shown in Figure 23-2.

The goal of kriging is to reduce the degree of variance error in the estimation across the surface. The variance error is a measure of the accuracy of the fit of the model and neighborhood parameters to the sample data, not the actual measured surface. One can only interpret this information in terms of knowledge about how well the sample data represents the actual surface. The more uniform the fit of the spatial model, the more likely it is good. The variance error is used to identify problems in the sample data, in the model parameters, and in the definition of the local neighborhood. It is not a measure of surface accuracy.

In IDRISI, two tools are available to assess the fit of the model to the sample data. First, the cross-validation tool iteratively removes a sample data point and interpolates a new value for the location. A table is produced to show the difference between the predicted attributes and the known attributes at those locations. Second, a variance image is produced that shows the spatial variation of uncertainty as a result of the fitted model. The variance image provides information to assist in identifying the problem areas where the relationship between the fitted model and the sample data points is poor.

Cokriging is an extension of kriging that uses a second set of points of different attributes to assist in the prediction process. The two attributes must be highly correlated with each other to derive any benefit. The description of spatial variability of the added variable can be used in the interpolation process, particularly in areas where the original sample points are sparse.

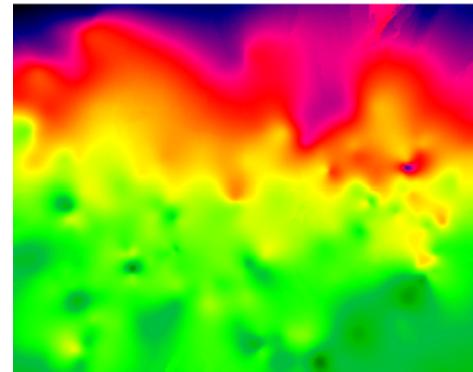


Figure 23-7

In conditional simulation, a non-spatially dependent element of variability is added to the model previously developed. The variability of each interpolated point is used to randomly choose another estimate. The resulting surface maintains the spatial variability as defined by the semivariogram model, but also represents pixel-by-pixel variability. The resulting surface is not smooth. Typically many of these surfaces (perhaps hundreds) are produced, each representing one model of reality. The surfaces differ from each other because of the random selection of estimates. Conditional simulation is best suited for developing multiple representations of a surface that may serve as inputs to a Monte Carlo analysis of a process model.

Summary

Geostatistics provides a large collection of tools for exploring and understanding the nature of a data set. Rather than simply seeking to produce a visually-pleasing interpolated surface, one engages in geostatistical analysis with the foremost purpose of understanding why various methods produce particular and different results. Interpretation of the information presented through the various techniques is dependent upon knowledge of other data characteristics and the actual surface. While spatial variability measures themselves are relatively simple descriptive statistics, understanding how they may be used with data sets that diverge from ideal assumptions requires practice and experience.

References / Further Reading

Geostatistical analysis is a well developed field and much literature is available. The brief list that follows should provide a good introduction to geostatistical exploration for those who already have a good command of statistics.

- Burrough, P., and McDonnell, R., 1998. *Principles of Geographical Information Systems*, 98-161, Oxford University Press, Oxford.
- Cressie, N., 1991. *Statistics for Spatial Data*, John Wiley and Sons, Inc., New York.
- Cressie, N., and Hawkins, D., 1980. Robust Estimation of the Variogram, *Journal International Association of Mathematical Geology*, 12:115-125.
- Deutsch, C., and Journel, A., 1998. *GSLIB Geostatistical Software Library and User's Guide, 2nd Edition*, Oxford University Press, Oxford.
- Goovaerts, P., 1997. *Geostatistics for Natural Resources Evaluation*, Oxford University Press, Oxford.
- Issaks, E., and Srivastava, R., 1989. *Applied Geostatistics*, Oxford University Press, Oxford.
- Journel, A., and Huijbregts, C., 1978. *Mining Geostatistics*, Academic Press, New York.
- Myers, J., 1997. *Geostatistical Error Management: Quantifying Uncertainty for Environmental Sampling and Mapping*, Van Nostrand Reinhold, New York.
- Pebesma, E., 1991-1998. Gstat, GNU Software Foundation.
- Pebesma, E., and Wesseling, C., 1998. Gstat: A Program for Geostatistical Modelling, Prediction and Simulation, *Computers and Geosciences*, 24(1): 17-31.
- Soares, A., Gómez-Hernandez, J., and Froidevaux, R., eds., 1997. *geoENV – Geostatistics for Environmental Applications*, Kluwer Academic Publishers, Dordrecht, The Netherlands .
- Solow, A., and Ratnick, S., 1994. Conditional Simulation and the Value of Information, In: *Geostatistics for the Next Century*, R. Simitrakopoulos (ed.), Kluwer Academic Publishers, Dordrecht, The Netherlands.

Appendix 1: Ellipsoid Parameters

ELLIPSOID NAME	MAJOR SEMI-AXIS	MINOR SEMI-AXIS
Airy	6377563.396	6356256.909
Modified Airy	6377340.189	6356034.448
Australian National	6378160.000	6356774.719
Average Terrestrial System 1997	6378135	6356750.305
Bessel 1841 (Ethiopia, Indonesia, Japan, Korea)	6377397.155	6356078.963
Bessel 1841 (Namibia)	6377483.865	6356165.383
Bessel Modified	6377492.018	6356173.509
Clarke 1858	20926348	20855233
Clarke 1866	6378206.400	6356583.800
Clarke 1866 Michigan	20926631.531	20855688.674
Clarke 1880	6378249.145	6356514.870
Clarke 1880 Benoit	6378300.789	6356566.435
Clarke 1880 IGN	6378249.2	6356515
Clarke 1880 (SGA 1922)	6378249.2	6356514.997
Everest - India 1830	6377276.345	6356075.413
Everest - India 1956	6377301.24	6356100.23
Everest - Pakistan	6377309.61	6356108.57
Everest - Sabah and Sarawak	6377298.56	6356097.55
Everest - West Malaysia 1969	6377295.66	6356094.67
Everest - West Malaysia and Singapore 1948	6377304.063	6356103.039
Everest (1830 definition)	20922931.8	20853374.58
Everest 1830 (1962 definition)	6377301.243	6356100.23
Everest 1830 (1967 definition)	6377298.556	6356097.55
Everest 1830 (1975 definition)	6377299.151	6356098.145
Fischer 1960	6378166.000	6356784.280
Modified Fischer 1960	6378155.000	6356773.320
Fischer 1968	6378150.000	6356768.340
GEM 10C	6378137	6356752.314
GRS 1967	6378160	6356774.516

ELLIPSOID NAME	MAJOR SEMI-AXIS	MINOR SEMI-AXIS
GRS 1980	6378137.000	6356752.314
Helmert 1906	6378200.000	6356818.170
Hough	6378270.000	6356794.343
Indonesian 1974	6378160.00	6356774.50
International 1924	6378388.000	6356911.946
Krassovsky 1940	6378245.000	6356863.019
NWL 9D	6378145	6356759.769
OSU86F	6378136.2	6356751.517
OSU91A	6378136.3	6356751.617
Plessis 1817	6376523	6355862.933
SGS 85	6378136.000	6356751.300
South American 1969	6378160.000	6356774.719
Sphere	6371000	6371000
Struve 1860	6378298.3	6356657.143
War Office	6378300.583	6356752.27
WGS 60	6378165.000	6356783.290
WGS 66	6378145.000	6356759.770
WGS 72	6378135.000	6356750.520
WGS 84	6378137.000	6356752.314

Appendix 2: Datum Parameters

The following table contains the constants required for the Molodensky Datum Transformation procedure. The ΔX , ΔY and ΔZ values are the three values (in that order) that should be specified in the "Delta WGS84" field of the Reference System Parameter File. These values represent the three-dimensional difference in position of the datum ellipsoid from that of WGS84. The values listed here were taken from the European Petroleum Survey Group Database.

DATUM	LOCATION	ELLIPSOID	DX	DY	DZ
ABIDJAN 1987	Cote D'Ivoire (Ivory Coast)	Clarke 1880	-124.76	53	466.79
ADINDAN	MEAN FOR Ethiopia, Sudan	Clarke 1880	-166	-15	204
	Burkina Faso		-118	-14	218
	Cameroon		-134	-2	210
	Ethiopia		-165	-11	206
	Mali		-123	-20	220
	Senegal		-128	-18	224
	Sudan		-161	-14	205
AFGOOYE	Somalia	Krassovsky 1940	-43	-163	45
AIN EL ABD 1970	Bahrain Island	International 1924	-150	-250	-1
	Saudi Arabia		-143	-236	7
AMERICAN SAMOA 1962	American Samoa Islands	Clarke 1866	-115	118	426
AMERSFOORT	Netherlands	Bessel 1841	593.16	26.15	478.54
ANNA 1 ASTRO 1965	Cocos Islands	Australian National	-491	-22	435
ANTIGUA ISLAND ASTRO 1943	Antigua (Leeward Islands)	Clarke 1880	-270	13	62

DATUM	LOCATION	ELLIPSOID	DX	DY	DZ
ARC 1950	MEAN FOR Botswana, Lesotho, Malawi, Swaziland, Zaire, Zambia, Zimbabwe	Clarke 1880	-143	-90	-294
	Botswana		-138	-105	-289
	Burundi		-153	-5	-292
	Lesotho		-125	-108	-295
	Malawi		-161	-73	-317
	Swaziland		-134	-105	-295
	Zaire		-169	-19	-278
	Zambia		-147	-74	-283
	Zimbabwe		-142	-96	-293
ARC 1960	MEAN FOR Kenya, Tanzania	Clarke 1880	-160	-6	-302
ASCENSION ISLAND 1958	Ascension Island	International 1924	-205	107	53
ASTRO BEACON E 1945	Iwo Jima	International 1924	145	75	-272
ASTRO DOS 71/4	St. Helena Island	International 1924	-320	550	-494
ASTRO TERN ISLAND (FRIG) 1961	Tern Island	International 1924	114	-116	-333
ASTRONOMICAL STATION 1952	Marcus Island	International 1924	124	-234	-25
AUSTRALIAN GEODETIC 1966	Australia & Tasmania	Australian National	-133	-48	148
AUSTRALIAN GEODETIC 1984	Australia & Tasmania	Australian National	-134	-48	149
AYABELLE LIGHTHOUSE	Djibouti	Clarke 1880	-79	-129	145
BATAVIA	Indonesia (Sumatra)	Bessel 1841	-377	681	-50
BELLEVUE (IGN)	Efate & Erromango Islands	International 1924	-127	-769	472
BERMUDA 1957	Bermuda	Clarke 1866	-73	213	296
BISSAU	Guinea - Bissau	International 1924	-173	253	27
BOGOTA OBSERVATORY	Colombia	International 1924	307	304	-318
BUKIT RIMPAH	Indonesia (Bangka & Belitung Islands)	Bessel 1841	-384	664	-48
CAMP AREA ASTRO	Antarctica (McMurdo Camp Area)	International 1924	-104	-129	239
CAMPO INCHAUSPE	Argentina	International 1924	-148	136	90
CANTON ASTRO 1966	Phoenix Islands	International 1924	298	-304	-375
CAPE	South Africa	Clarke 1880	-136	-108	-292
CAPE CANAVERAL	Bahamas, Florida	Clarke 1866	-2	151	181
CARTHAGE	Tunisia	Clarke 1880	-263	6	431
CHATHAM ISLAND ASTRO 1971	New Zealand (Chatham Island)	International 1924	175	-38	113
CHTRF95	Liechtenstein, Switzerland	GRS1980	0	0	0

DATUM	LOCATION	ELLIPSOID	DX	DY	DZ
CHUA ASTRO	Paraguay	International 1924	-134	229	-29
CORREGO ALEGRE	Brazil	International 1924	-206	172	-6
DABOLA	Guinea	Clarke 1880	-83	37	124
DECEPTION ISLAND	Deception Island, Antarctica	Clarke 1880	260	12	-147
DJAKARTA (BATAVIA)	Indonesia (Sumatra)	Bessel 1841	-377	681	-50
DOMINICA 1945	Dominica	Clarke 1880	725	685	536
DOS 1968	New Georgia Islands (Gizo Island)	International 1924	230	-199	-752
EASTER ISLAND 1967	Easter Island	International 1924	211	147	111
EGYPT 1907	Egypt	Helmer 1906	-130	110	-13
EST92	Estonia	GRS80	0.055	-0.541	-0.185
ETRF89	Europe	GRS80	0	0	0
EUROPEAN 1950	MEAN FOR Austria, Belgium, Denmark, Finland, France, West Germany, Gibraltar, Greece, Italy, Luxembourg, Netherlands, Norway, Portugal, Spain, Sweden, Switzerland	International 1924	-87	-98	-121
	MEAN FOR Austria, Denmark, France, West Germany, Netherlands, Switzerland		-87	-96	-120
	MEAN FOR Iraq, Israel, Jordan, Lebanon, Kuwait, Saudi Arabia, Syria		-103	-106	-141
	Cyprus		-104	-101	-140
	Egypt		-130	-117	-151
	England, Channel Islands, Ireland, Scotland, Shetland Islands		-86	-96	-120
	Finland, Norway		-87	-95	-120
	France		-84	-97	-117
	Greece		-84	-95	-130
	Iran		-117	-132	-164
	Italy (Sardinia)		-97	-103	-120
	Italy (Sicily)		-97	-88	-135
	Malta		-107	-88	-149
	Portugal, Spain		-84	-107	-120
	Tunisia		-112	-77	-145
	United Kingdom UKCS offshore east of 6 deg. west		-89.5	-93.8	-123.1

DATUM	LOCATION	ELLIPSOID	DX	DY	DZ
EUROPEAN 1979	MEAN FOR Austria, Finland, Netherlands, Norway, Spain, Sweden, Switzerland	International 1924	-86	-98	-119
Fahud					
FD58	Iran (Kangan district)	Clarke 1880	-241.54	-163.64	396.06
FORT THOMAS 1955	Nevis, St. Kitts (Leeward Islands)	Clarke 1880	-7	215	225
GAN 1970	Republic of Maldives	International 1924	-133	-321	50
GEODETIC DATUM 1949	New Zealand	International 1924	84	-22	209
GGRS87	Greece	GRS 1980	-199.87	74.79	246.62
GRACIOSA BASE SW 1948	Azores (Faial, Graciosa, Pico, Sao Jorge, Terceira)	International 1924	-104	167	-38
GRENADA 1953	Grenada	Clarke 1880	72	213.7	93
GUAM 1963	Guam	Clarke 1866	-100	-248	259
GUNUNG SEGARA	Indonesia (Kalimantan)	Bessel 1841	-403	684	41
GUX 1 ASTRO	Guadalcanal Island	International 1924	252	-209	-751
HERAT NORTH	Afghanistan	International 1924	-333	-222	114
HJORSEY 1955	Iceland	International 1924	-73	46	-86
HONG KONG 1963	Hong Kong	International 1924	-156	-271	-189
HU-TZU-SHAN	Taiwan	International 1924	-637	-549	-203
INDIAN	Bangladesh	Everest 1830	282	726	254
INDIAN	India, Nepal	Everest 1956	295	736	257
INDIAN	Pakistan	Everest	283	682	231
INDIAN 1954	Thailand	Everest 1830	217	823	299
INDIAN 1960	Vietnam (near 16°N) Con Son Island (Vietnam)	Everest 1830	198 182	881 915	317 344
INDIAN 1975	Thailand	Everest 1830	209	818	290
INDONESIAN 1974	Indonesia	Indonesian 1974	-24	-15	5
IRELAND 1965	Ireland	Modified Airy	506	-122	611
ISTS 061 ASTRO 1968	South Georgia Islands	International 1924	-794	119	-298
ISTS 073 ASTRO 1969	Diego Garcia	International 1924	208	-435	-229
JOHNSTON ISLAND 1961	Johnston Island	International 1924	189	-79	-202
KANDAWALA	Sri Lanka	Everest 1830	-97	787	86
KERGUELEN ISLAND 1949	Kerguelen Island	International 1924	145	-187	103
KERTAU 1948	West Malaysia & Singapore	Everest 1948	-11	851	5

DATUM	LOCATION	ELLIPSOID	DX	DY	DZ
KUSAIE ASTRO 1951	Caroline Islands, Federal States of Micronesia	International 1924	647	1777	-1124
L. C. 5 ASTRO 1961	Cayman Brac Island	Clarke 1866	42	124	147
LEIGON	Ghana	Clarke 1880	-130	29	364
LIBERIA 1964	Liberia	Clarke 1880	-90	40	88
LUZON	Philippines (Excluding Mindanao)	Clarke 1866	-133	-77	-51
	Philippines (Mindanao)		-133	-79	-72
MAHE 1971	Mahe Island	Clarke 1880	41	-220	-134
MASSAWA	Ethiopia (Eritrea)	Bessel 1841	639	405	60
MERCHICH	Morocco	Clarke 1880	31	146	47
MIDWAY ASTRO 1961	Midway Island	International 1924	912	-58	1227
MINNA	Cameroon	Clarke 1880	-81	-84	115
	Nigeria		-92	-93	122
MONTSERRAT ISLAND ASTRO 1958	Montserrat (Leeward Islands)	Clarke 1880	174	359	365
M'PORALOKO	Gabon	Clarke 1880	-74	-130	42
NAHRWAN	Oman (Masirah Island)	Clarke 1880	-247	-148	369
	Saudi Arabia		-243	-192	477
	United Arab Emirates		-249	-156	381
NAPARIMA BWI	Trinidad & Tobago	International 1924	-10	375	165

DATUM	LOCATION	ELLIPSOID	DX	DY	DZ
NORTH AMERICAN 1927	MEAN FOR Antigua, Barbados, Barbuda, Caicos Islands, Cuba, Dominican Republic, Grand Cayman, Jamaica, Turks Islands	Clarke 1866	-3	142	183
	MEAN FOR Belize, Costa Rica, El Salvador, Guatemala, Honduras, Nicaragua		0	125	194
	MEAN FOR Canada		-10	158	187
	MEAN FOR Continental US (CONUS)		-8	160	176
	MEAN FOR CONUS (East of Mississippi River) Including Louisiana, Missouri, Minnesota		-9	161	179
	MEAN FOR CONUS (West of Mississippi River)		-8	159	175
	Alaska (Excluding Aleutian Islands)		-5	135	172
	Aleutian Islands (East of 180°W)		-2	152	149
	Aleutian Islands (West of 180°W)		2	204	105
	Bahamas (Except San Salvador Island)		-4	154	178
	Bahamas (San Salvador Island)		1	140	165
	Canada (Alberta, British Columbia)		-7	162	188
	Canada (Manitoba, Ontario)		-9	157	184
	Canada (New Brunswick, Newfoundland, Nova Scotia, Quebec)		-22	160	190
	Canada (Northwest Territories, Saskatchewan)		4	159	188
	Canada (Yukon)		-7	139	181
	Canal Zone		0	125	201
	Cuba		-9	152	178
	Greenland (Hayes Peninsula)		11	114	195
	Mexico		-12	130	190
NORTH AMERICAN 1983	Alaska (Excluding Aleutian Islands), Canada, Central America, CONUS, Mexico	GRS 80	0	0	0
	Aleutian Islands		-2	0	4
	Hawaii		1	1	-1
NORTH SAHARA 1959	Algeria	Clarke 1880	-186	-93	310
OBSERVATORIO METER- EO 1939	Azores (Corvo & Flores Islands)	International 1924	-425	-169	81
OLD EGYPTIAN 1907	Egypt	Helmer 1906	-130	110	-13

DATUM	LOCATION	ELLIPSOID	DX	DY	DZ
OLD HAWAIIAN	MEAN FOR Hawaii, Kauai, Maui, Oahu Hawaii Kauai Maui Oahu	Clarke 1866	61 89 45 65 58	-285 -279 -290 -290 -283	-181 -183 -172 -190 -182
OMAN	Oman	Clarke 1880	-346	-1	224
ORD. SURVEY GREAT BRITIAN 1936	MEAN FOR England, Isle of Man, Scotland, Shetland Islands, Wales England England, Isle of Man, Wales Scotland, Shetland Islands Wales	Airy	375 371 371 384 370	-111 -112 -111 -111 -108	431 434 434 425 434
PICO DE LAS NIEVES	Canary Islands	International 1924	-307	-92	127
PITCAIRN ASTRO 1967	Pitcairn Island	International 1924	185	165	42
POINT 58	MEAN FOR Burkina Faso & Niger	Clarke 1880	-106	-129	165
POINTE NOIRE 1948	Congo	Clarke 1880	-148	51	-291
PORTO SANTO 1936	Porto Santo, Madeira Islands	International 1924	-499	-249	314
PROVISIONAL S. AMERICAN 1956	MEAN FOR Bolivia, Chile, Colombia, Ecuador, Guyana, Peru, Venezuela Bolivia Chile (Northern, Near 19°S) Chile (Southern, Near 43°S) Colombia Ecuador Guyana Peru Venezuela	International 1924	-288 -270 -270 -305 -282 -278 -298 -279 -295	175 188 183 243 169 171 159 175 173	-376 -388 -390 -442 -371 -367 -369 -379 -371
PROVISIONAL S. CHILEAN 1963	Chile (South, Near 53°S) (Hito XVIII)	International 1924	16	196	93
PUERTO RICO	Puerto Rico, Virgin Islands	Clarke 1866	11	72	-101
PULKOVO 1942	Russia	Krassovsky 1940	28	-130	-95
QATAR NATIONAL	Qatar	International 1924	-128	-283	22
QORNOQ	Greenland (South)	International 1924	164	138	-189

DATUM	LOCATION	ELLIPSOID	DX	DY	DZ
REUNION	Mascarene Islands	International 1924	94	-948	-1262
ROME 1940	Italy (Sardinia)	International 1924	-225	-65	9
S-42 (PULKOVO 1942)	Hungary	Krassovsky 1940	28	-121	-77
SANTO (DOS) 1965	Espirito Santo Island	International 1924	170	42	84
SAO BRAZ	Azores (Sao Miguel, Santa Maria Islands)	International 1924	-203	141	53
SAPPER HILL 1943	East Falkland Island	International 1924	-355	21	72
SCHWARZECK	Namibia	Bessel 1841 (Namibia)	616	97	-251
SELVAGEM GRANDE 1938	Salvage Island	International 1924	-289	-124	60
SGS 85	Soviet Geodetic System 1985	SGS 85	3	9	-9
S-JTSK	Czechoslovakia (prior to 1 Jan. 1993)	Bessel 1841	589	76	480
SOUTH AMERICAN 1969	MEAN FOR Argentina, Bolivia, Brazil, Chile, Colombia, Ecuador, Guyana, Paraguay, Peru, Trinidad & Tobago, Venezuela	South American 1969	-57	1	-41
	Argentina		-62	-1	-37
	Bolivia		-61	2	-48
	Brazil		-60	-2	-41
	Chile		-75	-1	-44
	Colombia		-44	6	-36
	Ecuador (Excluding Galapagos Islands)		-48	3	-44
	Ecuador (Baltra, Galapagos)		-47	26	-42
	Guyana		-53	3	-47
	Paraguay		-61	2	-33
	Peru		-58	0	-44
	Trinidad & Tobago		-45	12	-33
	Venezuela		-45	8	-33
SOUTH ASIA	Singapore	Modified Fischer 1960	7	-10	-26
TANANARIVE OBSERVATORY 1925	Madagascar	International 1924	-189	-242	-91
TIMBALAI 1948	Brunei, East Malaysia (Sabah, Sarawak)	Everest (Sabah & Sarawak)	-679	669	-48

DATUM	LOCATION	ELLIPSOID	DX	DY	DZ
TOKYO	MEAN FOR Japan, Okinawa, South Korea	Bessel 1841	-148	507	685
	Japan		-148	507	685
	Okinawa		-158	507	676
	South Korea		-146	507	687
TRISTAN ASTRO 1968	Tristan da Cunha	International 1924	-632	438	-609
VOIROL 1960	Algeria	Clarke 1880	-123	-206	219
VITI LEVU 1916	Fiji (Viti Levu Island)	Clarke 1880	51	391	-36
WAKE-ENIWETOK 1960	Marshall Islands	Hough	102	52	-38
WAKE ISLAND ASTRO 1952	Wake Atoll	International 1924	276	-57	149
WGS 1972	Global Definition	WGS 72	0	0	0
YACARE	Uruguay	International 1924	-155	171	37
ZANDERIJ	Surinam	International 1924	-265	120	-358

Appendix 3: Supplied Reference System Parameter Files

Geodetic (Latitude/Longitude)

A single REF file named LATLONG is supplied for geodetic coordinates. This file is based on the WGS84 datum. For other datums, use the COPY function in the IDRISI File Explorer to copy this file to another name and then use Metadata or Edit along with the data in **Appendices 1 and 2** to enter the new datum information.

Universal Transverse Mercator (UTM)

160 REF files are supplied for the UTM system—60 for the northern hemisphere using the WGS84 datum, 60 for the southern hemisphere using the WGS84 datum, 20 for North America based on NAD27 and 20 for North America based on NAD83. The northern WGS84 group has names ranging from UTM-01N to UTM-60N, while the southern WGS84 group has names ranging from UTM-01S to UTM-60S. The NAD27 group (covering zones 1-20) has names ranging from US27TM01 to US27TM20 while those for NAD83 range from US83TM01 to US83TM20. Note that the North American groups all use a North American mean value for the Molodensky constants.

For other datums, the module UTMREF may be used to create the reference system parameter file. In addition, several very specific instances of the UTM system are available in the Miscellaneous group listed later in this appendix.

US State Plane Coordinate System 1927

REF files are supplied for all US State Plane Coordinate Systems based on the Transverse Mercator and Lambert Conformal Conic projections for NAD27 and NAD83. The following table lists these files for the NAD27 datum. The *Projection* column indicates the projection upon which the system is based (L=Lambert Conformal Conic / TM=Transverse Mercator).

State	File name	Title	Projection
Alabama	SPC27AL1	Alabama State Plane Coordinate System Eastern Zone	TM
	SPC27AL2	Alabama State Plane Coordinate Western Zone	TM
Alaska	SPC27AK0	Alaska State Plane Coordinate System Zone 10	TM
	SPC27AK2	Alaska State Plane Coordinate System Zone 2	TM
	SPC27AK3	Alaska State Plane Coordinate System Zone 3	TM
	SPC27AK4	Alaska State Plane Coordinate System Zone 4	TM
	SPC27AK5	Alaska State Plane Coordinate System Zone 5	TM
	SPC27AK6	Alaska State Plane Coordinate System Zone 6	TM

State	File name	Title	Projection
	SPC27AK7	Alaska State Plane Coordinate System Zone 7	TM
	SPC27AK8	Alaska State Plane Coordinate System Zone 8	TM
	SPC27AK9	Alaska State Plane Coordinate System Zone 9	TM
Arizona	SPC27AZ1	Arizona State Plane Coordinate System Eastern Zone	TM
	SPC27AZ2	Arizona State Plane Coordinate System Central Zone	TM
	SPC27AZ3	Arizona State Plane Coordinate System Western Zone	TM
Arkansas	SPC27AR1	Arkansas State Plane Coordinate System Northern Zone	L
	SPC27AR2	Arkansas State Plane Coordinate System Southern Zone	L
California	SPC27CA1	California State Plane Coordinate System Zone I	L
	SPC27CA2	California State Plane Coordinate System Zone II	L
	SPC27CA3	California State Plane Coordinate System Zone III	L
	SPC27CA4	California State Plane Coordinate System Zone IV	L
	SPC27CA5	California State Plane Coordinate System Zone V	L
	SPC27CA6	California State Plane Coordinate System Zone VI	L
	SPC27CA7	California State Plane Coordinate System Zone VII	L
Colorado	SPC27CO1	Colorado State Plane Coordinate System Northern Zone	L
	SPC27CO2	Colorado State Plane Coordinate System Central Zone	L
	SPC27CO3	Colorado State Plane Coordinate System Southern Zone	L
Connecticut	SPC27CT1	Connecticut State Plane Coordinate System Zone 1	L
Delaware	SPC27DE1	Delaware State Plane Coordinate System Zone 1	TM
Florida	SPC27FL1	Florida State Plane Coordinate System Eastern Zone	TM
	SPC27FL2	Florida State Plane Coordinate System Western Zone	TM
	SPC27FL3	Florida State Plane Coordinate System Northern Zone	TM
Georgia	SPC27GA1	Georgia State Plane Coordinate System Eastern Zone	TM
	SPC27GA2	Georgia State Plane Coordinate System Western Zone	TM
Hawaii	SPC27HI1	Hawaii State Plane Coordinate System Zone 1	TM
	SPC27HI2	Hawaii State Plane Coordinate System Zone 2	TM
	SPC27HI3	Hawaii State Plane Coordinate System Zone 3	TM
	SPC27HI4	Hawaii State Plane Coordinate System Zone 4	TM
	SPC27HI5	Hawaii State Plane Coordinate System Zone 5	TM
Idaho	SPC27ID1	Idaho State Plane Coordinate System Eastern Zone	TM
	SPC27ID2	Idaho State Plane Coordinate System Central Zone	TM
	SPC27ID3	Idaho State Plane Coordinate System Western Zone	TM

State	File name	Title	Projection
Illinois	SPC27IL1	Illinois State Plane Coordinate System Eastern Zone	TM
	SPC27IL2	Illinois State Plane Coordinate System Western Zone	TM
Indiana	SPC27IN1	Indiana State Plane Coordinate System Eastern Zone	TM
	SPC27IN2	Indiana State Plane Coordinate System Western Zone	TM
Iowa	SPC27IA1	Iowa State Plane Coordinate System Northern Zone	L
	SPC27IA2	Iowa State Plane Coordinate System Southern Zone	L
Kansas	SPC27KS1	Kansas State Plane Coordinate System Northern Zone	L
	SPC27KS2	Kansas State Plane Coordinate System Southern Zone	L
Kentucky	SPC27KY1	Kentucky State Plane Coordinate System Northern Zone	L
	SPC27KY2	Kentucky State Plane Coordinate System Southern Zone	L
Louisiana	SPC27LA1	Louisiana State Plane Coordinate System Northern Zone	L
	SPC27LA2	Louisiana State Plane Coordinate System Southern Zone	L
	SPC27LA3	Louisiana State Plane Coordinate System Offshore Zone	L
Maine	SPC27ME1	Maine State Plane Coordinate System Eastern Zone	TM
	SPC27ME2	Maine State Plane Coordinate System Western Zone	TM
Maryland	SPC27MD1	Maryland State Plane Coordinate System Zone 1	L
Massachusetts	SPC27MA1	Massachusetts State Plane Coordinate System Mainland Zone	L
	SPC27MA2	Massachusetts State Plane Coordinate System Island Zone	L
Michigan	SPC27MI1	Current Michigan State Plane Coordinate System Northern Zone	L
	SPC27MI2	Current Michigan State Plane Coordinate System Central Zone	L
	SPC27MI3	Current Michigan State Plane Coordinate System Southern Zone	L
	SPC27MI4	Old Michigan State Plane Coordinate System Eastern Zone	TM
	SPC27MI5	Old Michigan State Plane Coordinate System Central Zone	TM
	SPC27MI6	Old Michigan State Plane Coordinate System Western Zone	TM
Minnesota	SPC27MN1	Minnesota State Plane Coordinate System Northern Zone	L
	SPC27MN2	Minnesota State Plane Coordinate System Central Zone	L
	SPC27MN3	Minnesota State Plane Coordinate System Southern Zone	L
Mississippi	SPC27MS1	Mississippi State Plane Coordinate System Eastern Zone	TM
	SPC27MS2	Mississippi State Plane Coordinate System Western Zone	TM
Missouri	SPC27MO1	Missouri State Plane Coordinate System Eastern Zone	TM
	SPC27MO2	Missouri State Plane Coordinate System Central Zone	TM
	SPC27MO3	Missouri State Plane Coordinate System Western Zone	TM
Montana	SPC27MT1	Montana State Plane Coordinate System Northern Zone	L

State	File name	Title	Projection
	SPC27MT2	Montana State Plane Coordinate System Central Zone	L
	SPC27MT3	Montana State Plane Coordinate System Southern Zone	L
Nebraska	SPC27NE1	Nebraska State Plane Coordinate System Northern Zone	L
	SPC27NE2	Nebraska State Plane Coordinate System Southern Zone	L
Nevada	SPC27NV1	Nevada State Plane Coordinate System Eastern Zone	TM
	SPC27NV2	Nevada State Plane Coordinate System Central Zone	TM
	SPC27NV3	Nevada State Plane Coordinate System Western Zone	TM
New Hampshire	SPC27NH1	New Hampshire State Plane Coordinate System Zone 1	TM
New Jersey	SPC27NJ1	New Jersey State Plane Coordinate System Zone 1	TM
New Mexico	SPC27NM1	New Mexico State Plane Coordinate System Eastern Zone	TM
	SPC27NM2	New Mexico State Plane Coordinate System Central Zone	TM
	SPC27NM3	New Mexico State Plane Coordinate System Western Zone	TM
New York	SPC27NY1	New York State Plane Coordinate System Eastern Zone	TM
	SPC27NY2	New York State Plane Coordinate System Central Zone	TM
	SPC27NY3	New York State Plane Coordinate System Western Zone	TM
	SPC27NY4	New York State Plane Coordinate System Long Island Zone	L
North Carolina	SPC27NC1	North Carolina State Plane Coordinate System Zone 1	L
North Dakota	SPC27ND1	North Dakota State Plane Coordinate System Northern Zone	L
	SPC27ND2	North Dakota State Plane Coordinate System Southern Zone	L
Ohio	SPC27OH1	Ohio State Plane Coordinate System Northern Zone	L
	SPC27OH2	Ohio State Plane Coordinate System Southern Zone	L
Oklahoma	SPC27OK1	Oklahoma State Plane Coordinate System Northern Zone	L
	SPC27OK2	Oklahoma State Plane Coordinate System Southern Zone	L
Oregon	SPC27OR1	Oregon State Plane Coordinate System Northern Zone	L
	SPC27OR2	Oregon State Plane Coordinate System Southern Zone	L
Pennsylvania	SPC27PA1	Pennsylvania State Plane Coordinate System Northern Zone	L
	SPC27PA2	Pennsylvania State Plane Coordinate System Southern Zone	L
Puerto Rico & Virgin Islands	SPC27PR1	Puerto Rico & Virgin Islands State Plane Coordinate System Zone 1	L
	SPC27PR2	Puerto Rico & Virgin Islands State Plane Coordinate System Zone 2 (St. Croix)	L
Rhode Island	SPC27RI1	Rhode Island State Plane Coordinate System Zone 1	TM
Samoa	-none-	-not supported-	L
South Carolina	SPC27SC1	South Carolina State Plane Coordinate System Northern Zone	L
	SPC27SC2	South Carolina State Plane Coordinate System Southern Zone	L

State	File name	Title	Projection
South Dakota	SPC27SD1	South Dakota State Plane Coordinate System Northern Zone	L
	SPC27SD2	South Dakota State Plane Coordinate System Southern Zone	L
Tennessee	SPC27TN1	Tennessee State Plane Coordinate System Zone 1	L
Texas	SPC27TX1	Texas State Plane Coordinate System Northern Zone	L
	SPC27TX2	Texas State Plane Coordinate System North Central Zone	L
	SPC27TX3	Texas State Plane Coordinate System Central Zone	L
	SPC27TX4	Texas State Plane Coordinate System South Central Zone	L
	SPC27TX5	Texas State Plane Coordinate System Southern Zone	L
	SPC27UT1	Utah State Plane Coordinate System Northern Zone	L
	SPC27UT2	Utah State Plane Coordinate System Central Zone	L
	SPC27UT3	Utah State Plane Coordinate System Southern Zone	L
Vermont	SPC27VT1	Vermont State Plane Coordinate System Zone 1	TM
Virginia	SPC27VA1	Virginia State Plane Coordinate System Northern Zone	L
	SPC27VA2	Virginia State Plane Coordinate System Southern Zone	L
Washington	SPC27WA1	Washington State Plane Coordinate System Northern Zone	L
	SPC27WA2	Washington State Plane Coordinate System Southern Zone	L
West Virginia	SPC27WV1	West Virginia State Plane Coordinate System Northern Zone	L
	SPC27WV2	West Virginia State Plane Coordinate System Southern Zone	L
Wisconsin	SPC27WI1	Wisconsin State Plane Coordinate System Northern Zone	L
	SPC27WI2	Wisconsin State Plane Coordinate System Central Zone	L
	SPC27WI3	Wisconsin State Plane Coordinate System Southern Zone	L
	SPC27WY1	Wyoming State Plane Coordinate System Eastern Zone	TM
	SPC27WY2	Wyoming State Plane Coordinate System East Central Zone	TM
	SPC27WY3	Wyoming State Plane Coordinate System West Central Zone	TM
	SPC27WY4	Wyoming State Plane Coordinate System Western Zone	TM

US State Plane Coordinate System 1983

REF files are supplied for all US State Plane Coordinate Systems based on the Transverse Mercator and Lambert Conformal Conic projections for NAD27 and NAD83. These are located in the GEOREF subdirectory of your CartaLinx program directory. The following table lists these files for the NAD83 datum. The *Proj* column indicates the projection upon which the system is based (L=Lambert Conformal Conic / TM=Transverse Mercator). Note that Samoa did not make the change from NAD27 to NAD83.

State	File name	Title	Proj
Alabama	SPC83AL1	Alabama State Plane Coordinate System Eastern Zone	TM
	SPC83AL2	Alabama State Plane Coordinate Western Zone	TM
Alaska	SPC83AK0	Alaska State Plane Coordinate System Zone 10	TM
	SPC83AK2	Alaska State Plane Coordinate System Zone 2	TM
	SPC83AK3	Alaska State Plane Coordinate System Zone 3	TM
	SPC83AK4	Alaska State Plane Coordinate System Zone 4	TM
	SPC83AK5	Alaska State Plane Coordinate System Zone 5	TM
	SPC83AK6	Alaska State Plane Coordinate System Zone 6	TM
	SPC83AK7	Alaska State Plane Coordinate System Zone 7	TM
	SPC83AK8	Alaska State Plane Coordinate System Zone 8	TM
	SPC83AK9	Alaska State Plane Coordinate System Zone 9	TM
	SPC83AZ1	Arizona State Plane Coordinate System Eastern Zone	TM
	SPC83AZ2	Arizona State Plane Coordinate System Central Zone	TM
	SPC83AZ3	Arizona State Plane Coordinate System Western Zone	TM
Arkansas	SPC83AR1	Arkansas State Plane Coordinate System Northern Zone	L
	SPC83AR2	Arkansas State Plane Coordinate System Southern Zone	L
California	SPC83CA1	California State Plane Coordinate System Zone I	L
	SPC83CA2	California State Plane Coordinate System Zone II	L
	SPC83CA3	California State Plane Coordinate System Zone III	L
	SPC83CA4	California State Plane Coordinate System Zone IV	L
	SPC83CA5	California State Plane Coordinate System Zone V	L
	SPC83CA6	California State Plane Coordinate System Zone VI	L
Colorado	SPC83CO1	Colorado State Plane Coordinate System Northern Zone	L
	SPC83CO2	Colorado State Plane Coordinate System Central Zone	L
	SPC83CO3	Colorado State Plane Coordinate System Southern Zone	L
Connecticut	SPC83CT1	Connecticut State Plane Coordinate System Zone 1	L
Delaware	SPC83DE1	Delaware State Plane Coordinate System Zone 1	TM
Florida	SPC83FL1	Florida State Plane Coordinate System Eastern Zone	TM
	SPC83FL2	Florida State Plane Coordinate System Western Zone	TM
	SPC83FL3	Florida State Plane Coordinate System Northern Zone	TM
	SPC83GA1	Georgia State Plane Coordinate System Eastern Zone	TM
	SPC83GA2	Georgia State Plane Coordinate System Western Zone	TM

State	File name	Title	Proj
Hawaii	SPC83HI1	Hawaii State Plane Coordinate System Zone 1	TM
	SPC83HI2	Hawaii State Plane Coordinate System Zone 2	TM
	SPC83HI3	Hawaii State Plane Coordinate System Zone 3	TM
	SPC83HI4	Hawaii State Plane Coordinate System Zone 4	TM
	SPC83HI5	Hawaii State Plane Coordinate System Zone 5	TM
Idaho	SPC83ID1	Idaho State Plane Coordinate System Eastern Zone	TM
	SPC83ID2	Idaho State Plane Coordinate System Central Zone	TM
	SPC83ID3	Idaho State Plane Coordinate System Western Zone	TM
Illinois	SPC83IL1	Illinois State Plane Coordinate System Eastern Zone	TM
	SPC83IL2	Illinois State Plane Coordinate System Western Zone	TM
Indiana	SPC83IN1	Indiana State Plane Coordinate System Eastern Zone	TM
	SPC83IN2	Indiana State Plane Coordinate System Western Zone	TM
Iowa	SPC83IA1	Iowa State Plane Coordinate System Northern Zone	L
	SPC83IA2	Iowa State Plane Coordinate System Southern Zone	L
Kansas	SPC83KS1	Kansas State Plane Coordinate System Northern Zone	L
	SPC83KS2	Kansas State Plane Coordinate System Southern Zone	L
Kentucky	SPC83KY1	Kentucky State Plane Coordinate System Northern Zone	L
	SPC83KY2	Kentucky State Plane Coordinate System Southern Zone	L
Louisiana	SPC83LA1	Louisiana State Plane Coordinate System Northern Zone	L
	SPC83LA2	Louisiana State Plane Coordinate System Southern Zone	L
	SPC83LA3	Louisiana State Plane Coordinate System Offshore Zone	L
Maine	SPC83ME1	Maine State Plane Coordinate System Eastern Zone	TM
	SPC83ME2	Maine State Plane Coordinate System Western Zone	TM
Maryland	SPC83MD1	Maryland State Plane Coordinate System Zone 1	L
Massachusetts	SPC83MA1	Massachusetts State Plane Coordinate System Mainland Zone	L
	SPC83MA2	Massachusetts State Plane Coordinate System Island Zone	L
Michigan	SPC83MI1	current Michigan State Plane Coordinate System Northern Zone	L
	SPC83MI2	current Michigan State Plane Coordinate System Central Zone	L
	SPC83MI3	current Michigan State Plane Coordinate System Southern Zone	L
	SPC83MI4	old Michigan State Plane Coordinate System Eastern Zone	TM
	SPC83MI5	old Michigan State Plane Coordinate System Central Zone	TM
	SPC83MI6	old Michigan State Plane Coordinate System Western Zone	TM
Minnesota	SPC83MN1	Minnesota State Plane Coordinate System Northern Zone	L

State	File name	Title	Proj
	SPC83MN2	Minnesota State Plane Coordinate System Central Zone	L
	SPC83MN3	Minnesota State Plane Coordinate System Southern Zone	L
Mississippi	SPC83MS1	Mississippi State Plane Coordinate System Eastern Zone	TM
	SPC83MS2	Mississippi State Plane Coordinate System Western Zone	TM
Missouri	SPC83MO1	Missouri State Plane Coordinate System Eastern Zone	TM
	SPC83MO2	Missouri State Plane Coordinate System Central Zone	TM
	SPC83MO3	Missouri State Plane Coordinate System Western Zone	TM
Montana	SPC83MT1	Montana State Plane Coordinate System Single Zone	L
Nebraska	SPC83NE1	Nebraska State Plane Coordinate System Single Zone	L
Nevada	SPC83NV1	Nevada State Plane Coordinate System Eastern Zone	TM
	SPC83NV2	Nevada State Plane Coordinate System Central Zone	TM
	SPC83NV3	Nevada State Plane Coordinate System Western Zone	TM
New Hampshire	SPC83NH1	New Hampshire State Plane Coordinate System Zone 1	TM
New Jersey	SPC83NJ1	New Jersey State Plane Coordinate System Zone 1	TM
New Mexico	SPC83NM1	New Mexico State Plane Coordinate System Eastern Zone	TM
	SPC83NM2	New Mexico State Plane Coordinate System Central Zone	TM
	SPC83NM3	New Mexico State Plane Coordinate System Western Zone	TM
New York	SPC83NY1	New York State Plane Coordinate System Eastern Zone	TM
	SPC83NY2	New York State Plane Coordinate System Central Zone	TM
	SPC83NY3	New York State Plane Coordinate System Western Zone	TM
	SPC83NY4	New York State Plane Coordinate System Long Island Zone	L
North Carolina	SPC83NC1	North Carolina State Plane Coordinate System Zone 1	L
North Dakota	SPC83ND1	North Dakota State Plane Coordinate System Northern Zone	L
	SPC83ND2	North Dakota State Plane Coordinate System Southern Zone	L
Ohio	SPC83OH1	Ohio State Plane Coordinate System Northern Zone	L
	SPC83OH2	Ohio State Plane Coordinate System Southern Zone	L
Oklahoma	SPC83OK1	Oklahoma State Plane Coordinate System Northern Zone	L
	SPC83OK2	Oklahoma State Plane Coordinate System Southern Zone	L
Oregon	SPC83OR1	Oregon State Plane Coordinate System Northern Zone	L
	SPC83OR2	Oregon State Plane Coordinate System Southern Zone	L
Pennsylvania	SPC83PA1	Pennsylvania State Plane Coordinate System Northern Zone	L
	SPC83PA2	Pennsylvania State Plane Coordinate System Southern Zone	L
Puerto Rico & Virgin Islands	SPC83PR1	Puerto Rico & Virgin Islands State Plane Coordinate System Zone 1	L

State	File name	Title	Proj
	SPC83PR2	Puerto Rico & Virgin Islands State Plane Coordinate System Zone 2 (St. Croix)	L
Rhode Island	SPC83RI1	Rhode Island State Plane Coordinate System Zone 1	TM
Samoa	-none-	-not supported-	L
South Carolina	SPC83SC1	South Carolina State Plane Coordinate System, single zone	L
South Dakota	SPC83SD1	South Dakota State Plane Coordinate System Northern Zone	L
	SPC83SD2	South Dakota State Plane Coordinate System Southern Zone	L
Tennessee	SPC83TN1	Tennessee State Plane Coordinate System Zone 1	L
Texas	SPC83TX1	Texas State Plane Coordinate System Northern Zone	L
	SPC83TX2	Texas State Plane Coordinate System North Central Zone	L
	SPC83TX3	Texas State Plane Coordinate System Central Zone	L
	SPC83TX4	Texas State Plane Coordinate System South Central Zone	L
	SPC83TX5	Texas State Plane Coordinate System Southern Zone	L
Utah	SPC83UT1	Utah State Plane Coordinate System Northern Zone	L
	SPC83UT2	Utah State Plane Coordinate System Central Zone	L
	SPC83UT3	Utah State Plane Coordinate System Southern Zone	L
Vermont	SPC83VT1	Vermont State Plane Coordinate System Zone 1	TM
Virginia	SPC83VA1	Virginia State Plane Coordinate System Northern Zone	L
	SPC83VA2	Virginia State Plane Coordinate System Southern Zone	L
Washington	SPC83WA1	Washington State Plane Coordinate System Northern Zone	L
	SPC83WA2	Washington State Plane Coordinate System Southern Zone	L
West Virginia	SPC83WV1	West Virginia State Plane Coordinate System Northern Zone	L
	SPC83WV2	West Virginia State Plane Coordinate System Southern Zone	L
Wisconsin	SPC83WI1	Wisconsin State Plane Coordinate System Northern Zone	L
	SPC83WI2	Wisconsin State Plane Coordinate System Central Zone	L
	SPC83WI3	Wisconsin State Plane Coordinate System Southern Zone	L
Wyoming	SPC83WY1	Wyoming State Plane Coordinate System Eastern Zone	TM
	SPC83WY2	Wyoming State Plane Coordinate System East Central Zone	TM
	SPC83WY3	Wyoming State Plane Coordinate System West Central Zone	TM
	SPC83WY4	Wyoming State Plane Coordinate System Western Zone	TM

Gauss-Kruger

The Gauss-Kruger reference system is used primarily in the countries of the former Soviet Union and Eastern Bloc. REF

files are included for zones 1-32. All use the Pulkovo 1942 datum and Krassovsky 1940 ellipsoid. The names of these files are GK01_P42.ref through GK32_P42.ref. The Gauss-Kruger projection is identical to the Transverse Mercator projection. (Note that the alternate spelling “Gauss-Krueger” is also acceptable in a REF file.)

Miscellaneous

A small set of additional reference system parameter files has been provided, including:

Location	File Name	Title
United States	ALBERSUS	Alber's Equal Area Conic for Conterminous US
United States	ALBERSUS	Alber's Equal Area Conic for Conterminous US
United States	ALBERSUS	Alber's Equal Area Conic for Conterminous US
United States	ALBERSUS	Alber's Equal Area Conic for Conterminous US
United States	LAZEA	USGS Lambert Azimuthal Equal Area
South Africa	CAPE35	South Africa Universal Transverse Mercator Zone 35
Malawi	MALAWI36	Universal Transverse Mercator Zone 36 for Malawi
Africa	CLABSHA	Clark Labs Hammer-Aitoff Grid for Africa
Bangladesh	BANGTM46	Bangladesh Universal Transverse Mercator Zone 46
Thailand	THAITM47	Thailand Universal Transverse Mercator Zone 47
Vietnam	VIETTM48	Vietnam Universal Transverse Mercator Zone 48

Appendix 4: Error Propagation Formulas

Arithmetic Operations

In the formulas below, S refers to RMS error. Formulas are presented for each of the arithmetic operations performed by the OVERLAY and SCALAR modules in IDRISI. In OVERLAY operations, S_x would refer to the RMS error in Map X, S_y refers to the RMS error in Map Y, and S_z refers to the RMS error in the final map produced, Map Z. In SCALAR operations, K refers to a user-defined constant. Often, error is computed as a uniform value for the entire resulting map. However, in some cases, the formula depends upon the values in corresponding cells of the input maps. These are referred to as X and Y. In these instances, the error would vary over the face of the map and would thus need to be computed separately for each cell. Note that these formulas assume that the input maps are uncorrelated with each other.

Overlay Add / Subtract

$$(\text{e.g., } Z=X+Y \text{ or } Z=X-Y) \quad S_z = \sqrt{S_x^2 + S_y^2}$$

Overlay Multiply / Divide

$$(\text{e.g., } Z=X*Y \text{ or } Z=X/Y) \quad S_z = \sqrt{(S_x^2 \cdot Y^2) + (S_y^2 \cdot X^2)}$$

Scalar Add / Subtract

$$(\text{e.g., } Z=X+k \text{ or } Z=X-k) \quad S_z=S_x \quad \text{i.e., no change}$$

Scalar Multiply

$$(\text{e.g., } Z=X*k) \quad S_z=S_x*k$$

Scalar Divide

$$(\text{e.g., } Z=X/k) \quad S_z=S_x/k$$

Scalar Exponentiate

$$(\text{e.g., } Z=X^k) \quad S_z = \sqrt{k^2 \cdot X^{(2(k-1))} \cdot S_x^2}$$

Logical Operations

For Boolean operations, logical errors may be expressed by the proportion of cells that are expected to be in error (ϵ) in the category being overlaid. Since a Boolean overlay requires two input maps, the error of the output map will be a function of the errors in the two input maps and the logic operation performed as follows:

Logical AND

$$\epsilon_z = \epsilon_x + (1-\epsilon_x)*\epsilon_y \quad \text{or equivalently} \quad \epsilon_z = \epsilon_x + \epsilon_y - (\epsilon_x * \epsilon_y)$$

Logical OR

$$\epsilon_z = \epsilon_x * \epsilon_y$$

Index

Symbols

.ADC Files 56
.AVL Files 56
.MAP Files 58
.MDB Files 56
.RDC File 50
.RST Files 48
.SM0, .SM1, .SM2 Files 59
.SMP Files 59
.SMT Files 59
.VCT Files 53
.VDC Files 54

Numerics

3-D Display 88

A

Absorption 201
Accuracy 129
Accuracy Assessment 36, 112, 205, 206
Active Sensors 22, 229
Adaptive Box Filter 191
Add Layer 66
Advanced Very High Resolution Radiometer (AVHRR) 25, 30, 37, 203, 233
Aerial Photography 27
Aerial Videography 28
Aggregation of Fuzzy Measures 174
Alber's equal area conic projection 304
ALLOCATE 94, 262
Allocation 17
Amplitude 193
Amplitude Image 197
Analysis Menu 89
Analytical Hierarchy Process 153
Analytical Hillshading 103
Anisotropic Cost Distance 255
Anisotropic Function 255, 257, 258
Anisotropy (in variability) 280
API 122
Application Programming Interface 122
Arc/Info Files 87
ARCIDRIS 87
ArcView Files 87
AREA 90
ASCII File Type 50
Ashburn Vegetation Index (AVI) 239
ASPECT 103
ASSIGN 84, 89, 113

Atlas GIS Files 87
ATLIDRIS 87
ATMOSC 188, 223
Atmosphere 22, 187, 223, 229
Atmospheric Correction 187
Atmospheric Windows 22
Attribute Documentation Files 56
Attribute Files 48, 56
AUTOCORR 97
Automatic Display 64
Automatically-Generated Output File Name 43
Autoscaling 62, 63
AVI Files 74, 88
AVIRIS 26, 31, 202, 224

B

Background 71
Backscatter 229
Band Ratio 189
Band Sequential 131
Band-Interleaved-by-Line 131, 222
Band-Interleaved-by-Line (BIL) Files 86
Band-Interleaved-by-Pixel (BIP) Files 86
Basic Probability Assignment (BPA) 179, 214
BAYCLASS 35, 110, 205, 210, 211, 212, 213, 214
BAYES 99
Bayes' Theorum 211
Bayesian Probability Theory 110, 168, 173, 174, 175, 177, 180, 182, 209, 210, 211, 216
BELCLASS 35, 110, 205, 210, 212, 213, 214, 215
BELIEF 99, 111, 180, 181, 215
Belief 168, 173, 179, 180, 213, 214, 215, 218
Belief Interval 179, 180, 213
BILIDRIS 222
Binary File Type 50
BMP Files 72, 87
BMPIDRIS 87
Boolean Combination 148
Boolean Layer 15
Bounding Rectangle 14, 56
BRANCH 122
Bridge and Tunnel Edges 267, 269, 270
BUFFER 93
Buffer Zone 17
Byte Data Type 49

C

CA_MARKOV 101, 252, 253
Calculate 82
CALIBRATE 186, 190
CALL 122
CartaLinx 10, 113, 127, 128
CELLATOM 101, 252

Cellular Automata 101, 252
CENTER 97
Change / Time Series Submenu 100
Change Analysis 100, 247
Change Resolution 133
Change Rows and Columns 133
Change Vector Analysis 248, 251
Chlorophyll 22, 24, 25, 37
Choice Function 147
Choice Heuristic 147
Classification (of remotely-sensed images) 25, 32, 34, 35, 201, 203, 204, 205, 206, 216
Classification Error 206, 207, 208, 209
Classification Uncertainty 205, 210, 211, 213, 214, 222
Classification Uncertainty Image 205, 210
CLUSTER 36, 110, 205, 219, 220, 221
Cluster Analysis 36, 218, 219
Cokriging 281
Collection 46
Collection Linked Zoom 76
Collections 81
Color 23, 24
COLSPACE 38, 107
COM Server 122
Comments Field 53
Complementary Objectives 148, 149, 158
Composer 65, 68, 88
COMPOSITE 88, 107
Composite Generation 34
Composite Image 219
Compromise Solution 149, 158
Computer Assisted Design (CAD) 10
CONCAT 113
Conditional Probability 178
Conditional Simulation 281
Conflicting Objectives 148, 149, 158
Consistency Field 53
Consistency Ratio 154
Constrained Delaunay Triangulation 269
Constraint (in multi-criteria evaluation) 146
Context Operators 18
Context Operators Submenu 95
CONTOUR 104
CONTRACT 113
Contrast 33, 69
CONVERT 112
Convex Hull 267, 275
Conway's Game of Life 252
Coordinate Geometry (COGO) 10
Corrected Transformed Vegetation Index (CTVI) 235
CORRELATE 101, 250
COST 93, 255, 256, 262

Cost Distance 17, 255
COUNT 96, 99
CRATIO 97
Crisp Sets 149, 175
Criterion (in multi-criteria evaluation) 146
Criterion Scores 152
Criterion Weights 152, 163
Critical Point 270, 274
CRLF 86
Crossclassification 248
CROSSTAB 89, 97, 100, 248
Crosstabulation 248
CTG 87
Cursor Inquiry Mode 74
CURVATURE 103
Curve Fitting 102, 103

D

Data Compression 126
Data Entry Menu 113
Data Formats 126
Data Import 125, 129, 130
Data Input 125
Data Integration 133
Data Paths 41, 85
Data Tables 56
Data Type 49, 54, 58
Database 9, 13
Database Development 125
Database Management System 10
Database Query 15, 18, 81, 90
Database Query by Attribute 15, 90
Database Query by Location 15, 92
Database Query Submenu 89
Database Tables 48, 56
Database Uncertainty 149, 167, 169, 171
DATABASE WORKSHOP 114
Database Workshop 81, 90
Datum 136, 285
Datum parameters 285
Datum transformation 137, 143
Decision (definition) 146
Decision Risk 149, 150, 167, 168, 169, 171, 172, 182
Decision Rule 147
Decision Rule Uncertainty 149, 168, 169, 173, 174, 182
Decision Strategy Analysis 145
Decision Support 12, 19
Decision Support Submenu 98
DECOMP 94, 255, 256
Delaunay Triangulation 267
Delphi 123
DEMIDRIS 87

Dempster-Schafer Theory 168, 173, 174, 175, 178, 179, 180, 182
Dempster-Shafer Theory 110, 210, 212, 213, 214, 216, 221, 222
Derivative Mapping 18, 128
DESTRIPER 106, 187
Dialog Box 41
Difference Vegetation Index (DVI) 239
Digital Elevation Model 267
Digital Image Processing 32, 201
Digital Number (DN) 185
Digitizing 10, 127
DISAGGREGATE 101
Disbelief 179, 181
DISPERSE 94, 255, 256, 258, 259
DISPLAY Launcher 88
Display Launcher 61
Display Menu 88
Display System 10, 61
DISTANCE 93
Distance Operators 17
Distance Operators Submenu 93
Distance-Based Vegetation Indices 233, 236
Distance-Weighted Average 263
DLG 87
Documentation File 50, 54
Documentation Files 46
DRAWFILT 108, 196, 198
DXF Files 87
DXFIDRIS 87
E
EDIT 105, 113
Edit 89
EDITSIG 204, 218
Electromagnetic Energy 21, 22, 23, 201, 202
Electromagnetic Spectrum 233
Ellipsoid parameters 283
EMF Files 72, 87
Enhancement (of remotely-sensed images) 32, 33
Environ 41
Environment 41
Erdas Files 87
ERDIDRIS 87
ERRMAT 36, 99, 112, 169, 170, 206
Error 149, 167
Error Matrix 170
Error Propagation 167, 169, 170
Error Propagation Formulas 305
Errors of Commission 170, 206
Errors of Omission 170, 206
Euler's Formula 195
European Remote Sensing Satellite 229
European Remote Sensing Satellite (ERS-1 and ERS-2) 31

EXPAND 113
Export 86
EXTRACT 84, 89, 96
F
Factor (in multi-criteria evaluation) 146
Factor Weights 155
False Color Composite 25, 27
False origin 138
Feature Extraction 104
Feature Properties 72, 75
File Menu 85
File Names 45
File Type 50, 57
FILTER 95, 107, 191, 206, 230
FILTERFQ 108, 196, 198
Filtering 34
FISHER 210
Flag Definition 53
Flag Value 53
Flood Polygon Digitizing 203
Focus 66
Force (in cost distance) 255, 256
FOURIER 108, 196, 198
Fourier Analysis 108, 193
FRACTAL 103
FREQDIST 108, 196, 198
Frequency 193
Frequency Domain 195, 196, 197, 198
Frequency Domain Filter 198
Friction 17
Friction (in cost distance) 255, 256
FUZCLASS 35, 110, 205, 210, 215
FUZSIG 35, 109, 204, 216, 217, 218
FUZZY 99, 152, 160, 176, 179
Fuzzy Measure 173
Fuzzy Membership Functions 176
Fuzzy Partition Matrix 217
Fuzzy Set 147, 149, 150, 152, 167, 168, 173, 174, 175, 182
Fuzzy Set Membership 202, 210, 215, 217, 222
Fuzzy Set Theory 110, 210, 215, 218
Fuzzy Signatures 216, 217
G
Gain 248
Generalization 205, 218, 219
Geodesy 135
Geodetic coordinates 137, 295
Geographic Information System (GIS) 9
Geoid 135
Geometric Restoration 33, 191
GEOREF folder 140

Georeferencing 14, 33, 70, 129, 135–144, 191
Geostatistics 103, 264
GeoTIFF data import 86
Global Positioning System 76, 127, 128, 203
GPS 76
Graphic Insets 71
GRASS Files 87
GRASSIDR 87
Green Vegetation Index (GVI) 242
Grid north 138
Grid referencing system 138
GROUP 95
Group File 46
Gstat 102, 264, 277

H

Hammer-Aitoff grid for Africa 304
Hard Classifiers 35, 109, 202, 204, 206, 221
Hardener 202, 218
Hardening (image classification) 35
Harmonic 195
Haze 188
Heads-up Digitizing 127
Help Menu 114
Help System 44
HILLSHADE 103, 190
HISTO 88, 90, 96
HNTRLAND 96
H-Scatterplot 278
Hybrid Unsupervised/Supervised Classification 220, 221
HYPERABSORB 111, 112, 223, 226
HYPERAUTOSIG 36, 109, 111, 223
HYPERMIN 36, 111, 223, 224, 225
HYPEROSP 36, 111, 223, 225
HYPERSAM 36, 111, 223, 224
HYPERSIG 36, 109, 223
Hyperspectral Classification 202, 222, 224–??, 224, 225, ??–226
Hyperspectral Remote Sensing 26, 35, 202, 203, 222
Hyperspectral Signature File (.HSG) 223, 224
Hyperspectral Signature Library 202, 223, 224
HYPERUNMIX 111, 112, 225
HYPERUNMIX, 223
HYPERUSP 36, 111, 223, 226

I

Icon Bar 39
ID Type 55
Ideal Point 159
Idrisi File Conversion 88
IDRISI File Explorer 64
Idrisi File Explorer 85
Ignorance 175, 178, 179, 180, 181, 212, 213, 214

ILLUMINATE 88
Illumination Effects 189, 191, 234
Illumination Modeling 190
IMAGE CALCULATOR 90, 93, 104
Image Calculator 117, 120
Image Classification 32, 109, 110
Image Deviation 250
Image Differencing 247
Image Enhancement 106
Image Group File 210, 222, 224
Image Partitioning 190
Image Processing Submenu 105
Image Ratioing 247
Image Restoration 106, 185
Image Transformation 107
IMAGEDIFF 100
IMAGERATIO 100
Imaginary Part Image 195, 196
Import 86, 125, 129, 130
Indian Space Research Organization (IRS) 30
INITIAL 113
In-Process Classification Assessment (IPCA) 205, 212, 214, 215, 222
Input File 42
Instantaneous Field of View (IFOV) 28
Integer Data Type 49
Interaction Mechanisms 23, 201
Interactive Display Features 74
INTERCON 102, 264
INTERPOL 102, 263
Interpolation 102, 270, 274, 277, 281
ISOCLUST 36, 110, 219, 221
J
Japanese Earth Resource Satellite 229
Japanese Earth Resource Satellite (JERS) 31
J-Shaped Fuzzy Membership Function 176
K
Kappa 170
Kappa Index of Agreement 249, 253
Kauth-Thomas Tasseled Cap Transformation 242
Kriging 102, 264, 277, 281
Kriging and Conditional Simulation 102, 103
L
Lag 278
Lambert azimuthal equal area projection 304
LANDSAT 131, 233
Landsat data import 86
LANDSAT satellites and data 25, 29, 188, 202, 204
Latitude and longitude 137, 295
LATLONG REF file 295
Launcher 88

Layer 45
Layer Frame 65
Layer Properties 68
Layover Effect (in Radar data) 230
Least-Cost Pathway 17
Legend Categories 53
Legends 70
Limiting Factor 147
Lineage Field 53
Linear Discriminant Analysis Classifier 210
Linear Fuzzy Membership Function 177
Linear Spectral Unmixing 35
Link File 46
Linked-Table Collections 81
LINTOPNT 102, 268
Loadings (in PCA) 249
Logical Operations 15
LOGISTICREG 253
LOGITREG 97
M
Macro 85, 105
Macro Modeler 117
Magnitude/Direction Image Pair 256
MAKESIG 108, 204, 221
Map Algebra 16
Map Analysis Package Files 87
Map Composition 10
Map Composition Files 58
Map Grid 71
Map Layer 45
Map Layer Names 45
Map Properties 70
Map Window 65
MAPIDRIS 87
MapInfo Files 87
MARKOV 101, 251, 252
Markov Chain Analysis 251
Mathematical Operators Submenu 92
MAXBAY 212, 218
MAXBEL 218
MAXFRAC 218
MAXFUZ 218
Maximum Likelihood 209, 221
Maximum Value 52
MAXLIKE 35, 36, 110, 204, 205, 209, 210, 212, 221
MAXSET 35, 36, 110, 205, 214, 219, 221
MCE 98, 152, 154, 163, 170, 176
MDCHOICE 99
Measurement Error 167, 169, 171
Media Viewer 74, 101
Menu 85

Menu System 39
Metadata 46, 85
MIFIDRIS 87
MINDIST 35, 109, 204, 205, 206, 207, 209, 215
Minimum Distance 206
Minimum Value 52
Misra's Green Vegetation Index (MGVI) 243
MIXCALC 35, 111, 214, 215
Mixed Pixels 201, 202, 205, 212, 214, 215, 216, 221, 236
Mixed pixels 35
Model Specification Error 168
Modified Soil-Adjusted Vegetation Index (MSAVI) 241
MODIS 26, 31, 203
Modules 85
MOLA 99, 160, 164, 253
Molodensky transform procedure 143
Molodensky transformation constants 143, 285
Monte Carlo Simulation 171
MOSAIC 187
Motorolla Processor 129
Move 74
Multi-Criteria Evaluation 148, 151, 154, 155, 160, 163, 177
Multi-look Processing 230
Multi-Objective Decision 151, 158
Multi-Objective Evaluation 148, 158, 160
Multi-Objective Land Allocation 151, 159, 160, 164
MULTIREG 96
N
NADCON procedure 143
NADUSLAT 143
NADUSLON 143
NDVICOMP 106
NMEA Format 77
Noise Removal 190, 191, 193, 230
Non-Constrained Delaunay Triangulation 269
NORMALIZE 101
Normalized Difference 37
Normalized Difference Vegetation Index (NDVI) 235
North Arrow 71
Nugget 280
Nyquist Frequency 193, 197
O
Objective 147
Offset 248
On-Screen Digitizing 78, 203, 204
On-screen Digitizing 127
Order Weights 155
Ordered Weighted Average 155, 174, 176
Original Point 272
ORTHO 88

Orthogonal Tranformation Vegetation Indices 234, 242
Orthographic Display 88
Output File 42
OVERLAY 89, 92
Overlay 11
Overwrite Protection 43
P
Packed Binary File Type 50
Pairwise Comparison Matrix 153, 154
Palette Files 59, 62, 88
Palette Files Import 88
PALIDRIS 88
Pan 73, 75
Parallelepiped 204, 208, 209
PARE 86
Passive Sensors 21
PATHWAY 94
PATTERN 95, 96
PCA 38, 107
PCLASS 90, 99, 171, 172
Pearson Product Moment Coefficient of Correlation 101
Pearson Product-Moment Correlation 250
PERIM 90
Perpendicular Vegetation Index (PVI) 236, 237, 238, 239
Phase 193
Philosophy of GIS 20
Photogrammetry 28, 33
Pick List 42
PIPED 35, 109, 204, 208, 218
Placemarks 72, 76
Plane reference system 139
Plate Carrée projection 139
Plausibility 173, 179, 180, 181, 213, 215
PNTGEN 268
Position Error 52
Posterior Probability 178
Potential Model 263
Power Spectrum Image 195, 196, 197
Preferences 85
Principal Components Analysis 25, 38, 187, 188, 242, 249
Print Composition 72
Prioritized Solution 149, 158
Priority 66
Process Modelling 19
PROFILE 90, 100, 223, 224, 250
Profiling 250
Progress Report 40
PROJECT 112, 139, 143
Project 143
Project File 41
Projection 137

Proportional Error 167, 170

Q

QUADRAT 97

QUERY 90

Query by Attribute 90

Query by Location 92

R

Radar 22, 30, 31, 191

RADARSAT 86, 229

RADARSAT (module) 230

RADIANCE 106, 186

Radiance Calibration 185

Radiometric Restoration 30, 32, 185

RANDOM 98, 99, 171

Range 280

RANK 98, 158, 164

Raster 12, 13

Raster Documentation File 50

Raster Files 48

Raster Surface 267, 274, 275

Raster/Vector Conversion Submenu 113

Ratio Vegetation Index (RATIO) 234

Ratio Vegetation Index (RVI) 236

Real Data Type 49

Real Part Image 195, 196

RECLASS 89

Reference ellipsoid 135, 136, 283

Reference surface 135

Reference System 133, 135

Reference system parameter file (REF) 59, 139, 140, 295

Reflectance 23, 24, 201

Reformat Menu 112

REGRESS 96, 190

Regression Differencing 247

RELOCATE 94

Remote Sensing 11, 21, 26, 201

Remove Layer 67

Representative Fraction 75

Representative Fraction (RF) 40

RESAMPLE 33, 106, 112, 191

Resize 74

Resolution 52, 128, 133

Resource Folder 41

Restoration (of remotely-sensed images) 32

Restore Original Window 76

RESULTANT 94, 255, 256

Reverse-Transform PCA 187, 188

RF 40

RGB24 Data Type 50

Risk 155

Risk-aversion (in decision making) 148
ROC 253
Root Mean Square (RMS) Error 167, 171
Run Macro 85, 105
S
SAMPLE 36, 98, 99, 112, 169, 206
Sample Size (for Accuracy Assessment) 206
Sample Variogram Modeling 102, 103
Satellite Imagery 130
Satellite Platforms 28
Satellite Sensor Calibration 185
Saturation Points 69
Save Composition 72
SCALAR 93
Scale Bar 71
Scale of Display 40
Scan Line Drop Out 187, 191
Scanner 10, 127
Scanner Distortion Correction 191
SCATTER 109
SCREEN 223
Screen Dump 72
SDTS data import 87
Semivariogram 279
Sensor Band File (.SBF) 222, 223, 224
SEPARATE 88
SEPSIG 109
SHAPEIDR 87
SIGCOMP 109, 204
Sigmoidal Fuzzy Membership Function 176
Signature (in Remote Sensing) 23, 24, 26, 34, 201, 202, 204
Signature Development 108, 204, 216, 217, 222, 223
Signature File (.SIG) 204, 217
Signature Pixel File (.SPF) 204
Sill 280
Simulation 281
Skew Correction 191
SLOPE 103
Slope-Based Vegetation Indices 233, 234
Soft Classifiers 110, 202, 204, 205, 210
Soft Constraint 147
Soil Background 233
Soil Line 234, 236
Soil-Adjusted Vegetation Index (SAVI) 239
Spatial Continuity 277
Spatial Domain 197, 198
Spatial Resolution 28
Spatial Variability 277
SPDIST 93
Specification Error 149
Speckle 190

Spectral Angle Mapper 224
Spectral Curve File (.ISC) 224
Spectral Resolution 28
Spectral Response Pattern 23, 24, 34, 201, 224
SPOT data import 86
Spreadsheet Files 86
SQL 81, 82
SRFIDRIS 87
SSTIDRIS 86
STANDARD 98, 99
Standardization of Factors 160
Standardized Distance (in Classification) 207
State plane coordinate system 295
Statistical Analysis 11
Statistics Submenu 96
Status Bar 40
STCHOICE 101, 251
STRETCH 88, 107
Striping 186, 190, 191
Sub-Pixel Classification 35, 202, 212, 214
Supervised Classification 35, 108, 201, 203
SURFACE 95, 170
Surface 261
Surface Analysis Submenu 102
Surface Interpolation 102, 267, 274
Surface Model 261
Surfer Files 87
Surrogate Data 128
Surveying 127
Symbol Files 59, 62, 88
Symbol Workshop 73
Synthetic Aperture Radar 229
System Operation 39
System Pour L'Observation de la Terre (SPOT) 29, 131, 233
T
TASSCAP 38, 108, 233, 242, 244
Tasseled Cap 234, 243
Terminate 40
Tessellation 262
Text Inset 71
TEXTURE 38, 95, 108, 230, 231
Texture Analysis 230, 231
Thematic Mapper 131
THERMAL 38, 108
Thiam's Transformed Vegetation Index (TTVI) 235
THIESSEN 94, 103, 262
TIFF Files 87
TIFIDRIS 87
Time Series Analysis 38, 100, 249
TIN 102, 263, 265
TIN (module) 274

TINPREP 102, 268
TINSURF 102, 265, 274, 275
Titles 71
T-Norm/T-CoNorm 173, 174
Topographic Effects 189, 234
Topographic Variables 103
TOPOSHAPE 104
Tradeoff 155
Tradeoff (in multi-criteria evaluation) 147, 148
Training Sites 35, 202, 203, 204, 205, 206, 212, 214, 216, 217
TRANSFORM 93
Transformation (of remotely-sensed images) 32, 37
Transformed Soil-Adjusted Vegetation Index (TSAVI) 240
Transformed Vegetation Index (TVI) 235
Transmission (in Remote Sensing) 201
TRANSPOSE 113
TREND 97, 103, 262
Trend Surface Analysis 262
Triangulated Irregular Networks 263, 267, 270
True origin 138
TSA 38, 100, 249
U
Uncertainty 145, 149, 150, 167, 168, 179
Uncertainty Management 145
Universal Transverse Mercator (UTM) 295
UNMIX 35, 111, 205, 210, 211, 218
Unsupervised Classification 35, 36, 201, 218
UPDATE 113
US State plane coordinate system 295
User Preferences 85
UTMREF 113
V
VALIDATE 253
Value Error 52
Value Error Field 170, 171
Value Units 52
Values File 84
Values Files 48, 56
VAR2FIX 86
VARCOST 94, 256, 257, 258, 259
Vector 12, 13
Vector Documentation File 54
Vector Files 53
Vector Layer Type 68
Vegetation 233
Vegetation (in Remote Sensing) 201, 202
Vegetation Index 233
Vegetation Indices 37
VEGINDEX 38, 108, 233, 244
Video 74

VIEWSHED 95
Visibility of Layers 68
Visual Basic 123
Visual C++ 123
Voronoi Tessellation 262
W
WATRSHED 95
WAV Files 88
WAVES Subdirectory 223
Waypoints 77
WEIGHT 98, 154, 163
Weighted Difference Vegetation Index (WDVI) 241
Weighted Linear Combination 148, 154, 174
Wheeler-Misra Orthogonal Transformation 242, 243
WINDOW 112
Window List 114
WMF Files 72, 87
Working Folder 41
X
XYZIDRIS 86
Z
ZEROPAD 108, 196
Zoom 73, 75, 76