INFORMATION TO USERS

This material was produced from a microfilm copy of the original document. While the most advanced technological means to photograph and reproduce this document have been used, the quality is heavily dependent upon the quality of the original submitted.

The following explanation of techniques is provided to help you understand markings or patterns which may appear on this reproduction.

1. The sign or “target” for pages apparently lacking from the document photographed is “Missing Page(s)”. If it was possible to obtain the missing page(s) or section, they are spliced into the film along with adjacent pages. This may have necessitated cutting thru an image and duplicating adjacent pages to insure you complete continuity.

2. When an image on the film is obliterated with a large round black mark, it is an indication that the photographer suspected that the copy may have moved during exposure and thus cause a blurred image. You will find a good image of the page in the adjacent frame.

3. When a map, drawing or chart, etc., was part of the material being photographed the photographer followed a definite method in “sectioning” the material. It is customary to begin photoing at the upper left hand corner of a large sheet and to continue photoing from left to right in equal sections with a small overlap. If necessary, sectioning is continued again — beginning below the first row and continuing on until complete.

4. The majority of users indicate that the textual content is of greatest value, however, a somewhat higher quality reproduction could be made from “photographs” if essential to the understanding of the dissertation. Silver prints of “photographs” may be ordered at additional charge by writing the Order Department, giving the catalog number, title, author and specific pages you wish reproduced.

5. PLEASE NOTE: Some pages may have indistinct print. Filmed as received.

Xerox University Microfilms
300 North Zeeb Road
Ann Arbor, Michigan 48106
LEIGHTY, Robert Dwaine, 1930-
A LOGICAL APPROACH TOWARDS TERRAIN PATTERN RECOGNITION FOR ENGINEERING PURPOSES.
The Ohio State University, Ph.D., 1973
Engineering, civil

University Microfilms, A XEROX Company, Ann Arbor, Michigan

© 1973
ROBERT DWAINLEIGHTY
ALL RIGHTS RESERVED
A LOGICAL APPROACH TOWARDS TERRAIN
PATTERN RECOGNITION FOR
ENGINEERING PURPOSES

DISSERTATION
Presented in Partial Fulfillment of the Requirements for
the Degree Doctor of Philosophy in the Graduate
School of the Ohio State University

By
Robert Dwaine Leighty, B.S.C.E., M.S.C.E.

The Ohio State University
1973

Reading Committee:

Robert B. Lackey

C.E. Wann

Approved by

Glen W. Dyer
Adviser
Department of
Civil Engineering

C.E. Wann
Adviser
Department of
Electrical Engineering
A LOGICAL APPROACH TOWARDS TERRAIN
PATTERN RECOGNITION FOR
ENGINEERING PURPOSES

By
Robert Dwaine Leighty, Ph.D.
The Ohio State University, 1973
Professors O. W. Mintzer and C. E. Warren, Advisers

Automated pattern recognition techniques have predominantly ex­
ploited the statistical nature and the two-dimensional shape character­
istics of photographic tone patterns. For a given terrain-related
engineering problem, however, qualitative information is manually de­
duced mainly from geomorphological inferences drawn from pattern
analysis of surface configuration, and only secondarily from tone
inferences. This study has developed a logical approach towards
terrain pattern recognition for engineering purposes which models the
approach used by the engineer terrain analyst to derive information
from stereoscopic aerial imagery for the solution of his terrain-
related engineering problems. To facilitate the development and
communication of concepts used by the terrain analyst in matters re­
lated to terrain-related engineering problems and terrain organization,
a formalism in the language of predicate calculus is employed. These
concepts then form the base for development of the logical approach to
terrain pattern recognition and mapping which involves problem organ­
ization, terrain pattern description, strategy selection, and boundary
mapping. A simulated terrain pattern recognition and mapping problem
is developed and discussed in detail to illustrate the logical approach
on a digital computer system. To demonstrate the applicability of this approach to a practical problem, real terrain elevation and photographic tone data is used with essentially the same computer program to produce a coded map indicative of that produced by the engineer analyst as a part of his solution to a terrain-related engineering problem from the study of aerial imagery.
ACKNOWLEDGEMENTS

I gratefully acknowledge the assistance of my advisers: Professor O.W. Mintzer, Civil Engineering Department, Professors C.E. Warren and R.B. Lackey, Electrical Engineering Department, The Ohio State University; the support of the U.S. Army Cold Regions Research and Engineering Laboratory, Hanover, New Hampshire, and the U.S. Army Engineer Topographic Laboratories, Ft. Belvoir, Virginia, and my colleagues in the Photographic Interpretation Research Division; and the unselfish cooperation of my family.
VITA

1930 ................. Born - Johnstown, Pennsylvania

1952 ................. B.S.C.E., Virginia Military Institute, Lexington, Virginia


1954-1956 ............. Research Assistant, Joint Highway Research Project, Purdue University, Lafayette, Indiana

1956 ................. M.S.C.E., Purdue University, Lafayette, Indiana


1961-1970 ............. Research Civil Engineer, Photographic Interpretation Research Division, U.S. Army Cold Regions Research and Engineering Laboratory, Hanover, New Hampshire

1967-1968 ............. Graduate Student, The Ohio State University, Columbus, Ohio

1970 ................. Assistant Chief, Photographic Interpretation Research Division, U.S. Army Engineer Topographic Laboratories, Fort Belvoir, Virginia

REGISTRATION

Professional Engineer - Virginia, 1952

Professional Engineer - New Hampshire, 1969

PUBLICATIONS


iii


Contributor, "Photographic Interpretation in the Arctic and Subarctic." Proc. First International Conference on Permafrost, Session 7, Exploration and Site Selection, Purdue University, 1963.


"Remote Sensing for Engineering Investigation of Terrain - Introduction."

"Remote Sensing for Engineering Investigation of Terrain - Radar Systems."

FIELDS OF STUDY

Major Fields: Civil Engineering and Electrical Engineering

Studies in Engineering Photographic Interpretation. Professor Olin W. Mintzer

Studies in Communication Theory. Professor C. Earl Warren, Professor Robert B. Lackey, and Professor William Davis

Studies in Coherent Optics. Professor Stewart A. Collins
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Acknowledgments</th>
<th>ii</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vita</td>
<td>iii</td>
</tr>
<tr>
<td>List of Tables</td>
<td>viii</td>
</tr>
<tr>
<td>List of Figures</td>
<td>ix</td>
</tr>
<tr>
<td>Chapter</td>
<td></td>
</tr>
<tr>
<td>I. Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Statement of the Problem</td>
<td></td>
</tr>
<tr>
<td>1.2 Approach to the Problem</td>
<td></td>
</tr>
<tr>
<td>1.3 Terrain Analyst Model</td>
<td></td>
</tr>
<tr>
<td>1.4 Dissertation Organization</td>
<td></td>
</tr>
<tr>
<td>II. Remote Sensing Data and Terrain Pattern</td>
<td>5</td>
</tr>
<tr>
<td>Recognition Techniques</td>
<td></td>
</tr>
<tr>
<td>2.1 Remote Sensing Data</td>
<td></td>
</tr>
<tr>
<td>2.2 Qualitative Terrain Pattern Recognition</td>
<td></td>
</tr>
<tr>
<td>2.3 Statistical Pattern Recognition</td>
<td></td>
</tr>
<tr>
<td>2.4 Linguistic Pattern Recognition</td>
<td></td>
</tr>
<tr>
<td>2.5 Logical Pattern Recognition Approach</td>
<td></td>
</tr>
<tr>
<td>III. Symbolic Logic</td>
<td>31</td>
</tr>
<tr>
<td>3.1 Nature of Symbolic Logic</td>
<td></td>
</tr>
<tr>
<td>3.2 Basic Concepts and Definitions</td>
<td></td>
</tr>
<tr>
<td>3.3 Computer Programming of Symbolic Logic</td>
<td></td>
</tr>
<tr>
<td>IV. Terrain-Related Engineering Problems</td>
<td>53</td>
</tr>
<tr>
<td>4.1 Civil Engineering and the Landscape</td>
<td></td>
</tr>
<tr>
<td>4.2 Engineering Problems</td>
<td></td>
</tr>
<tr>
<td>4.3 Characterization of Terrain-Related Problems</td>
<td></td>
</tr>
<tr>
<td>V. Terrain Organization</td>
<td>87</td>
</tr>
<tr>
<td>5.1 Characterization of Geomorphic Processes and Terrain Materials</td>
<td></td>
</tr>
<tr>
<td>5.2 Terrain Classification</td>
<td></td>
</tr>
<tr>
<td>VI. Problem Organization</td>
<td>115</td>
</tr>
<tr>
<td>6.1 Preliminary Problem Organization</td>
<td></td>
</tr>
</tbody>
</table>
6.2 Applied Terrain Description Formulation
6.3 Mapping Strategies

VII. SIMULATED LOGICAL TERRAIN PATTERN RECOGNITION
AND MAPPING PROBLEM ........................................146

7.1 Simulation Problem
7.2 Simulation Data
7.3 Development of Pattern Recognition and
Mapping Strategies
7.4 Implementation of Pattern Recognition
and Mapping Strategies
7.5 LPRP Results

VIII. APPLICATION OF LOGICAL TERRAIN PATTERN RECOGNITION
PROGRAMMING TO REAL DATA ..............................195

8.1 Example Terrain Pattern Recognition Problem
8.2 Discussion of Example Results

IX. CONCLUSIONS AND RECOMMENDATIONS ..................205

9.1 Conclusions
9.2 Recommendations

REFERENCES..........................................................211

APPENDIX..........................................................216
LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Operations and Rules for Propositional Calculus</td>
<td>36</td>
</tr>
<tr>
<td>3.2</td>
<td>Comparison of Dissertation Notation and FORTRAN Symbolism</td>
<td>49</td>
</tr>
<tr>
<td>4.1</td>
<td>Examples of Terrain-Related Projects and Problems</td>
<td>55</td>
</tr>
<tr>
<td>6.1</td>
<td>Example of Problem Definition</td>
<td>118</td>
</tr>
<tr>
<td>7.1</td>
<td>Outline for Development and Implementation of Pattern Recognition and Mapping Strategies and Programs</td>
<td>149</td>
</tr>
<tr>
<td>7.2</td>
<td>Glossary of Major Terms Used in LPRP</td>
<td>174</td>
</tr>
</tbody>
</table>
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>The Overlay Nature of Study Area Data Matrices and the Definition of 'Spectral Vector'</td>
<td>8</td>
</tr>
<tr>
<td>2.2</td>
<td>Pattern Description with a Structural Graph and a Set of Grammar Rules</td>
<td>27</td>
</tr>
<tr>
<td>3.1</td>
<td>Computer Flowcharts Representing FORTRAN Computer Statements Associated with Logic Statements</td>
<td>51</td>
</tr>
<tr>
<td>4.1</td>
<td>Examples of Partitioned Area Graphs and Associated Maps Representing Results from Selective Area Partitioning at Different Levels</td>
<td>68</td>
</tr>
<tr>
<td>4.2</td>
<td>Matrices Associated with Producing BCM(m,n,z), the Boundary Coded Matrix</td>
<td>74</td>
</tr>
<tr>
<td>4.3</td>
<td>Composite Attribute Map</td>
<td>76</td>
</tr>
<tr>
<td>4.4</td>
<td>Schema for Partitioning Areas of a Selected ATN&lt;sub&gt;1&lt;/sub&gt; from CMM(m,n,z) for N Attributes AT&lt;sub&gt;1&lt;/sub&gt;</td>
<td>80</td>
</tr>
<tr>
<td>5.1</td>
<td>Hierarchy of Rock Classification Based Upon Geological Origin with Example Rock Types</td>
<td>92</td>
</tr>
<tr>
<td>6.1</td>
<td>Schematic Flowchart Associated with Solution of the Problem Stated in Table 6.1</td>
<td>120</td>
</tr>
<tr>
<td>6.2</td>
<td>Terrain Description Organization for Terrain-Related Engineering Problems</td>
<td>121</td>
</tr>
<tr>
<td>6.3</td>
<td>Terrain Pattern Description Hierarchy</td>
<td>124</td>
</tr>
<tr>
<td>6.4</td>
<td>A Possible Study Area Pattern Description with a Structural Graph and a Set of Grammar Rules</td>
<td>129</td>
</tr>
<tr>
<td>6.5</td>
<td>Two Examples of Contiguity Matrices with Associated Transition Graphs and a Possible Map for Each</td>
<td>131</td>
</tr>
<tr>
<td>6.6</td>
<td>A Simplification of the Structural Graph and Grammar Rules from those of Fig. 6.4</td>
<td>135</td>
</tr>
<tr>
<td>6.7</td>
<td>Modified Map, Structural Graph, and Grammar Rules for Figs. 6.4 and 6.6</td>
<td>136</td>
</tr>
<tr>
<td>6.8</td>
<td>Definitions Associated with Fig. 6.7</td>
<td>139</td>
</tr>
</tbody>
</table>
6.9 Flowchart Associated with a Basic Set of
Operation and Decision Rules of a
Mapping Strategy ................................................. 143
7.1 Model Map Showing Spatial Distribution of Landforms ...... 148
7.2 Spatial Elevation Matrix, IZ(IX, IY) ............................ 151
7.3 Spatial Tone Matrix, IT2(IX, IY) ............................... 152
7.4 Spatial Tone Matrix, IT3(IX, IY) ............................... 153
7.5 Landform Contiguity Matrix IC(I, J) and Terrain
Materials-Landform Implication Matrix MTLI(K, J) .......... 158
7.6 Algorithm for Simulation Model Start State Location
and Verification ......................................................... 160
7.7 Flowchart for Preliminary Boundary Mapping
Algorithm and Sequential Stages of its
Application to Produce MAP(IX, IY) ......................... 163
7.8 Material Classifications from a Partitioning of a
Plot of Spectral Tone Vectors ................................. 167
7.9 Material Classification Map, MCL(IX, IY) ...................... 168
7.10 In-Context Algorithmic Definition of TERRACE
as Used in the Boundary Mapping Algorithm ............... 171
7.11 Flowchart of Main Program, LPRP ............................ 175
7.12 Flowchart of SUBROUTINE STRTS ............................ 177
7.13 Generalized Flowchart for SUBROUTINE LFCMP ............ 179
7.14 Detailed Flowchart for SUBROUTINE LFCMP ................. 180
7.15 Generalized Flowchart for SUBROUTINE MOVE ............. 182
7.16 Movement Indicator Weights IA(I, J) and Examples
of SUBROUTINE MOVE Applied to Selected
MFLD(I, J) Inputs .................................................... 183
7.17 Generalized Flowchart for SUBROUTINE SMSRCH ............ 185
7.18 Program Branching in SUBROUTINE SMSRCH ................ 187
7.19 Generalized Flowchart for SUBROUTINE LFCLAS ............ 188
7.20 Generalized Flowchart for SUBROUTINE TCLAS ............ 190
7.21 MAP(IX, IY) Output of LPRP for Input Data in
Fig. 7.2 through 7.4 ............................................ 191
7.22 Stages of Boundary Mapping of MAP(IX, IY) ................. 192
7.23 Final IBMAP(IX, IY) .......................................... 193
8.1 Photograph of \( IZ(IX,IY) \) Expanded to 256x256 and Quantized to 64 Shades of Grey .................. 197

8.2 Photograph of \( IT1(IIX,IIY) \) as Sampled in a 256x256 Matrix and Quantized to 64 Shades of Grey ........................................................... 199

8.3 Generalized Flowchart of PROGRAM LPRPWR ........................................... 201

8.4 Photograph of MAP(IIX,IIY), the Results of PROGRAM LPRPWR ......................... 203
CHAPTER I
INTRODUCTION

1.1 Statement of the Problem

Until recently terrain pattern recognition from aerial imagery has been accomplished solely by manual, qualitative techniques. The introduction of automated approaches has predominantly exploited the statistical nature and two-dimensional shape characteristics of photographic tone patterns. These have been useful in the military, forestry, agriculture and other disciplines where photographic tone patterns are directly related to the surface terrain patterns of interest. For a given terrain-related engineering problem, however, qualitative information is manually derived mainly from geomorphological inferences drawn from pattern analysis of surface configuration, and only secondarily from tone inferences. Clearly, automation of terrain pattern recognition for engineering purposes requires a new approach.

1.2 Approach to the Problem

This study presents an initial approach to modeling qualitative terrain pattern recognition techniques in a manner which could lead to automated or interactive applications for problem oriented engineering purposes. The study has two goals. The first goal is to develop a logical formalism of concepts used by the engineering image analyst in obtaining information from aerial imagery. The formalisms, in the language of predicate calculus, pertain to terrain-related engineering problems, and terrain organization, and are required to facilitate communication in a common formal language. The formalisms relate spatial
elevation and spectral tone data to terrain information and allow pattern definition in terms of variables and relations.

The second goal is to develop the logical approach to terrain pattern recognition for engineering purposes. The approach models the engineer analyst's terrain pattern recognition strategies for problem organization, terrain description, strategy selection and boundary mapping. A simulation terrain pattern recognition problem is discussed in detail to illustrate the approach on a digital computer system and the results of a pattern recognition problem with real-world data is presented to demonstrate the approach.

1.3 Terrain Analyst Model

How does a rational engineer obtain information from aerial imagery for aid in the solution of his terrain-related problems? What aspects of an engineering problem are related to the terrain? How does the analyst organize terrain for his needs? What aspects of an aerial image are related to solutions of engineering terrain-related problems? How is the image data precessed of yield the required information? The aim of the terrain analyst model is to take steps toward answering these and other related questions. Our guiding assumption is that these questions can be satisfactorily answered only by a formalism which addresses various aspects of the terrain pattern recognition activities. We use the term 'model' as a heuristic device in the sense of 'theory' or 'picture', rather than the sense of 'realization'. The important aspects of our model are the definitions; for heuristic purposes we regard these as hanging together to form a theory, and hence a model, of terrain pattern recognition.

No claim is made at this stage that our model is a good picture of actual behavior; it is certainly not exhaustive. The intent is to present the major concepts in a more definitive manner than previously
available. These concepts have been selected from two relatively distinct areas of the analyst's mental activities: problem organization, and terrain organization. Though automated or semi-automated realization of the logical approach involves explicit formulation of activities related to pattern recognition, activities related to problem and terrain organization provide the implicit context for terrain pattern recognition. Without this context, terrain pattern recognition is but a blind application of one or more irrational pattern recognition techniques.

Realization of the logical approach to terrain pattern recognition in terms of computer output for a given problem may be considered rather crude at this stage. This should not detract from the formalism of the terrain analyst activities, but rather indicates present deficiencies in adapting a discrete system to perform tasks which may be complex for humans. The power of a digital computer system in our logical pattern recognition approach lies in the capability of performing an enormous number of processing and decision operations in a relatively short time period. Pattern recognition and delineation tasks accomplished through heuristic and intuitive processes by the analyst must be explicitly defined in proper sequence for effective programming on a digital computer. The simulated terrain pattern recognition and delineation problem presented in Chapter 7 not only illustrates the automated approach but when interpreted with the data by sequentially following operations and decisions through the source program, one obtains an appreciation for the mental and physical processes accomplished by the human and disdain for quick acceptance of approximate analogue solutions. In this sense, the analyst can profit by studying the computer operations of the simulation program.
1.4 Dissertation Organization

The dissertation is organized in the following manner. Chapter 2 reviews qualitative and statistical terrain pattern recognition techniques from aerial imagery and introduces the logical terrain pattern recognition approach, after first discussing the two basic types of data in which terrain patterns are embedded. Chapter 3 introduces the basic concepts of symbolic logic from which subsequent formalisms will be developed. Chapters 4 and 5 respectively, develop the formalism for terrain-related engineering problems, and terrain organization. In Chapter 6 problem organization is discussed. Here an approach to applied terrain description formulation and recognition strategies are presented for a practical terrain pattern recognition system. A simulated logical terrain pattern recognition and mapping computer program is presented in Chapter 7 which simply illustrates the concepts of Chapter 6. Then in Chapter 8 an example problem with real-world data is presented to demonstrate the method. Finally, in Chapter 9 the conclusions and recommendations of the study are presented.
CHAPTER II
REMOTE SENSING DATA AND TERRAIN
PATTERN RECOGNITION TECHNIQUES

The purpose of this chapter is to review important concepts from the qualitative and quantitative approaches to terrain pattern recognition and to introduce the approach to be used in this study. Particular attention is given to qualitative terrain pattern recognition for it is a general objective of this dissertation to model aspects of this approach. The nature of statistical terrain pattern recognition techniques are outlined and precautions are given for their use with aerial sensor data. Then, as a means to introduce the dissertation approach, aspects of linguistic pattern recognition applicable to pattern description and pattern recognition languages are discussed. The last section contains an introduction to our synthesized pattern recognition approach which utilizes concepts from qualitative, statistical, and linguistic pattern recognition.

We must preface the discussion of pattern recognition techniques with a discussion of aerial remote sensing data.

2.1 Remote Sensing Data

Success of a pattern recognition technique is highly dependent upon the data in which the patterns are embedded. We will consider our data to be contained in vertical images of the terrain obtained with some aerial sensing system. Our interest is not in the image data per se, but rather in the terrain-related information which can be inferred from patterns in the data for aid in the solution of our terrain-
related engineering problems. And in this sense, the information derivable from an aerial image of the terrain is ubiquitous for it applies to the "meaning" of the image pattern and pattern relationships. In general it can be said for image data that what are one man's patterns are another man's noise. In a later chapter we will define "pattern" in a rigorous manner, but for our purposes here let us consider a pattern to be an assemblage of identifiable attributes in the image or data. Thus if we are to recognize our patterns of interest we must be able to identify pattern attributes in the data. Our prime interest is in those patterns which represent terrain objects or their conditions, however we must realize that the sensor system and environment can affect the data and distort the attributes and patterns. Additionally we must acknowledge distortions in our use of the data.

In general, we will speak of multidimensional spatial data, whether it be our primary data or image, or our secondary data sampled from the image. Our data will be intrinsically spatial in two orthogonal directions. Upon this two-dimensional data base we can then superimpose our multidimensional data. This multidimensional data will be of two classes: spectrally coded data and elevation data. The spectrally coded data will consist of one or more images from different portions of the electromagnetic spectrum, while the elevation data can be considered as a single image.

We will now discuss the nature of these two classes of data. For this discussion we will consider the images to be primary data, though in reality these are also samples, while the samples obtained from the images will be secondary data.

2.1.1 **Spectrally Coded Spatial Data**

Many types of aerial sensors can be used to image an area of the earth's surface. For example, there are camera systems using photographic emulsions with differing spectral sensitivities, i.e.,
panchromatic, panchromatic infrared, color, and color infrared, and each emulsion can be used with any one of a number of band pass filters which can serve to shape the system response, and thus spectrally code the primary images. We will use the terms "multiband photography" or "multiband imagery" in referring to the primary data or images produced by photographic systems in different spectral bands. The term "multispectral imagery" is more general and also includes scan imagery from any portion of the electromagnetic spectrum, e.g., radar, passive microwave, and thermal infrared imagery. This study will focus on multiband photography, however the approach can be extended to use of multispectral data.

In the general case of secondary data obtained from a photographic image, sample data values are measured at intersection points of an imaginary regular grid superimposed on the image. In a multiband system the same grid is registered to each of the images. Thus \((x,y)\) represents the spatial coordinates in the arrays of size \(x=1,X\) by \(y=1,Y\). Each band of the multiband camera system has a different spectral transfer function; thus each of the \(I\)-number of spatial arrays in the spectral matrix has a different spectral coding. For example, the spectral matrix for the terrain which was imaged may have been obtained from measuring the optical transmission for each array coordinate point in three images resulting from three film-filter systems. A three element spectral vector can be constructed for any selected matrix point. Thus the spectral vector could be an ordered triple \((S_1(x_0,y_0),S_2(x_0,y_0),S_3(x_0,y_0))\) with \((x_0,y_0)\) representing a fixed spatial coordinate. Fig. 2.1 shows the general spectral vector with \(I\) elements.

The purpose of multiband and multispectral imagery is to enable recognition of natural and cultural terrain objects or materials and conditions by spectral signatures related to the vector elements at
Spectral vector = \{ S_1(x_0, y_0), S_2(x_0, y_0), \ldots, S_I(x_0, y_0) \}
for matrix point \((x_0, y_0)\)

Fig. 2.1 The overlay nature of study area data matrices and the definition of 'spectral vector'.
any point of the image. The rationale is that an I-dimensional measurement space for all values of $S_j(x,y)$ for $i=1,I$, $x=1,X$ and $y=1,Y$ can be partitioned by discriminant functions related to spectral signatures of known terrain materials of interest. Simple as this may seem there are many possibilities for ambiguity and error if care is not exercised. This will be addressed presently.

The image or primary data must be considered as sample data. That is, the general nature of spectral sampling is due to the spectral sensitivity of the photographic emulsion-filter combination and the electronic detector sensitivity. The images are also samples from the temporal and spatial viewpoints. For example, the images are records for natural and physical processes existing at the time of imaging. Many of the natural and physical processes are time variant; some processes are quasi-periodic, e.g., diurnal solar illuminance or seasonal variation of vegetation and snow cover; some processes are aperiodic, e.g., variation of soil moisture with time for a given terrain material or variation of atmospheric transmission properties with time; while yet other natural and physical processes may be characterized by impulse-like processes, e.g., clearing vegetation, damming a river, flooding, or cloud shadowing. Temporal sampling effects on terrain information derived from imagery must be considered when using any spectral image, but is of prime importance when using images obtained at different exposure dates.

Aerial images can also be considered as spatial samples because the imaged terrain will be spatially distorted, i.e., the spatial patterns will not have their correct geometric shape or spatial position. A planimetrically correct image is an orthophotograph; each point in the image is positioned as if it were the nadir point of the image. Camera geometry and perspective of the three dimensional surface onto two
dimensional presentations cause the major spatial distortions which are corrected by photogrammetric techniques.

Let us now consider sampling parameters for obtaining secondary data from a photographic image. Suppose we have a regular orthogonal grid placed over the photographic image. For a given grid spacing, not necessarily the same spacing in both directions, consider an aperture of some finite size to exist at each grid intersection in the image area. Now suppose we can measure the average tone within each aperture with a densitometer and map these measurements into a set of discrete real integer numbers having a finite range, e.g., consider our range to be from white to black in \( n \) steps so that any densitometer measurement can be quantized and assigned to a specific step.

If we are to speak of pattern recognition with a digital computer we must be able to describe the patterns in terms of the \( n \)-step data within the apertures at the grid intersections. To gain an intuitive feel for the effect of the sampling parameters of grid spacing, aperture size, and number of measurement steps, consider an exercise where a small hole in a piece of opaque paper is moved at uniformly spaced intervals across the photographic image and the object is to identify the patterns in the photographic image in terms of the average tone at each position. There will be certain minimum criteria which the sampling parameters must satisfy so that pattern representation is possible in the data. For example, if the sampling aperture is too large the spectral vector resulting from the average tones may not be representative of the spectral vector of any terrain material in the image. This might also be said of too few quantization steps in the measurement data. Whereas coarse sample spacing mainly effects spatial resolution of pattern representations. Cheng and Ledley (1968) developed a theory for picture digitization which treats spatial sampling parameters but
2.1.2 Spatial Elevation Data

This type of data has not been used in automated terrain pattern recognition. We shall see in the following section that an engineering photographic interpreter engaged in problem solving activities will insist upon stereoscopic study of terrain from conjugate photographic images.

Analogous to spectrally coded spatial data, spatial elevation data can be thought of as primary and secondary types. The primary spatial elevation data would be the continuous optical images in the two stereoscopic models from three sequential vertical aerial photographs. Thus \( Z_j \) might represent the set of two primary stereo models associated with the \( j-1, j \) and \( j+1 \) photographs along the flight line. It should be stressed that \( Z_j \) must be optically reconstructed and that it does not explicitly exist in the \( j \)-th image.

Thus \( Z_j(x,y) \) will represent a regular matrix of terrain elevation data when mapped onto the terrain and not onto any one or a combination of the images. We assume that our terrain elevation samples are results of automatic photogrammetric mapping equipment or measurements from topographic maps which have been compiled within photogrammetric standards.

Spatial elevation data has little or no temporal variation of significance in temperate climates. There will be landslides, and earth moved in construction projects, but these are relatively infrequent in temporal and spatial occurrence. There is the continual natural mass wasting processes, but in general these accumulated changes will be imperceptible to the eye or the photogrammetric instruments.

The sampled spatial elevation data \( Z_j(x,y) \) is affected by sample parameters much as with the sampled spectrally coded spatial data. Sample density, sample spot size and quantization again are the im-
portant parameters. Though sample density and quantization have the same interpretation as before, sample spot size may be different. Here the elevation sample could be estimated from a point on the surface, i.e., very small spot size, or it could be from the average surface elevation obtained from scanning a small spot along some raster. In any event, the sampling may destroy high frequency spatial elevation variations and this could affect terrain component recognition.

Assuming that our sampled elevation data is adequate for our needs, recognition of terrain components in spatial elevation data requires a different approach than that used for analysis of spectral data. An elevation chosen randomly from the spatial elevation matrix for a given area of terrain has little significance. To be of value the elevation must be associated with surrounding elevations in the matrix or one must know the context information related to the terrain component name and sample location on that terrain component.

2.2 **Qualitative Terrain Pattern Recognition**

In this discussion qualitative terrain pattern recognition shall be equivalent to the image analysis techniques usually associated with aerial photographic interpretation. We shall first discuss image interpretation in general, then we will discuss important concepts concerned with image interpretation for engineering purposes.

2.2.1 **Image Interpretation-General**

Prior to the advent of automated image analysis there were two commonly accepted means for data extraction from aerial photographs; photogrammetry and photo interpretation.

**Photogrammetry** is the science and art of obtaining reliable measurements from aerial photography. Usually photogrammetry is regarded as being synonymous with the compilation of topographic maps from aerial photographs. However, measurements of volume, height,
area, density, rate of change, or some other factor of interest may not lead directly to topographic map production but yet fall within the purview of photogrammetry.

Photo interpretation is the act of examining photographic images for the purpose of identifying objects and judging their significance. We can generalize this definition to apply to all aerial photography. In the interpretation, the identification of objects can be considered as two distinct processes: the recognition process and the classification process. Recognition will imply the act of discerning an object pattern or condition pattern while searching within the image and classification is the act of naming the pattern from a set of acceptable names. The assessment of pattern significance is a value judgment related to the photo interpretation problem at hand.

The loosely defined criteria for image interpretation cause it to be more of an art process than a scientific process. The scientific aspects of image interpretation are related to systematic approaches to image analysis and within each professional discipline will be found accepted analysis techniques. There are few texts dealing with photographic interpretation techniques, e.g., Miller (1961) pertains to photogeology and Lueder (1959) discusses photo interpretation with respect to a number of professions, but one soon learns the art-type nature of image interpretation when surveying the literature. The methods of photo interpretation outlined in texts are pedagogical and not the practical method. In general, the literature deals with examples of what has been accomplished with image interpretation techniques rather than with the details of how the analysis was conducted. Thus it is often difficult to see or reason to the author's conclusion. This is analogous to a painting displayed by an artist as proof of his capabilities as a painter. The severe question which must be asked of
each example-type article in the literature is whether the information was derived from analysis of the imagery or from field sampling or a prior knowledge of the study area.

2.2.2 Image Interpretation for Engineering Purposes

An engineer is interested in image interpretation as a source for terrain information of value for planning construction projects which interface with the terrain. As an engineering project transitions through several phases from initial conception to finished construction each successive phase requires a more sophisticated type of engineering terrain information. The preferred method for obtaining quantitative terrain information, especially for the latter design phases of a project, concerns physical sampling in the project area and subsequent laboratory testing of the samples. However, acquisition of information in this manner is costly and time consuming.

In the initial phases of a project the information requirements are more general because the decisions relate to alternatives in location and design strategies and organization of the overall planning effort. To assist in these initial phases the engineer developed a technique for obtaining terrain information from aerial photography (Belcher, 1946). The technique is generally applicable; having been successfully applied for more than three decades in many areas of the world on numerous types of engineering problems. Perhaps its greatest utility is for application to terrain-related engineering problems in remote and/or relatively inaccessible areas. To give a selected and very limited sampling of the early literature we mention the following: Frost and Mintzer (1950) used aerial photographic interpretation techniques to identify permafrost in Alaska; Leighty (1962a, 1962b) discusses engineering application in arctic areas; Miles et. al. (1963) presents an airphoto approach to forecasting soil trafficability;
Johnson (1951) interpreted engineering significance of sand areas; Browning (1951) mapped geologic formations; and Mintzer and Frost (1952) used aerial photographs for material surveys.

More recently the bulk of the literature deals with interpretation of imagery from newer types of aerial sensors, e.g., color photography, panchromatic infrared photography, color infrared photography, infrared thermal imagery, radar imagery, and passive microwave imagery. Selected examples from the literature in these areas are: Rib (1967) presented a multisensor approach for engineering soil mapping; Tanguay and Miles (1969) discuss multispectral data interpretation for engineering soil mapping; and Leighty et. al. (1968) discuss remote sensing for engineering terrain investigation with photographic, and infrared, radar and microwave imagery.

Leighty (1966) showed that the airphoto interpretation technique could be used to obtain terrain information from high-altitude, side-looking radar imagery. This demonstrated the power and generality of the technique for obtaining terrain information. The image was assumed to come from a black box sensor, i.e., little or no use was made of information related to the reflectance properties of the terrain at the radar transmitter frequency or the signal processing within the sensor system, and only terrain configuration information inferred from the radar returns and shadows was used to detect terrain components and in turn infer the geomorphology of the image areas.

With this in mind we will now look closer at important concepts of the aerial photographic interpretation technique as a systematic approach to obtaining qualitative terrain information. First we will present three basic principles for determination of soil and rock characteristics from aerial photographs. These are more or less informal theorems of the interpretation technique. Then we present the six airphoto patterns elements used for identifying and analyzing soils.
and rocks from aerial photographs. This material is quoted from Frost et al. (1953), an out of print document which is significant to this discussion.

2.2.2.1 Basic Principles for Soil and Rock Determination

There are three basic principles which govern determination of characteristics of soils and rocks from aerial photographs.

1. The aerial photograph records the results of natural processes in the development of residual soils and occurrence of transported soils, by reflecting surface and subsurface features. Successful analysis and interpretation is contingent on recognition and proper evaluation of natural environmental aspects of an area, which are reflections of certain natural processes, particularly those of a geologic and pedologic nature. The process of mapping and evaluating the soils and rocks by a study of aerial photographs requires use of logic and understanding of the "rhyme and reason" of how a given deposit developed. Being able to trace the causes and events responsible for the development of a given soil deposit makes the process of soil evaluation with a aerial photograph not a chance occurrence but an expected reality.

2. Earth materials can be grouped together to form a pattern which is composed of recognizable surface features. It is an important fact that soil deposits of similar physical characteristics occur over a considerable area and that areas of similar materials may be composed of definite recognizable physical features. These patterns can be considered as areal soil patterns. Often a single aerial photograph contains sufficient information to determine the sequence of events which contributed to the development of a certain physiographic condition. However, within any given physiographic region, a variety of patterns is likely to occur as a result of local variations within the province.

3. Soil and rock patterns and their reflected aerial photographic patterns are repetitive in nature, i.e., similar materials create similar photographic patterns, unlike materials reflect unlike patterns. Any two materials that are derived from the same soil and rock parent material, or deposited in a similar manner, and both occupying the same relative position, and existing under the same climatic conditions, will have similar engineering properties and will exhibit the same air-photo patterns.

2.2.2.2 Photo Pattern Elements

The tools which the interpreter uses in identifying and analyzing soils and rocks from aerial photographs are referred to as elements of
the photo pattern. They represent identifiable patterns in the aerial images.

1. **Landform**: The topographic expression of the landscape is composed of distinct and recognizable features superimposed at different scales. Each physical feature can broadly be termed as a landform developed by erosional or depositional processes. Empirical and explanatory descriptions are used to relate information about landforms. The empirical description of a region uses simple commonplace expressions, with no attempt at explanation. For example, hills, mountains, valleys, and other features are described in terms of shape, size, position, and color. Explanatory descriptions employ terms which convey some idea concerning the genesis of the landform. Such terms as dune, esker, cinder cone, etc., convey more than the simple fact of origin, for they convey or note also much regarding the matter of shape, size, composition, as well as location with relation to other landforms. An understanding of landforms and their origin makes it possible to bound areas of like materials and to predict textures and physical characteristics of the materials that make up the deposits. Of all the natural elements of the photographic pattern, the landform often is the most important because it is so closely associated with origin of the materials and the subsequent erosional history.

2. **Regional Drainage**: The general character of the surface of a given terrain reflects a regional drainage pattern. The overburden and strike, dip, and type of bedrock. The drainage pattern is directly related to the porosity or permeability of the soil profile, for instance, porous sands show an absence of surface drainage development whereas clays and silts resist penetration of moisture, promoting run-off accompanied by soil erosion. Thus the complete absence of surface drainage indicates porous materials whereas a highly integrated system with branching tributaries is inferred as associated with poor internal drainage characteristics.

3. **Erosional Features and Gully Characteristics**: Gully systems, or local arrangement of erosional features, are closely associated with particular soil types and are rarely duplicated in unlike materials. Gully characteristics usually fall into classes associated with soil texture groups. Granular soils have V-shaped gullies with short, steep gradients. Non-granular soils have broad, rounded cross-sections. Sand-clay erosional features have flat bottomed gullies with sharp, steep-sided slopes and low flat gradients. Cohesive soils have rounded saucer-shaped gullies with long, low gradients. Combinations of these have no special classification when there are layered or stratified soils with strong profiles.
4. **Photo Tones:** On panchromatic aerial photography soil color tones are represented as a series of grey tones in a scale which ranges from black to white. These gray scales, in addition to presenting a representation of soil color, may at times indicate differences in soil textures, moisture content, elevations, and vegetation. Soil color tones range from extensive areas of one shade, where the areal distribution of a soil type is great, to a minute speckled effect created by the variation in texture and thickness of the top soil occurring in extensive areas of uniform soil parent materials. The element of photo tone can be unreliable if the interpreter has failed to properly evaluate the climatic conditions.

5. **Vegetation:** Contrasts in vegetation usually reflect a change in the natural and environmental conditions for plant growth. The interpreter must decide whether these conditions are natural or artificial and whether they are associated with soil texture, soil moisture, or with topography. Often study of the other pattern elements along with the vegetation will aid in the determinations. Correlation of specific types of vegetation with soil characteristics and conditions has been accomplished to a limited extent and care must be exercised in extrapolation to other locals without field checking.

6. **Land Use and Special Landforms:** Man adapts himself to his environment or alters it to suit his needs and his efforts related to land use are recorded on aerial photographs. Ditches, levees, flood walls, contour plowing, orchards, borrow pits are but a few of the patterns and features recognizable on airphotos which yield information. There are also certain natural features which are indicators of soil and rock types, such as sinkholes in limestone areas, vertical road cuts, pinnacles and terraces in loessial soils, and blowouts in aeolian sands.

This has given us the definite flavor for engineering photographic interpretation. The principles tell us that terrain is not random but organized by natural processes which result in repetitive photographic patterns. We have seen that much of the information concerns the configuration of the terrain surface and can now appreciate the engineer's need for stereoscopic imagery. The interrelated nature of the information derived from study of individual pattern elements for a given pattern implies that the empirical interpretation technique is implicitly based on application of informal logic processes. The interpretation is more than reading a photograph, it is the final step which
follows both the identification of photo elements of the patterns and the collation and analysis of spatial information.

We will now outline the important concepts related to the procedures of airphoto study. Of prime importance is the premise that the interpretation procedure is based on the desired end result, such as a solution to a given requirement for engineering information. For a formal study, the end result is usually a map showing the spatial distribution of coded descriptions related to the problem at hand and a report describing the purpose of the study and information and descriptions supplemental to that presented in the map form.

The photo study procedure will generally be one of two types: (1) a known area procedure and (2) an unknown area procedure. In both cases the study begins with a random scan or search of the stereoscopic image to normalize photo tones, photo scale and vertical exaggeration and then to locate a terrain feature (photo pattern element) which places the interpreter in the terrain context of the image. Then if the interpreter is knowledgable of the terrain patterns in the imaged area and experienced with the problem at hand, he will converge rapidly on the terrain component of interest to the problem by sequentially searching out key terrain features (photo pattern elements) of terrain components associated with the already identified terrain component until the problem requirements have been satisfied. If a map is to be prepared based upon a given information coding, then a second and more careful study is performed beginning at a known point in the image and drawing boundaries around patterns or pattern combinations representative of the information code.

The procedures used in study of known areas are mainly intuitive and strongly conditioned by experience. For the study of unknown areas, given that the interpreter is experienced with the engineering
problem at hand, the study is a recursive procedure with first principles. That is, beginning from a known terrain component, cultural features, or photo pattern elements in the image, e.g., an erosional gully, a point in the drainage system, a local topographic high point, etc., a search is made for photo pattern elements which are hypothesized as being associated with the known pattern. When this is satisfied the other photo pattern elements associated with that pattern are located and the spatial extent of the pattern is mentally or physically bounded. Knowing this new pattern may then suggest possible adjacent patterns to be located and identified. When the photo pattern elements of the new pattern are not congruent, the procedure and logic must retrace to the last verified hypothesis and a new next hypothesis formulated knowing what was learned from the latest sampling. If stymied with no fresh hypothesis in mind, the practice then dictates selection of a new known point or pattern elsewhere in the image to begin the analysis from the beginning or from some previously successful hypothesis.

Sometimes the terrain is such that gross partitioning is possible prior to attempting to identify specific patterns. For example, in areas of diverse topographic relief, gross boundaries of a preliminary nature may be drawn based on arbitrary slope classes. With these general gross patterns defined the above procedure is used for each gross pattern.

Each study of an unknown area is a puzzle unto itself. Some being easy to solve, others being difficult to solve, and sometimes there are patterns which are such that only field sampling will reveal the correct solution. The frequency of this happening with an experienced interpreter is very low, but could occur with anyone.

What we have outlined for study procedures could be considered as
analysis strategies. We have attempted to outline the more common systematic analysis strategies, but it should be recognized that one could employ a random analysis strategy, though the useful purpose of such is not known. Other than for a few general approaches, which are mainly implicit to the interpretation technique, the literature is devoid of definition of analysis strategies.

2.3 Statistical Pattern Recognition

The statistical pattern recognition techniques become important when one begins to think of using a digital computer to assist in pattern recognition problems. The secondary data resulting from sampled imagery $S_i(x,y)$, $x=1,X$, $y=1,Y$, $i=1,I$, as discussed in Section 2.1.1, are real numbers from which the pattern or information of interest is to be extracted. Thus we can think of statistical pattern recognition and quantitative pattern recognition as being synonymous.

The statistical approach to pattern recognition is founded on two assumptions: (1) that the object or event of interest can be represented by a finite-dimensional vector of features, and (2) that the problem is to assign any given vector to one of a finite number of categories.

2.3.1 Classical Statistical Pattern Recognition

The classical statistical model of pattern recognition divides the problem into two stages: the extraction of significant characterizing features, followed by the classification of the resulting feature vector. This feature extraction-classification model has proven useful in organizing and focusing efforts on the study and development of classification algorithms and solving a variety of real life problems in automatic pattern recognition, e.g., in communication theory applications, voice recognition, character recognition, etc. Feature extraction is still relatively undefined, but generally thought to be
the heart of the pattern recognition problem. Empirical work on feature extraction has been surveyed by Levine (1969) and Tou (1969) and theoretical work on classification has been surveyed by Ho and Agrawala (1968) and Nagy (1968b). Excellent surveys of the total classical statistical pattern recognition field were given by Nagy (1968a) and Duda and Hart (1971).

### 2.3.2 Applied Statistical Pattern Recognition

The early literature from the remote sensing programs was symptomatic of a newly expanding field. In the beginning the literature presented unique examples intended to demonstrate sensor system capabilities in various portions of the electromagnetic spectrum, and little attention was given to limitations or detailed analysis. The result of this has been a type of over-selling which has led to the philosophy of flying as many sensors as possible so as not to miss any potentially useful information. These new data acquisition techniques engendered new data analysis techniques. The spatial data from electromechanical scanners was processed and analyzed by time series and other techniques familiar to statistical communications theory (Rosenfeld, 1963 and Fu et al., 1969). Classical statistical pattern recognition techniques were used on sampled data $S_i(x,y)$ (Dalke, 1966; Harlick, 1969; Harlick and Kelly, 1969; and Smedes et al., 1971) and coherent optical techniques on image data $S_i$ (Pincus, 1969; Lendaris and Stanley, 1970; and Eppes and Rouse, 1971). Then as it became apparent that the aerial data acquisition capability was well advanced beyond the data analysis capability, attention was given to automated pattern recognition (Spooner, 1965; Centner and Hietanen, 1969; and Richardson, 1971). At this stage researchers are beginning to give attention to the limitations of the techniques and definition of the natural and physical parameters which affect the data and data analysis (Steiner and Haefner,
1965; Kriegler, 1969; Crane, 1971; Nagy et. al., 1971; Ready, 1971; Silverman, 1971; and Turner, 1971). Also attention is now being given to the combination of man and machine in the data analysis tasks (Haralick, 1970; Eppler et. al., 1971).

2.3.3 Precautions for Statistical Terrain Pattern Recognition

We are now interested in statistical pattern recognition for obtaining terrain information for sampled aerial sensor data. Our sampled data will, as mentioned previously, be of the type $S_i(x,y)$, $x=1,X$, $y=1,Y$, $i=1,I$, for sampled photographic image data.

If one is to use terrain spectral vectors for pattern recognition there are some precautions which should be evaluated and possibly corrected in light of the two mentioned basic assumptions for the statistical approach. To satisfy these assumptions with spatial data we would have to add an additional assumption which considers the spatial data as position invariant. We will now heuristically discuss some cases where a position invariant assumption may not be valid.

For this discussion consider an aerial photographic system to provide our primary data and boldly assume an isotropic reflectance spectra for each diffuse reflecting material in the terrain. Now we can assume position variant data to result from two sources; one being the photographic system and the second being related to the terrain and nature. As for the photographic system there is the inevitable vignetting which causes increases attenuation of energy as a function of angular distance from the lens center axis. This can be corrected or minimized by lens calibration and use of anti-vignetting filter or correcting the sample data as a function of spatial position (Silvestro, 1968).

The remaining causes of position variance must be handled separately. Consider the sun at a given position; we will then have shadows
cast by natural and cultural features in the image. Shadow areas are illuminated by blue skylight with the result of changing the spectral vector for any given material originally determined from incident solar spectra. Correcting for this source of position variance would be quite laborious and only approximate (Krieeler et al., 1969). The best solution is to obtain the imagery with the sun as close to the zenith as possible; therefore minimizing shadow areas.

The geometric aspect of the terrain surface with respect to the sun provides an additional source of variation which is directly related to surface configuration. A surface normal to the sun will receive the maximum energy per unit area and the energy incident at any other angle will be a function of that angle. Corrections can be made for this topographic variation in spectral data by use of the elevation spatial matrix, $Z_j(x,y)$ (Steiner and Haefner, 1965).

The last type of spatially variant information pertains to natural variation in the surface or surface conditions which alter the inherent spectral reflectance of the terrain material. This type of variation is relatively easily perceived by an interpreter but may appear as noise to an automated system. Examples of natural variations would be high soil moisture contents in local depressions while local high areas of the terrain are relatively dry. Rather than correct for this type of variation, it should be considered as information if detectable.

One may ask why we bother to consider spectral data for obtaining engineering terrain information. Certainly the acquisition, storage, and processing of spectrally coded spatial matrices is expensive and time consuming and only large processing systems could be expected to handle these large volumes of data in a somewhat efficient manner. Our reply to this question would be that we wish to present a general approach but more specifically it is for our convenience. Some terrain
patterns have relatively high correlation with spectral vectors which are relatively stationary, e.g., water, sand, snow, etc., and as we shall later show, our suggested approach to automated terrain recognition can conveniently use spectral information for description and analysis on selected occasions.

2.4 Linguistic Pattern Recognition

In addition to the previously discussed methods of qualitative and quantitative pattern recognition, in this section we will discuss selected concepts of linguistic pattern recognition. This relatively new approach to pattern recognition is yet in its embryo stage of development in terms of the sophistication of patterns it can handle, but unlike quantitative pattern recognition, it offers the potential for utilizing pattern structure for pattern description and analysis. Researchers in scene analysis have recognized a large body of problems that require descriptive, as opposed to classificatory, methods for their solution.

In a broad sense linguistics is the study of languages and languages consist of vocabulary and grammars. If we think of the vocabulary as a set of signs and the grammar as the relational structure between signs, it is not difficult to visualize the application of linguistics to line type pictures, such as flow charts, mechanical drawings, bubble chamber photographs, computer graphics etc. Miller and Shaw (1968) and Fu and Swain (1969) review published work in pattern recognition using linguistic techniques. The novice will soon find that to discuss the grammar as meaningful relations among subparts of pictures the linguist has brought to the pattern recognition literature his own language with its associated vocabulary and grammar.

Narasimhan (1970) states, "One can only identify such complex pictures by 'articulating' their several aspects. That is, one can
only identify a tree as a tree by recognizing that it has a part that looks like a trunk, from which issue parts which look like leaves, and so on." Thus, to recognize a tree it is necessary to analyze it into its structured subparts and the relations between them.

Narasimhan (1970), Evans (1969) and Miller and Shaw (1968) assert the feature extraction-classification model in statistical pattern recognition is not adequate to handle complex picture structure. Evans (1969) states that the limitations inherent in the ability of the classical statistical model to treat complex patterns are based on its inability to cope with what we think of intuitively as the structure of the pattern and to concentrate its attention as necessary on the subpatterns whose relationships form the picture. The model is a "global" one, capable only of computing a set of properties defined on the whole input pattern, then making a choice based on them.

To get the flavor for one type of formal description, we present in Fig. 2.2 an illustration given by Evans (1971). We will have use for this approach in a later discussion.

Evans was concerned with grammatical inference in pattern analysis and here his input pattern represents (after preprocessing) a face-like drawing. He has a language for his descriptions which is composed of a set of symbols (English letters and numbers) and a grammar, which is a finite set of rules, each of which consists of three parts:

1. The arbitrary name of the pattern of construct being defined (in this case 'face').
2. A list of dummy variable names for the component parts for reference to (3) (here 'head', 'features', 'eyes', etc. are dummy variables).
3. A condition, written in terms of the available predicates, i.e., the properties and relation, that must be satisfied by
Fig. 2.2 Pattern description with a structural graph and a set of grammar rules (after Evans, T.C., 1971).
the component parts in order that we have an instance of the pattern object being defined. This is demonstrated in Fig. 2.2 and the first grammar rule can be read as: a 'face' is defined to be an object (pattern) made up of two components (call them \( x \) and \( y \)) such that \( x \) is of the object type 'features' and \( y \) is of the object type 'head' and the predicate 'inside(\( y, x \))' is satisfied. Thus his fundamental symbol set is the set of dummy variables circle(\( x \)), dot(\( x \)), square(\( x \)), and lineseg(\( x \)), and his fundamental predicate set is inside(\( x, y \)), above(\( x, y \)), left(\( x, y \)), and horiz(\( x \)). Evans' example ends here, but if he was to use these descriptions for recognition and identification of faces, then the elements of his fundamental symbol set and his fundamental predicate set would have to be programmable, i.e., each symbol and predicate expressed in terms of elementary data elements and machine operations.

More sophisticated than the concept of pattern structure description would be a picture language which considered picture analysis based upon picture descriptions. Miller and Shaw (1967) developed a picture calculus for artificial pictures such as line drawings, flow charts, block letters, etc. Their picture description language permits one to describe the concatenation of primitive line elements of their class of pictures. A set of generation rules (syntax) formalized these concatenation descriptions and allowed construction or generation of a valid set of pictures. In picture recognition applications the set of grammar rules enabled analysis (parsing) of the picture of a given class. We are not interested in picture generation for pattern recognition, but rather in pattern recognition for picture analysis.

Narasimhan (1970) has another approach to pattern recognition. He defines a pattern as an organization: it is an organization of subpatterns, or objects, or elements, or whatever. A subpattern is again an
organization of further subpatterns, or objects. And so on. An organization is a complex of relationships that subsist between the elements which are recognized. Thus a method of recognizing a pattern is to see it as a particular organization, i.e., see it as particular objects satisfying particular relationships, by analyzing it into its primitives and their relationships. He calls this method "compositional description".

Clowes (1969) states, "The interpretation of pictures involves a further syntactic structure of the event being pictorially depicted which constitutes the semantics of the picture...we should regard the semantics of the pictures as concerned with the exhibition of relationships of a nonpictorial kind. These relationships must be regarded as defining the character of the physical world. Pictures express events in the physical world."

From this we imply the interpretation to involve not only the structure (syntax) of the patterns in the image but also the meaning (semantics) of the image pattern to the real world object represented by the pattern in the image.

2.5 Logical Pattern Recognition Approach

Our goal is to develop an automated approach to terrain pattern recognition for engineering purposes based upon the concepts used in qualitative image interpretation for engineering purposes. These qualitative concepts, when formalized, allow us to organize and structure the pattern recognition programs in a logical rather than an ad hoc manner. Implementation of the qualitative concepts via FORTRAN IV utilize selected aspects of statistical and linguistic pattern recognition in an inference logic framework. This accounts for our approach being termed as a "logical approach" to terrain pattern recognition.

The purpose of the terrain pattern recognition system is to obtain terrain information for the solution of engineering terrain-related problems. Each engineering terrain-related problem will have a solu-
tion related to the terrain in the geographic area of study. Thus each
terrain-related problem has virtually a unique solution requiring a
unique pattern recognition program. The programmer must be familiar
with the terrain-related requirements for solution of engineering prob-
lems and the manner in which terrain patterns are organized for the
solution of problems. In Chapters 4 and 5 the concepts used by the
terrain analyst are formalized. The formalisms will be in the language
of predicate calculus from symbolic logic but the computer programs are
prepared in FORTRAN IV, which is a somewhat similar formalism allowing
definition of pattern components and structure to be formulated in terms
of conditional statements containing logical and relational operators.

To develop and implement the logical approach for practical pat-
tern recognition a schema similar to Fig. 2.2 expresses the structure
of terrain components of value to the problem solution which are ex-
pected to be found in the study area. Definitional descriptions pre-
pared for each terrain component in terms of primitive properties and
relations are selectively used to recognize and identify terrain
features in the spatial elevation data and the spectrally coded spatial
data. A simple classification space is used to assign terrain material
classes from spectral vector data.

Operation of the sequential, goal-oriented recognition and mapping
is accomplished under control of the program strategy proceeding from
an identified location in the data to map the boundary of the associated
pattern. An adjacent pattern is then identified by inference logic and
verification of the definitional descriptions with elevation and spectral
data is accomplished. The process is repeated until the terrain com-
ponent associated with the problem goal is recognized and its boundary
mapped.
CHAPTER III
SYMBOLIC LOGIC

This chapter presents the basic tools of symbolic logic used in succeeding chapters to formalize qualitative concepts of the engineer terrain analyst interested in obtaining information from aerial imagery for solution of his terrain-related problems. Our attention is given first to a short discussion of the nature of symbolic logic and then to the basic concepts of propositional and predicate calculus and commands. The chapter concludes with our approach to computer programming of symbolic logic formalisms for our pattern recognition activities.

3.1 Nature of Symbolic Logic

Logic deals with the results of thinking, not with the thinking processes themselves. Once a result of thinking is obtained, we can reorder our thoughts in a cogent way, constructing a chain of thoughts between point of departure and point of arrival.

The rational reconstruction process is bound to linguistic form and for that reason logic is closely connected with language. Only after thinking processes have been cast into linguistic form do they attain the precision that makes them accessible to logical tests. An essential instrument for such clarification is supplied by the method of symbolization, which has given its name to the modern form of logic. Simple logical operations can be performed without the help of symbolic representation; but the structure of complicated relations cannot be seen without the aid of symbolism. As an analogy, consider mathematical statements being limited to words without symbolization. Symbolization eliminates the specific meaning of words and expresses the
general structure that controls these words, allotting to them their places within comprehensive relations. The twin advantages of symbolic logic are exactness and brevity. Had the mathematician been confined to words and denied the use of numerals and other special symbols, the development of mathematics to its present high level would have been not merely more difficult, but psychologically impossible. The symbolic method gives mathematics an advantage in its investigation of numbers, numerical functions, etc.; symbolic logic seeks this same advantage in full generality for its treatments of concepts of any kind.

There are many texts available on the subject of symbolic logic and there is a wide range in methods used to present the subject. Ackermann (1970) paints a popular picture which introduces concepts but avoids formalism. An intermediate text in paperback book form by Reichenbach (1966) is oriented towards developing the connection between logic and language. Another paperback, by Carnap (1958) is rigorous and often overly concise in presentation of a wealth of fundamentals. Texts by Church (1956) and Kleene (1952) are often quoted and present the subject at the mathematical logician level.

3.2 Basic Concepts and Definitions

Selected concepts from the field of symbolic logic are presented in the following discussion. No attempt is made to discuss any given concept in detail; for this we can refer to the above referenced texts. The task we have is to develop an introduction to the tools required for formulating and processing coded information. We shall see that our system for formulating and processing becomes more sophisticated as we increase the information carrying potential of our symbol set. Thus the discussion proceeds from the more general concept level where we talk of languages, through a discussion of the calculus for handling language statements (propositional calculus), to the higher level dis-
33

cussion of statement components (functional calculus and class calculus).

3.2.1 Language and Signs

Language consists of signs and rules for coordinating their use. Let us see how this might relate to our study. Signs are physical things: ink marks on paper, chalk marks on a blackboard, sound waves produced by the human throat. They are signs because of their intermediary position between an object and the sign user, i.e., a person. The person, in the presence of a sign, takes account of an object; the sign therefore appears as the substitute for the object with respect to the sign user. A photo interpreter uses an aerial photograph in this manner. To him the terrain patterns and their components are signs representing terrain objects arranged according to the rules of nature and man's activities. Thus terrain patterns and their organization can be viewed as a language. When the terrain signs are properly read (recognized) and their meaning understood (classified), then significance (values) can be attached to the signs for the engineering problem at hand. Hence, the terrain language can be related to engineering problems.

We have said that signs are physical objects coordinated to other things by certain rules. The process of coordination can be repeated, and we may introduce signs referring to signs. This interaction of the coordination process leads to languages at different levels. Thus we can have terrain languages pertaining to the atomic and molecular levels as in soil and rock minerology or at the higher levels where terrain masses represent signs, as in geomorphology or structural geology. The language which is the object of study is called the object language and the language used in speaking about the object language is called the metalanguage. From the metalanguage we proceed to metalanguages of higher levels by introducing signs denoting signs of signs. As an ex-
ample when an interpreter chooses to communicate the results of his photo study he uses a metalanguage which is oral, written and/or graphic. Each is a different language.

Metalanguage is divided into three parts. The first is called syntax and deals with relations between signs only, and thus concerns the structural properties of the object language. The second part, semantics, refers to both signs and objects; in particular, it therefore includes statements concerning the truth-value of propositions, since truth is a relation between signs and objects. The third part, pragmatics, adds reference to persons; therefore this part refers to things, signs and persons.

The rules needed to define a language are also of three types. The first are formation rules which tell us under what conditions a set of signs are meaningful. Of this kind are the grammatical or syntax rules. The second are truth rules which tell us what kind of truth-values a proposition can have and how these truth-values determine the truth-values of compound expressions. And the third type of rules is called derivation rules, which tell us ways of deriving new propositions, e.g., the rule of inference.

The most important unit among conventional signs in a spoken or written language is the proposition (the terms 'sentence' or 'statement' are also used interchangeably with 'proposition'). A proposition is usually composed of several words and is the smallest unit of language to which a truth-value can be assigned. As such they are termed the atoms of language. As atoms of matter are themselves composed of sub-particles having structure which can be studied; similarly, the inner structure of propositions can be investigated. These considerations led modern logicians to a general division of logic into two parts. The first part, the calculus of propositions, which deals with the
operations combining propositions as wholes; the second part, the calculus of functions, treats the inner structure of propositions, relating this analysis to the results of the first part.

In pattern recognition the 'patterns' are the 'propositions' and the 'pattern features' or 'pattern elements' correspond to the 'words'. Thus a given set of pattern features can have some spatial arrangement into patterns to which a truth-value can be assigned when the available signs are properly coordinated to object characteristics. A meaningful image will consist of a number of patterns having intra- and inner-structure. In the following discussion we can substitute 'pattern' for 'proposition'. We will talk of the properties of the patterns and this is their inner-structure. We will talk of relations of patterns and this is their intra-structure. We present the remainder of the chapter in the language of symbolic logic.

3.2.2 The Calculus of Propositions

Selected concepts from the calculus of propositions will now be presented.

3.2.2.1 Propositional Operations

The fundamental operations used to construct molecular sentences from atomic sentences, viz., compound sentence, are expressed by the words 'not', 'or', 'and', 'implies', and 'equivalent'. These are called propositional operations. If we use letters 'a', 'b', etc., for the propositional variables we can use the symbolic notation given in Table 3.1 to illustrate the propositional operations.

3.2.2.2 Formation Rules

From the formation rules for the operations given in Table 3.1 it follows that every propositional expression must be constructed in steps and must therefore consist of units, divided into subunits, and so on.
### TABLE 3.1

**OPERATIONS AND RULES FOR PROPOSITIONAL CALCULUS**

#### Propositional Operations

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
<th>Equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \neg a )</td>
<td>not a</td>
<td>(3.1)</td>
</tr>
<tr>
<td>( a \lor b )</td>
<td>or b</td>
<td>(3.2)</td>
</tr>
<tr>
<td>( a \land b )</td>
<td>and b</td>
<td>(3.3)</td>
</tr>
<tr>
<td>( a \Rightarrow b )</td>
<td>implies b</td>
<td>(3.4)</td>
</tr>
<tr>
<td>( a \equiv b )</td>
<td>equivalent to b</td>
<td>(3.5)</td>
</tr>
</tbody>
</table>

#### Formation Rules

1. Every elementary proposition or propositional variable is a propositional expression.
2. When a negation symbol precedes a propositional expression, the resulting expression is a propositional expression.
3. Every combination of two propositional expressions by means of a binary operation is a propositional expression.

#### Elementary Truth Tables

<table>
<thead>
<tr>
<th>Monary</th>
<th>Binary Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operation</td>
<td>( a )</td>
</tr>
<tr>
<td>( \neg a )</td>
<td>T</td>
</tr>
<tr>
<td>( T )</td>
<td>T</td>
</tr>
<tr>
<td>( F )</td>
<td>F</td>
</tr>
</tbody>
</table>

#### Derivation Rules

- \( a \lor b \equiv b \lor a \) (3.6)
- \( a \land (b \lor c) \equiv (a \land b) \lor c \equiv a \land b \lor c \) (3.7)
- \( a \land (b \lor c) \equiv (a \land b) \land (a \lor c) \equiv a \land b \land c \) (3.8)
- \( a \lor (b \land c) \equiv (a \lor b) \land (a \lor c) \) (3.9)
- \( -a \land b \equiv -a \lor -b \) (3.10)
- \( -(a \land b) \equiv -a \lor -b \) (3.11)
- \( a \lor b \equiv -a \lor b \) (3.12)
- \( a \lor b \equiv -a \land -b \) (3.13)
- \( a \lor b \equiv -a \lor b \) (3.14)
- \( a \equiv (a \land b) \lor (-a \lor -b) \) (3.15)
- \( a \equiv -(\neg a) \) (3.16)

**Substitution Rule:** It is permissible to assert a formula resulting from a tautology when a propositional expression, which is arbitrarily chosen, is substituted in the place of any free elementary propositional variable, provided that this substitution is made in all places where the original variable occurs.

**Fundamental Rule of Inference:** If \( 'a \Rightarrow b' \) is true, and \( 'a' \) is true, then \( 'b' \) can be asserted.
3.2.2.3 Truth Rules

The truth rules of logic are formulated by means of truth tables. By stating the truth-values coordinated to certain combinations of propositions by propositional operations, these rules are directives defining the meaning of the operations. Thus propositional operations establish a relation between the truth-values of the molecular sentence and the truth-values of the atomic sentences.

An interpretation of a compound sentence involves the assignment of a truth-value to each atomic sentence. The truth-value of a compound sentence can then be determined recursively from the values of its elementary sentences by means of truth tables. If the resulting value of the statement (compound or elementary) is true for a given interpretation we say that this interpretation satisfies the statement.

3.2.2.4 Derivation Rules

An advantage of the symbolic technique is that it allows us to manipulate logical formulas like mathematical formulas. The laws of commutativity, associativity, and distributivity supply the reason that these manipulations closely resemble algebraic transformations. The tautologies given in Table 3.1 illustrate these laws.

The commutativity of 'or' and 'and' are shown in (3.6) and (3.8), respectively. (3.7) and (3.9) illustrate the associativity of 'or' and 'and', respectively, while (3.10) and (3.11) illustrate distributivity. De Morgan's rules are given in (3.12) and (3.13). The definitions of implication and equivalence, respectively, are given in (3.14) and (3.15), in terms of 'not', 'or', and 'and'. (3.16) is the rule of double negation.

The rule of substitution is stated in Table 3.1. As an example consider the tautology in (3.12), which is

\[-(a \& b) \equiv -a \lor -b\] (3.17)
Now if

\[ a \equiv c \lor d \]

we can make this substitution in both sides of (3.12) and the result

\[ -((c \lor d) \land b) \equiv -(c \lor d) \lor \neg b \]

(3.18)

is still a tautology.

The fundamental rule of inference (often called "modus ponens") is stated in Table 3.1. This rule is usually indicated by the schema

\[
\begin{array}{c}
\text{a} \\
\hline
\text{b}
\end{array}
\]

(3.19)

The first two lines of the schema are the premises (hypotheses) and the third line is called the conclusion; the order of the premises can be changed without changing the conclusion. Though the rule as stated above appears to be an implication within an implication, we must remember that an implication is a statement; it is used for inferences, but cannot take the place of an inference because an inference is a procedure and not a statement. The only requirement of the rule of inference is that it lead from true statement to true statement.

In the analytical derivation of logical formulas the only admissible premises are analytic formulas or tautologies which are always true. However, when we are interested in derivations in general, we are only concerned with the establishment of true formulas; and it is therefore permissible to introduce initial formulas from every source of knowledge. Of course, derivations from synthetic formulas, i.e., formulas which can be true or false, require that the initial formulas be verified by methods other than logical. They may be verified by direct observations; or they may be established by more comprehensive methods including inductive inferences, such as used in constructing scientific knowledge. The verification, however achieved, will be of
an empirical nature. Our pattern recognition formalisms will be mainly synthetic to be verified or denied by direct observations from our data sets.

Now if we rephrase the fundamental inference rule in an empirical context we might have the following. If 'a' represents a valid or well-formed statement in our language, then 'a' has a value of 'true' or 'false'. If 'a' can be experimentally validated, i.e., if evidence exists from sampled data for the determination that 'a' is true, then we assert 'a'. Also, if we know that the conditional statement 'a + b' has the value 'true', then we can assert or claim 'b'.

Then if we can test the truth of 'b' and know that 'b + c' is 'true', then we can claim 'c', if 'b' is 'true'. Thus

\[
\begin{align*}
& \frac{a}{a + b} \\
& \frac{b}{b + c} \\
& \frac{c}{(a + b) + (b + c) + \cdots + (p + q) + q + r}
\end{align*}
\]

When generalized, this schema becomes the chain rule of inference, i.e.,

\[
\begin{align*}
& a \\
& a + b \\
& b + c \\
& \vdots \\
& \vdots \\
& p + q \\
& q + r \\
& r
\end{align*}
\]

Now suppose we have a compound statement 'b + (c ∨ d)', with the symbol '∨' representing the exclusive-or operator. If we introduce the tautology

\[
b + (c ∨ d) ≡ (b + c) ∨ (b + d)
\]

then we can read the compound statement as: 'Statement b implies statement c or statement b implies statement d, but statement b does not imply both statement c and statement d'. This could lead to a branching inference schema such as:
If 'c' is 'true' then we can assert 'e', given that 'c + e' is 'true' and we can additionally assert that 'd' is 'false', given the truth of 'c' and the exclusive-or operator; and vise versa for 'd'. This schema will be applied to our recognition strategies in Chapter 6.

3.2.3 The Calculus of Functions

The propositional or statement calculus deals with sentences as a unit and as such is quite inadequate, i.e., the propositional calculus does not break down a sentence into sufficiently "fine" constituents for most purposes. On the other hand, with three additional notions, called terms (things), functions (property predicates), and quantifiers, it has been found that much of everyday and mathematical language can be symbolized in such a way as to make possible an analysis of an argument.

In the sphere of signs we can introduce variables, i.e., signs which are not restricted to denoting one thing or one property, but which can be used to represent any thing or any property. As variables denoting things we can use the letters 'x', 'y', 'z', etc.; as constants, the letters 'a', 'b', 'c', etc.; and as predicate variables letters 'F', 'G', etc. The introduction of variables suggest the analogy to a mathematical function and indeed the symbolization

\[ F(x) \]  

is generally accepted. The property is conceived as a function of the thing, which in turn is regarded as the argument of the function.

Argument and property, in combination, determine a situation. When the
argument varies while the function remains constant, we obtain a set of situations. It is this variation of the situation with the argument which constitutes the root of the name 'function', in analogy to mathematical functions. The expression 'F(x)', including both the predicate and argument will be called a functional.

There are also functions of two or more variables. The two place functions are usually called relations, which corresponds to the term 'property' in one place functions. Examples of two place functions are: 'smaller than', 'above', 'beside'. Thus 'a is higher than b' can be symbolized as

\[ F(a, b) \] (3.24)

where 'F' represents the function 'higher than'. Examples of three place functions are: 'gives', 'between', etc.

We now introduce the operation of specialization by which we put a special value 'x₁' in place of the argument in the propositional function to construct a proposition 'F(x₁)'. We use the subscript in 'x₁' to indicate that the variable has been specialized. A specialized variable is also called a constant; it represents a specific interpretation of the variable.

3.2.3.1 Quantification

Another way of constructing a propositional function other than the operation of specialization is given by operations called quantification, which are methods for binding variables to constant values.

One such operation is generalization. In this operation we proceed from the propositional function 'F(x)', where 'x' is a free variable, i.e., without definite truth-value, to the proposition 'for all x, F(x)' which we write

\[ (x)F(x) \] (3.25)
we call \((x)\)', preceding the functional, the all-operator, and the variable \('x'\), the bound variable. For a given meaning of \('F'\) this is a proposition for \('x'\), and is similar to the dummy variable of integration in a definite integral in mathematics. The bound variable has a definite truth-value, while the \('x'\) variable assumes each value, one after the other. The truth-value of an all-expression representing a proposition is true only if each of the values it assumes results in a true value, and is false otherwise.

Another form of quantization pertains to an operation which obtains when we go from the propositional \('F(x)'\) to the proposition 'there is and \(x\) such that \(F(x)'\). This is symbolized as

\[(Ex)F(x)\]

The symbol \('(Ex)'\) is the existential operator and the total expression is called the existential statement. It is important to realize that an existential statement contains a bound variable, like the all-statement, and likewise refers to a totality; it says 'among the totality of things there is at least one thing that has the property \(F'\).

Existence is formulated by an operation in the object language; its corresponding expression in the metalanguage is truth. Whenever a sentence is true, it can be translated into a statement that something exists in the sphere of objects. This relation between truth and existence can be given a correlate within the object language when we use, instead of the statement that the sentence is true, the asserted sentence itself. The resulting statement is expressed by the general formula:

\[F(x_1) \equiv (Ex)F(x) \& (x = x_1)\]

The equality statement here represents identity, a relation holding between arguments, whereas the sign of equivalence stands between propositions.
The two operators for quantification are related in terms of negation. This relation is stated by the following two formulas:

\[-(x)F(x) \equiv (\exists x)-F(x)\]  \hspace{1cm} (3.28)
\[-(\exists x)F(x) \equiv (x)-F(x)\]  \hspace{1cm} (3.29)

These two statements make use of the difference between denying the whole statement and denying the operand.

### 3.2.3.2 Complex Functions and Definitions

A function like 'F(x)' is called an elementary function when the symbol 'F' is not reducible to other symbols. For those expressions in which the symbol 'F' possesses an inner structure, i.e., is defined in terms of other functions or arguments, we will use the term complex functions.

The introduction of new terms as a function of known terms is called a definition. In symbolic language we use here the sign '\(=_{df}\)' meaning 'equivalence by definition'. In general the sign '\(=_{df}\)' stands between sentences, not between words or phrases. The symbolization being defined is called the definiendum; the definiens is the expression, conventionally placed to the right of '\(=_{df}\)', with the older terms.

A definition must enable us to eliminate the new sign for any given sentence containing it, i.e., to transform the given sentence back into a sentence containing the older terms. The definition is then an abbreviation for the combination of more primitive terms.

Definitions are arbitrary, and we may include whatever predicates we wish in a defined term; but after the definition is given there always remains the question whether there is a corresponding thing. The answer lies in the form (Ex)F(x) and requires verification by empirical methods. The importance of definitions is the fact that they allow us to define and abstract terms by reference to concrete terms, whereas the scholastic definition through higher genus and specific differences
determines an abstract term by reference to more abstract terms.

3.2.4 The Calculus of Classes

Apart from separate things or individuals, logic is concerned with classes of things. In everyday life and mathematics, classes are often referred to as **sets**. Arithmetic, for instance, frequently deals with sets of numbers, and in geometry our interest attaches itself not so much to single points as to point sets (namely, to geometric configurations). Classes of individuals are called **classes of the first order**, while **classes of the second order involve classes which consist, not of individuals, but classes of the first order, and similarly there are higher order classes. Here we will concern ourselves with classes of the first order.**

A **class** $F$ can be defined as the totality of objects, individuals, or things having a certain property $F$, i.e., of the objects $x$ for which the expression 'F(x)' is true. Here the class is denoted by a capital letter corresponding to the letter denoting the function. When an object $x$ **belongs to** a class $F$, or is a member of $F$, we write

$$x \in F$$

We read the symbol '$\in$' then as 'belongs to' or 'is a member of'.

The definition of classes in terms of functions can be expressed by the rule that (3.30) means the same as 'F(x)'; i.e.,

$$x \in F =_{df} F(x)$$

from this definition it is clear that classes can be dispensed with; they can be eliminated by use of functions. It is convenient for many purposes, however, to have class notation. This notation corresponds more closely to those forms of conversational language in which the word 'is' is combined with a noun expressing a one place function. When we say 'this is an alluvial terrace' we are stating that 'this terrace belongs to the things in class alluvial'.
3.2.4.1 Class Operations

The application of propositional operations leads to the definition of operations with classes which are usually represented on a Venn diagram. It should be noted that the expressions of the form 'F v G' and 'F & G', denote classes but do not constitute propositions. When we wish to write propositions we must either add the symbol 'x' and 'e', as has been done above, or express the proposition under consideration in the form of a statement of the identity of two classes. Whereas we have used the identity of physical things as a primitive term, we can easily define the identity of classes by the following relation:

\[(F = G) = \text{df} (x)[(x \in F) \equiv (x \in G)]\] (3.32)

This definition states: two classes are identical when they have the same members. The identity of classes is thus defined in terms of the equivalence of propositions.

Similar to (3.27) we can express existence in class notation, i.e.,

\[(x \in F) \equiv (\exists y)[(y \in F) \& (y = x)]\] (3.33)

where the equality statement represents identity things, whereas the equivalence stands between propositions. (3.33) can be used to point to or indicate specific class members under consideration.

3.2.4.2 Abstraction

The notation of class finds an important application in the interpretation of a logical operation which traditionally has been called abstraction. Given a group of objects, we sometimes wish to speak of a common property of all these objects. We then abstract this property from the complex descriptive expression of the various objects; i.e., we disregard the differences between the objects and select the common property as the focus of our attention. The term 'abstraction' has here the meaning of 'separation from'. Thus the property of redness can be defined by abstracting it from a group of red objects.
3.2.5 Commands

The calculus of propositions, functions and classes can be termed the calculus of assertions, where an assertion is a sentence, statement, or proposition used in some context to make a factual claim. Our language requirements are slightly broader and must allow use of a subclass of imperatives pertaining to commands. A command is interpreted to be a statement in the metalanguage which has the form of an order, directive, instruction, or a prohibition or 'negative' command. A command is neither true or false, e.g., 'close the door', but it can be well-formed (valid) or invalid (illogical), e.g., 'write me on the second Tuesday of next week'. Rescher (1966) gives historical observations and his approach to the theory of commands. The following has been adapted from his presentation to satisfy our formalism requirements and to facilitate the transition to computer programming.

Each command is considered a complex function and every command has a source, a recipient, a requirement and an execution precondition. The command may also contain, or have associated with it, a terminating statement which is true when the command requirement is satisfied. The minimum command, given in context, may have only the command requirement explicitly stated, whereas the other command components are implicit from the context. For example, the context command 'close the door' states only the requirement while the command 'I order you, John, to close the door in the south wall of this room now' explicitly mentions each of the command components. An associated terminating statement could then be 'the door is closed'. Additionally we can illustrate the complex nature of a command with this simple example by noting that the method, i.e., the actions and decisions, required to 'close the door' have not been enumerated for 'John'. Thus it is implicit in the command that 'John' knows of or can find an effective
method for accomplishing the requirement. The command could then be
decomposed into a series of elemental commands each with an associated
terminal statement. If 'John' is a 'small boy' our command require­
ments may be a series such as 'go to the door', 'grasp the door',
'swing the door closed'. If 'John' is a 'robot' the command require­
ments must be defined and programmed at the elemental and decision
levels. Commands for our automated pattern recognition approach must
be defined and programmed at this elemental level to be effective in
a digital computer object language. If, however, we can formulate ef­
fec tive commands in object language we can perhaps combine elemental
actions and decisions to form algorithms or subroutines which can be
named and compiled in a higher language. Continuing this recursively
in context we attain the level of everyday speech or formalism where
each statement or statement component represents a complex group of
well defined actions and decisions in the elemental object language.

Our command formalisms will be given at this higher level as propo­
sitional functions of one or more variables. Each will be defined in
the metalanguage and implicitly required to have an associated defin­i­
tion in the elemental object language. Our symbolic convention for
commands can be illustrated by

\[ \text{CL!}(x, y, z) \]

which reads as 'classify x as y given z'. Here 'CL' is the symbol for
'classify' and the exclamation mark '!' indicates the command to classify,
though this mark may be deleted for convenience. In the above command
'x', 'y', and 'z' are free variables, i.e., are defined as propositional
function variables of indefinite value, but when the command is in con­
text with an assertion from the calculus of classes its free variables
become fixed and it becomes a statement similar to a terminal statement
for the command. For example,
\[(t \in APM) =_{df} (\exists x)(z) \left[ (x \in \text{TIME}) \& (z = x) \& (1200 \leq z < 1800) \& \text{CL}(x, t, z) \right] \quad (3.34)\]

is a definition of time which is classified as being afternoon 'APM'.
It is a directive read as 'classify the time as being in the afternoon period if the time is between the hours of 1200 and 1800'.

3.3 Computer Programming of Symbolic Logic

In this section we wish to outline our approach to the use of digital computers for programming symbolic logic statements. We will utilize this approach in programming our logical pattern recognition problems in Chapters 7 and 8. Table 3.2 presents the comparison between the dissertation notation and FORTRAN symbolism.

3.3.1 Propositional Operations

General purpose digital computers have as primitive logical operators the three binary operations of 'not', 'or', and 'and', given by (3.1), (3.2) and (3.3), respectively. These operators link into logical expressions the logical variables or constants. Logic expressions, however, cannot be processed in a computer system; instead the FORTRAN assignment statement is used. The left side of this statement has a logical variable which assumes the equivalent truth value for the defined but arbitrarily complex expression on the right side.

Table 3.2 indicates that the assignment statement is also used for equivalence and for definitions. Equivalence, usually defined by (3.15), will have a restricted meaning in our approach, i.e., experimental verification of synthetic formulas will limit equivalence to an assignment statement or a test for equality, e.g., in a logical IF statement.

Implication statements can be treated in two ways. The first treats an implication statement as a conditional statement and uses the FORTRAN logical IF statement, i.e., 'C \rightarrow D' is treated equivalent to
TABLE 3.2

COMPARISON OF DISSERTATION NOTATION
AND FORTRAN SYMBOLISM

<table>
<thead>
<tr>
<th>LOGICAL OPERATORS</th>
<th>DISSERTATION</th>
<th>FORTRAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>And</td>
<td>C &amp; D</td>
<td>C .AND. D</td>
</tr>
<tr>
<td>Or</td>
<td>C v D</td>
<td>C .OR. D</td>
</tr>
<tr>
<td>Not</td>
<td>-C</td>
<td>.NOT. C</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ASSIGNMENT STATEMENT</th>
<th>Logical Statement</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Equivalence</td>
<td>A = C</td>
</tr>
<tr>
<td></td>
<td>Definition</td>
<td>A = df C</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IMPLICATION</th>
<th>IM(C,D) or C + D</th>
<th>IF(C)R or IF(C)C</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>RELATIONAL OPERATORS</th>
<th>LT(R,S)</th>
<th>R .LT. S</th>
</tr>
</thead>
<tbody>
<tr>
<td>Less Than</td>
<td>LE(R,S)</td>
<td>R .LE. S</td>
</tr>
<tr>
<td>Less Than or Equal To</td>
<td>EQ(R,S)</td>
<td>R .EQ. S</td>
</tr>
<tr>
<td>Equal To</td>
<td>GE(R,S)</td>
<td>R .GE. S</td>
</tr>
<tr>
<td>Greater Than or Equal To</td>
<td>GT(R,S)</td>
<td>R .GT. S</td>
</tr>
<tr>
<td>Greater Than</td>
<td>NE(R,S)</td>
<td>R .NE. S</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>QUANTIFICATION</th>
<th>(Ex)[(...x...)]</th>
<th>DO ___x=N,NN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Existential</td>
<td>(x)[(...x...)]</td>
<td>DO ___x=N,NN</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>COMMANDS</th>
<th>READ(a,b)</th>
<th>PRINT(a,b)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>GOTO a</td>
<td>CALL(...)</td>
</tr>
</tbody>
</table>

where:
A and B are logical variables and constants,
C and D are valid logical expressions,
R and S are valid numerical expressions, and
a and b are indicators and statement numbers.
'IF(C)D' where 'C' is a logical expression and 'D' is a logical assignment statement. The second method of treating an implication statement has the implication statement in the notation 'IM(C,D)' and this is defined by the definition for material implication given in (3.14). In general we use the first method, i.e., the conditional statement, because we are interested in the fundamental rule of inference which derives an assertion from a true statement 'C' and an implication statement having 'C' as its antecedent.

3.3.2 Class and Functional Operations

FORTRAN can be adapted to program higher order logic statements. For example '(Ex)(Ey)[(x e X) & (y e Y) & LT(x,y)]' might be read as 'some things x are larger than, LT, some things y'. This requires an ordered list for 'X' and 'Y' and a two-dimensional matrix of logical values relating elements of 'X' and 'Y' for the 'larger than' relation. To determine the truth value of the expression we can use two assignment statements, two nested FORTRAN 'DO loops', and a logical IF statement within the interloop as shown in the upper flow chart of Fig. 3.1. Universal quantification is accomplished in a similar manner.

FORTRAN is very convenient for handling relational operators, which are mathematical relations used to make comparisons between fixed values of numerical variables or constants. The value of the expression as determined in an IF statement will either be TRUE or FALSE depending upon the comparison being made and the numerical values of the expression at the time of comparison. The six common relational operators given in Table 3.2 show the prefix notation used in the dissertation and the associated FORTRAN infix notation.

Equality of variables within a logical expression such as $F(x_1) = (Ex)[(x e X) & (x = x_1) & F(x)]$ is handled in two ways. If 'X' is a list of numbers the 'x_1' can be directly handled in an assignment
Fig. 3.1 Computer flowcharts representing Fortran computer statements associated with logic statements.
statement, but if 'X' is an ordered list of object names then elements of the list 'X' must be matched to locate 'x₁' and thus determine the truth value of the statement. In most cases, however, we are not interested in the truth value but in the list position which results in the truth value, so that we can further process 'F(x₁)'.

3.3.3 Command Operations

Several commands in context are simply accomplished in FORTRAN. Table 3.2 indicates examples to be READ, PRINT, GOTO, CALL, and DO. Each has one or more associate requirement and terminating condition. In the sense of the command format of Section 3.2.5, an execution precondition established by an IF statement can precede any of these commands.

Commands in context not available as FORTRAN statements can be prepared. The lower portion of Fig. 3.1 shows a simple flow chart associated with (3.34). Here the termination is associated with the end of the 2400 time so that all time increments could be considered in the same manner. More complicated commands and statements associated with commands can be fabricated into subroutines and algorithms in FORTRAN.

In Fig. 3.1 there may appear to be a disparity between the elements of the flowchart and the elements of the associated logic statement. This is due to the language differences. The programs represented by the flowcharts however, will accomplish the intent of the logic statements. Further demonstration of logic programming with FORTRAN statements are found in Chapters 7 and 8.
CHAPTER IV

TERRAIN-RELATED ENGINEERING PROBLEMS

The purpose of this chapter is to describe and formulate the nature of terrain-related engineering problems. The first section relates the civil engineer to the landscape. The second sets forth an informal characterization of types of engineering problems and solutions to terrain-related problems. The last section contains the formalism for the concepts of the second section. The formalism or model begins with the general ideas and then as implicit terrain and problem parameters are defined for terrain description and terrain partitioning a level is reached where all necessary problem criteria are related to the terrain and the problem solutions are obtained from terrain data in the form of coded and boundary matrices. Having obtained this level of formalism, several concepts are relatively simply discussed and formalized.

This method of characterizing terrain-related engineering problems can be illuminating to the engineer for it forces one to explicitly relate the important parameters of his problem to data from terrain attributes. The model is not complete, however; it does not consider the natural organization of terrain, nor the relation of observable attributes to characteristics and properties of terrain materials. These are the topics of the next chapter. Also our discussion and formalism grosses over topics related to organization of the problem solving effort, strategy selection, etc., which are left for discussion in later chapters.

4.1 Civil Engineering and the Landscape

Civil engineers plan, design, and supervise the construction of
all types of buildings, bridges, dams, transportation facilities including highways, railways, waterways, airports, pipe lines, harbour works, water works, waste disposal facilities, and similar essential works for our modern society. It is evident that the civil engineer must be knowledgeable of terrain because the foundations of his facilities must interface directly with the earth at or beneath its surface. Also the engineer selectively uses the natural earth materials as construction materials for concrete aggregate, fill, etc. He must be knowledgeable of the physical characteristics and conditions of the natural materials to select the equipment most efficient for excavation, transportation, placement and compaction in the construction stages of his projects. And above all he must be cognizant of the spatial variability of the various terrain characteristics which have effect on the design and construction of his facilities.

4.2 Engineering Problems

An engineer is basically a problem solver. The terrain-related problems of interest concern the planning and construction phases of engineering projects. We will now discuss aspects of the engineering terrain-related problem domain and how a problem is organized for solution.

4.2.1 Types of Projects

There is no general list of engineering terrain-related projects and problems. In Table 4.1 we present a list of example project types and problems grouped by commonality of approach. We will consider a project to be a complex of engineering problems. Under the special problems group are the studies and problems which commonly reoccur and are important enough to be mentioned. The arbitrary nature of the list is evident when one considers that associated with a highway project could be problems concerned with site selection for bridges, access
**TABLE 4.1**

EXAMPLES OF TERRAIN-RELATED PROJECTS AND PROBLEMS

<table>
<thead>
<tr>
<th>Route Type Projects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Highway</td>
</tr>
<tr>
<td>Railway</td>
</tr>
<tr>
<td>Conduit</td>
</tr>
<tr>
<td>Waterway</td>
</tr>
<tr>
<td>Communication and Power Lines</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Site Type Projects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Airport</td>
</tr>
<tr>
<td>Town</td>
</tr>
<tr>
<td>Military Installation</td>
</tr>
<tr>
<td>Industry</td>
</tr>
<tr>
<td>Dam</td>
</tr>
<tr>
<td>Bridge</td>
</tr>
<tr>
<td>Antenna Facility</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Natural Material Source Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Concrete Aggregate</td>
</tr>
<tr>
<td>Bulk Fill</td>
</tr>
<tr>
<td>Select Fill</td>
</tr>
<tr>
<td>Base Course</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Water Resources and Control Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Water Supply</td>
</tr>
<tr>
<td>Runoff and Surface Drainage</td>
</tr>
<tr>
<td>Ground Water</td>
</tr>
<tr>
<td>Infiltration</td>
</tr>
<tr>
<td>Flood Control</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Special Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Construction</td>
</tr>
<tr>
<td>Slope Stability</td>
</tr>
<tr>
<td>Trafficiability</td>
</tr>
<tr>
<td>Municipal, Urban, or Regional Studies</td>
</tr>
<tr>
<td>Planning Field Investigations</td>
</tr>
<tr>
<td>Equipment Scheduling</td>
</tr>
</tbody>
</table>
roads, materials location for concrete aggregate and bulk fill, surface
drainage and flood control problems and problems dealing with unusual
construction practices. Also one must realize that within a project
type there could be a wide range of approaches to location, design, and
construction, e.g., there are many types of highways, from earth roads
to interstate expressways.

4.2.2 Solving Terrain-Related Problems

A problem is usually defined as a perplexing question demanding a
solution. We shall think of a terrain-related engineering problem as
an engineering problem requiring terrain information for its solution.
Problems can occur at many levels of complexity. The simplest problems
we define as atomic or elementary terrain-related problems having a
solution in terms of a description and/or a design for a particular
set of terrain conditions. Problems more complex can be divided into
elementary problems, each of which has a solution when necessary cri-
teria are satisfied. Thus given an engineering terrain-related problem
we must formulate our approach to its solution in terms of solutions to
the more elemental problems. In essence this requires a systematized
statement of the problem requirements and constraints on methods for
achieving the required solutions. We require that a problem statement
contain:

1. a list of problem goals,
2. specification of the problem area, and
3. a list of problem constraints.

The problem goals define the specific question which must be
answered and each goal may be interpreted as a subproblem with many
lower level problems. A hierarchy of problems exists below the problem
statement level which extends to the elementary problem levels.

Our problems are terrain-related and cannot be solved knowing only
the problem requirement. We thus must have specified the geographic area of terrain for our study.

The list of problem constraints are required in the problem statement to establish the scope and level of effort to be applied towards the solution as well as evaluation criteria to be used. Examples of fundamental criteria for a problem include a maximum allowable cost, types of designs desired, desired locations with respect to existing transportation facilities, utilities, etc.

Analogous to the mathematical techniques in our purview are the available terrain-related problem solving techniques associated with terrain information acquisition and analysis. The ordered basic operations and decisions on the set of terrain data lead to specific designs or final data necessary for problem solving. For simple problems, the solutions may be relatively simple descriptions obtained by partitioning the terrain into a few classes. For complex problems, a number of approaches may be required. These are usually applied sequentially and are selected by some judgement or decision process based upon the desired output or value of the previous levels of solution and the desired output for that step in the sequence. The final output is a synthesis of lower level solutions in the form of numbers representing design specifications or descriptions of the terrain-problem situation necessary for subsequent problem solving decisions. Generally the output of an engineering terrain-related problem solving effort is a map showing the spatial distribution of classes of terrain information which may or may not be accompanied by a report.

4.3 Characterization of Terrain-Related Problems

In this section we will present a formalization of selected major concepts discussed in this chapter. We will, in this sense, be constructing a model of terrain-related problems encountered in the physical
world prior to adding specific elements of terrain context. The significant results obtained from the formalism involve the explicit definition of parameters important to engineering terrain-related problems in one expression. This basic expression, (4.38), relates problem criteria, image attributes, sampling parameters, terrain attributes, and coding functions to produce a coded spatial matrix which is easily processed to obtain a boundary matrix. Several concepts are then defined by these results and the formalism leading to these results. The major underlying concept of the formalism is the description of terrain in terms of the problems and the data; the formalism develops the parameters required to describe the terrain attributes in terms of its data in a manner such that the data is easily coded in terms of the fundamental problem criteria. The following chapter discusses terrain organization and concepts of terrain classification for engineering purposes are formalized.

4.3.1 Preliminary Definitions

Our model, M, characterizing terrain-related problems will begin by presenting some elemental ideas from which more complicated or complex ideas can be defined or discussed. The formalism explicitly points to the properties and relations of factors necessary for definition of the concepts.

We consider an engineering problem P to be a basic concept of M which does not require complete definition in elemental terms, however, we must indicate, at least heuristically, the scope of P. In general a problem is a requirement for information or a solution, while information is raw or processed data of value for decision making, and a solution to a problem is more inclusive in that it results from decision making operations using information as input data.

If we think of the totality of all problems being the problem
universe PU, then in the sense of set or class theory, engineering problems P form a very small properly included class of this universe. Likewise, our solutions of P require a more restricted class of information and decisions. Thus

\[ P \subseteq \text{PU} \]  \hspace{1cm} (4.1)

and

\[ (x)[(x \in P) \land (x \in \text{PU})] \subseteq (x)\neg[(x \in P) \equiv (x \in \text{PU})] \]  \hspace{1cm} (4.2)

This indicates that all members of P are also members of PU, but there are members of PU which are not members of P.

If SN is a class of all solutions associated with PU as given by

\[ (x)(\exists y)[(x \in \text{PU}) \land (y \in \text{SN}) \land R(y,x)] \]  \hspace{1cm} (4.3)

where 'R(y,x)' is the relation 'y relates to x', (4.3) says that every problem of the universe of problems has at least one solution and perhaps more.

For the class of engineering problems we have

\[ (u)(\exists v)[(u \in P) \land (v \in \text{SN}) \land R(v,u)] \]  \hspace{1cm} (4.4)

where SN remains the class of solutions associated with PU. From (4.2), (4.3), and (4.4) it is reasonable to assume that the portion of solution space SN associated with P is smaller than that associated with PU.

We now introduce the concept of a terrain-related engineering problem, PT, as being an engineering problem, P, requiring terrain information, IT, for its solution, SN. Thus terrain-related engineering problems PT are properly contained in the class of engineering problems P,

\[ \text{PT} \subseteq P \]  \hspace{1cm} (4.5)

and as in (4.4) we have

\[ (w)(\exists z)[(w \in \text{PT}) \land (z \in \text{SN}) \land R(z,w)] \]  \hspace{1cm} (4.6)

from which we assert that the portion of solution space associated with PT to be smaller than that associated with P.
While this is not profound, when we consider that the solutions to $PT$ require a more restricted class of information $IT$, operations $O$, and decisions $D$ than those required for $P$, it is apparent that the differences could be significant when considering systems or approaches to terrain-related engineering problem solving.

4.3.1.1 Tentative Definition of 'SN'

Suppose, without proof, we tentatively adopt the definition of a solution, i.e., $(x \in SN)$, a member of a set of solutions to an engineering terrain-related problem $PT$, as

$$(x \in SN) =_{df} (\forall y)(\forall z)[(y \in IT) \land O(y,z) \land D(z,x)]$$

(4.7)

where $O(y,z)$ represents the command relation $y$ is operated upon to yield $z$ and $D(z,x)$ represents the command 'given $z$ decide $x$'. We can interpret (4.7) as: 'If it is true that $x$ is a solution to a given problem $PT$ then it is true that some elements called terrain information $IT$ are operated upon to yield $z$ output elements which result in a solution decision $x$. To understand what constitutes a solution decision $x$ we must adequately define $IT$. This is not a simple task to accomplish; we must first develop a number of primitive concepts of $M$.

4.3.1.2 Preliminary Terrain-Related Problem Definitions

Consider an area $AG$ as some portion of the earth's surface. In general we are interested in the land area associated with the continents, islands, and other locations where civil engineers have occasion to practice. If we consider a specific geographic area $A_1G$, then we imply an associated set of boundaries delimiting the areal extent of $A_1G$ which can be described in some appropriate coordinate system. In a similar manner we will want to talk about elements or portions of $AG$ and $A_1G$ which will be symbolized as $EAG$ and $EA_1G$, respectively.

Associated with a land area $AG$ will be the terrain $T$ which we will consider to be the set of natural and cultural things or elements of the
environment on or beneath the earth's surface. Associated with $A_1G$ will be a specific terrain $T_1$, i.e., specific subsets of $T$.

To describe $T$ or $T_1$ we abstract characteristics or properties, in the sense of Section 3.2.4.2, which we call 'terrain attributes' $AT$. These descriptions are then used in solving terrain-related problems $PT$.

$PT$. The definition of an engineering terrain-related problem can be symbolized as

$$\forall (x \in PT) = \text{df} (\exists x) (\exists y) (\exists z) [(x \in P) \& (y \in SN) \& R(y, x) \& (z \in IT) \& RQ(y, z)] \quad (4.8)$$

where '$RQ(y, z)' is a relational directive reading 'y requires z'. In a popular form this could read as 'Terrain-related engineering problems $PT$ are defined as those engineering problems $P$ requiring terrain information $IT$ for their solution $SN$'. The similarity between (4.8) and (4.4) is easily seen. This is yet a complex definition and is valid only if the relations '$R(y, x)' and '$RQ(y, z)' exist.

$PT_1$. An engineering terrain-related problem $PT$, when associated with a specific geographic area of terrain or terrain context, $A_1G$, is a terrain-problem situation, $PT_1$. From this definition we have

$$\forall (x \in PT_1) = \text{df} \exists x [(x \in PT) \& A_1G \& AW(x, A_1G)] \quad (4.9)$$

where '$A_1G$' is considered a constant, and '$AW(x, A_1G)$' represents the relation 'associated with' and is read as 'x is associated with $A_1G$'

$PT_1$ is thus a subset of engineering terrain-related problems associated with a given geographic area. In the next chapter it will be shown that by association with a given area $A_1G$, and hence a specific set of terrain classes $T_1$, $PT_1$ is a very small subset of $PT$. Thus $PT_1$ will be a proper subset of $PT$. We indicate this here to give insight to the nature of $PT_1$ but leave the justification until later.

(4.9) is an implicit definition in the sense it assumes one to be knowledgeable of which portion of a $PT$ is associated with $A_1G$. A more explicit definition of $PT_1$ would be
Thus by substituting (4.8) into (4.9) and associating the constant $A_{1G}$ with the terrain information $IT$ required for the solution $SN$ related to the engineering problem $P$ we have a definition which is more explicit than (4.9), however, there remain implied generalities pertaining to $A_{1G}$, $IT$, and $SN$ which should be clarified to make the definition useful.

4.3.2 Terrain Descriptions

We are interested in terrain descriptions because they will be used in the solution of our terrain-problem situations.

$A_{1G}$ and $T$. Earlier we stated that terrain $T$ was associated with a land area $AG$. This association is not one of set inclusion for $T$ and $AG$ are from two different but associated superclasses. Thus if we speak of

$$(Ex)(Ey)[(x \in T) \& (y \in AG) \& AW(x,y)]$$

we are talking in generalities about terrain $T$ and geographic areas $AG$. We can also speak in general of terrain $T$ associated with a specific area $A_{1G}$, as in

$$(Ex)[(x \in T) \& A_{1G} \& AW(x,A_{1G})]$$

But we are most specific when we speak of specific classes of terrain in a specific geographic location. Thus

$$T_{1} \subset T$$

and

$$(x)[(x \in T_{1}) \rightarrow (x \in T)] \& (x)-[(x \in T_{1}) \equiv (x \in T)]$$

$T_{1}$ is considered a proper subset of $T$, i.e., $T_{1} \subset T$, since by convention we will never have occasion to enumerate all classes of $T$ when discussing $T_{1}$, as we shall see in the next chapter. Thus

$$(Ex)[(x \in T_{1} \& A_{1G} \& AW(x,A_{1G})]$$

is not equivalent to (4.12).
4.3.2.1 Qualitative Terrain Description

Elements of terrain \((x \in T)\) or \((x \in T_1)\) are usually represented by names, e.g., river, hill, road, etc. When qualitative descriptors of properties and characteristics are appended to the names to provide the character for intra and inner element differences we have components of a qualitative terrain description. The set of descriptors will be called terrain attributes, \(AT\), and a specific subset from this set will be symbolized as \(AT_1\). For example, 'a wide, deep, and slow-moving river' is distinguished from 'a narrow, shallow, and turbulent river', while both are 'rivers'. The specific set of terrain attributes \(AT_1\) of width, depth, and velocity here qualify the terrain element 'river'. The qualitative values or qualifiers which limit the interpretation of the attribute 'width' are 'wide' and 'narrow'. If we think of 'wide' and 'narrow' as representing the intervals of the range of values for 'width', then we call this a particular set of attribute class intervals \(ATI_1\) from the set of all attribute class intervals associated with all attributes \(ATI\). Further we will refer to the intervals within the sets of attribute class intervals as \(ATII\), and a particular interval, usually associated with a particular set of class intervals, e.g., 'wide' or 'narrow', will be shown as \(ATII_1\).

While qualitative descriptions suffice for much of our everyday life, the classification ambiguity when describing attributes of natural terrain features with more than a few qualitative intervals introduces classification uncertainty. Experience indicates that qualitative descriptions for purposes of discrimination in a particular terrain-problem situation \(PT_1\) should have attributes chosen to fit the situation. For each attribute a well chosen set of class intervals \(ATI_1\) with a minimum of class intervals \(ATII\) will reduce the possibilities for classification ambiguity.
Mixed terrain descriptions. Quantitative data from sample measurements on selected attributes can be used in qualitative descriptions. This type of mixed description reduces many of the difficulties associated with the pure qualitative descriptions. Analytical models utilize mixed terrain descriptions for their solutions.

Terrain descriptions for specific geographic areas \( A_{1G} \) will be related to spatial points or area elements. Two effective methods of presenting these descriptions are by codings \( CD \) on a map \( MP \) and/or in a report \( R \). In the map form, areas containing the same descriptions are similarly coded and separated from areas with different descriptions by boundaries.

4.3.2.2 Beginning Description Formalism

We begin this discussion on terrain description formalism by indicating that there are usually specific sets of terrain attributes associated with the characterization of a terrain element from \( T \) or \( T_1 \). For convenience we will only consider the case where we are interested in specific classes of terrain \( T_1 \) in specific geographic areas \( A_{1G} \), as in (4.15). If we consider \( AT \) to be a list or set of all possible terrain attributes, then we have

\[
(\exists x)(\exists y)[(x \in AT) \land (y \in T_1) \land AW(x,y) \land A_{1G} \land AW(y,A_{1G})] \quad (4.16)
\]

\( AT \) represents a very large set of terrain attributes which must be reviewed for association with each element of \( T_1 \). In the next chapter we will show that this is only accomplished once for each \( PT_1 \) in \( A_{1G} \).

Adding \( ATI \). To coordinate sets of attribute intervals from \( ATI \) to their associated attributes in (4.16) we simply add its set member relation and the connecting relation '\( AW(.,,.) \)' . Thus

\[
(\exists x)(\exists y)(\exists z)[(x \in ATI) \land (y \in AT) \land AW(x,y) \land (z \in T_1) \\
\land AW(y,z) \land A_{1G} \land AW(y,A_{1G})] \quad (4.17)
\]
Adding PT\textsuperscript{i} and FTC. (4.17) assumes that each set of attributes class intervals \textsuperscript{ATI} and terrain attributes \textsuperscript{AT} have been chosen to fit the terrain-problem situation PT\textsuperscript{i}. Fundamental terrain criteria FTC are associated with terrain-problem situations PT\textsuperscript{j} and are utilized to select attribute class intervals \textsuperscript{ATI} for a terrain attribute of the set \textsuperscript{AT}. Explicitly expressed this would be

\[(Ev)(Ew)(Ex)(Ey)(Ez)\left[(v \in \textsuperscript{ATI}) \& (w \in \textsuperscript{AT}) \& AW(v,w) \& (x \in \textsuperscript{PT}_{1}) \& (y \in \textsuperscript{FTC}) \& AW(v,y) \& AW(y,x) \& (z \in \textsuperscript{T}_{1}) \& AW(w,z) \& A_{1G} \& AW(z,A_{1G})\right] \quad (4.18)\]

Adding AT\textsuperscript{II}. To talk about intervals of terrain attribute class intervals \textsuperscript{ATII} we must add to (4.18) as

\[(Eu)(Ev)(Ew)(Ex)(Ey)(Ez)\left[(u \in \textsuperscript{ATII}) \& (v \in \textsuperscript{ATI}) \& AW(u,v) \& (w \in \textsuperscript{AT}) \& AW(v,w) \& (x \in \textsuperscript{PT}_{1}) \& (y \in \textsuperscript{FTC}) \& AW(v,y) \& AW(y,x) \& (z \in \textsuperscript{T}_{1}) \& AW(w,z) \& A_{1G} \& AW(z,A_{1G})\right] \quad (4.19)\]

Adding P\textsuperscript{T}_{1} and FTC\textsuperscript{l}. To indicate a specific problem in a terrain-problem situation PT\textsuperscript{i} we use the symbol P\textsuperscript{T}_{1}. When addressing a P\textsuperscript{T}_{1} we only require a specific set of fundamental terrain criteria from FTC, which we symbolize as FTC\textsuperscript{l}. Thus

\[(P\textsuperscript{T}_{1} \in \textsuperscript{PT}_{1}) \equiv (Ex)\left[(x \in \textsuperscript{PT}_{1}) \& (x = P\textsuperscript{T}_{1})\right] \quad (4.20)\]

and

\[(FTC\textsuperscript{l} \in \textsuperscript{FTC}) \equiv (Ex)\left[(x \in \textsuperscript{FTC}) \& (x = FTC\textsuperscript{l})\right] \quad (4.21)\]

using the right sides of (4.20) and (4.21) in (4.19)

\[(Eu)(Ev)(Ew)(Ex)(Ey)(Ez)\left[(u \in \textsuperscript{ATII}) \& (v \in \textsuperscript{ATI}) \& AW(u,v) \& (w \in \textsuperscript{AT}) \& AW(v,w) \& (x \in \textsuperscript{PT}_{1}) \& (x = P\textsuperscript{T}_{1}) \& (y \in \textsuperscript{FTC}) \& (y = FTC\textsuperscript{l}) \& AW(v,y) \& AW(y,x) \& (z \in \textsuperscript{T}_{1}) \& AW(w,z) \& A_{1G} \& AW(z,A_{1G})\right] \quad (4.22)\]

Adding AT\textsuperscript{II}, to (4.19) and (4.22). The same approach as used in
(4.20) can be applied to (4.19) and (4.22) to address a particular class interval ATII\(^1\). It will be useful for the following discussion to show the results of specialization to both statements. Thus

\[(\text{ATII}\_1 \in \text{ATII}) \equiv (\text{Ex})[(x \in \text{ATII}) \& (x = \text{ATII}\_1)] \quad (4.23)\]

when applied to (4.19)

\[(\text{Eu})(\text{Ev})(\text{Ew})(\text{Ex})(\text{Ey})(\text{Ez})[(u \in \text{ATII}) \& (u = \text{ATII}\_1) \&
\quad \text{AW}(u,v) \& (w \in \text{AT}) \& \text{AW}(v,w) \& (x \in \text{PT}\_1) \&
\quad (y \in \text{FTC}) \& \text{AW}(v,y) \& \text{AW}(y,x) \& (z \in \text{T}\_1) \&
\quad \text{AW}(w,z) \& A\_1G \& \text{AW}(z,A\_1G)] \quad (4.24)\]

and when applied to (4.21)

\[(\text{Eu})(\text{Ev})(\text{Ew})(\text{Ex})(\text{Ey})(\text{Ez})[(u \in \text{ATII}) \& (u = \text{ATII}\_1) \&
\quad (v \in \text{AT}) \& \text{AW}(u,v) \& (w \in \text{AT}) \& \text{AW}(v,w) \&
\quad (x \in \text{PT}\_1) \& (x = \text{P}\_1\text{T}\_1) \& (y \in \text{FTC}) \& (y = \text{FTC}\_1)
\& \text{AW}(v,y) \& \text{AW}(y,x) \& (z \in \text{T}\_1) \& \text{AW}(w,z) \& A\_1G
\& \text{AW}(z,A\_1G)] \quad (4.25)\]

There are a few final generalizations required to make these statements useful for terrain descriptions; we must add the concepts of space, time and terrain data.

4.3.3 Terrain Partitioning

4.3.3.1 Selective Area Partitioning at Different Levels

First we will consider space. The prior expressions pertain to geographic areas in general, i.e., AG, or to the whole of a specific geographic area, i.e., A\(_1\)G. These are insufficient for terrain descriptions which must have capabilities for referencing points and areas in general. For example, we may wish to speak of elements, i.e., portions, of our specific geographic area A\(_1\)G instead of it as a whole. We do not consider it a contradiction to consider A\(_1\)G a constant in (4.9) and (4.10) and now talk about partitioning a constant, as long as the sum of the final partition is equal to that of the whole. The need exists for considering only the final partitioning at different
levels.

Without regard to the criteria for partitioning an area, we will represent the set of area elements resulting from the first partitioning of $A_1 G$ as $E A_1 G$. This set is included in $A_1 G$ as in (4.16). Thus

$$E A_1 G \subseteq A_1 G$$

which can be represented by

$$(x)[ (x \in EA_1 G) \lor (x \in A_1 G) ]$$

In general, we cannot speak of the subset of area elements partitioned from a set of higher level elements as an included set for selective area partitioning at different levels. This is apparent from the upper portion of Fig. 4.1, which is an example of an unsymmetric partitioned area graph representing the results of the general case of area partitioning, and the lower portion of Fig. 4.1, which is a special case of area partitioning at different levels. For Fig. 4.1 consider the initial partitioning of $A_1 G$ to yield the set $E A_1 G$ containing four members, i.e., 1, 2, 3, and 4. The next lower, or second, level of partitioning yields the set $E^2 A_1 G$ with five members contained in two subsets. No members are associated with nodes numbered '1' and '4', hence these are terminal nodes; two members have node '2' as a parent or higher level node; and three members have '3' as a parent. The partitioning continues until only terminal nodes remain. Results of the final partitioning has the partitioned areas coded with the names of the terminal nodes $TN$ associated with the partitioned area graph.

4.3.3.2 Addressing Points and Areas

The location of points within $A_1 G$ can be kept in list $L_0$ with $L_{0_1}$ being a particular point location. Thus

$$(L_{0_1} \in L_0) \equiv (Ex)[(x \in L_0) \land (x = L_{0_1})]$$

A general statement which allows one to address a specific element
Fig. 4.1 Examples of partitioned area graphs and associated maps representing results from selective area partitioning at different levels.
area after completion of area partitioning and/or a specific point location would be

\[(\text{Ex})[(x \in TN \land (x = TN_1)) \lor ((x \in L) \land (x = LO_1))] \quad (4.29)\]

(4.29) requires that lists $TN$ and $LO$ each contain the null name $\emptyset$ so that it can be addressed by $TN_1$ or $LO_1$ when that particular reference is not required, i.e., if we want to address $TN_1$ but not $LO_1$, then for $LO_1$, we address $\emptyset$.

There are a number of references to combinations of points and areas useful in terrain description. For example a reference to all points of $LO$ is contained in $TN_1$,

\[(\text{Ex})(\text{Ey})[(x \in TN \land (x = TN_1) \land (y \in LO) \land CI(y, x))] \quad (4.30)\]

where 'CI(y, x)' reads as 'y is contained in x'. For an area from $TN$ containing a point $LO_1$,

\[(\text{Ex})(\text{Ey})[(x \in TN \land (y \in LO) \land (y = LO_1) \land CI(y, x))] \quad (4.31)\]

or the areas of $TN$ containing the points of $LO$,

\[(\text{Ex})(\text{Ey})[(x \in TN \land (y \in LO) \land CI(y, x))] \quad (4.32)\]

These are but a few of the possible examples which have the same general structure but result in a different reference to points and areas.

4.3.3.3 Terrain Data and Area Partitioning

Our discussions of area partitioning have not concerned the data being partitioned nor the criteria for partitioning. We will now develop these ideas.

We assume a set of raw or preprocessed data $DT$ exists for all or one or more portions of $A_1G$. Raw data is that data associated with the initial sampling technique. Preprocessed data is that data which has been passed through some elementary data processing operation to prepare it for subsequent processing and perhaps preliminary decision processes. The raw and preprocessed terrain data will have implicit or explicit spatial references to points or areas within $A_1G$. 
Before we relate the data to the terrain attributes let us first consider another useful spatial data sampling structure. We want to consider the case of discrete sampling at regular grid intersections superimposed on an area. This has application to digital sampling and processing. If our one-dimensional list of points $L_0$ was obtained from points at grid intersections, it contains an implicit set of sampling parameters $SP$, e.g., sampling interval or sample spacing, location of coordinate reference within the grid, grid directions, number of samples in each grid direction, etc. For each point in $L_0$ we can have an associated data value in $DT$ which will allow us to construct the grid of values. Thus

$$(x)(Ey)[(x \in L_0) \& (y \in DT) \& AW(y,x)] \quad (4.33)$$

where the relation 'associated with' is a one-one binary matrix which relates each point in $L_0$ with only one value in $DT$. This is inefficient for most purposes. The technique generally used has the set $DT$ as an ordered list of data values and an associated specific set of sampling parameters $SP_1$ which allows processing the list $DT_1$ into a two-dimensional matrix and associating each value with a geographic point in $A^1_G$. This can be represented by

$$(GD_1(u,v) =_{df} (x)[(x \in DT_1) \& SP_1 \& GC(x,SP_1,u,v)] \quad (4.34)$$

where 'GD$_1(u,v)' represents a specific set of grid data points according to the set of specific sampling parameters $SP_1$ and 'GC(x,SP$_1$,u,v)' is a grid construction algorithm which accepts elements of $DT_1$ and the set of sampling parameters $SP_1$ as input and labels the output points $GD_1(u,v)$, with $u$ and $v$ being the spatial coordinates.

(4.34) is important for it allows us to view spatial data as a discrete two-dimensional set of values and, perhaps more important, it is a utility for discussing continuous two-dimensional data within sample images. This is possible if we allow sample spacing to approach
zero as the number of samples in the rows and columns of the matrix approach infinity in our set of sampling parameters $SP_1$. Obviously this is infeasible, but it can be a useful ploy.

We must relate image data $DI$ and terrain data $DT$ to terrain attributes $AT$. It was mentioned earlier that raw data was data most closely associated with the initial sampling technique. We now state that the sampling techniques should be closely associated or correlated with the terrain attribute for the possible variations in the data to be representative of the possible attribute variations. This association between data and terrain attributes is of utmost importance if remote sensing is to have worth as a source of terrain information for engineering purposes.

We now make reference to image data $DI$, image parameters $IP$, and attributes of aerial images $AI$. If we let $AI$ signify the total set of image attributes and $AI_1$ be a specific set of attributes associated the sensor image, and $DI$ be the total set of image data and $DI_1$ be a specific set of directly measurable image data, then we can symbolize the relation between terrain data $DT_1$ and the image attributes, $AI_1$ image data $DI_1$, and terrain attributes $AT_1$ as the definition

\[
(z \in DT_1) = \text{def}(w)(Ex)(Ey)[(w \in DI_1) \& (x \in AI_1) \& AW(w,x) \& IP_1 \& (y \in AT_1) \& AW(x,y) \& AW(z,y) \& CF(z,w,IP_1)]
\]

(4.35)

where $IP_1$ are the associated image parameters, e.g., image scale, image size, image spectral sensitivity etc., and $'CF(z,w,IP_1)'$ reads as '$z$ is computed from $w$ and $IP_1'$. Thus we can define elements of a specific set of terrain data to be all elements of measurable specific image data that is associated with an image attribute which is in turn associated with a specific terrain attribute associated with the terrain data and is computed from elements of the image data and image parameters.
When the terrain attributes $\text{AT}_1$ are not directly measurable in an image but can be inferred from image data we must use another definition instead of (4.35). Thus

$$\{z \in \text{DT}_1\} = \text{df} \ (w)(\text{Ex})(\text{Ey})[(w \in \text{DI}_1) \ & \ (z \in \text{AI}_1) \ & \ \text{AW}(w,x) \ & \ \text{IP}_1 \ & \ (y \in \text{AT}_1) \ & \ \text{IM}(x,y) \ & \ \text{IM}(w \ & \ x \ & \ \text{IP}_1, z) \ & \ \text{INF}(z, (w \ & \ x \ & \ \text{IP}_1), \ \text{IM}(w \ & \ z \ & \ \text{IP}_1), z)] \quad (4.36)$$

where 'IM(x,y)' reads as 'x implies y' and 'INF(z, (w \ & \ x \ & \ \text{IP}_1), \ \text{IM}(w \ & \ x \ & \ \text{IP}_1), z)' reads as 'z is inferred from premise (w \ & \ x \ & \ \text{IP}_1) and the premise IM((w \ & \ x \ & \ \text{IP}_1), z)'. Comparing (4.36) and (4.35) we see that for terrain data inferred from aerial sensor data we have implications and inferences instead of direct associations and analytical formulations.

Now we can substitute (4.36) and/or (4.35) into (4.34) and have $\text{GD}_1(u,v)$ represent grid points associated with data sampled from aerial imagery to yield the grid sample data $\text{GD}_1(u,v,z)$. We will substitute (4.36) into (4.34) to obtain

$$\text{GD}_1(u,v,z) = \text{df} \ (w)(\text{Ex})(\text{Ey})[(w \in \text{DI}_1) \ & \ (x \in \text{AI}_1) \ & \ \text{AW}(w,x) \ & \ \text{IP}_1 \ & \ (y \in \text{AT}_1) \ & \ \text{IM}(x,y) \ & \ \text{IM}((w \ & \ x \ & \ \text{IP}_1), z) \ & \ \text{INF}(z, (w \ & \ x \ & \ \text{IP}_1), z) \ & \ \text{SP}_1 \ & \ \text{GC}(z, \text{SP}_1, u,v)] \quad (4.37)$$

The sampling parameters $\text{SP}_1$ now refer to the image data $\text{DI}_1$ through the variable $z$, whose value results from the inference operation.

4.3.3.4 Coded Spatial Matrices and Boundary Matrices

We are now in position to express area partitioning for grid sample $\text{GD}_1(u,v,z)$ obtained from aerial imagery for a terrain attribute from $\text{AT}_1$ and applied to a particular problem or a terrain-problem situation $\text{PT}_1$. The data to be partitioned is represented by (4.37) and the
criteria for partitioning is given in (4.22), though other data and criteria expressions could be used. Our partitioning will be a two step process, with the first step producing a two-dimensional coded matrix and the second step producing boundaries or partitions in the coded matrix. The explicit coding expression can be defined as

\[
CM(m,n,z) = \text{def} \ (o)(Ep)(Eq)(Er)(Es)(Eu)(Ev)(Ew)(Ex)
\]

\[
(Ey)[(o \in DI_1) \& (r \in AI_1) \& AW(o,r) \&
IP_1 \& (s \in AT_1) \& IM(r,s) \& IM(o \& r \&
IP_1),a_1) \& INF(a_1,(o \& r \& IP_1),
IM((o \& IP_1,a_1)) \& SP_1 \&
GC(a_1,SP_1P,g) \& (u \in ATII) \& CI(a_1,u) \&
(v \in ATI) \& AW(u,v) \& AW(v,s) \& (w \in PT_1) \&
(w = P_1T_1) \& (x \in FTC) \& (x = FTC_1) \& AW(u,x) \&
AW(x,w) \& (y \in T^1) \& AW(s,z) \& A_1G \&
AW(y,A_1G) \& MC(u,z,m,n) \& EQ(m,p) \&
EQ(n,q)]
\]

(4.38)

where \(CM(m,n,z)\) is the representation of the coded matrix with \(m\) and \(n\) being the spatial coordinates equivalent to \(p\) and \(q\), and \(z\) representing the code variable. We employed in (4.38) a new functional relation, \(MC(u,z,m,n)\), which is a matrix coding function reading as '\(u\) is matrix coded as \(z\) at spatial coordinates \(m\) and \(n\)'. Thus if \(EQ(u,z)\) was included in (4.38), the coded values would be the intervals of \(ATII\). Otherwise, the coded values are some other functional relation to the intervals of \(ATII\).

The final step in producing boundaries in the coded matrix produces for an area \(A_1G\) of arbitrary shape imbedded in a field of zeros from upper left part of Fig. 4.2. The boundary matrix associated with \(A_1G\) is shown in upper right part of Fig. 4.2 and the results of three-value coding data associated with \(A_1G\) via (4.38) is shown in the lower left
Fig. 4.2 Matrices associated with producing BCM(m,n,z), the boundary coded matrix.
portion of Fig. 4.2. Then $BCM(m,n,z)$ results from extracting only the boundary elements for each of the three coding values of $CM(m,n,z)$.

One of the four following conditions must be satisfied for each element of the boundary coded matrix $BCM(m,n,z)$. The first condition is satisfied if $(m,n)$ is an element of the boundary matrix $BM$ associated with $A_{ij}$ and is stated as

$$\begin{align*}
(m)(n)(BM) & [(BM(m,n) \in BM) \& AW(BM,A_{ij}), (m,n) \in BCM(m,n,x)) \& \\
(x = z) \& INC(z,CM(m,n,z))] (4.39)
\end{align*}$$

This reads as: 'If $(m,n)$ is an element of boundary matrix $BM$ which is associated with $A_{ij}$, then $(m,n)$ is an element of the boundary coded matrix $BCM(m,n,z)$ where $x$ is equal to the $z$ included in coded matrix $CM(m,n,z)$ for that element $(m,n)$'.

The second condition is satisfied if the element $(m,n)$ of $CM(m,n,z)$ has a different $z$ value than that of element $(m,n+1)$ of $CM(m,n,z)$ and is stated as

$$\begin{align*}
(m)(n)(Ew)(Ex)(Ey) & [(CM(m,n,z) \in CM) \& INC(z,CM(m,n+1,z)) \& \\
INC(w,CM(m,n+1,w)), BCM(m,n,x, (x = z) \& \\
INC(z,CM(m,n,z)) \& BCM(m,n+1,y, (y = y) \& \\
INC(v,CM(m,n+1,v)))] (4.40)
\end{align*}$$

This reads as: 'If $z$ associated with $CM(m,n,z)$ is not equivalent to $v$ included in $CM(m,n+1,v)$, then the boundary coded matrix $BCM(m,n,z)$ has values for both elements $(m,n)$ and $(m,n+1)$ and each is equivalent to the values associated with the codes at their respective points in $CM(m,n,z)$'.

The third condition is similar to the second but pertains to the $z$ values of elements $(m,n)$ and $(m+1,n)$ of $CM(m,n,z)$ and, similar to (4.41)

$$\begin{align*}
(m)(n)(Ew)(Ex)(Ey) & [(CM(m,n,z) \in CM) \& INC(z,CM(m,n,z)) \& \\
INC(c,CM(m+1,n,w)), (BCM(m,n,x) \& (x = z) \& \\
INC(z,CM(m,n,z)) \& BCM(m+1,n,y, (y = v) \& \\
INC(v,CM(m,n+1,v)))
\end{align*}$$
Matrix of Possible Combinations of Attribute Maps A and B with Actual Combinations Underlined

\[
\begin{array}{cccc}
1,1 & 1,2 & 1,3 & 1,4 \\
2,1 & 2,2 & 2,3 & 2,4 \\
3,1 & 3,2 & 3,3 & 3,4 \\
\end{array}
\]

Recoding Equivalences

<table>
<thead>
<tr>
<th>Composite Recode</th>
<th>Original Codes</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
</tr>
<tr>
<td>1</td>
<td>1 &amp; 1</td>
</tr>
<tr>
<td>2</td>
<td>1 &amp; 2</td>
</tr>
<tr>
<td>3</td>
<td>1 &amp; 3</td>
</tr>
<tr>
<td>4</td>
<td>2 &amp; 2</td>
</tr>
<tr>
<td>5</td>
<td>2 &amp; 3</td>
</tr>
<tr>
<td>6</td>
<td>3 &amp; 4</td>
</tr>
</tbody>
</table>

Fig. 4.3 Composite attribute map recoding.
CM(m+1,n,v))])] (4.41)

and reads similar to (4.40).

The fourth and last condition accounts for all \( \emptyset \)-valued element entries in the boundary coded matrix \( BCM(m,n,z) \). Thus

\[
(m)(n)\left[ IM\left( ((m,n) \in BM) \& AW(BM,A_1G) \& EQ(z,v) \&
INC(z,CM(m,n,z)) \& INC(v,CM(m,n+1,v)) \& EQ(z,w) \&
INC(w,CM(m+1,n,w))) \right), (BCM(m,n,x) \& (x = \emptyset)) \right]
\]

(4.42)

which reads as: 'If \( (m,n) \) is not a member of the boundary matrix \( BM \) associated with \( A_1G \) and the \( z \)-coded values of \( CM(m,n,z) \) are equal for the elements \( (m,n) \), \( (m+1,n) \), and \( (m,n+1) \), then the boundary coded matrix has the value of \( \emptyset \) for element \( (m,n) \).'

The boundary coded matrix \( BCM(m,n,z) \) is then produced from an inference involving a disjunctive combination of (4.39) through (4.42) as an implication premise and the necessary data from \( CM(m,n,z) \) as the other premise. Schematically, this is

\[
BCM(m,n,z) =_{df} (m)(n)\left[ INF(z,IM((4.40) \vee (4.41) \vee
(4.42) \vee (4.43), CM(m,n,z),
CM(m,n,z)) \right]
\]

(4.43)

4.3.3.5 Composite Attribute Maps

We must realize that (4.38) and Fig. 4.2 pertain to the coding of only one attribute from \( AT \). Additional attributes can be coded and results superimposed to form composite attribute or factor maps. The process of compiling a composite attribute map will involve recoding as indicated in Fig. 4.3. In this the boundaries are continuous for clarity rather than discrete elements as in Fig. 4.2. Note that all possible attribute coding combinations do not occur in the final map. This is the general case, as will be seen in the next chapter.

Briefly, the formalism concerned with producing a composite attribute map involves for each element \( (m,n) \) a logical anding of code sym-
bols from the coded map matrix $\text{CM}(m,n,z)$ for each map and these form premises along with the recoding equivalence $\text{RE}(x)$ for inferring the proper recode symbol. First we define $\text{RE}(x)$ as

$$\text{RE}(x) = \exists u \exists v \left[\text{IM}((u \land v) \land \text{INC}(u, \text{CM}(m,n,u)) \land \text{INC}(v, \text{CM}(m,n,v)), x)\right]$$ (4.44)

From Fig. 4.3 we see the composit recode symbol '6' is associated with the recoding equivalence

$$3 \land 4 \leftrightarrow 6$$

which is a two way implication. Thus in (4.44) we would have

$$\text{RE}(6) = \text{IM}((3 \land 4) \land \text{INC}(3, \text{CM}(m,n,3)) \land \text{INC}(4, \text{CM}(m,n,4)), 6)$$

where it is implicit that coded matrix A precedes B.

The composit map matrix is

$$\text{CMM}(m,n,z) = \exists m \exists n \left[\text{INF}(z, (u \land v) \land \text{INC}(u, \text{CM}(m,n,u)) \land \text{INC}(v, \text{CM}(m,n,v)), \text{RE}(z))\right]$$ (4.45)

or using the right side of (4.44)

$$\text{CMM}(m,n,z) = \exists m \exists n \exists u \exists v \left[\text{INF}(z, (u \land v) \land \text{INC}(u, \text{CM}(m,n,u)) \land \text{INC}(v, \text{CM}(m,n,v)),
\text{IM}((u \land v) \land \text{INC}(u, \text{CM}(m,n,u)) \land \text{INC}(v, \text{CM}(m,n,v)), z)\right]$$ (4.46)

For all purposes $\text{CMM}(m,n,z)$ can be treated as an $\text{CM}(m,n,z)$. Thus we can apply (4.43) to any $\text{CMM}(m,n,z)$ to form a boundary coded matrix $\text{BCM}(m,n,z)$.

### 4.3.3.6 Sequential Partitioning

We conclude discussion of area partitioning with the process of selected sequential partitioning. The graph of Fig. 4.1 indicates that areas coded as '22' were partitioned into '221' and '222' without concerning areas '-22', i.e., areas not coded as '22.' This is an example of selective sequential partitioning and is equivalent to extracting areas associated with a given AT and producing a partitioning
or a new \(\text{CMM}(m,n,z)\) at the next lowest level. Thus instead of talking about areas we can associate one or more attributes \(AT_1\) with each sub-area and form an attribute graph having terminal nodes \(\text{ATN}_1\).

Now suppose we want to further partition areas of a selected \(AT_1\) from \(\text{CMM}(m,n,z)\). The scheme of the subroutine-like process can be represented by the flow chart in Fig. 4.4 with the significant portions represented by circled numbers to the left of the blocks. The process accepts the existing \(\text{CMM}(m,n,z)\) and its associated list of point coordinates \(LO\) associated with areas of \(\text{ATN}_1\). The areas of \(\text{CMM}(m,n,z)\) associated with \(\text{ATN}_1\) are extracted in \(\circ\). Next for \(\bullet\) data \(DI_1\) is extracted from aerial imagery for the required areas and attribute \(AT_1\). In \(\circ\) the data \(DI_1\) is processed for \(GD_1(u,v,z)\) according to (4.37), then in \(\circ\) this is processed with the \(PT_1\) parameters associated with FTC to form \(\text{CM}(m,n,z)\). If only one attribute is being coded then \(\circ\) prefixes the previous attribute code \(\text{ATN}_1\) to the new \(\text{CMM}(m,n,z)\) and \(\circ\) inserts this into the old \(\text{CMM}(m,n,z)\) and \(\circ\) updates the attribute partitioning graph and adds new \(\text{ATN}_1\) points to \(LO\) to complete the process.

If, however, more than one attribute \(AT_1\) is concerned with the partitioning, then \(\circ\), \(\circ\), and \(\circ\) are repeated once for each attribute or combination of attributes and 5 recodes and forms a new combined \(\text{CMM}(m,n,z)\) for each sequence prior to \(\circ\), \(\circ\), and \(\circ\). We will now symbolically indicate some steps of this process.

For \(\circ\) the extraction of points \((m,n)\) included in areas of \(\text{CMM}(m,n,z)\) associated with \(\text{ATN}_1\) we define \(\text{EXA}(m,n)\) as

\[
\text{EXA}(m,n) = \text{df} (Em)(En)(Ex)\left[ (x \in \text{ATN}) \& (x = \text{ATN}_1) \& ((m,n) \in \text{LO}) \& \text{AW}(L,\text{ATN}_1) \& \text{INT}((m,n), \text{BN}(\text{ATN}_1)) \& \text{EQ}(\text{ATN}_1z_1) \& \text{INC}(z_1, \text{CMM}(m,n,z_1)) \right]
\]  

(4.47)

Thus (4.47) can be used to partition one particular attribute coded area.
Note: 'N' is the number of specific attributes $AT_1$ for partitioning $ATN_1$.

Fig. 4.4 Schema for partitioning areas of a selected $ATN_1$ from $CMM(m,n,z)$ for $N$ attributes $AT_1$. 
For\(^{(2)}\) and\(^{(3)}\) we modify (4.37) to represent $GD_1(u,v,z)$ associated with $EXA(u,v)$. Thus

$$GD_1(u,v,z) \& AW(GD_1(u,v,z),EXA(u,v)) = \text{df}$$

$$(EM)(En)(w)(Ex)(Ey)(z)(w \in DI_1) \& x \in A1 \&$$

$$IP_1 \& (y \in AT_1) \& IM(x,y) \& IM(w \& x \& IP),$$

$$z) \& INF(z,(w \& x \& IP_1), IM((x \& w \& IP_1),$$

$$z)) \& SP_1 \& CG(z,SP_1, u,v) \& EXA(u,v) \&$$

$$INC((u,v), EXA(u,v))] \quad (4.48)$$

For\(^{(4)}\) we classify the data in (4.48) into the attribute data according to the fundamental terrain criteria $FTC_1$ associated with a particular terrain-problem situation $P_jT_1$. Thus, modifying (4.38) we have

$$CM(m,n,r) = (Ep)(Eq)(Ex)(Eu)(Ev)(Ew)(Ex)(Ey)(Ez)$$

$$[GD_1(p,q,z) \& AW(GD_1(p,q,z), EXA(p,q)) \&$$

$$(v \in AT_II \& CI(z,u) \& (v \in ATI) \&$$

$$AW(u,v) \& (s \in AT_1) \& AW(v,s) \& (w \in PT_1 \&$$

$$(w = P_1T_1) \& (x \in FTC) \& (x = FTC_1 \&$$

$$AW(u,x) \& AW(x,w) \& (y \in T_1) \& AW(s,r) \&$$

$$A_1G \& AW(y,A_1G) \& MC(u,r,m,n) \& EQ(m,p) \&$$

$$EQ(n,q)] \quad (4.49)$$

For\(^{(5)}\) the recoding is given in (4.44), i.e., only two $CM$'s are combined at one time to form a $CMM(m,n,z)$ with (4.46).

For\(^{(6)}\) the last $CMM(m,n,z)$ from\(^{(5)}\) undergoes an additional recoding involving prefixing the original attribute code $ATN_1$ to the new code. For example if for a given point $(m,n)$ in the old $CMM(m,n,z)$ we had $z = '211'$ and in the new $CMM(m,n,z)$ we have $u = '5'$. Then the recoding would have a value of $x = '2115'$ for that point. For this example we have

$$x = (z \ast 10) + 5$$
Thus with a function 'MULT(x, y)' reading as 'x multiplied by y' and a function 'PLUS(x, y)' reading as 'x plus y' we would have a recoding function similar to (4.44).

\[
\text{RE}(x) \equiv (\text{Eu}(\text{Ev})[\text{IM}(u \& v) \& \text{INC}(u, \text{CMM}(m, n, u)) \& \\
\text{INC}(v, \text{CMM}(m, n, v)), x = \text{PLUS}(<\text{MULT}(u, 10, v))])
\]

We would realize this by using (4.50) in (4.45). Then these are accomplished in a straightforward manner not to be shown here.

4.3.4 Time

Time can be used in many ways in terrain descriptions. Here we only mention examples without formalism.

Time must be a consideration when discussing certain natural terrain attributes. For example, soil moisture contents are associated with, among other things, the time since last precipitation; vegetation, which mantles most of the earth's surface in the form of grass, brush, and trees has seasonal, if not diurnal variations in color. Solar illumination is time dependent, but its effects at the terrain surface are also a function of sky and atmospheric conditions. Thus time affects terrain attributes in more of a random than a periodic manner, and when little historic data is available, only expected values of current attribute data can be used.

Time is important in establishing priorities for an engineering project, for sequencing operations, etc. Though in reality, problems are solved in continuum, solutions are usually represented in a discrete manner. For example, in selective sequential attribute partitioning of an area, an interpreter continuously flows from start to finish, whereas our representations are within a discrete context, i.e., a process is segmented into discrete steps and a step begins at some time \( t_n \) and ends at some later time \( t_{n+m} \).
Time is only implied in our previous formalism. Time could be expressed explicitly but we will not undertake that task here.

4.3.5 Further Comment on Concepts

The foregoing discussion of terrain descriptions related to the solution of terrain-problem situations will now aid us in the further comments and formalization of selected concepts.

We can modify (4.38) to give gross spatial cost estimates if we first relate \( z \) of \( \text{CMM}(m,n,z) \) to an index value \( \text{ICEM} \) concerned with construction effort and/or materials \( \text{CEM} \), then we can compute an estimated cost \( EC \) from the index value and a reference cost \( RC \).

\[
\begin{align*}
(\text{Ex})(\text{Ey})(\text{Ez})[R(z,x) & \land (x \in \text{ICEM}) \land (y \in \text{CEM}) \land \\
& AW(x,y) \land AW(y,P_1T_1) \land \text{INC}(z,\text{CMM}(m,n,z))] \quad (4.51)
\end{align*}
\]

since '\( R(z,x) \)' indicates '\( z \) is related to \( x \)', perhaps by a functional, we can imply \( x \) for \( z \) and when given \( z \) we can then infer \( x \). Thus

\[
(\text{Ex})(\text{Ey})[\text{INF}(x,\text{IM}(z,x),z)] \quad (4.52)
\]

and \( c \) and element or value of cost \( EC \) can be computed from \( x \) and \( RC \) by some functional, thus

\[
(\text{Ec})[\text{CF}(c,x,RC) \land (c \in \text{EC}) \land RC] \quad (4.53)
\]

then allows an inference of cost \( c \) similar to (4.52). Combining these with \( \text{CMM}(m,n,z) \) from (4.38) we have a matrix of expected costs per resolution element, \( \text{MEC}(m,n,c) \). For each \( (m,n) \) then

\[
\begin{align*}
\text{MEC}(m,n,c) = & \text{df} \ (\text{Ec})(\text{Ex})(\text{Ey})(\text{Ez})[\text{CMM}(m,n,z) \land \\
& \text{INC}(z,\text{CMM}(m,n,z)) \land R(z,x) \land (x \in \text{ICEM}) \land \\
& (y \in \text{CEM}) \land AW(x,y) \land AW(y,P_1T_1) \land \text{IM}(z,x) \\
& \land \text{INF}(x,\text{IM}(z,x),z) \land \text{CF}(c,x,RC) \land \\
& (c \in \text{EC}) \land RC \land \text{IM}(x \land RC,c) \land \\
& \text{INF}(c,\text{IM}(x \land RC,c),x \land RC)] \\& (4.54)
\end{align*}
\]

Suppose we now define the concept of terrain information as:

information \( I \), of terrain attributes, \( AT \), for solution of a terrain-problem situation, \( SN \), is defined as terrain information, \( IT \). If we
further define information as data of value for decision making, then the data contained in the coded matrix $CM(m,n,z)$ of (4.38) and the composit map matrix $CMM(m,n,z)$ of (4.46) must be considered as terrain information $IT$. These formalisms explicitly consider a specific terrain-problem situation $P_IT_j$, its fundamental terrain criteria $FTC_1$, one or more terrain attributes $AT_j$, spatial image data $DI_j$, etc., and can be considered terrain information for our purposes. Thus

$$(x \in IT) \equiv CMM(m,n,z) \quad (4.55)$$

could be substituted into (4.7) to give a more explicit definition of a solution to a terrain-related problem.

The concept of a terrain problem statement can be given by the requirement that a terrain-related problem statement, $DS(PT)$, must contain: 1) a list of problem goals, $G$, 2) a list of problem constraints, $PTC$, and 3) specification of the problem's geographic area of interest, $A_{Gj}$. Thus

$$DS(PT) = df (Ex)(Ey)[(x \in G) \& (y \in PTC \& A_{1G} \& AW(x,y) \& AW(x, A_{1G}) \& AW(y, A_{1G})] \quad (4.56)$$

If we consider the set of goals $G$ to be equivalent to a set of problem solutions $SSN$, i.e.,

$$(SN \in G) \equiv (Ex)(Ey)[(x \in G) \& (y \in SSN) \& AW(y,x)] \quad (4.57)$$

then we can define a list of problem constraints as the constants and variables in (4.38) which are necessary for solution of a $P_IT_j$ as defined in (4.20) and (4.21) and the constraints on the operations $Q$ and decisions $D$ in (4.7). With each goal $G$ in (4.57) we then have one or more problem solutions $SSN$, indicating that the problem or set of problems $P_IT_j$ can be factored into elementary terrain problems $EPT$ each having an associated fundamental terrain criteria $FTC$ as given in (4.18) and (4.22). These criteria are used to select terrain attributes $AT$ for partitioning the study area and determining the composit
map matrix $C_M(m,n,z)$ of (4.46). This is the information, according to (4.55), necessary for decision making when combined with values $ICEM$ from the problem constraints $PTC$ to yield a matrix of expected costs $MEC(m,n,c)$. The result is a solution $SSN$ and when the various solutions to the elemental terrain-related problems $EPT$ are properly combined, the goals are met and the solution $SN$ to the stated problem is found. An example is given in Section 6.1.1 to illustrate these concepts.

4.3.7 Summary

It behooves us to summarize the results of our formalism of terrain-related engineering problems.

In (4.8), (4.10), and (4.20), respectively, we defined a terrain-related engineering problem $PT$, a terrain-problem situation $PT_i$, and a specific problem in a terrain-problem situation $PT_jT_i$. In each case terrain information $IT$ is required for the solution $SN$, but we see that as the problem becomes more specific we address specific sets of fundamental terrain criteria $FTCi$ for the solution in terms of terrain attribute classes $AT$ and particular class intervals $ATII_i$. If we grid sample the terrain for a given attribute, e.g., slope, as in (4.34) we obtain $GD_1(u,v,z)$ by direct measurement, whereas if the sampled data is from an aerial image then the terrain data must be inferred to yield $GD_1(u,v,z)$ representative of terrain data in (4.37). This terrain attribute data can be partitioned according to the attribute class intervals $ATII$ associated with the fundamental terrain criteria $FTCi$ for the particular problem $PT_iT_i$ at hand. This yields the two-dimensional coded matrix $CM(m,n,z)$ in (4.38) from which a boundary coded matrix $BCM(m,n,z)$ in (4.43) can be formed for the single attributes. Coded matrices for multiple attributes can be combined by a recoding equivalence function $RE(x)$ in (4.44) to produce a discrete composite attribute map as in Fig. 4.3, or its equivalent, a composit map matrix $CMM(m,n,z)$
as in (4.46). Selected areas of this map can be sequentially partitioned to provide the required degree of refinement as in (4.49). We can then relate terrain information IT to the composit map matrices. An expected cost matrix $MEC(m,n,c)$ can be determined from $CMM(m,n,z)$ as in (4.54). If the problem has more than one plausible solution we now could choose a "best" solution based upon minimum cost criteria.
CHAPTER V

TERRAIN ORGANIZATION

The previous chapter addressed terrain-related engineering problems with major emphasis on considerations necessary for formulation of problem concepts. In this chapter we will be interested in terrain organization for civil engineering purposes. Our objective will be a formulation of terrain classification for engineering purposes.

Following the formalization of natural and geomorphic processes, and terrain materials, we define terrain features based upon the ability to delineate and partition terrain according to its attributes. A terrain component is defined to be a partition of a terrain feature associated with distinct attributes. Then a unit landform is defined in terms of its terrain features and natural processes responsible for the attribute. The genetic link between the natural and geomorphic processes and the physical characteristics of the terrain materials is defined along with the inferences which may be known about terrain materials associated with landforms. The physiographic landform hierarchy is defined and terrain classification is shown to be the classification of terrain descriptions. The formalism concludes with the definition of the three approaches to engineering terrain classification. The pragmatic approach is primarily an ad hoc, problem-oriented, adaption of genetic concepts based upon three premises for solution of a terrain-related engineering problem. The parametric approach is a general problem-oriented approach similar to that defined in Chapter 4. The genetic approach is general and non-problem-oriented, and serves to partition terrain to supply information but not answers for
problems.

5.1 Characterization of Geomorphic Processes and Terrain Materials

5.1.1 Introduction

Basic to the concept of terrain organization are the concepts of terrain description and terrain classification. We define terrain classification as the classification of qualitative and/or quantitative terrain descriptions into unique categories. In Section 4.3 we discuss aspects of terrain descriptions related to terrain-related engineering problems and their relation to terrain data, space and time. Our discussions were mainly concerned with the partitioning of terrain data $D_T$ from one or more matrices into a coded matrix $CM(m,n,z)$ based upon fundamental terrain criteria $FTC_j$ for a given problem situation $P_jT_j$. This amounted to sorting or classifying the terrain data into preselected terrain attribute class intervals $ATII$ and producing a coded matrix $CM(x,y,z)$ with a matrix coding function $MC(u,z,x,y)$. Thus for each point $(x,y)$ data which belonged to class interval $u$ were classed as $z$. In all cases our criteria $FTC_j$, class intervals $ATII$, etc. were numerical data descriptions $DS$ of the terrain $T$. Though well suited for computer manipulation and classification, these descriptions are not common to an engineer's terrain description language $L$. An engineer uses quantitative data to solve problems but qualitative data to organize terrain from a background in geomorphic processes and their related terrain materials. We are now interested in formalizing these concepts.

5.1.2 Earth Materials

The four essential parts of the earth $EARTH$ are the lithosphere $LITHSP$, the hydrosphere $HYDSP$, the atmosphere $ATMSP$, and the biosphere $BIOSP$. Thus we can define the earth $EARTH$ by these four components:

$$(x \in EARTH) =_{df} (x \in LITSP) \lor (x \in HYDSP) \lor (x \in ATMSP) \lor (x \in BIOSP)$$
\[(x \in \text{BIOSP})\] \hspace{1cm} (5.1)

The engineer is mainly interested in the earth materials \(EM\) at or near the surface of the lithosphere \(\text{LITSP}\). These are the rocks \(RX\), the soil \(SO\), the water \(HTO\), and vegetation \(VEG\). We shall use them as components to define a member of the set of earth materials \(EM\) as

\[(x \in EM = \text{df} (x) [(x \in RX) \lor (x \in SO) \lor (x \in HTO) \lor (x \in VEG)] \] \hspace{1cm} (5.2)

where

\[(x \in \text{LITSP}) = \text{df} (x) [(x \in RX) \lor (x \in SO)] \] \hspace{1cm} (5.3)

and the water \(HTO\) of interest is a portion of the hydrosphere \(HYDSP\), thus

\[HTO \subseteq HYDSP\] \hspace{1cm} (5.4)

therefore

\[(x) [(x \in HTO) \lor (x \in HYDSP)] \] \hspace{1cm} (5.5)

and likewise, the vegetation \(VEG\) is a part of the biosphere \(BIOSP\), thus

\[VEG \subseteq BIOSP\] \hspace{1cm} (5.6)

therefore

\[(x) [(x \in VEG) \lor (x \in BIOSP)] \] \hspace{1cm} (5.7)

therefore

\[(x) [(x \in EM) \lor (x \in \text{EARTH})]\] \hspace{1cm} (5.8)

Defining earth materials \(EM\) as being composed of rocks \(RX\) and soil \(SO\) and water \(HTO\) and vegetation \(VEG\) is too general for this discussion, thus we will neglect the water materials \(HTO\) and the vegetation \(VEG\). Then (5.2) becomes

\[(x \in EM = \text{df} (x) [(x \in RX) \lor (x \in SO)] \] \hspace{1cm} (5.9)

In essence we are saying we are only interested in the materials of the lithosphere \(\text{LITSP}\) since the right sides of (5.3) and (5.9) are identical the left sides must be equivalent. To be able to say only that a sample of earth material \(EM\) is either soil \(SO\) or rock \(RX\) is not
definitive enough for engineering needs. We thus look to the geological sciences for a better definition of both soil SO and rock RX.

5.1.3 Geological Origin

Given a point anywhere on the lithosphere's surface we can say that the earth materials EM at that point \((x,y)\) are related to its geological origin GEOO. For our purposes we consider geological origin synonymous with geomorphological origin with the following definition.

Geological origin GEOO is defined as a natural process NP, or combination of natural processes, responsible for the formation of earth materials EM having typically different repetitive characteristics AT related to such items as topography TOPO, mode of occurrence MO, composition COMP, lithology LITH, etc.

This can be formalized as

\[
(v \in GEOO) = \text{df} \ (Ew)(Ex)(Ey)(z)[(w \in NP) \& (x \in EM) \& (y \in AT) \& \\
RES(w,x,y) \& ((z = TOPO) \vee (z = MO) \vee (z = COMP) \vee \\
(z = LITH) \vee \ldots) \& R(y,z)]
\]

which reads as 'Geological origin GEOO is defined as some (one or more) natural processes NP responsible for some terrain materials EM having terrain attributes AT related to topography TOPO, and/or mode of occurrence MO, and/or composition COMP, and/or lithology LITH and/or etc.' This definition describes the total set of geological origins GEOO, whereas a particular geological origin GEOO, e.g., glacial or fluvial origin, may be defined as

\[
(GEOO_1 \in GEOO) \equiv (Ex)[(x \in GEOO) \& (x = GEOO_1)]
\]

which indicates the definition of GEOO to be the same as (5.10) except that for each GEOO there is a particular subset of natural processes NP and terrain attributes AT. These will become apparent as the discussion progresses.
Rocks. All rocks $\mathbb{R}X$ can be divided into three groups based upon their origin, these often called the 'great rock groups.'

\[(x)(x \in \mathbb{R}X) \equiv (x)[(x \in \mathbb{R}I) \lor (x \in \mathbb{R}S) \lor (x \in \mathbb{R}M)] \tag{5.12}\]

expresses the division of $\mathbb{R}X$ into three disjoint classes; igneous rocks $\mathbb{R}I$, sedimentary rocks $\mathbb{R}S$, and metamorphic rocks $\mathbb{R}M$.

The information given by an element of one of the three classes is, of course, greater than that contained in an element of the parent class. For our purposes this information difference is due to the association with a particular geological origin $\mathbb{GEOO}_i$. Therefore, given any arbitrary piece of rock, we can express its group classification if we know its geological origin $\mathbb{GEOO}_i$.

\[(x)(y)(z)([(x \in \mathbb{R}X) \land (y \in \mathbb{GEOO}_i) \land \mathbb{AW}(x,y)] \to (\mathbb{CL}(x,z,y) \land ((z = \mathbb{RI}) \lor (z = \mathbb{RS}) \lor (z = \mathbb{RM})))] \tag{5.13}\]

where the relation '$\mathbb{CL}(x,z,y)$' reads as 'x is classified as z given y.' This is shown at the first level of the graph in Fig. 5.1. An additional level of classification, based only on additional geological origin information is shown at the second level of the graph. We can express this sub-group classification at the second level in terms of the group classification at the first level and additional geological origin information in a manner similar to (5.13). For example,

\[(x)(y)(z)([(x \in \mathbb{RI}) \land (y \in \mathbb{GEOO}_i) \land \mathbb{AW}(x,y)] \to (\mathbb{CL}(x,z,y) \land ((z = RIP) \lor (z = RIV)))] \tag{5.14}\]

Below the second level of the graph, rocks $\mathbb{R}X$ are classified by rock type $\mathbb{R}XT$ when given additional information for the type of source $\mathbb{PEM}$ and environment during formation $\mathbb{ENV}$ in addition to geological origin $\mathbb{GEOO}_i$. Our formalism here will not continue the hierarchy of (5.13) and (5.14); instead we will define a member of a rock type $(z \in \mathbb{R}XT)$ as a classified rock $\mathbb{R}X$, then

\[(z \in \mathbb{R}XT) =_{df} (v)(w)(x)(y)(z)([(v \in \mathbb{R}X) \land (w \in \mathbb{GEOO}_i) \land \mathbb{AW}(x,y)] \to ((v \lor \mathbb{CL}(x,z,y) \land ((z = RIP) \lor (z = RIV)))] \tag{5.14}\]
Universe of consolidated natural materials

Three great rock groups based on geological origin GEOO_1 information

Subgroups based upon additional GEOO_1 information

Principle rock types based upon additional information of GEOO_1, PEM, and ENV

Symbol Key:

<table>
<thead>
<tr>
<th>RX</th>
<th>Rocks</th>
<th>RXT</th>
<th>Rock Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>I.</td>
<td>RI</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A.</td>
<td>RIP</td>
<td>RIG</td>
<td>Granite</td>
</tr>
<tr>
<td>B.</td>
<td>RIV</td>
<td>RIL</td>
<td>Lava</td>
</tr>
<tr>
<td>II.</td>
<td>RS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A.</td>
<td>RSF</td>
<td>RSS,</td>
<td>Sandstone, Shale</td>
</tr>
<tr>
<td>B.</td>
<td>RSC</td>
<td>RSL</td>
<td>Limestone</td>
</tr>
<tr>
<td>III.</td>
<td>RM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A.</td>
<td>RMI</td>
<td>RMG</td>
<td>Gneiss</td>
</tr>
<tr>
<td>B.</td>
<td>RMS</td>
<td>RMS</td>
<td>Schist</td>
</tr>
</tbody>
</table>

Fig. 5.1 Hierarchy of rock classification based upon geological origin with example rock types.
with \[\text{AW}(v, w \in x \in y)\] and \[\text{CL}(v, z, w \in x \in y)\] indicating the interrelated nature of the information for classification.

5.1.4 Landform

A unit landform LF is a terrain feature TF or terrain habit TH created by natural processes NP in such a way that it may be described DS(LF) and recognized ID(LF) in terms of typical attributes AT wherever it may occur, and which, then identified, provides information IT concerning its own structure STRR and composition or texture TXTR. Discussion of this definition will be treated in two parts; the first being the definition of LF and the second as the implications related to landform identification ID(LF). We must begin with an explanation of the terms 'terrain feature' TF and 'terrain habit' TH. A terrain feature TF is a distinct natural topographic form, such as a kame, a terrace, a cirque, etc., which can be delineated and partitioned into terrain components TC according to their shape attributes AT. 'Delineate' and 'partition' are similar functionals, DEL(x) and PART(x), respectively. To speak of the process of delineating or delimiting we must speak about a boundary BNDY based upon differentiating or classifying attributes AT of the thing interior to the boundary from the surrounding things not within the boundary. Thus,

\[
\text{DEL}(w) = \text{df} (Ex)(Ey)(Ez)[(x \in \text{BNDY}) \& \text{INT}(w, x) \& (y \in T_1) \& (z \in AT_1) \& \text{CL}(y, w, z)]
\]

Then to delineate \(w\) for terrain \(T_1\) involves some boundary \(\text{BNDY}\) with \(w\) on the interior and attributes \(AT_1\) which allows \(T_1\) to be classified as \(w\) given \(AT_1\). Partitioning involves recursive delineation

\[
\text{PART}(w) = \text{df} (Ex)(Ey)(Ez)[(x \in \text{BNDY}) \& \text{INT}(w, x) \& (y \in T_1) \& (z \in AT_1) \& \text{CL}(y, u, z) \& \text{CL}(u, w, z = AT_{11} \vee AT_{12}) \& (AT_{11} \in AT_1) \& AT_{12} \in AT_1)]
\]
Thus, subsets $AT_{11}$ and $AT_{12}$ of the attribute $AT_1$ used to delineate $w$ are used to partition $w$. Then a terrain feature $TF$ is equivalent to

$$(x \in TF) \equiv (Ex)(Ey)[(y \in T_1) \& \text{DEL}(x) \& \text{PART}(x) \& \text{AW}(x,y)]$$  (5.18)

and a terrain component $TC$ is then a partition of $TF$

$$(x \in TC) \equiv (Ex)(Ey)[(y \in TF) \& \text{PART}(x) \& \text{AW}($$ \text{PART}(x),y)]$$  (5.19)

Terrain habit $TH$ is defined by (5.16) through (5.19) when $TH$ is substituted for $TF$. The difference between a terrain feature $TF$, which is a well defined topographic form, and a terrain habit $TH$ is that the boundaries and partitions of $TH$ are usually ill-defined, or are not included in the study area.

We will use the following terminology associated with 'landform': the 'total set of landforms' will be designated by '$LFS$', a general member of $LFS$ will be called a 'unit landform' or simply a 'landform' with the designation '$LF$', while a 'specific landform' will be named, e.g., kame or esker, and will be symbolized as $LFS_i$. Thus the symbolism is

$$(LF \in LFS) \equiv (Ex)[(x \in LFS) \& (x = LF)]$$  (5.20)

$$(LF_1 \in LFS) \equiv (Ex)[(x \in LFS) \& (x = LF_1)]$$  (5.21)

Then a definition of a unit landform $LF$ in general could be

$$(w \in LF =df (Ex)(Ey)(Ez)[((x \in TF) \lor (x \in TH)) \& (y \in NP) \& (z \in AT) \& \text{RES}(y,x,z))]$$  (5.22)

which says that a landform $LF$ is a terrain feature $TF$ or terrain habit $TH$ with the terrain attributes $AT$ resulting from one or more natural processes $NP$. Here the terrain attributes $AT$ has the general meaning pertaining to any attribute of the landform, e.g., surface configuration, size, internal structure, composition, etc. Description of a landform $DS(LF)$ could then be a description of its associated terrain components which could be a listing of attributes $AT$ associated with the terrain components $TC$ that are the parts of the landform $LF$. This
in general is an insufficient description for recognition and identification ID(LF) of a landform. A sufficient description contains a description of the parts plus a description of the spatial arrangement of the parts. We will use a simple but adequate descriptor of spatial arrangement or the parts; the contiguity relation CON(x,y).

At this point in our discussion of terrain description we will adopt the following definition of landform description:

\[
DS(LF) = \text{df} \ (Ew)(Ex)(Ey) \ [ (w \in \text{AT}) \& (x \in \text{TC}) \& (y \in \text{TC}) \& (z \in \text{LF}) \& \text{AW}(w,x) \& \text{AW}(x,z) \& \text{CON}(x,y) \& \text{EQ}(x,-y)] \tag{5.23}
\]

Identification of the landform ID(LF) is a similar definition with a specific reference to terrain \( T_1 \), i.e.,

\[
ID(LF) = \text{df} \ (Ev)(Ew)(Ex)(Ey) \ [ (v \in \text{TE}_1) \& (w \in \text{AT}) \& (x \in \text{TC}) \& (y \in \text{TC}) \& \text{AW}(v,w) \& \text{AW}(w,x) \& \text{AW}(x,z) \& \text{CON}(x,y) \& \text{EW}(x,-y)] \tag{5.24}
\]

Thus identification ID(LF) involves matching the description to the specific terrain \( T_1 \).

Presently we will discuss natural processes NP and geomorphic processes GP. Here in our discussion on landforms we want to indicate the relation between the landform LF and the characteristics of the terrain materials EM contained within them.

Landforms LF, because of their genetic link to the morphological processes GP, provides information related to terrain materials EM, their texture TXTR and their structure STRR. Thus,

\[
(Ex)(Ey)(Ez) \ [ (x \in \text{GEOO}_1) \& \text{IM}(x,y) \& (z \in \text{GP}) \& \text{IM}(y,z)] \tag{5.25}
\]

says that a landform LF implies a geologic origin GEOO which in turn implies one or more geologic processes GP. By the chain rule of inference this indicates that the landform LF implies the geologic process GP, i.e., IM(LF, GP). And subsequently we shall show the geologic
processes \( GP \) to be responsible for the terrain material \( EM \) having the physical characteristics related to texture \( TXTR \) and structure \( STRR \), i.e., \( RES(GP,EM,TXTR&STRR) \). If this is tentatively accepted, then the following statement indicates that landform \( LF \) implies texture \( TXTR \) and structure \( STRR \) of its materials \( EM \).

\[
(Ev)(Ew)(Ex)(Ey)(Ez)[(v \in LF) \& (w \in GP) \& \text{IM}(v,w) \& (x \in EM) \& \text{CI}(x,v) \& (y \in TXTR) \& (z \in STRR) \& \text{RES}(w,x,y,z) \text{ IM}(v,y,z)] \tag{5.26}
\]

It should therefore be evident that given (5.26) to be valid, along with a means of identifying that a landform does exist \( ID(LF) \) from (5.24), then we could draw conclusions about the properties of the terrain materials contained in that landform.

**Landform classes.** We now want to look closer at the natural processes \( NP \) responsible for landform formation or origin \( LF0 \). Every landform can be classified as either constructional \( LFC \) or residual \( LFR \). Thus

\[
(x)(Ey)[(x \in LFS) \& \text{CL}(x,y) \& ((y = LFC) \lor (y = LFR))] \tag{5.27}
\]

Constructional landforms \( LFC \) contain materials \( EM \) produced by constructive natural processes \( NPC \).

\[
(Ex)(Ey)(Ez)[(x \in NPC) \& (y \in EM) \& (z \in LFC) \& \text{RES}(x,y,z)] \tag{5.28}
\]

Here the natural constructive processes \( NPC \) are mainly the processes related to accumulation \( NPCD \) or diastrophism \( GPCS \). The process of accumulation or deposition \( NPCD \) pertain mainly to unconsolidated terrain materials, i.e., \( SO \), and the diastrophism processes \( GPCS \) usually pertain to rock \( RX \). These ideas can be incorporated into (5.28) as

\[
(Ex)(Ey)(Ez)[((x \in NPCD) \& (y \in SO)) \lor ((x \in NPCS) \& (y \in RX)) \& (z \in LFC) \& \text{RES}(x,y,z)] \tag{5.29}
\]

The residual landforms \( LFR \) result from removal of earth materials \( EM \), i.e., soil \( SO \) or rock \( RX \), by a destructive natural process \( NPD \).

\[
(Ex)(Ey)(Ez)[(x \in NPD) \& (y \in -EM) \& (z \in LFR) \& \text{RES}(x,y,z)] \tag{5.30}
\]
The destructive natural processes $NPD$ are mainly the processes of erosion $NPDE$, certain phases of weathering $NPW$, and the destructive phases of volcanism $GPDV$. These processes operate on all types of earth materials $EM$, therefore we cannot associate a material with a process as in (5.29) but we can state

$$(Ex)[(x \in NPD) \equiv (x \in NPDE) \lor (x \in NPW) \lor (x \notin GPDV)] \tag{5.31}$$

and the right side of the equivalence in (5.31) can replace the left side of (5.31) in (5.30).

The engineer is mainly interested in the constructional landforms $LFC$ because his terrain-related projects interface with the materials in these landforms. His interest in residual landforms $LFR$ relates mainly to inferences which can be drawn about the properties of the material removed by the destructive processes $NPD$.

5.1.5 Natural and Geomorphic Processes

We were careful in defining geologic origin $GEOO$ in (5.10) and landform $LF$ in (5.22) so as to indicate that natural processes $NP$ were responsible for the origin and the landforms. The major natural processes related to landform origin $LFO$ are weathering $NPW$, erosion $NPDE$, and deposition $NPCD$. Combinations of these complex natural processes $NP$ form the geomorphic processes $GP$. Before explaining this further we must define the three natural processes.

**Weathering.** The natural weathering process $NPW$ involves mechanical processes $NPWM$ and chemical processes $NPWC$ acting on the surface materials $EM$ to cause disintegration $DNG$ and decomposition $DCP$, respectively. Thus,

$$(w \in NPW) =df (Ex(Ey)(Ez)[((x \in NPWM) \lor (y \in NPWC)) \land$$

$$(z \in EM) \land (RES(x,z,DNG(z)) \lor RES(y,z,DCP(z))]) \tag{5.32}$$

Mechanical and chemical weathering are independent, i.e., the product of mechanical weathering is disintegration of the materials $DNG(EM)$.
with no chemical change, whereas the product of chemical weathering involves inplace chemical change of the minerals or decomposition DCP(EM), but both are present everywhere and the importance of each is dependent mainly upon the climate.

Erosion. Erosion NPDE is a complex natural destructive process NPD which concerns the excavation or removal EXC(EM) and transportation TRA(EM) of surface materials EM. Thus,

\[(w \in \text{NPDE}) = df (Ex)(Ey) [(x \in \text{NPD}) \& (y \in \text{EM}) \& RES(x,y,EXC(y)\&TRA(y))]\]  \hspace{1cm} (5.33)

Erosion NPDE produces residual landforms LFR;

\[(Ex)(Ey)(Ez) [(x \in \text{NPDE}) \& (y \in \text{T}) \& (z \in \text{LFR}) \& RES(x,y,z)]\]  \hspace{1cm} (5.34)

Erosion processes NPDE can be grouped by the agents TRAA that cause the transportation TRA(EM). These are water HTO, windWND, ice ICE, volcanism VO, and gravity GR. A fluvial erosion process GPFE may then be defined as

\[(v \in \text{GPFE}) = df (Ew)(Ex)(Ey)(Ez) [(w \in \text{NPDE}) \& (x \in \text{GP}) \& (y \in \text{TRA}) \& (y = \text{HTO}) \& (z \in \text{EM}) \& RES(y,TRA(z),w\&x)]\]  \hspace{1cm} (5.35)

and a glacial erosion process GPGE may be defined as

\[(v \in \text{GPGE}) = df (Ew)(Ex)(Ey)(Ez) [(w \in \text{NPDE}) \& (x \in \text{GPG}) \& (y \in \text{TRA}) \& (y = \text{ICE}) \& (z \in \text{EM}) \& RES(y,TRA(z),w\&x)]\]  \hspace{1cm} (5.36)

We can define the aeolian erosion process GPAE for (TRA = WND); the volcanic erosion process GPVE for (TRA = VO); and the colluvial erosion process GPCE for (TRA = GR). If the transportation was (TRA = HTO) in a marine environment, i.e., saltwater, then the marine erosion process GPME would replace the fluvial erosion process GPFE in (5.35) and if it was a coastal environment where both marine and
fluvial erosion processes were operative, the definition would be a combination of both processes.

**Deposition.** Deposition \( \text{NP}_{\text{CD}} \) is a complex natural constructional process \( \text{NP}_{\text{C}} \) which results in deposits \( \text{DP} \) of natural materials \( \text{EM} \) settling in bodies \( \text{SEDBD} \) of water \( \text{HTO} \), ice \( \text{ICE} \), and air \( \text{AIR} \). We will not be concerned here with chemical precipitation of materials.

Our formalism is then

\[
(v \in \text{NP}_{\text{CD}}) \equiv (w \in \text{NP}_{\text{C}}) \& (x \in \text{EM}) \& (y \in \text{DP}) \& (z \in \text{SEDBD}) \& \text{RES}(w,gz,x,CI(x,y))
\]

(5.37)

This may be read as 'the deposition process \( \text{NP}_{\text{CD}} \) is defined as a combination of a natural constructional process \( \text{NP}_{\text{C}} \) and a sedimentation environment \( \text{SEDBD} \) which is responsible for the terrain materials \( \text{EM} \) being contained in deposit \( \text{DP} \).

Deposition processes produce constructional landforms \( \text{LFC} \)

\[
(\text{Ex})(\text{Ey})(\text{Ez})[(x \in \text{NP}_{\text{CD}}) \& (y \in \text{T}) \& (z \in \text{LFC}) \& \text{RES}(x,y,z)]
\]

(5.38)

Deposition processes can be classed according to their sedimentation environment \( \text{SEDBD} \) and their geomorphic processes \( \text{GEOP} \), similar to erosional processes \( \text{GPFE} \) and \( \text{GPGE} \) in (5.35) and (5.36), respectively.

A fluvial deposition process \( \text{GPFD} \) may then be defined as

\[
(u \in \text{GPFD}) \equiv (v \in \text{NP}_{\text{CD}}) \& (w \in \text{GPF}) \& (x \in \text{SEDBD}) \& (x = \text{HTO}) \& (y \in \text{EM}) \& (z \in \text{DP}) \& \text{RES}(x,CI(y,z),v,gw)
\]

(5.39)

Hence, we have defined the three natural processes \( \text{NP}_{\text{W}} \), \( \text{NP}_{\text{DE}} \), and \( \text{NP}_{\text{CD}} \) and by symbolism have indicated the association of the geomorphic processes \( \text{GEOP} \) to each. We see from (5.35), (5.36) and (5.39) that the geomorphic processes \( \text{GEOP} \) are descriptive modifiers of their associated natural processes. We could now return to (5.10) and substitute the appropriate combinations of \( \text{NP}_{\text{W}}, \text{NP}_{\text{DE}}, \) and \( \text{NP}_{\text{CD}} \) for \( \text{NP} \) to
obtain a more definitive expression of geologic origin GEOO. Similarly, we could better define landform LF in (5.22).

5.1.6 Landform Origin

When speaking about landform origin we should describe the three natural processes and their associated geomorphic processes. This might be represented as

\[
(v \in \text{LFO}) \overset{df}{=} (Ew)(Ex)(Ey)(Ez)(w \in \text{LF}) \& (x \in \text{NPW}) \& (y \in \text{NPDE}) \& (z \in \text{NPCD}) \& R(w,x,y,z) \quad (5.40)
\]

Thus if we were to describe the origin of a particular landform we might say that the landform results from the combination of natural weathering, erosion, and deposition processes. And if we wished to be more definitive we could substitute (5.32), (5.33), and (5.37) properly into (5.40). Though this may be proper, the method is seldom used in practice. Instead, a restricted interpretation of landforms origin is either implied by the landform name or an appended origin descriptor is used in conjunction with the landform name.

Given a list of geomorphic processes GEOP, the constructional and residual landform sets can be partitioned by origin. For example, 'kame' is a constructional landform with an implicit glacial origin; 'cirque' is a residual landform with an implicit glacial origin, whereas 'terrace' and 'valley' are terrain feature names TF which could have more than one geological origin GEOO, but 'fluvial terrace' or 'kame terrace' are landforms with fluvial GPF and glacial GPG origins, respectively. Likewise 'fluvial valley' or 'glacial valley' are residual landforms with explicit origins. The pattern should now be clear; landform origin LFO has been used to indicate the geomorphic process GEOP which accomplishes the erosion and/or transportation of the earth materials.

Therefore where origin is implied
reads as 'a landform name LF implies the geomorphic process GEOP is responsible for the origin LFO of the landform LF.' For the other case, when an origin process GEOP must be appended to the terrain feature name TF, a landform name expression which satisfies (5.41) is

\[(x \in LF) = (Ey)(Ez)[(y \in GEOP) \& (z \in LF) \& AW(y,z)] \quad (5.42)\]

This reduces ambiguity and allows partitioning of both constructional and residual landform sets based upon geomorphic processes or origin. Thus this more limited sense of landform origin LFO may be defined as

\[[(w \in LFO) \equiv (w \in GEOP) \equiv (Ex)(Ey)(Ez)[(x \in LF) \& (y \in NPDE) \& (z \in GEOP) \& AW(x,y) \& AW(y,z) \& AW(x,z)] \quad (5.43)\]

5.1.7 Soils

Soils SO are earth materials EM that result from natural weathering processes NPW acting on rock RX. This can be symbolized as

\[(w \in SO) = (Ex)(Ey)(Ez)[(w \in EM) \& (x \in RX) \& (y \in NPW) \& RSFG(x,y,z)] \quad (5.44)\]

with 'RSFG(x,y,z)' being the relation which reads 'x results from y given z.' A more descriptive definition is obtained by substituting the definition of a natural weathering process NPW from (5.34) into (5.43) and the appropriate rock type RX replacing the general rock term 'RX' to obtain

\[(w \in SO) = (Ex)(Ey)(Ey)(Ez)[(w \in EM) \& (x \in RX) \& ((y \in NPWM) \lor (z \in NPWC)) \& RSFC(w,x,(RES(y,w,DNG(x)) \lor RES(z,w,DCP(x)))))] \quad (5.45)\]

which reads as 'soil SO is defined as the earth materials EM resulting from rock RX given: 1) that mechanical weathering NPWM is responsible for the earth material being disintegrated rock DNG(RX) or 2) that
chemical weathering \(NPWC\) is responsible for the earth materials \(EM\) being decomposed rock \(DCP(RX)'\).

**Residual soil.** A first approximation to the definition of residual soil \(SOR\) is given by (5.45), but a better definition would concern soil weathering. Thus, first phrasing the definition similar to (5.44)

\[
(w \in SOR) = df (Ex)(Ey)(Ez)[(x \in SO) \& (y \in RXT) \& (z \in NPW) \& PARM(y,x) \& RSFG(x,x\&y,z)]
\]

(5.46)

which says the rock type \(RXT\) is the parent material of the soil \(SO\) which results from the weathering of the rock and the weathering of the soil. The relation \(PARM(y,x)'\), indicating that \(y\) is the parent material of \(x\) is the common method of indicating that \(x\) is derived from \(y\) without reference to erosion and deposition processes but with the general process of weathering \(NPW\) implied. Again a more descriptive definition is obtained by substituting the definition of the natural weathering process \(NPW\) from (5.34) into (5.46) to obtain

\[
(v \in SOR) = df (Ev)(Ew)(Ex)(Ey)(Ez)[(v \in SO) \& (w \in SOR) \& (x \in RXT) \& ((y \in NPWM) \& (z \in NPWC)) \& PARM(x,w) \& RSFG(w,w\&x,(RES(y,w,\text{NG}(w\&x)))\& RES(z,w,DCP(w\&w))))]
\]

(5.47)

**Transported soils.** Soils \(SO\) resulting from a sequence of erosional \(NPDE\) and depositional \(NPCD\) processes are classed as transported soils \(SOT\). The deposits \(DP\) upon which erosion \(NPDE\) acts can contain either residual soils \(SOR\) or previously transported soils \(SOT\).

\[
(u \in SOT) = df (Ev)(Ew)(Ex)(Ey)(Ez)[(v \in SO) \& (w \in SOR) \& (x \in SOT) \& (y \in NPDE) \& (z \in NPCD) \& RSFG(v,z,RSF(EXC(wvx)\&TNSP(wvx),y))]
\]

(5.48)

This says that a transported soil \(SOT\) is a soil \(SO\) resulting the natural process of deposition \(NPCD\) given that the excavation and transportation of the residual soil \(SOR\) or transported soil \(SOT\) resulted from the
natural process of erosion NPDE. This allows us to talk about different erosional and depositional sequences. For example, consider an original residual soil deposit DPR subjected to glacial erosion GPGE and deposition GPGD and subsequently subjected to fluvial erosion GPFE and deposition GPFD.

Soil properties. The engineering properties of soils are those properties related to engineering performance, such as bearing capacity, shear strength, consolidation properties, etc. The physical properties most often correlated with engineering properties concern texture of the soil, structure of the soil, soil moisture, and mineral composition. Soil texture TXTR and structure STRR are of importance to us here because natural processes NP allow qualitative prediction of texture and structure of soils.

Texture TXTR pertains to the soil particle sizes and shapes TXTRS, including size classification of soil such as the Unified Soil Classification System USCS. Structure STRR pertains to particle uniformity STRU, particle cohesiveness STRC, and mass density STRD. We will refer to these as the general class of physical characteristics of soils PHCHS. Further we can use these structure properties STRR to describe the physical characteristics of rock PHCHR. Thus a definition for the physical characteristics of soil is

\[(w \in PHCHS) =_{df} (Ex)(Ey)(Ez)[(x \in SO) \& (y \in TXTR) \& (z \in STRR) \& PRYO(y,z,x)] \quad (5.49)\]

with 'PRYO(y,z,x)' being the relation 'y and z are properties of x' and the set memberships for TXTR and STRR being

\[(x \in TXTR) \equiv (Ey)(Ez)[(y \in USCS) \& (z \in TXTRS) \& AW(x,y,z)] \quad (5.50)\]

and

\[(w \in STRR) \equiv (Ex)(Ey)(Ez)[(x \in STRU) \& (y \in STRC) \& \]
which can be substituted back into (5.49).

Physical properties PHCHS of residual soils SOR contained in residual deposits DPR are mainly related to the parent rock types RXT and the process of weathering NPW. Thus we can state

\[(Ev)(Ew)(Ex)(Ey)(Ez)[(v \in PHCHS) \& (w \in SOR) \& (x \in DPR) \& (y \in RXT) \& (z \in NPW) \& PARM(y,w) \& R(PRTYO(v,x)\& CI(w,x),RXT\&NPW)]\] (5.52)

This can be expanded in a definition manner by substituting (5.49), (5.50) and (5.51) for PHCHS; (5.47) for SOR; and (5.32) for NPW. Similarly we might attempt to define the relation between soil properties PHCHS and rock properties PHCHR for use in (5.52).

Physical properties PHCHS of transported soils SOT contained in constructional soil deposits DPC are mainly related to the processes of erosion NPDE and deposition NPCD. This we can state as

\[(Ev)(Ew)(Ex)(Ey)(Ez)[(v \in PHCHS) \& (w \in SOT) \& (x \in DPC) \& (y \in NPDE) \& (z \in NPCD) \& R(PRTYO(v,w)\& CI(w,x),y\&z)]\] (5.53)

Expansion is accomplished by substituting (5.49), (5.50), and (5.51) for PHCHS; (5.48) for SOT; and (5.33) and (5.37) for NPDE and NPCD, respectively. Additionally (5.35), or a similar expression for a geomorphic erosional process, may be substituted for the natural destruction process NPD of (5.33); and (5.39) or a similar expression for a geomorphic destruction process may be substituted for the natural construction process NPC of (5.37). As an example of this type of expansion we use (5.35), (5.39), (5.47), (5.49), (5.50), and (5.51) for substitution into (5.53) with the following result.

\[(Ez)(Eb) \ldots (En)[(a \in SO) \& (b \in USCS) \& (c \in TXTRS) \& AW(a,b\&c) \& (d \in STRU) \& (e \in STRC) \& (f \in STRD) \&\]
\( AW(a,d\infty e,f) \land PRTY0(b\infty c\infty d\infty e\infty f,a) \land ((g \in SOR) \lor (h \in SOT) \land (i \in GPFE) \land (j \in GPF) \land (k \in TRAA) \land (k = HTO) \land RES(k,TRA(a),i\&j) \land (l \in GPFD) \land (m \in GPF) \land (k \in SEDBD) \land (n \in DP) \land RES(k,DI(a,n),l\&m) \land RSFG(a,l\&m,RSF(EXC(gvh),i\&j)) \land R(PRTY0(b\infty c\infty d\infty e\infty f,a)\&CI(a,n),(i\&j)\&(l\&m))) \)  

(5.54)

(5.52) and (5.53) indicate the relation between the physical characteristics \( PHCHS \) of the soils \( SO \) and the natural \( NP \) and geomorphic \( GNP \) processes which are responsible for the deposits \( DP \) containing the soils \( SOT \). This then is an explanation and justification for the implication in (5.26) when we note that \( (DP = LF) \).

We now add a few heuristic comments. First, no two soil deposits resulting from the same geomorphic processes are identical. However, they usually have similarities in terms of their physical properties when contrasted to those physical properties of a deposit resulting from a different set of geomorphic processes. For example, any two aeolian sand dunes anywhere on earth will have a set of physical characteristics which match more closely than when the properties of an aeolian sand dune and any glacial terminal moraine are compared. Second, the information related to the physical properties of a soil contained in a deposit formed by any geomorphic process is not directly related to the exterior configuration of the deposit, i.e., physical measurements of the deposit will not directly reveal the physical properties of the materials. However, by associating the deposit shape, i.e., surface configuration pattern for the terrain feature, with a landform type, a geomorphic process can be inferred, which in turn allows inference as to the qualitative physical characteristics of the deposit. Or equivalently, if we know the origin of a terrain feature and we can associate its shape properties with a landform type, then the qualitative physical properties associated with that landform type
may be inferred for the terrain feature.

Thus we have come the full circle; in (5.18) we defined a terrain feature TF based upon the ability to delineate and partition T according to attributes AT. In (5.19) terrain component TC was given to be a partition of distinct portion of a terrain feature TF with distinct attributes AT. Then in (5.22) we defined a unit landform LF in terms of its terrain features TF and natural processes NP responsible for the attribute AT. (5.25) indicated the genetic link between the geomorphic process GP and the physical characteristics of unconsolidated terrain materials PHCHS and (5.26) pertained to inferences which could be drawn about terrain materials EM associated with landforms LF.

5.1.8 Physiographic Units

A physiographic unit PP or PS of the earth's surface is an area A within which the major topographic features LF2 have a single geomorphic history GEOH, a definite general structure STRR, certain physical characteristics PHCH, and a predictable general pattern of lower order landforms LF3. This defines a physiographic unit, i.e., a physiographic province PP or a physiographic section PS, in terms of an area A containing major topographic features LF2, e.g., mountains, plains, plateaus, etc., with a single geomorphic history GEOH. Lobeck (1939) classified the continental areas as first order landforms LF1 and major quasi-homogeneous divisions of continents were physiographic provinces PP containing major topographic features, termed as second order landforms LF2. The geomorphic history GEOH is a time ordering of the geomorphic processes GP which acted in area A as a whole, e.g., tectonics, glaciation, marine embayment, etc. These major geomorphic processes were responsible for the basis character of the landscape within the province. Phases or variations of the major processes GP are responsible for variations in general terrain structure STRR and
certain other physical characteristics PHCH, hence a province PP can be partitioned into physiographic sections PS, within which the characteristics are more uniform. Third order landforms LF3, are the terrain features TF and deposits DP we have been discussing. The third order landforms LF3 are superimposed upon second order forms LF2. For example, in mountainous areas of LF2 we have alluvial terraces, kames, eskers, etc., which are third order landforms LF3. The first and second order landforms of the United States have been defined by Fenneman (1931 and 1938).

There are a few notions associated with the definition of a physiographic unit which deserve formalization. The order of landform hierarchy is

\[(Ex)(Ey)(Ez)[(x \in LF1) \& (y \in LF2) \& (z \in LF3) \& CI(y,x) \& CI(z,y) \& CI(z,x)]\]  (5.55)

A physiographic unit PU can be represented by

\[(u \in PU) = df (Ev)(Ew)(Ex)(Ey)(Ez)[(v \in A) \& (w \in GEOH) \& (x \in LF1) \& (y \in LF2) \& (z \in PHCH) \& PART0(v,x) \& CI(z,v) \& AW(u,w\&z)]\]  (5.56)

The geomorphic history GEOH can be linked to the second order landforms LF2 by a sequence SEQ of geomorphic processes GP by

\[(Ew)(Ex)(Ey)(Ez)[(w \in LF2) \& (x \in GEOH) \& (y \in GP) \& (z \in SEQ) \& AW(w,x) \& RSFG(y\&x,y)]\]  (5.57)

Finally, given a physiographic unit PU one can infer the geomorphic processes GEOP associated with the area, from which the third order landforms LF3 are implied.

\[(Ex)(Ey)(Ez)[(x \in PU) \& (y \in GEOP) \& (z \in LF3) \& AW(y,x) \& INF(c,y) \& IM(y,z)]\]  (5.58)

5.2 Terrain Classification

Terrain classification TC is defined as the classification of
qualitative and/or quantitative terrain descriptions $DS(T)$ into unique categories $CD$, i.e.,

$$(w \in TC) = df (Ex)(Ey)[(x \in T) \& (y \in CD) \& CL(DS(x),y)]$$

whereas an engineering terrain classification technique $ETCT$ is defined as a method or plan for generating and classifying terrain descriptions $DS(T)$ for engineering purposes $PT$. Thus,

$$(w \in ETCT ) = df (Ex)(Ey)(Ez)[(x \in T) \& (y \in CD) \& (z \in PT) \& R(CL(DS(x),y),z)]$$

which reads as 'engineering terrain classification $ETC$ is defined as a classification of terrain descriptions $DS(T)$ into categories $CD$ related to engineering terrain-related problems $PT$'. An engineering terrain classification technique then involves

$$(w \in ETCT) = df (Ex)(Ey)(Ez)[(x \in T) \& (y \in CD) \& (z \in PT) \& R(GEN(DS(x)),R(CL(DS(x),y),z))]$$

This summary-type definition indicates engineering terrain classification techniques generate terrain descriptions related to the classification of terrain descriptions into categories which are in turn related to engineering terrain-related problems.

The three basic approaches to terrain classification for engineering purposes $ETCT$ are:

1. the pragmatic approach $ETCTPG$,
2. the parametric approach $ETCTPA$, and
3. the genetic approach $ETCTGN$.

Each differs in the manner of generating and classifying terrain descriptions for engineering problems. We will now discuss these three approaches.

5.2.1 The Pragmatic Approach

The pragmatic approach to terrain classification for engineering purposes is an ad hoc, problem-oriented approach characterized by the
lack of formal organization of the terrain or parameterization which allows extension from one project to another. The main objective is to obtain only that information useful to the problem at hand by whatever data source and analysis technique is most applicable. Thus the range and depth of schemes for terrain classification falling within this approach are dependent upon the ability and experience of the practicing engineers.

We assume, for purposes of formalism, the pragmatic approach to be an ad hoc adaption of the genetic concepts and based upon the following three premises:

1. The physiographic unit \( PU \) implies geomorphic processes \( GEOP \), which in turn imply possible landforms \( LF \), which in turn imply terrain attributes \( AT \), which include physical characteristics \( PHCH \) of terrain materials \( EM \).

\[
(Eu)(Ev)(Ex)(Ey)(Ez)[(u \in PU) \land (v \in GEOP) \land (w \in LF) \land (x \in AT) \land (y \in PHCH) \land (z \in EM) \Rightarrow IM(IM(IM(u,v),w),x\in INC(AW(y,z),x))] 
\] (5.62)

2. The terrain-problem situation \( P,T,T' \) implies the fundamental terrain criteria \( FTC \) for the problem.

\[
(Ex)(Ey)[(x \in P,T,T') \land (y \in FTC) \land IM(x,y)] 
\] (5.63)

3. The terrain attributes \( AT \), including the physical characteristics \( PHCH \) of the terrain materials \( EM \), can be classified \( CD \) when the fundamental terrain criteria \( FTC \) are given.

\[
(Ev)(Ew)(Ex)(Ey)(Ez)[(v \in AT) \land (w \in PHCH) \land (x \in EM) \land (y \in CD) \land (z \in FTC) \land CLG(v\in INC(AW(w,x),v),y,z)] 
\] (5.64)

which can be rewritten as an implication statement as follows

\[
(Ev)(Ew)(Ex)(Ey)(Ez)[(v \in AT) \land (w \in PHCH) \land (x \in EM) \land (y \in CD) \land (z \in FTC) \land CLG(v\in INC(AW(w,x),v),y,z)] 
\] (5.64)
Then when given these three premises in addition to a terrain-related problem $PT_1$ in a geographic area $A$, a set of codings $CD$ for the terrain attributes $AT$ or physical characteristics $PHCH$ of the materials $EM$ contained in a desired set of landforms $LF$ can be inferred from

$$(\alpha \in A) \& (\beta \in PU) \& (\gamma \in GEOP) \& (\delta \in LF) \& (\varepsilon \in AT) \& (\zeta \in PHCH) \& (\eta \in EM) \& (\theta \in P_1T_1) \& (\iota \in FTC) \& (\kappa \in CD) \&$$

$$INF(j, INF(e \& INC(AW(f,g),e), INF(d, INF(c, CI(a,b), IM(b,c)), IM(c,d)), IM(d, e \& INC(AW(f,g),e)) \&$$

$$INF(i, h, IM(h,i)), IM(e \& INC(AW(f,g),e) \& i))$$

(5.66)

In essence this inference represents an a priori code to terrain $T$ expected to be found in area $A$. The complete solution to the terrain-related problem $PT_1$ is then obtained only if experimental evidence verifies the existence of the items in the inference (5.66) and specifically located the landforms $LF$ or terrain attributes $AT$ in area $A$. Then a search strategy is adopted for locating and identifying the desired $LF$ in $A$. This could be a strategy for use of aerial photographic interpretation techniques or field exploration. Regardless, field sampling is usually accomplished to verify inferences associated with physical characteristics of terrain materials.

From (5.61) we now see that the pragmatic approach generates a priori terrain descriptions, while classification of terrain descriptions involves fitting terrain data to the description. This approach places an engineering problem into a terrain morphology context. This approach will be discussed and illustrated in detail in the following chapters.
5.2.2 The Parametric Approach

Whereas the pragmatic approach ETCPG is ad hoc and problem-oriented, the parametric approach ETCTPA is general and problem-oriented. The main objective is to abstract ABST spatial data DT related to terrain attributes AT for storage STOR in a database DB to be used in analytical models ANMD. This can be represented as

\[(Ev)(Ew)(Ex)(Ey)(Ez)[(v \in A) \& (w \in T) \& (x \in AT) \& (y \in DT) \& (z \in DB) \& ABST(AW(y,x),CI(w,v)) \& STOR(y,z)] \quad (5.67)\]

The terrain attribute data AW(DT,AT) for all of area A is stored in matrix format, and to provide a general system capable of solving a range of engineering problems in an area requires a large number of attributes with data of high spatial resolution. However, given that the attribute data and an analytical model are available for a problem, a solution is relatively easy to obtain following much of the approach outlined in the formalism of Chapter 4. For example, the stored data DT referred to in (5.67) is equivalent to the grid data GDj(u,v,z) of (4.37). When given a terrain-problem situation P'T1, a set of fundamental terrain criteria FTC are generated to establish attribute intervals ATI for the attributes AT1 associated with the analytical model ANMD and a matrix coding function MC(u,z,m,n) codes the data u as class z at point (m,n). This is important because prestored attribute data must often be transformed for use in the model. The resulting coded matrix CM(m,n,z), as given in (4.38) then represents the spatial distribution for one model parameter or attribute. More than one CM(m,n,z) can be combined to form a coded map matrix CMM(m,n,z) as per (4.46) to form a terrain factor TRNF for use in the analytical model. We will not reproduce this formalism of Chapter 4 but wish to indicate that the WES parametric system (Benn and Grabau, 1968; Grabau, 1968) represents a practical example of our formalism scheme.
To relate our discussion of the parametric approach to terrain
classification ETCTPA to the general definition of terrain classification
ETCT given in (5.61) we see the terrain descriptions $DS(T)$ to be
of two types: 1) is the attribute data in grid format $GD(u,v,z)$ in
the general storage and 2) is the coded matrices of $CM(u,v,z)$ for use
in the models. Both types concern the generation of the description
$GEN(DS(x))$, while the classification of the descriptions for a given
problem is accomplished by the analytical model.

5.2.3 The Genetic Approach

Whereas the pragmatic approach ETCTPG is ad hoc and problem-orien­
ted and the parametric approach ETCTPA is general and problem-oriented,
the genetic approach ETCTGN is general and non-problem-oriented. Its
purpose is to spatially delineate and describe divisions of natural
terrain features in a manner that an engineer could interpret for
solution of his problem. We will use the pattern, unit, component
evaluation or PUCE scheme (Aitchison and Grant, 1968) as an example of
this approach.

A terrain feature hierarchy for the PUCE scheme is given in four
levels as the terrain component TCP, the landform unit LF, the terrain
pattern TP, and the province PGEOL. We will now formalize the five
steps to the PUCE classification procedure in general terms since the
procedure involves image interpretation II.

1. Identification $ID(LF)$ and description $DS(LF)$ of landforms LF
is accomplished from aerial photographs $AE$.

$AE(u)(Ev)(Ew)(Ex)(Ey)(Ez)[(u \in A) \& (v \in T) \& (w \in AT) \& (x \in LF)$
$\& (y \in AI) \& (z \in AE) \& CI(v,u) \& AW(w,v) \& AW(w,x) \&$
$AW(y,w) \& AW(y,z) \& ID(x) \& DS(x)]$ (5.68)

where $LF$ is defined in (5.22) and $ID(x)$ and $DS(x)$ are defined in (5.23)
and (5.34), respectively.
2. Delineate **DEL** and describe **DS** groups **GRP** of landforms **LF** having the same origin, i.e., originating from the same geomorphic processes **GEOP**, as aerial photographic terrain patterns **TPAP**.

\[(EW)(Ex)(Ey)(Ez)\[(w \in LF) \& (x \in GEOP) \& (y \in IA) \& (z \in TPAP) \& DEL(CLG(GRP(ID(w)),z,\Delta W(w,x)\land \Delta W(w,y)\land \Delta W(y,z)) \& DS(CLG(GRP(ID(w)),z\Delta W(w,x)\land \Delta W(w,y)\land \Delta W(y,z)))\] (5.69)

where **DEL** is defined by (5.16).

3. Group **GRP** aerial photograph patterns **TPAP** into provinces **PGEOL** based upon geological information **ITG** obtained from photographic interpretation **II**.

\[(EW)(Ex)(Ey)(Ez)\[(w \in TPAP) \& (x \in PGEOL) \& (y \in ITG) \& (z \in II) \& CLG(GP(w),x,\Omega F(y,z))\] (5.70)

4. Code **CD** the landforms **LF**, aerial photograph terrain patterns **APTP**, and provinces **PGEOL** from the above steps and associated identification criteria **IDC**.

\[(Eu)(Ev)(Ex)(Ex)(Ey)(Ez)\[(u \in PGEOL) \& (v \in TPAP) \& (w \in LF) \& (x \in CD) \& (y \in IDC) \& (z \in IA) \& CLG(u,x\land \Delta W(x,u),u\land \Delta W(y,\Delta W(u,z))) \& CLG(v,x\land \Delta W(x,v),v\land \Delta W(y,\Delta W(v,z))) \& CLG(w,x\land \Delta W(x,w),w\land \Delta W(y,\Delta W(w,z)))\] (5.71)

5. Check the validity **V** of the classified areas: if the coding **CD** or boundaries **BNDY** are different than those obtained through field studies **FS** then make corrections.

\[(Eu)(Ev)(Ex)(Ey)(Ez)\[(u \in CD) \& (v \in BNDY) \& (w \in PGEOL) \& (x \in TPAP) \& (y \in LF) \& (z \in FS) \& (\Delta W(uv,v,w)) \lor \Delta W(uv,v,x)) \lor \Delta W(uv,v,y)) \& IM(\Delta W(uv,v,w),\Delta W(w,u\lor v,z),\Delta W(uv,v)) \& IM(\Delta W(uv,v,x),\Delta W(x,u\lor v,z),\Delta W(uv,v)) \&
To relate this discussion of the genetic approach to terrain classification ETCTGN to the general definition of terrain classification ETCT given in (5.61) we see the PUCE scheme yields only general descriptions DS(T) not oriented to a specific engineering problem PjTj. Thus the classification is also general so that the codings CD and boundaries BNDY can be applied to a wide range of problems. The PUCE scheme therefore does not supply answers to problems, but it does supply terrain information IT for solving problems.
CHAPTER VI
PROBLEM ORGANIZATION

In Chapter 4 we discussed and formalized concepts of terrain-related engineering problems. We devoted considerable attention to defining a problem in terms of its fundamental criteria and the attributes of terrain and the partitioning of an area. In Chapter 5 we discussed and formalized concepts for terrain organization for engineering problems. We developed the formalism of material characteristics in terms of the landforms and genetic processes responsible for their being and justified our use of the genetic approach to terrain classification and formalized three types of engineering terrain classification techniques. The purpose of this chapter is to discuss problem organization for a terrain pattern recognition system. It will bring together the concepts of the previous chapters and complete our pattern recognition model. Then in Chapter 7 we develop a simulated logical terrain pattern recognition and mapping problem which illustrates the concepts of this chapter. Chapter 8 applies these techniques to real world data.

The objective of problem organization is to analyze the various aspects of the problem goals, the terrain in the area of interest and the problem constraints so as to 1) select terrain components or target landforms which may serve as solutions to the goals, 2) define the gross terrain pattern structure and pattern attributes, 3) select a sequential strategy for locating and delineating the boundary of each target landform and 4) define the pattern attributes and their relations in terms of the data for those patterns indicated by the strategy
selection. Our premise is that target landforms will seldom have unique attributes detectable by simple search. This requires that the recognition strategy begin at some defined initial state in the data and sequentially transition to recognition and boundary delineation of the target landform.

This chapter is divided into three sections. The first section pertains to preliminary problem organization for selection of the target landforms which may satisfy the problem goals. The second section deals with applied terrain description formulation and the third section discusses the general nature of strategy selection. In the following chapter development and implementation of pattern recognition and mapping strategies is discussed and illustrated in detail for a simulated problem.

6.1 Preliminary Problem Organization

We now want to discuss organization of the problem solving effort when given a valid problem statement. This organization serves to indicate the terrain patterns and their related hierarchy to be defined by the descriptions for recognition purposes. The organization results are used in development of the strategy to physically accomplish the problem solving and will be discussed further in a subsequent section.

In general, problem organization and its formalism was the topic of Chapter 4. Here we want to stress the mechanics of problem organization rather than the formal characterization. These problem organization mechanics are important because no two solutions to a given terrain-related problem are identical for different terrain situations. Therefore each terrain-problem situation requires a renewed look at problem organization.

6.1.1 The Problem Statement

Problem statements and their factoring into elementary terrain-
related problems, each requiring a solution, was formalized in (4.52) and indicated a problem statement $\text{DS}(\text{PT})$ to consist of the problem goals $G$, the geographic area of interest $A_iG$ and a list of problem constraints $\text{PTC}$. Associated with each goal there is one or more elementary terrain-related problem $\text{EPT}$ which itself will have associated one or more fundamental terrain criteria $\text{FTC}$ from (4.18) and (4.22). When the information from the $\text{EPTs}$ is properly combined, the result is the required solution to the stated problem.

We submit as an example the simple problem statement in Table 6.1. It is simple in the sense that this problem could be a part of a more complex problem or project such as associated with route or site projects. From the problem statement the goal, geographic area and constraints have been extracted. The problem type, a local reconnaissance for natural material sources, suggests seven subproblem considerations and the associated problem graph indicates five elementary terrain-related problems which can be defined in terms of their fundamental terrain criteria. It may appear strange that ownership and landuse codes are classes under fundamental terrain criteria, however if the land is not zoned for materials excavation in the vicinity of built-up areas we should not consider more detailed phases of the reconnaissance in that area.

6.1.2 Problem Definition

In the example of Table 6.1 we define the problem and its solution in terms of estimated costs. Major cost considerations are confined to estimates of four items. These items may be taken to the desired estimation level dictated by the context in which the problem is presented for solution. For example, if a haul road must be constructed the estimate of its cost can range from a rule-of-thumb estimate per unit length, to a complete itemizing and estimation of costs associated
TABLE 6.1

EXAMPLE OF PROBLEM DEFINITION

**PROBLEM STATEMENT:** Locate a potential source of at least VOYD cubic yards of concrete aggregate in area A^G. The aggregate size and quality requirements are ASQR. The aggregate is to be used at LOC(x,y). The existing processing capabilities are APC. The cost per cubic yard CPCY of aggregate delivered to the use site is not to exceed ACOST.

**GOAL:** To locate a source of VOYD cubic yards of quality concrete aggregate for delivery to LOC(x,y) at a cost not to exceed ACOST.

**GEOGRAPHIC AREA:** A^G

**CONSTRAINTS:** LOC(x,y), VOYD, ASQR, APC, ACOST.

**SUBPROBLEM CONSIDERATIONS:**
1. Source location, SLOC(x,y)
2. Materials size and quality estimates, MSQE
3. Estimate volume of available source materials, EVOL
4. Ownership and landuse codes, OLUC
5. Accessibility, ACBL
6. Source workability, SWK
7. Processing requirements, MPRO

**PROBLEM GRAPH:**

![Diagram of problem graph]

**MAJOR COST CONSIDERATIONS:**
1. Property purchase CPCYP
2. Access
   a. Haul distance CPCYA
   b. Haul roads to be constructed HAULC
3. Source workability
   a. Site preparation CONSTC
   b. Work type and equipment CPCYW
4. Material processing CPCYWPT
5. Processing costs CPCYP
with efforts and materials required for the construction, e.g., clearing, earthwork, drainage, structures, equipment, manpower, etc. The schematic flowchart associated with the solution of the materials location problem is given in Fig. 6.1. We leave undefined the subroutines for cost estimation. Flowcharting serves to formalize selected aspects of problems. The flowchart approach to problem definition forces one to think in terms of the sequential aspects and decision criteria of problem components, but lack problem definitions present by problem graphs.

6.1.3 Target Landform Selection

Here we define the mechanics for a priori selection of landforms with the highest potential for having attributes representing solutions to all or parts of our problem. We are thus adding the terrain organization concepts of Chapter 5 to our problem organization context.

These results will be used in our pattern description and recognition languages and the recognition strategies to be discussed in later sections of this chapter. For this discussion we introduce Fig. 6.2 to illustrate the schema of terrain description organization.

6.1.3.1 Subproblem Landform List

From the problem statement we can define the terrain-related subproblem set $\{P_iT\}$, $i=1,I$, as indicated in Fig. 6.2. The value of $I$, or for any 'i', is in a large extent determined by context in which the problem statement is presented, as discussed in Section 6.1.2.

Given the subproblem set, for each subproblem we can generate a list of landforms which could possibly serve as solutions to the subproblem. We will denote a subproblem landform list as $\{LF_{ij}\}$, $j=1,J$, containing 'J' landform names or symbols. For the subproblem set $\{P_iT\}$ we then have an associated set of subproblem landform lists $LF_{ij}$. 

2. Are size and quality of materials acceptable?

3. Is the estimated material volume \( EVOL \) greater than \( YQYD \)?

4. Are ownership and landuse codes acceptable?

5. Subroutine ACBL: Computes estimated cost of access \( CPCYA \) from estimates of haul costs \( HAULC \) and construction costs \( CONSTC \).

6. Subroutine SWK: Computes estimated cost of workability \( CPCYW \) from estimates of site preparation \( CPCYWP \) and work type \( ADN \) equipment \( CPCYWT \) costs.

7. Subroutine MPRQ: Computes estimated processing costs.

8. Sums all estimated unit costs.


10. List of acceptable sources by location, type, and total estimated unit costs.

11. Counts to \( NN \) the number of members in list of item 10 above.

12. Selects best (minimum unit costs) source from list in item 10 above.

Fig. 6.1 Schematic flowchart associated with solution of the problem stated in Table 6.1.
Fig. 6.2 Terrain description organization for terrain-related engineering problems.
6.1.3.2 Study Area Landform List

The problem statement contains the geographic location of the study area $A_1 G$ allowing one to use the physiographic approach discussed in Sections 5.1.8 and 5.2.1 to develop a list of landforms which can be expected to be found in the study area. Briefly, physiographic units $PU$ are associated with geographic area $A$ and, when given a specific geographic area $A_1 G$, one or more contiguous physiographic units $PU$ are indicated. Given this information, the geomorphic history $GEOH$ of the area is indicated and in turn the geomorphic history indicates from (5.57) the type of landforms $\{LF_k\}$ which can be expected in the study area. Thus, from the chain rule of inference (3.20)

$$A_1 G$$
$$A_1 G \rightarrow PU$$
$$PU \rightarrow GEOH$$
$$GEOH \rightarrow LF_k$$

(6.1)

we obtain the study area landform list.

6.1.3.3 Target Landform List

Given the subproblem landform lists and the study area landform list, we can match lists for a given subproblem to obtain another list of landforms of interest, as shown in Fig. 6.2. This list can be ordered by criteria associated with the subproblem $PTC$ pertaining to the physical characteristics of the landform materials. The resulting list will be headed by the landform of highest priority, which we shall refer to as the target landform $TLF$.

If we use Table 6.1 as an example, our list of landforms of interest contains names or symbols indicating land units which can be found in the study area having characteristics useful to the solution of the subproblem. For our example suppose these names were 'UPLAND',
'TERRACE', 'FLOOD PLAIN', and 'RIVER'. Suppose further that we know the general physical characteristics of these landforms and their materials. The TERRACE contains sands and gravels requiring no preprocessing and can be worked easily; the FLOOD PLAIN contains silts, sands, and gravels and will require minor preprocessing and working conditions may be hindered by high water table conditions; RIVER contains sands and gravels which can only be removed by dredging; and UPLAND contains clay, silt, and sands with little gravel and rock which must be quarried and crushed. Our target landform list would then be ordered as TERRACE, FLOOD PLAIN, RIVER, UPLAND; with TERRACE being our target landform TLF.

6.1.4 Terrain Description Organization

In this section we briefly outline our organization of terrain descriptions for problem organization. From the previous section we have a target landform list as well as a list of all expected landforms to be found in the study area prepared from the problem statement and our apriori knowledge of the study area. We have not as yet exhausted our apriori knowledge of the study area. Here we want to indicate the structural organization of the descriptions of the spatial distribution of landforms, their components and their component attributes.

6.1.4.1 Terrain Pattern Hierarchy

Similar to the description of landform in the formulation of (5.23) we can define the terrain pattern description hierarchy for a study area as shown in Fig. 6.3. For a given geographic area this hierarchy serves to define a set of expected terrain patterns and their relations similar to that of the PUCE system. In general, terrain patterns are spatial groupings of landforms having a common geomorphic process, e.g., alluvial patterns, glacial patterns, residual patterns. Our terrain description organization of Fig. 6.2 is similar to that of our pattern
Fig. 6.3 Terrain pattern description hierarchy.
hierarchy below the terrain pattern level. Terrain patterns and their relations may be formulated into special use descriptions for some problems, however the more general case is to partition an area by landforms. At the lower level of our hierarchy we have landform component attributes and their relations which can be described by groups of terrain measurements. Thus the study area can be described in terms of its attribute data as in the parametric classification system of Section 5.2.2. However if we do this then we neglect all relational information from the description hierarchy.

6.1.4.2 Matrix Organization

Our descriptions will be stored in matrices and statements. One dimensional matrices are used for name lists, i.e., the study area landform list and the landform component list, whereas multidimensional matrices are used for relations so that it is possible to state a given relation between any two elements being described. Usually our study area list has only a few landform names for any given problem and the associated lists and relations required to describe and identify these are rather small. Thus we are not storage limited in terrain description organization.

Further aspects of structure and details associated with the description matrices and their organization will become apparent in our later discussions.

6.1.7 Summary

We can briefly summarize this portion of the chapter formally by (5.66). This formalization explains all of the considerations for problem organization but none of the realization mechanics; which were described here. Problem definition, preparation of the target landform list, and the terrain pattern hierarchy are all indispensable in preparation for solving pattern recognition problems. The image analyst
usually accomplishes these tasks mentally from intuition, but we must be more formal for machine processing.

We have only considered the problem organization relative to the terrain and the engineering problem. But at this stage we know what landforms will serve as solutions to our given problem. It remains for us to describe the terrain features of interest in terms suitable to our recognition system and then devise a strategy to locate the target landforms in the study area from sample data.

6.2 Applied Terrain Description Formulation

Terrain descriptions are important in our approach to terrain pattern recognition because classification of the land units is dependent upon the ability to define the land units in terms of a set of fundamental symbols representing properties and relations.

In general we are not interested in descriptions of all landforms LF from \{LF_k\}; only those which link the initial or start state \(S_0\) to the target landform TLF. To further simplify the descriptions of those landforms of \{LF_k\} that are of interest, we realize that the descriptions need only be defined to distinguish between contiguous members of \{LF_k\}, i.e., we are not interested in complete descriptions, rather our descriptions should be sufficient to distinguish between contiguous members of \{LF_k\}.

Here we are interested solely in defining terrain units for recognition purposes; in this sense it is a discriminating language. Our discussion will be limited to the general nature of the description of terrain patterns and their structure at various levels of Fig. 6.3. We omit from the discussion references of methods for selecting definitions because one must have the strategy in mind to adequately select the definitions to assist in producing the simplest recognition strategy. This will be clear from the discussion in Section 6.3.
A terrain pattern discrimination language will now be outlined.

6.2.1 Terrain Pattern Discrimination Language

The purpose of the discrimination language to be outlined is to produce valid definitions of terrain patterns in terms of a set of fundamental symbols representing properties and relations. The descriptions will be used in the pattern recognition system for discriminating between patterns in the sampled data produced by the pattern recognition system. This language will allow us to formally describe the terrain pattern elements used for terrain pattern recognition in a manner similar to that used by the engineer photo interpreter as discussed in Section 2.2.2.

A language, as indicated in Section 3.2.1 consists of a vocabulary of signs and a grammar or rules for their use.

6.2.1.1 Evans' Pattern Description Language

Evans (1971) was concerned with grammatical inference in pattern analysis in Fig. 2.2 where his input pattern represents (after preprocessing) a face-like drawing. He has a language for his descriptions which is composed of a set of symbols (English letters and numbers) and a grammar, which is a finite set of rules. His approach was presented in the discussion of linguistic pattern recognition in Section 2.4. Evan's example ended without a method of implementation, but to use these descriptions for recognition and identification of faces, the elements of his fundamental symbol set and his fundamental predicate set would have to be programmable, i.e., each symbol and predicate expressed in terms of elementary data elements and machine operations. If one were to use Evans' language to identify individuals or classes of individuals, his fundamental symbol set would require extension to be descriptive of class differences, i.e., indicative of discriminating facial characteristics. If the faces were allowed orientations other
than the direct, upright and frontal view of Fig. 2.2, then other fundamental predicates would be required.

6.2.1.2 Extending Evans' Description Language to Terrain Patterns

First we will translate Evans' grammar rules of Fig. 2.2 to definitions in the formalism of our previous chapters.

\[
\text{FACE} = \text{df} (\exists x)(\exists y)[(x \in \text{HEAD}) \land (y \in \text{FEATURES}) \land \text{INSIDE}(y,x)] \quad (6.2)
\]

\[
\text{HEAD} = \text{df} (\exists x)[(x \in \text{CIRCLE})] \quad (6.3)
\]

\[
\text{FEATURES} = \text{df} (\exists x)(\exists y)(\exists z)[(x \in \text{EYES}) \land (y \in \text{NOSE}) \land (z \in \text{MOUTH}) \land \text{ABOVE}(x,y) \land \text{ABOVE}(x,z) \land \text{ABOVE}(x,y)] \quad (6.4)
\]

\[
\text{EYES} = \text{df} (\exists x)(\exists y)[(x \in \text{DOT}) \land (y \in \text{DOT}) \land \text{LEFT}(x,y)] \quad (6.5)
\]

\[
\text{NOSE} = \text{df} (\exists x)[(x \in \text{SQUARE})] \quad (6.6)
\]

\[
\text{MOUTH} = \text{df} (\exists x)[(x \in \text{LINESEG}) \land \text{HORIZ}(x)] \quad (6.7)
\]

Likewise we can return to Evans' grammar from these expressions. We can also combine by substitution the expressions to remove some of the mid-level names. For example, the definition of 'FEATURES' and 'FACE' in terms of the fundamental symbol set are:

\[
\text{FEATURES} = \text{df} (\exists a)(\exists b)(\exists c)(\exists d)[(a \in \text{DOT}) \land (b \in \text{DOT}) \land (c \in \text{SQUARE}) \land (d \in \text{LINESEG}) \land \text{ABOVE}(a\&b\&\text{LEFT}(a,b),c) \land \text{ABOVE}(a\&b\&\text{LEFT}(a,b),d) \land \text{ABOVE}(c,d)] \quad (6.8)
\]

\[
\text{FACE} = \text{df} (\exists a)(\exists b)(\exists c)(\exists d)(\exists e)[(a \in \text{DOT}) \land (b \in \text{DOT}) \land (c \in \text{SQUARE}) \land (d \in \text{LINESEG}) \land (e \in \text{CIRCLE}) \land \text{INSIDE}(\text{ABOVE}(a\&b\&\text{LEFT}(a,b),c) \land \text{ABOVE}(a\&b\&\text{LEFT}(a,b),d) \land \text{ABOVE}(c,d),e)] \quad (6.9)
\]

We present a possible study area pattern description in Evans' grammar in Fig. 6.4. This graph and grammar will be simplified shortly but for now let us look at the characteristics of this approach. The map shows the study area to be composed of four basic landforms and the structural graph indicates each landform to have at least one landform component and each landform component has a fundamental dummy
Map Legend:
1 River Channel
2 Flood Plain
3 Terrace
4 Upland

**STRUCTURAL GRAPH**

**GRAMMAR RULES:**

Study Area → (w,x,y,z): River Channel(w), Flood Plain(x), Terrace(y), Upland(z): PADJ(w,x) & PADJ(w,y) & PADJ(w,z) & PADJ(x,y) & PADJ(y,z) & LTHAN(w,x) & LTHAN(x,y) & LTHAN(y,z)

River Channel → (x): Channel(x): BIFURCATING(x)

Channel → (x): WATER(x): WIDTH(x) & LOWGRAD(x)

Flood Plain → (x): Surf1(x)

Surf1 → (x): PLANE(x): LEVEL(x)

Terrace → (x,y): Surf2(x), Surf3(y): TADJ(x,y)

Surf2 → (x): PLANE(x): LEVEL(x) & -DRAIN(x)

Surf3 → (x): STEP(x): RAMP(x) & RIVDIR(x)

Upland → (x,y): Surf4(x), Drainage(y): IN(y,x) & LOCLOW(y)

Surf4 → (x): IRR(x)

Drainage → (x): DEND(x)

Fig. 6.4 A possible study area pattern description with a structural graph and a set of associated grammar rules.
symbol as an identifiable attribute. The grammar rules then define the study area and landform set in terms of the fundamental symbol set and the predicate set. These are indicated by capital letter names in Fig. 6.4, whereas the dummy variables are lower case letters.

Thus the predicate descriptions at all levels must be defined in terms of primitives; whereas the land unit descriptions require definition only at the lowest level. We will show shortly that the 'fundamental' symbols of Fig. 6.4 are complex but can be defined in terms of basic data. Now we want to return to Fig. 6.2 to complete the discussion of description organization.

6.2.1.3 Description Organization

We mentioned that the terrain description organization of Fig. 6.2 was similar to the terrain description hierarchy of Fig. 6.3, however the contiguous matrices $C(I,J)$ or start state descriptions $Sp$ have not been defined. From the list $\{LF_k\}$ and knowledge of the physiographic setting and geomorphic history we can construct a symmetric matrix $C(LF_k,LF_{k'})$, $k,k'=1,K$, which denotes the contiguity relationship between terrain components that must be traversed to form a path from $Sp$ to TLF. Fig. 6.5 shows two examples of contiguity matrices and two possible map representations. The number symbols in the maps represent code symbols for the terrain components and the lines represent discrete boundaries between terrain components. Thus from the matrix one can see the allowable set of neighbors for points in the map (image).

For a given TLF we select a start state $Sp$ which, if it exists, will uniquely provide a reference point in the geomorphic modeling, i.e., is unambiguously associated with a terrain component of $\{LF_k\}$. From $Sp$ to LFC a "best path" will be indicated by the contiguity matrix $C(LF_k,LF_{k'})$, however the set of all paths from $Sp$ to TLF must be con-
CONTIGUITY MATRIX

\[ C(T_k, T_{k'}) \]

\[
\begin{array}{cccc}
S_0 & 1 & 2 & 3 & 4 \\
S_0 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 1 & 0 & 0 \\
2 & 0 & 1 & 1 & 1 & 0 \\
3 & 0 & 0 & 1 & 1 & 1 \\
4 & 0 & 0 & 0 & 1 & 1 \\
\end{array}
\]

Fig. 6.5 Two examples of contiguity matrices with a possible map for each.
sidered in the strategy. This is seen from the lower map in Fig. 6.5 where, for example, if the start state $S_0$ is associated with $T_k = LF_k = 1$ and our $TLF = T_k = LF_k = 3$, then from the contiguity matrix we know that $C(1,3) = 1$ is the best path, but we also must be interested in the combination of $C(1,2)$ and $C(2,3)$, as well as the combination $C(1,4)$ and $C(4,3)$, if we wish to detect all possible occurrences of $T_k = LF_k = 3$.

This then is the heart of the strategy—to begin from a start state $S_0$ and sequentially recognize and classify contiguous terrain components until the TLF are detected and classified. In the general case there will be more than one occurrence of a TLF in an area.

Now the hierarchical organization of landform descriptions in Fig. 6.3 should be transport and structure is seen to be similar to Evans' grammar statements. Thus the study area has an associated set of landforms $\{LF_k\}$ and these have an associated set of predicates in the form of a logical contiguous matrix $C(LF_k, LF_k')$ and one or more landform components $LF_{C_k}$ and predicates $C(LF_{C_k}, LF_{C_k'} )$ and $R(LF_{C_k}, LF_{C_k'})$. At the lowest description level we find $\phi_{k1m}$ attributes defined in terms of fundamental predicates $\phi_{k1m}(DT)$ and $C(\phi_{k1m}, \phi_{k1m'})$.

Therefore we can discuss or describe the study area at three levels. At the landform level, gross pattern properties and relations between patterns can be discerned. At the landform component level, the major parts of the landforms and their predicates relations are describable.

We can use two basic forms of relation descriptions. One type is the binary symmetric matrix as indicated above, e.g., $R(LF_k, LF_k')$. Here $R$ could represent surface configuration differences such as relative slopes or elevations from the topographic data, or spectral density relations from the tone data. Thus we could use $LTZ(LF_k, LF_k')$ to represent the assertion that $LF_k$ was lower than $LF_{k'}$ when the relation
LTZ(LF_L, LF_L) has a '1' entry; and conversely a '0' entry would indicate LF_L was greater than or equal to the elevation of LF_L. This type of binary symmetric matrix is a logic matrix having 'true' represented by '1' and 'false' by '0', and this same concept can be generalized to relations having more than two members, e.g., the relation BTZ(LF_L, LF_L, LF_L) might indicate that the elevation of LF_L is between the elevations of LF_L and LF_L when the respective element of the logic matrix is '1'.

The second type of relation description would be the relation description statement formed by a statement of combined predicates. For example, level surface and density in the blue spectral band greater than 1.5 and elevation less than 100 ft., etc. We will illustrate both forms of relation descriptions shortly.

6.2.2 Description Learning

It may not be possible in all cases to prepare discriminatory definitions to allow recognition and classification of terrain units from apriori knowledge. Or upon cycling the pattern recognition program, errors may be detected which can be removed by refined definitions. Where no prior knowledge exists a clustering approach to finding discriminations between patterns may be attempted. Two often used references are those of Nilsson (1965), who showed how almost all of the classification procedures in common use could be reduced to the problem of computing discriminant functions, and Higleyman (1962) is an often-quoted paper on the application of linear discriminant functions in pattern recognition.

An approach to learning descriptions may be had by selecting spatial samples from data matrices of known patterns which are to be discriminated. Then some statistics of the samples may allow selection of discriminant functions for the samples. This approach should be an
expedient means for obtaining discriminants between spectral reflectances of known natural materials rather than accomplish normalization and correction of distortions due to the environment and the imaging and sampling systems. To improve the results of clustering for certain types of terrain conditions it may be important to accomplish noise removal processing, e.g., correction for angular variations in terrain reflections. Such corrections may increase the signal to noise ratio for better discrimination results.

6.2.3 Formulating Descriptions

The general nature of description formulation will now be illustrated, remembering that the exact nature of the formulation will depend upon the problem, the terrain and the strategy to solve the problem. We will use Fig. 6.6 and Fig. 6.7, which are modifications of Fig. 6.4, to show the initial formulation and Fig. 6.6 to illustrate the fundamental definitions. Evans' grammar scheme will be used because of comparative formulation ease.

6.2.3.1 Conceptual Map

Given the study area landform list \( \{LF_y\} \) and the target landform \( TLF \), the first step in formulating effective descriptions is to sketch a conceptual map of the spatial arrangement or structure of the landforms of \( \{LF_k\} \). The problem may be such that selected landforms from \( \{LF_k\} \) can be grouped into a larger class, e.g., in Fig. 6.4, Fig. 6.6 and Fig. 6.7 the landforms in the UPLAND class have been grouped. Essentially we are indicating that no individual members of this class will affect our problem solution and that the definition of the class will be sufficient for all its members. This will reduce the number of members of \( \{LF_k\} \) for which we must give later attention.

Our main concern in developing the conceptual map is to indicate all possible combinations of landform contiguity which may be expected
Map Legend:
1 River Channel
2 Flood Plain
3 Terrace
4 Upland

**STRUCTURAL GRAPH**

```
Study Area
  River Channel   Flood Plain   Terrace   Upland
    WATER        PLANÉE         PLANÉE  STEP  IRREGULAR
```

**GRAMMAR RULES:**

Study Area $\rightarrow (w,x,y,z)$: River Channel$(w)$, Flood Plain$(x)$, Terrace$(y)$, Upland$(z)$: MAT(C) & MAT(LTHAN)

River Channel $\rightarrow (x)$: WATER$(x)$

Flood Plain $\rightarrow (x)$: PLANÉE$(x)$: LEVEL$(x)$

Terrace $\rightarrow (x,y)$: PLANÉE$(x)$, STEP$(y)$: LEVEL$(x)$ & SLOPING$(y)$ & ADJ$(x,y)$

Upland $\rightarrow (x)$: IRREGULAR$(x)$

Fig. 6.6 A simplification of the structural graph and grammar rules associated with Fig. 6.6.
So

Map Legend:
1 River Channel (R)
2 Flood Plain (FP)
3 Terrace (T)
  3a Terrace Face (TF)
  3b Terrace Top (TT)
4 Upland

STRUCTURAL GRAPH

Study Area

River Channel Flood Plain Terrace Upland

WATER PLANE PLANE STEP (IRREGULAR)

GRAMMAR RULES:

Study Area \(\rightarrow\) \((w,x,y,z)\): River Channel\((w)\), Flood Plain\((x)\), Terrace\((y)\),
Upland\((z)\): MAT\((C)\) & MAT\((LTHAN)\)

River Channel \(\rightarrow\) \((x)\): WATER\((x)\)

Flood Plain \(\rightarrow\) \((x)\): LEVEL PLANE\((x)\)

Terrace \(\rightarrow\) \((x,y)\): Terrace Face\((x)\), Terrace Top\((y)\)

Terrace Face \(\rightarrow\) \((x)\): SLOPING STEP\((x)\)

Terrace Top \(\rightarrow\) \((x)\): LEVEL PLANE\((x)\)

Upland \(\rightarrow\) \((x,y,z)\): NOT River Channel\((x)\), NOT Flood Plain\((y)\),
NOT Terrace\((z)\)

Fig. 6.7 Modified map, structural graph, and grammar rules
associated with Figs. 6.4 and 6.6.
in our real world data. We are not concerned with the fact that our conceptual map may not be the true map of the spatial distribution of landforms produced from the real world data. Given the conceptual map we can prepare the landform contiguity matrix $C(LF_k, LF_{k'})$ with $k$ being the element of the reduced set $\{LF_k\}$ if landform grouping has been accomplished.

From the map a unique point or area is selected to represent the start state $S_0$. As indicated above this will reference the geomorphic model to the real world data and serve to initialize the strategy for the pattern recognition.

As in Figs. 6.4 through 6.6, coding of landform names is useful and serves to make subsequent symbolization simpler. The landform names and its coding representation will only be dummy variables in our descriptions and final output.

6.2.3.2 Structural Graph

The next step is to begin a structural graph by indicating the landforms depicted in the conceptual map. The purpose served by the structural graph is to indicate the hierarchy of names of landforms, landform components, and landform attributes which require grammar rule definition. The lowest level of the structural graph contains names of landform attributes or landform component attributes which must be defined in terms of fundamental operations and decision functions within the capability of our pattern recognition system. Intermediate between the landform level and the attribute level may be landform components which must be recognized to identify the landforms also.

Figs. 6.6 and 6.7 show two different structural graphs for the same area. These will require definitions and the differences in the graphs must be reflected also in the grammar rules.

6.2.3.3 Formulating Grammar Rules
Each name in the structural graph requires a grammar rule definition. The grammar rules of Fig. 6.6 are similar to those of Evans' (1971) shown in Fig. 2.2, whereas those of Fig. 6.7 contain no explicit predicates in the grammar. Instead all predicates are contained in the fundamental definition, which is complex. Capital letter names in the definitions all represent primitive functions or operators. For example, $EQ(a,b)$ indicates a logical function which is 'true' if the values of $a$ and $b$ are equal; $PEQ(a,b)$ is a logical function which is 'true' if the names of $a$ and $b$ are identical; $H_{20}$ is a proper classification name, i.e., a constant; $NEQ$ means 'not equal'; $LTE$ is 'less than or equal to'; $GT$ is 'greater than'. The operator $INFER$ is an exception, as is $FCLASS$, because they are not primitive functions. In the case of $INFER$ we refer to the inference process which results in a conclusion statement when given the premises in the form of arguments. We will consider $FCLASS$ to be a function or subroutine which by statistical decision techniques yields a classification constant for a spectral vector argument. These cause us no problem so long as the expressions are well formed and can be programmed.

The definitions of Fig. 6.8 will now be given a general interpretation; a specific interpretation results when specific values are given to all variables. The definition of $S_0$ states: There exist some things $x$ and $y$ such that the minimum elevation, i.e., $FMINZ$, occurs in the study area elevation matrix at coordinate $(x,y)$ and at this coordinate the spectral vector classification is $H_{20}$, which is identical to the $i$-th pattern in the pattern list $P$ and the name of the $i$-th pattern is also identical to the inference conclusion having $S_0$ and its associated conditional statement from the contiguity matrix as premises. The premise $S_0$ and the definition statement $S_0$ are quite different. The premise represents a logical statement, whereas the definition is
DEFINITIONS:

\[ S_o = \text{df} \ (Ex)(Ey) [\text{EQ}(FMINZ,Z(x,y)) \& \ (E1)[\text{PEQ}(P(i),\text{PEQ}(FCLASS(S1(x,y),S2(x,y)),H2O) \\
& (j)[\text{PEQ}(P(i),\text{INFER}(S_o,P(j)))]]]] \]

WATER = \text{df} \ (Ex)(Ey) [\text{PEQ}(FCLASS(S1(x,y),S2(x,y)),H2O)]

LEVEL PLANE = \text{df} \ (Ex)(Ey)(Eu)(Ev)[\text{EQ}(Z(x,y),Z(u,v)) \& \text{NEQ}((x,y),(u,v))] = \text{df} \ (Ex)(Ey)[\text{EQ}(Z(x,y),Z(f(x),g(y)))]

SLOPING STEP = \text{df} \ (Ex)(Ey)(Eu)(Ev)[((\text{LTE}(Z(x,y),Z(u,v)) \& \text{GT}(Z(x,y),Z(-u,-v))) \lor \\
(\text{LTE}(Z(x,y),Z(-u,v)) \& \text{GT}(Z(x,y),Z(u,-v))) \lor \\
(\text{LTE}(Z(x,y),Z(u,-v)) \& \text{GT}(Z(x,y),Z(-u,v))) \lor \\
(\text{LTE}(Z(x,y),Z(-u,-v)) \& \text{GT}(Z(x,y),Z(u,v)))) \& \text{NEQ}((x,y),(u,v))]]

\[
\begin{array}{|c|c|c|c|c|c|}
\hline
& S_o & R & FP & TF & TT & U \\
\hline
S_o & 0 & 1 & 0 & 0 & 0 & 0 \\
R & 0 & 0 & 1 & 1 & 0 & 1 \\
FP & 0 & 1 & 0 & 1 & 0 & 0 \\
TF & 0 & 1 & 1 & 0 & 1 & 0 \\
TT & 0 & 0 & 0 & 1 & 0 & 1 \\
U & 0 & 1 & 0 & 0 & 1 & 0 \\
\hline
\end{array}
\]

\[
\begin{array}{|c|c|c|c|c|c|}
\hline
& S_o & R & FP & TF & TT & U \\
\hline
S_o & 3 & 3 & 3 & 3 & 3 & 3 \\
R & 3 & 3 & 1 & 1 & 1 & 1 \\
FP & 3 & 0 & 3 & 1 & 1 & 1 \\
TF & 3 & 0 & 0 & 3 & 1 & 1 \\
TT & 3 & 0 & 0 & 0 & 3 & 1 \\
U & 3 & 0 & 0 & 0 & 0 & 3 \\
\hline
\end{array}
\]

Fig. 6.8 Definitions associated with Fig. 6.7.
extra-logical in that it has values derived or validated empirically.

The definition **WATER** states: There exist somethings \( x \) and \( y \) such that the spectral vector at coordinate \((x,y)\) is classified identical to \( H_2O \).

The definition **LEVEL PLANE** states: There exist somethings \( x, y, u, \) and \( v \) such that the elevation value at coordinate \((x,y)\) equals the elevation value at coordinate \((u,v)\) and the coordinates \((x,y)\) and \((u,v)\) are different. Here \( u \) and \( v \) are some functions of \( x \) and \( y \), respectively, and each has a restricted range. Of the two definitions given in Fig. 6.8, it is proper to use the second definition to indicate this functional dependence, however when determining the truth value of the statement we use each member of the functional coordinate set. For example, the functional coordinate \((f(x),g(y))\) could have the coordinate set:

\[
(f(x),g(y)) = (x-1,y-1),(x-1,y),(x-1,y+1) \\
(x,y-1),(x,y+1),(x+1,y-1), \\
(x+1,y),(x+1,y+1)
\]

for the definition to have the value 'true', the conjunction of \( f(x,y) \) with every member of \( f(x,y) \) must be 'true'.

The definition **SLOPING STEP** states: There exist somethings \( x, y, u, \) and \( v \) such that the elevation at coordinate \((x,y)\) is less than that of one neighbor and greater than the conjugate neighbor coordinate. The objective of this definition is best met by not using the functional notation.

The matrices at the bottom of Fig. 6.8 represent another definition form. The left matrix defines the pattern and pattern feature contiguity relationships. This type of matrix has been discussed previously. The remaining matrix, here called the **LTHAN** matrix, is coded to indicate whether one pattern is higher than or lower than another pattern. If a matrix element has a value '0' then the associated i-th pattern is
higher than the associated j-th pattern; a '1' indicates a lower than relationship; and a '3' should be disregarded.

This completes the discussion of our definitions in Fig. 6.8. We now must consider the strategies and the use of the definitions in the mapping process.

6.3 Mapping Strategies

In this section we address the general nature of the strategies which control the operation of our pattern recognition system. The following chapter deals in detail with the development and implementation of a mapping strategy.

Mapping strategies consist of a set of operation and decision rules designed to accomplish the location and delineation of the target landforms in a given area. These rules must be developed for the problem and implemented through software for realization on a digital computer.

We can indicate a basic set of operation rules useful in discussing any terrain pattern recognition strategy as:

1. Locate the start state, \( S_p \).
2. Map the boundary of a given state, MLF.
3. Transition to the next state.
4. Identification of the stopping state, \( S_s \).

The decision rules for any strategy associated with these operations rules will be the logical tests of the terrain descriptions with the input data and validity tests for the operation rules. Here we consider a state as a possible combination of data fitting a terrain description. The set of terrain descriptions thus specify our mapping landform states, MLF. The start state \( S_p \), as indicated in Section 6.2.1.3, uniquely provides an initial reference point between the data and the geomorphic model and is unambiguously associated with a particular terrain component, terrain condition, or landform. The stopping
state $S_g$ is the state or data condition which exists when a problem goal has been satisfied. Fig. 6.9 is a flowchart associated with the basic set of operation and decision rules. For any particular problem and geographic area, each block of this illustration is considered complex and requires significant expansion to become representative of operational software for the problem. The nature of this expansion will be evident in the next chapter.

6.3.1 Boundary Mapping

We have repeatedly referred to boundary mapping in our previous discussions and with this is implied the location and classification of the outer perimeter of the MLFs. The operations and decisions required to map a boundary are dependent upon the approach. For example, we might employ a boundary following algorithm which tracks the sequential elements of the MLF boundary in the data. Or a global mapping algorithm might be used to scan the data on some raster pattern, i.e., a TV line scan pattern, and the data at each element are classified and then the boundary coded matrix $BCM(m,n,z)$ of Fig. 4.2 is created.

Two important reasons are given here for support of the boundary following approach in our terrain pattern recognition system. The prime reason being that an MLF will seldom have unique attributes, therefore adjacent patterns must be detected, identified, and checked for validity of spatial association with the landform being mapped. This would necessitate a complicated set of operation and decision rules for software implementation for the global mapping algorithms. Another important reason for selecting the boundary following approach is that it models the method of the qualitative interpreter.

Now returning to the second operation rule, i.e., to map the boundary of a given MLF. With the boundary following approach boundary mapping is a sequential point by point tracking through the data from a
Fig. 6.9  Flowchart associated with a basic set of operation and decision rules of a mapping strategy. Decision rules have double-lined boundaries.
start point to an end point, which is in general the reoccupied start point. Upon completion of a boundary delineation phase, the strategy calls for a transition to the next MLF. This usually involves checking the classification of the landforms contiguous to those which have been mapped, and is accomplished with the contiguity matrix as discussed in Section 6.2.1.3 and shown in Fig. 6.5. If the newly identified landform can be connected to the goal landform in the contiguity matrix, it is then designated as the MLF to be mapped by the boundary following algorithm. This process is continued until the stop state $S_5$ is obtained. The stop state $S_5$ usually occurs when the boundaries of the goal landforms have been mapped.

6.3.2 Start State Selection

We must say a few words about selection of the start state $S_0$. $S_0$ is usually selected with apriori knowledge of terrain organization and the data matrices. Given the choice of defining $S_0$ with the elevation data or tone data, the choice should always favor the elevation data due the lower associated ambiguity. Usually one can associate a landform uniquely with the highest of lowest elevation in a given area. If the contiguity matrix representation of the terrain landform structure indicates a path between the landform in question and the goal landform then this is a logical start state. The possibilities of other start states should always be considered, especially if there is a chance to shorten the path to the goal landform.

6.3.3 Preprocessing

To improve the operational efficiency of pattern recognition systems, preprocessing of the input data is often employed. In our logical pattern recognition system preprocessing would be part of the mapping strategy. The preprocessing must be simple because it is usually applied globally to the data.
Let us consider a few simple examples where preprocessing might simplify the strategy. Consider the problem of siting a microwave relay tower in a mountainous area. From an apriori knowledge of the terrain elevations in the study area, an elevation threshold can be established to delete lower areas from consideration in the following strategy. In an opposite sense, if one were interested in alluvial materials in this area he may wish to delete all terrain above a given threshold elevation from future consideration.

Though we will not utilize preprocessing in the later chapters which demonstrate the logical pattern recognition system, its application to logical pattern recognition is potentially valuable and should not be overlooked.
CHAPTER VII
SIMULATED LOGICAL TERRAIN PATTERN RECOGNITION
AND MAPPING PROBLEM

A computer simulation model is developed in this chapter to illustrate in detail the development and implementation of a logical terrain pattern recognition and mapping program. The emphasis here is on the expansion of concepts presented in Chapter 6. These include terrain organization for a given terrain-problem situation, definition of terrain components for recognition and mapping, and computer programming of recognition and mapping strategies. The model is an analytical representation of strategies which the terrain analyst might employ to recognize and delimit selected terrain features from stereoscopic aerial imagery. If desired, the operation of the simulation model can be followed with the data provided and the software listings in the appendix. In the next chapter these techniques are applied to real data.

7.1 Simulation Problem

The problem to be modeled is that of landform mapping, i.e., the classification and boundary delineation of major landforms in a study area. This is an intermediate but general type of problem in the sense that the results can be used in a wide range of more specific engineering problems.

We retain this generality in modeling the geographic setting and terrain which could represent a wide class of terrain patterns in temperate regions. The terrain model simulates a portion of an alluvial valley and its adjacent uplands as schematically represented in map
form in Fig. 7.1.

The task then is to develop and implement a pattern recognition and mapping strategy to produce results similar to Fig. 7.1 from input data representing terrain surface configuration and multiband aerial photography of the study area. Table 7.1 outlines the strategy development and program implementation now briefly discussed, and forms the guide for subsequent detailed discussions.

Development of the analysis strategy begins with the conceptual organization of terrain components. This involves a definition of type and spatial association of terrain components and relations between the input data and terrain components expected in the study area. Next, a sequential mapping strategy is outlined. This involves specification of criteria for recognition of a start state in the input data to initialize the pattern recognition algorithms. Given the start state, a plan or algorithm must be formulated to move along the boundaries of the selected terrain components. When a boundary is completed, an algorithm is required to select and identify a new boundary to be mapped or identify the stop state which reflects the condition that all terrain components have been mapped, i.e., classified and delineated. The last stage in the development of the analysis strategy involves preparation of statements which define the terrain components and their relations in terms of the input data. These definitions need only be sufficient to provide in-context decision criteria for logical recognition of terrain component classes.

Implementation of the pattern recognition strategy involves preparation of the program software and its testing on a digital computer system. The program is modularized into subroutines to accomplish separate recognition and mapping tasks under control of a main program. The model is explicitly defined in the FORTRAN computer language for
Fig. 7.1 Model map showing spatial distribution of landforms.

Key:
X  Start State
2  River
3  Flood Plain
4  Terrace
5  Upland
TABLE 7.1

OUTLINE FOR DEVELOPMENT AND IMPLEMENTATION
OF PATTERN RECOGNITION AND MAPPING
STRATEGIES AND PROGRAMS

I. DEVELOPMENT
   A. Organization of terrain components, LFC.
      1. List type and spatial associations of
         expected LFCs.
      2. Define gross relations between data and
         LFCs.
   B. Outline sequential mapping strategy.
      1. Specify start state and its validation.
      2. Sketch boundary mapping algorithm.
      3. Sketch algorithm for location and
         identification of new boundaries to be mapped.
      4. Define stop state condition.
   C. Define MLFs and their relations in terms of the data.
      1. Define MCL classes and their discriminators.
      2. define start state.
      3. Prepare MLF definitions.
      4. Prepare MLF relational definitions.

II. IMPLEMENTATION
   A. Prepare modular software program for
      1. Input
      2. Start state location and verification
      3. Boundary mapping
      4. Boundary search and
      5. Output.
   B. Test and modify program.
      1. Test program
      2. Modify program where required.
operation on a time-share system.

To illustrate the development and implementation of pattern recognition and mapping strategy we develop the model in a manner of a real world problem. This scenario is then a model of model development. We will employ four means of describing the model: a verbal or written description, a terse listing of algorithms, flowcharts, and computer program source listings.

7.2 Simulation Data

The simulation data is of two types: 1) a spatial elevation matrix, \( IZ(IX,IY) \) and 2) two spatial tone matrices, \( IT2(IX,IY) \) and \( IT3(IX,IY) \). For both types of data, the matrices are square and coincident with elements \( IX=1,32 \) and \( IY=1,32 \). The distance between data points on the ground in IX and IY directions is 500 ft; thus the size of the study area is slightly less than 9 sq. mi.

The spatial elevation data shown in Fig. 7.2 are given in decimal integer values varying from 2 to 60. Each element represents a point elevation measurement truncated in ten foot intervals, i.e., the unit foot values have been wasted, giving the data an appearance of a filled contour matrix. At any point, \((IX,IY)\) in the spatial elevation matrix the true ground elevation data can range from \(20<IZ<30\) to \(600<IZ<610\) ft.

Tone matrices \( IT2(IX,IY) \) and \( IT3(IX,IY) \) are given in Figs. 7.3 and 7.4, respectively. Matrix elements represent a linear quantization of the optical transmission for points in an aerial photographic image and each spatial tone matrix represents data from a different spectral band, i.e., a different photographic filter-emulsion combination. We will assume that the tone data has been corrected and normalized to remove effects of ground slope and aspect, sun angle and intensity, photographic systems nonuniformity, chemical processing, etc.
### Fig. 7.2 Spatial elevation matrix, $IZ(IX, IY)$, $IX=1, 32$, $IY=1, 32$. Matrix elements should be multiplied by 10 to obtain estimate of surface elevation in feet.
Fig. 7.3 Spatial tone matrix $I_{2}(IX,IY)$. 
|     | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
|-----|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 32  | *32|   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 31  | *31|   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 30  | *30|   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 29  | *29|   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 28  | *28|   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 27  | *27|   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 26  | *26|   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 25  | *25|   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 24  | *24|   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 23  | *23|   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 22  | *22|   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 21  | *21|   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 20  | *20|   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 19  | *19|   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 18  | *18|   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 17  | *17|   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 16  |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 15  |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 14  |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 13  |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 12  |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 11  |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 10  |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 9   |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 8   |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 7   | *7 |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 6   |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 5   | *5 |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 4   |   | *4 |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 3   | *3 |   |   | *3 |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 2   | *2 |   |   |   |   | *2 |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1   | *1 |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |

Fig. 7.4 Spatial tone matrix IT3(IX,IY).
Thus at any point of the ground indicated by a data matrix sample point (IX, IY) we have three data values: an elevation of the terrain surface, \( IZ(IX, IY) \), and two values representative of terrain reflectance in two spectral bands, \( IT2(IX, IY) \) and \( IT3(IX, IY) \), respectively.

7.3 Development of Pattern Recognition and Mapping Strategies

The discussion will now follow the outline of Table 7.1, for the development of pattern recognition and mapping strategies.

7.3.1 Conceptual Terrain Organization

Knowing we are dealing with a study area in a rural location containing an alluvial valley and its associated uplands, from our knowledge of geomorphology we can list four types of expected landforms: a river, flood plain, terraces, and undifferentiated upland. We assume that only one level of terraces exist in the area. The spatial association of these landforms can be represented in a contiguity matrix IC (I,J) indicating all allowable contiguous landform situations.

\[
\begin{array}{c|cccc}
  & R & F & T & U \\
\hline
RIVER (R) & 0 & 1 & 1 & 1 \\
FLOOD PLAIN (F) & 1 & 0 & 1 & 0 \\
TERRACE (T) & 1 & 1 & 0 & 1 \\
UPLAND (U) & 1 & 0 & 1 & 0 \\
\end{array}
\]

where '1' matrix elements indicate positive contiguity relations and '0' indicates no contiguity relation. It is useful to draw a simple map which represents IC(I,J).
And of course if the map is first conceived, its representative contiguity matrix can be defined. Thus the river can be expected to be found adjacent to all landforms except another river, whereas the flood plains can be expected contiguous to the river and terraces, but not other flood plains or the upland, etc.

Again from our knowledge of geomorphology we can hypothesize gross elevation relationships between landforms in the input data. Thus the river will have the lowest topographic elevation in its locality, the flood plain will be higher than the river but lower than the terrace or upland. Similarly the terraces are higher than the flood plain or river but lower than the upland and, lastly, the upland will be locally the highest landform. This is expressed as an elevation relation matrix \( \text{RZ}(I,J) \)

\[
\begin{array}{ccccc}
R & F & T & U \\
R & 0 & LT & LT & LT \\
F & GT & 0 & LT & LT \\
T & GT & GT & 0 & LT \\
U & GT & GT & GT & 0 \\
\end{array}
\]

where 'LT' is 'less than' and 'GT' is 'greater than'.

If we assume mid-summer aerial photography we can relate the tone data to gross terrain material types. Four major material types are chosen for the rural study area: water, light soil, dark soil, and vegetation. We assume training samples will be collected for partitioning spectral tone vector classification space to verify this assumption; if the assumption is not verified additional material classes are required.

Next we propose a terrain material-landform implication matrix \( \text{MTLI}(k,J) \) which associates the above terrain materials with landforms.
This MTLI(k,J) matrix indicates that water is expected only in the river, whereas light soil is associated with the alluvial soils of the terrace and flood plain, dark soils only with the upland and vegetation with all land areas.

This completes our formulation of apriori terrain organization concepts for the study area.

7.3.2 Sequential Mapping Strategy

A sequential mapping strategy will now be outlined for the model according to Table 7.1. The result of applying this strategy is supposed to produce a matrix MAP(IX,IY), IX=1,32, IY=1,32, containing zeros except for elements classified with landform boundary class codes.

7.3.2.1 The Start State $S_0$

Our apriori concepts of terrain organization indicated in the contiguity matrix IC(I,J), the elevation relation matrix RZ(I,J), and the terrain material landform implication matrix MTLI(K,J) can assist in the selection of a start state $S_0$ which when located and validated in the input data matrices will initialize the pattern recognition algorithms to follow. There are two choices for a start state $S_0$; a start state associated with the river R or the upland U. To make the selection we review our terrain organization concepts related to R and U. From RZ(I,J), R is lower than any other landform class, LFC, whereas U is higher than any other LFC. From IC(I,J), R could be contiguous to any LFC, whereas U is contiguous only to R and T. Lastly, from MTLI(K,J), R is only implied from WATER, whereas U is implied by either
DARK SOIL or VEGETATION. We choose $S_0$ associated with $R$ because of simpler recognition logic requirements and our approach is to locate a point of minimum elevation in $IZ(IX,IY)$, which we call $IZMIN$, and at this point $(IX,IY)$ define $IZMINX=IX$ and $IZMINY=IY$. Then with spectral tone data at this point, i.e., $IT2(IZMINX,IZMINY)$ and $IT3(IZMINX,IZMINY)$, when plotted in a terrain material classification space $TCLASS$ (yet to be developed) a material class $MCL$ of WATER will result.

It is convenient to define the relation between $S_0$ and landform classes in the contiguity matrix $IC(I,J)$ and the material classes $MCL$ in the material-landform implication matrix $MTLI(MCL,LFC)$ as shown in Fig. 7.5. Then since $IC(S_0,R)$ is the only valid state, i.e., $IC(S_0,R) = TRUE$ and $IC(S_0,-R) = FALSE$, $S_0$ is located and verified. This algorithm is shown more succinctly in Fig. 7.6.

7.3.2.2 The Boundary Mapping Algorithm

Following Table 7.1 we must now sketch the boundary mapping algorithm. We define a boundary as a spatial partition between two elements of different landform classes to be mapped $MLF$, i.e., the landforms which have been defined. We make the distinction between $MLF$ and $LFC$ for the reason that, through choice or incomplete apriori knowledge, the set of classes $MLF$ representing the landforms to be mapped and the set $LFC$ indicating the landforms which exist in the area may not be identical. For our model $(x) [(x \in MLF) \equiv (x \in LFC)]$, however, we will refer to $MLF$ instead of $LFC$ when boundary mapping is considered.

The boundary mapping algorithm must have two basic capabilities: 1) it must be capable of distinguishing which elements of the input data are boundary elements and 2) it must be capable of moving sequentially around the boundary of a given $MLF$. To realize these capabilities we must adopt a number of conventions or rules which provide a
**LANDFORM CONTIGUITY MATRIX IC(I,J)**

**LFC**

<table>
<thead>
<tr>
<th>LFC</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>IC(I,J)</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>LFC 3</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

**LFC Codes:**

- 1 $=_{df}$ START STATE, $S_O$
- 2 $=_{df}$ River, $R$
- 3 $=_{df}$ Flood Plain, $F$
- 4 $=_{df}$ Terrace, $T$
- 5 $=_{df}$ Upland, $U$

**TERRAIN MATERIALS-LANDFORM IMPLICATION MATRIX MTLI(MCL,LFC)**

**LFC**

<table>
<thead>
<tr>
<th>MCL</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>MTLI(K,J)</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>MCL 3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

**MCL Codes:**

- 1 $=_{df}$ WATER
- 2 $=_{df}$ LIGHT SOIL
- 3 $=_{df}$ DARK SOIL
- 4 $=_{df}$ VEGETATION

**Fig. 7.5** Landform contiguity matrix IC(I,J) and terrain materials-landform implication matrix MTLI(K,J) containing $S_O$ as an LFC.
deterministic order for our algorithm. The start state algorithm of Fig. 7.6 is a set of such rules. The rules are instructions which must be followed to obtain predictable results. To be effective an algorithm must contain a complete set of rules covering the range of conditions presented in the data. When an unforeseen condition is experienced for which no rule exists, an error usually results. Thus we are interested in defining a complete set of rules for our boundary mapping algorithm.

**Boundary mapping conventions.** Our first boundary mapping convention pertains to the amount of input data used at any one time to distinguish boundary elements. According to our informal definition of a boundary we can state that a minimum of two points must be classified to determine the existence of the boundary; each point being a different element of the MLF set, i.e.,

\[(x \in \text{MLF}) \& (y \in \text{MLF}) \& \text{NE}(x,y)\]

where 'NE(x,y)' says 'x is not equal to y'.

To account for all boundary positions with respect to a point in a square matrix we must consider eight adjacent points. This we will call the mapping field of view matrix MFLD(M,N), M=1,3, N=1,3 and using the coordinate convention

\[
\begin{array}{ccc}
(3,1) & (3,2) & (3,3) \\
(2,1) & (2,2) & (2,3) \\
(1,1) & (1,2) & (1,3) \\
\end{array}
\]

We can think of MFLD(M,N) as a 3x3 element matrix aperture superimposed upon the three data matrices allowing access to data only within the aperture. As a second boundary mapping convention, it is assumed that the element (2,2) is always a boundary element and contains data which has been classed as the MLF being mapped, and that one or more of the neighbor elements represent MLF of different classes. A third rule
START STATE LOCATION AND VERIFICATION ALGORITHM

1. Find $IZMIN$ in $IZ(IX,IY)$.

2. Let $IZMINX = IX$ and $IZMINY = IY$.

3. If $TCLASS$ for point $(IT2(IZMINX,IZMINY),IT3(IZMINX,IZMINY))$ indicates $MCL = WATER$, then go to 4; otherwise goto 1.

4. If $MTLI(\text{WATER, ?})$ implies $R$, then goto 5; otherwise goto 1.

5. If $IC(S_o,R)$ is TRUE, then goto 6; otherwise goto 1.

6. $S_o$ has been located and verified.

Fig. 7.6 Algorithm for simulation model start state location and verification.
indicates that we are not interested in classifying points of MFLD(M,N) which are not identical to the MLF being mapped. A fourth rule indicates that only data from element (2,2) is put into the output matrix representing the boundary map MAP(IX, IY) however all new data is transferred to a temporary boundary matrix IBMAP(IX, IY). Thus we give MFLD (M,N) elements not equivalent to MFLD(2,2) a temporary symbol to be stored in IBMAP(IX, IY).

MFLD(M,N) Movement conventions. The following conventions pertain to the sequential movement of MFLD(M,N) around the boundary of a given MLF. First we assume for the general case that MLF boundaries are closed curves. This necessitates a second convention where the study area boundaries have truncated the LFC boundaries; the study area boundaries are then considered to be MLF boundaries. Mapping of lines or points, which represent LFCs having local area dimensions below the data matrix resolution capabilities, will be considered as special cases.

An important convention for the boundary mapping algorithm assumes the initial boundary point and its MLF class is provided by another algorithm. Given the initial boundary point we adopt a clockwise movement or mapping convention around the boundary. Development of criteria for directional movement of MFLD(M,N) will be postponed until the software preparation stage is discussed. The boundary is considered to be effectively mapped when the algorithm returns MFLD(2,2) to the initial point or when a special case stopping condition for a line or point obtains. This completes the preliminary sketch of the boundary mapping algorithm.

7.3.2.3 Locating and Identifying a New Boundary

From Table 7.1 the next task is to sketch an algorithm for locating and identifying a new boundary to be mapped. Here we adopt a simple
scan search for a temporary unclassified boundary code which the mapping algorithm has inserted into IBMAP(IX, IY). The scan convention is a rowwise raster scan terminating when the unclassified boundary is detected. The matrix address of this symbol is then the initial point of the next boundary to be mapped after the new boundary has been classified.

At this stage of strategy development we are not prepared to completely define the algorithm for classification of the new boundary but we can sketch the general considerations. Given an unclassified point at element address (IX, IY) in IBMAP(IX, IY) we know that at least one immediate neighbor contains a classified MLF representative of a point on a boundary previously classified and mapped. This classified point is found and used in the contiguity matrix IC(I, J) to infer one or more possible MLF classes which might represent the classification at (IX, IY). For example, if the classed adjacent point was MLF = 2, from IC(2, J), J=1, 5, we know that our unknown is MLF = 3, 4, or 5, i.e., the river is adjacent to flood plains, terrace or upland. The next step in our identification procedure would test definitions (yet to be discussed), and MTLI(K, J) of the MLFs to determine which is the correct MLF. When this is accomplished, the algorithm for location and identification of a new boundary is complete and a transfer is made to the boundary mapping algorithm.

A stop state condition S5 obtains when no unclassified boundary points are located. This indicates that the mapping has been successfully completed and a transfer is then made to the output algorithm.

Fig. 7.7 is a flowchart for this preliminary boundary mapping algorithm and stages of its application for the production of MAP(IX, IY).

7.3.3 MLF Definition Formulation

Explicit MLF definitions are required to implement any recognition
Fig. 7.7 Flowchart for preliminary boundary mapping algorithm and sequential stages of its application to produce a completed map, MAP(IX, IY).
and mapping algorithm in our model. According to Table 7.1 we must define the MLFs and required relations in terms of the data to be classified. These definitions will be formulated as logical statements to be tested with study area data.

From previous discussion in this chapter we saw that definitions are required for the start state $S_0$, for identifying MLF membership in the boundary mapping algorithm and identifying MLF membership and relations in locating new boundaries to be mapped. Before these definitions are presented we review briefly the main aspects of terrain description formulation.

There are three prime considerations for the formulation of MLF definitions:

1. The definitions need only be sufficient to establish class membership. The simplest definition which always allows correct identification of a MLF from a set of MLFs is the desired definition for that MLF.
2. The MLF definition can use sequential conditional logic. Definitions can be prepared as IF-type statements appropriate with the structure of a decision tree.
3. The structural organization of the study area is

and we are interested in formulating mutually exclusive definitions of the terrain patterns to be mapped, i.e., MLFs.

Study area data representative of a given MLF can be expected to have some variability and thus the definitions must be responsive within
a range of data values. Our model uses three data sources, I2(IX,IY), IT2(IX,IY), and IT3(IX,IY), and we must know the data ranges of each which are necessary for our definitions. Sources of terrain elevation variability could result from local natural erosion and deposition processed or the nature of our sampling and quantizing technique relative to the terrain surface configuration. Sources of photographic tone variations could result for natural variations in the reflectance properties of the terrain materials, e.g., soil moisture content or terrain slope or vegetation soil cover variations, or time of day, seasonal or meteorological effects, sky conditions, etc.; while recording system variations, chemical processing, sampling and quantizing are also sources of possible variability.

There are two basic approaches to obtaining the data range necessary for the definitions. The first is the apriori approach which involves a preliminary estimate of the appropriate data values from existing knowledge, e.g., experience, maps, literature, etc., and the second is a learning procedure which involves selective sampling from the study area data to obtain estimates of the required values. If topographic maps are available elevation value estimates may be easily obtained, whereas value estimates for tone data can be obtained through a learning technique. With either approach errors may result from poor estimates and in practice an error correcting feedback cycle can be used which is based upon output evaluation. This could involve a complete model modification or simply changing the discriminant values in the definitions.

7.3.3.1 MCL Definitions

Tone data for our model was assumed to be corrected and normalized, and divisible into four gross material types, i.e., water, light, soil, dark soil, and vegetation, which can be identified by spectral tone vectors when plotted in the classification space such as shown in Fig.
7.8 We assume disjoint data which can be partitioned such that when a spectral vector \((IT2(IX, IY), IT3(IX, IY))\) for the point \((IX, IY)\) is plotted in this classification space a materials classification MCL results. Given the set of partitions we can prepare the simple MCL definitions given in Fig. 7.8. Though not required, it is a simple task to prepare and run a program which classifies the materials at each point in the study area from the definitions. This map, we call MCL(IX, IY), appears as Fig. 7.9 and can be used for additional spot checking the representativeness of the partitions to the data in the space domain.

According to our considerations for formulation of MLF definitions, the definitions given in Fig. 7.8 are simple and mutually exclusive to distinguish class membership. They use the sequential conditional logic and within the structural organization of the study area they relate study area data to terrain features to be mapped. The terrain material-landform implication matrix MTLI(K, J) given in Fig. 7.5 then relates the MCLs to the proper class or classes of landforms to be mapped MFL.

7.3.3.2 Start State Definition

The start state \(S_0\) has been described in Section 7.3.2.1 and an algorithm has been given in Fig. 7.6 for its location and verification on the study area data. The algorithm represents a sequential, conditional formulation of the definition. If we choose to represent the definition in a simple first order predicate calculus formulation, the conditional formulation is retained but we lose the aspects of sequential operations. It is illustrative to present this formulation using Fig. 7.6 as a guide.

First we define IZMIN to be the minimum value of the elevation matrix IZ(IX, IY) by

\[
IZMIN = \text{df} (x)(y)[(x \in X) \& (y \in Y) \& \text{MIN}(IZ(x, y))]
\]
Spectral vector classification space for determining material classifications, MCL.

1 = \textit{df} WATER = \textit{df} IT2(IX, IY) + IT3(IX, IY) \leq 10

2 = \textit{df} LIGHT SOIL = \textit{df} (IT2(IX, IY) + IT3(IX, IY) > 10) \& (IT2(IX, IY) + IT3(IX, IY) \leq 25)

3 = \textit{df} DARK SOIL = \textit{df} (IT2(IX, IY) + IT3(IX, IY) > 25) \& (IT3(IX, IY) - IT2(IX, IY) \geq 5)

4 = \textit{df} VEGETATION = \textit{df} (IT2(IX, IY) + IT3(IX, IY) > 25) \& (IT2(IX, IY) - IT3(IX, IY) > -5)

Material classification definitions.

Fig. 7.8 Material classifications from a partitioning of a plot of spectral tone vectors.
Fig. 7.9 Material classification map, MCL(IX,IX).

---

**Material Classification Map, MCL(IX,IX):**

- The diagram represents a material classification map, likely used in a specific context such as material science or engineering.
- The map is labeled with indices or categories, indicating different materials or properties.
- The diagram is detailed, with various symbols and lines suggesting the distribution or classification of materials according to specific criteria.
Then we define the coordinate values at which IZMIN occurs.

\[
IZMINX = \text{df} \ (Ex)[(x \in X) \& \text{EQ}(IZMIN,IZ(x,Y))] \\
IZMINY = \text{df} \ (Ey)[(y \in Y) \& \text{EQ}(IZMIN,IZ(x,y))] 
\]

and

Now the start state \( S_0 \) can be defined by

\[
(z = S_0) = \text{df} \ (Ex)(Ey)[(x \in MCL) \& (x = \text{WATER}) \& \\
(y \in MLF) \& (y = R) \& \text{IM}(CL((IT2(IZMINX,IZMINY), \\
IT3(IZMINX,IZMINY)),x),\text{IM}(MTLI(x,y), \\
\text{IM}(IC(S_0,y),z)))]
\]

which contains embedded implications, each states as 'IM(x,y)' reading as 'x implies y' and 'CL(x,y)' reads as 'x is classified as y'. For a precise logical definition of \( S_0 \) we could have used embedded inference expressions containing implications, however the added length and redundancy of expressions tends to obscure the details for this discussion.

7.3.3.3 MLF Definitions

The MLF definitions are formulated for and used in the context of the boundary mapping algorithm and the algorithm for location and identification of new boundaries to be mapped.

We choose a simple approach to defining the MLF classes which requires that representative data from IZ(IX, IY) be available for the elevation range of each MLF class and elevation variations between adjacent points on the surface of MLFs of a given class. MLF definitions also utilize IT2(IX, IY) and IT3(IX, IY) to define the MCL at a given point, and then to validate the MLF inferred by MCL, using the terrain material-landform implication matrix MTLI(MCL, MLF) as given in Fig. 7.6.

An example MLF definition for a terrace, i.e., MLF = TERRACE, would be

\[
\text{TERRACE} = \text{df} \ (IZ(IX, IY)>21) \& (IZ(IX, IY)<28) \& \\
\]

\[(\text{ABS}(\text{IZ}(\text{IX},\text{IY}) - \text{IZ}(\text{IIX},\text{IIY})) < 2) \&
(\text{MTLI}((\text{CL}((\text{IT}2(\text{IX},\text{IY}), \text{IT}3(\text{IX},\text{IY})), \text{MCL}), \text{T})))\]

where \((\text{IIX},\text{IIY})\) is a neighborhood point to \((\text{IX},\text{IY})\). Other \text{MLF} definitions can be formulated in a similar manner.

These definitions should not be used out-of-context if the logical pattern recognition model is to simulate the terrain analyst. The definition out-of-context indicates that by knowing the elevations at two adjacent points, say A and B, and the tone data at one of the points, say A, one can decide whether or not A belongs to the \text{MLF} class called \text{TERRACE}. If the definition is used out-of-context, there would be no need for boundary mapping algorithms; a simple raster scan where each successive matrix element is tested by the definitions is used with out-of-context definitions. This approach does not account for ambiguous data and thus is unusable in the general mapping case.

Fig. 7.10 indicates the in-context algorithmic definition of \text{TERRACE} as used in the boundary mapping algorithm. In such a case we know the \text{MLF} class of \text{MFLD(2,2)} and the object is to classify the neighbors which have not been previously classified.

### 7.3.3.4 \text{MLF} Relational Definitions

The contiguity matrix \(\text{IC(I,J)}\) represents one means for defining \text{MLF} relations. This type of relational definition, which also includes \text{MTLI(K,J)}, does not directly concern the relationship of data to the \text{MLFs}, but rather expresses allowable associations between \text{MLFs} or between intermediate data classes, i.e., \text{WATER}, \text{SOIL}, \text{VEGETATION}, and the \text{MLFs}.

Our model also includes relational definitions between \text{MLFs} which directly concern data. Thus for each positive association between \text{MLFs} if \(\text{IC(I,J)}\) we have a relational definition with data discriminators. These are used to imply and verify the existence of a specific
OBJECTIVE: To classify IBMAP(IIX,IIY) as MLF = TERRACE or MLF = UNID, i.e., temporarily unidentified.

CONTEXT:
1. If MFLD(2,2) = IBMAP(IX,IY) = MLF = TERRACE, then continue; otherwise an error exists.
2. Find a neighbor point (IIX,IIY) to (IX,IY) such that IBMAP(IIX,IIY) # MLF = RIVER, TERRACE, FLOOD PLAIN, or UPLAND.

ALGORITHM BODY:
3. If IZ(IIX,IIY) ≥ 21 and IZ(IIX,IIY) ≤ 28, then continue; otherwise go to 8.
4. If ABS(IZ(IX,IY) - IZ(IIX,IIY)) ≤ 2, then continue, otherwise go to 8.
5. If MTLI(CL((IT2(IIX,IIY),IT3(IIX,IIY)),MCL), TERRACE), then continue; otherwise go to 8.
6. IBMAP(IIX,IIY) = MLF = TERRACE.
7. The classification of IBMAP(IIX,IIY) is completed.
8. IBMAP(IIX,IIY) = UNID.
9. The classification of IBMAP(IIX,IIY) is temporarily coded as UNID.

Fig. 7.10 In-context algorithmic definition of 'TERRACE' as used in the boundary mapping algorithm.
boundary condition between MLFs from the data. For example, for IC(2,4), i.e., when a TERRACE is implied to be contiguous to a RIVER in Fig. 7.5, we have the conditional definition

\[
\text{IF } \text{IZ(IX, IY) - IZ(IIX, IYY) > 14 and}
\]
\[
\text{if } \text{IZ(IX, IY) - IZ(IIX, IYY) < 18, then}
\]
\[
a \text{TERRACE is inferred to be adjacent to}
\]
\[
a \text{a RIVER; otherwise MLF must be located}
\]
\[
\text{adjacent to the RIVER.}
\]

From context we should have known that the existence of RIVER was represented by IC(2,-) and IZ(IIX,IIY). Similarly, other MLF relational definitions can be prepared. Definitions must be prepared for each positive entry in IC(I,J) which does not pertain to the start state nor a diagonal matrix element, i.e., I=J in IC(I,J). An additional simplification exists due to the symmetry of IC(I,J) when the first row and column are disregarded, and the complementary relationship exists between MLFs. For example, if RIVER is lower than FLOOD PLAIN, the FLOOD PLAIN is higher than RIVER. Thus if IC(I,J) is a unit matrix, i.e., all elements have the value '1', except for the row and column containing the start state, it would indicate every MLF is possibly contiguous to every other MLF, then the number of relational definitions required would be \(N^2-3N+2\), where \(N\) is the number of row or column elements in the square matrix. Because of symmetry only half of this number represents the number of definitions requiring unique data relations. And of course if '0' elements occur there are no definitions required and each lessens the number of required definitions by one. In our model \(N=5\); the number of possible definitions would be 12; the number of actual definitions, 10; and the number of unique data definitions, 5.

7.4 Implementation of Pattern Recognition and Mapping Strategies
Implementation of pattern recognition strategies involves preparation of the program software and its testing with data in a digital computer system. The simulation program source listing in FORTRAN IV are included as an appendix. A glossary of major terms used in this program is given in Table 7.2.

The major aspects of simulation program and its modules will now be described with the aid of flowcharts. Specific references to the source listings will be given by line number as 'LINE ___' and/or statement number as 'STMT ___', both occur on the left edge of the listings.

7.4.1 LPRP - Logical Pattern Recognition Program

Program LPRP is the main program for the simulation problem. It has as a main purpose the illustration of logical pattern recognition programming using simulated data of terrain elevation and two photographic spectral bands. Fig. 7.11 shows a flowchart of the program to involve three basic functions: 1) to input the data IT(32,32), IT2(32,32), and IT3(32,32), and the relation matrices IC(5,5) and MTLI(4,5) into the machine and zero MAP(32,32) and IBMAP(32,32); 2) to process this data with a group of subroutines to produce MAP(IX,IY), I=1,32, J=1,32, where each element is a zero or a number code representing a MLF class; and 3) to print the landform boundary map MAP(IX,IY).

The processing of data begins with STRTS, which verifies the start state definition, and locates and classifies, with the aid of TCLASS, the start point with a MLF class indicating the boundary to be mapped. LFCMP then classifies all elements of the 3x3 matrix MFLD(I,J), which is centered on the start point, either as the same MLF class as the boundary being mapped or a '7' indicating an unclassified boundary point with MLF class different from that being mapped. This classification is accomplished by LFCLAS and TCLASS. When MFLD(I,J) has been processed its contents are put into IBMAP(IX,IY) at the proper addresses.
**TABLE 7.2**

GLOSSARY OF MAJOR TERMS USED IN THE SIMULATED LOGICAL TERRAIN PATTERN RECOGNITION AND MAPPING SOURCE PROGRAMS

**PROGRAM NAMES:**

- LPRP, Main program
- LFCLAS, SUBROUTINE
- LFCMP, SUBROUTINE
- MOVE, SUBROUTINE
- PRINT3, SUBROUTINE
- PRINT4, SUBROUTINE
- SMSRCH, SUBROUTINE
- STRTS, SUBROUTINE
- TCLASS, SUBROUTINE

**MAJOR VARIABLES:**

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEF</td>
<td>MLF definition value, logical</td>
</tr>
<tr>
<td>IA(I,J)</td>
<td>Movement indicator weights, (3,3)</td>
</tr>
<tr>
<td>IBMAP(IX,IY)</td>
<td>Temporary boundary map, (32,32)</td>
</tr>
<tr>
<td>IBSTOP</td>
<td>Special boundary mapping stop indicator</td>
</tr>
<tr>
<td>IC(I,J)</td>
<td>Contiguity matrix, (5,5)</td>
</tr>
<tr>
<td>IT2(IX,IY)</td>
<td>Photographic tone matrix #2, (32,32)</td>
</tr>
<tr>
<td>IT3(IX,IY)</td>
<td>Photographic tone matrix #3, (32,32)</td>
</tr>
<tr>
<td>IZ(IX,IY)</td>
<td>Elevation data matrix, (32,32)</td>
</tr>
<tr>
<td>IZMIN</td>
<td>Minimum elevation value in IZ(IX,IY)</td>
</tr>
<tr>
<td>IZMINX</td>
<td>x-element of coordinate associated with start state</td>
</tr>
<tr>
<td>IZMINY</td>
<td>y-element of coordinate associated with start state</td>
</tr>
<tr>
<td>LFC</td>
<td>Landform classification set</td>
</tr>
<tr>
<td>M</td>
<td>Movement indicator number</td>
</tr>
<tr>
<td>MAP(IX,IY)</td>
<td>Program boundary map, (32,32)</td>
</tr>
<tr>
<td>MCL</td>
<td>Material classification set</td>
</tr>
<tr>
<td>MFLD(I,J)</td>
<td>Boundary mapping field of view, (3,3)</td>
</tr>
<tr>
<td>MLF</td>
<td>Mapping landform classification set</td>
</tr>
<tr>
<td>MTLI(K,J)</td>
<td>Terrain material-landform implication matrix, logical, (4,5)</td>
</tr>
<tr>
<td>NB</td>
<td>Boundary counter</td>
</tr>
<tr>
<td>NM7</td>
<td>No more 7's indicator</td>
</tr>
</tbody>
</table>
Fig. 7.11 Flow chart of main program LPRP -- Logical Pattern Recognition Problem.
and the MLF class of element (2,2) of MFLD(I,J) is put into MAP(IX,IY) at the proper address. MOVE then depositions MFLD(I,J) in a clockwise manner along the boundary of MLF in IBMAP(IX,IY). When the initial point is reoccupied in IBMAP(IX,IY) by MFLD(2,2), the boundary has been mapped by LFCMP. Next SMSRCH accomplishes a simple scan search of IBMAP(IX,IY) for a '7' which will be the initial point of a new boundary to be mapped. If a '7' is detected set NM7=0 and this point (IX,IY) is given an MLF classification by LFCLASS and TCLASS then control branches to LFCMP and continues until no '7' is detected. If no '7' is detected, set NM7=1 and exit the subroutine and commence the output processes.

7.4.2 Subroutine STRTS

SUBROUTINE STRTS is called by LPRP to validate the start state, $S_0$, and select and classify the initial boundary mapping point. It is only used once; to initialize the boundary mapping algorithm of SUBROUTINE LFCMP. The approach used by STRTS is detailed in Sections 7.3.2.2 and 7.3.3.2, and Fig. 7.6 gives the algorithm for software preparation, while Fig. 7.12 presents a condensed flowchart for STRTS.

After IZNMIN is located as (IZMINX,IZMINY) in IZ(IX,IY), SUBROUTINE TCLASS is called to provide the MCL class at this point from the data in IT2(IX,IY) and IT3(IX,IY). Then given MCL, the MLF is found which satisfies both MTII(MCL,MLF) and IC(1,MLF) in LINE 1130 and thus verifies the start state definition. Finally, the MLF class is put into MAP(IX,IY) and IBMAP(IX,IY) at (IZMINX,IZMINY) and control returns to LPRP.

7.4.3 Subroutine LFCMP

SUBROUTINE LFCMP is called by LPRP to map the boundary of a given MLF in MAP(IX,IY) and IBMZP(IX,IY), beginning from a given starting point on the MLF boundary. In preparation of this subroutine the con-
Fig. 7.12 Flow chart for SUBROUTINE STRTS.
ventions and algorithm outlined in Section 7.3.2.2 was closely followed. Figs. 7.13 and 7.14 present the generalized and detailed flowcharts, respectively for LFCMP.

Upon entry, the initial boundary coordinate is reference coded (KX,KY) and its flag KF is set to zero to indicate the first boundary point. If NB, a boundary counter in LINE 1430, is zero, then the initial boundary is to be mapped and the 3x3 mapping field of view MFLD (I,J) is initialized with zero valued elements in LINE 1435. The MLF class determined by STRTS is then assigned to MFLD(2,2).

If NB has a value greater than zero, then MLF class data prepared in SUBROUTINE SMRSRC is available in IBMAP(IX,IY) which can be used in MFLD(I,J). This transfer is accomplished in LINE 1446.

Next a test is made of the current boundary coordinate and the initial boundary coordinate. If these are identical and KF=1, then the boundary has been completed and control is returned to LPRP; otherwise (IX,IY) is a new point on an uncompleted map boundary.

If (IX,IY) is located on the study area boundary, 8's are put in the MFLD(I,J) elements that overlap this boundary and a boundary flag IBF is set to unity.

Next the elements of MFLD(I,J) that are equal to '0', '7', or '9' must be MLF classified for the boundaries being mapped. With (IIX,IIY) being a neighbor point of (IX,IY), if IBMAP(IIX,IIY)=MFLD (2,2)=MLF of the boundary being mapped, then that element of MFLD(I,J) is given the MLF value. If IBMAP(IIX,IIY) has a MLF class value other than the one being mapped it is first checked for contiguity validity in IC(I,J) and if valid this value is placed in MFLD(I,J) at the appropriate element position; otherwise it is given the value of '9' in MFLD(I,J) and IBMAP(IIX,IIY) indicating it is unclassified. If IBMAP (IIX,IIY equals '0', '7', or '9', then SUBROUTINE LFCLAS is called to
Fig. 7.13 Generalized flow chart for SUBROUTINE LFCMP.
Fig. 7.14 Detailed flowchart of SUBROUTINE LFCMP.
verify or deny the definition of the MLF being mapped using data form 
(IIX,IIY). LFCLAS returns a logical value in logical variable DEF and 
if DEF is true, then the definition is verified and MFLD(I,J) is given 
the value of MLF: otherwise DEF is false and MFLD(I,J) corresponding 
to (IIX,IIY) is given the value '7' indicating the data to be tempo­
arily unclassified.

When all elements of MFLD(I,J) have been filled, SUBROUTINE MOVE 
is called for selection of a new coordinate center for MFLD(2,2) along 
the boundary being mapped. Upon return to LFCMP the MLF value for 
the new center is given to MAP(IX,IY) and IBMAP(IX,IY); KF is set to unity 
and if IBSTOP, a variable indicating the special boundary mapping stop­
ping conditions for points and lines, is equal to zero the algorithm 
for a new MFLD(I,J) is repeated; otherwise IBSTOP indicates the bound­
ary is completed and control transfers to LPRP.

7.4.4 Subroutine MOVE

This subroutine, called by LFCMP, has the purpose of moving the 
3x3 boundary mapping field of view MFLD(I,J) to the next position 
around the boundary of the MLF being mapped by MFCMP, and to fill the 
elements of MFLD(I,J) with MLF's classified at the previous position. 
Fig. 7.15 presents a generalized flowchart for MOVE.

Basic to the operation of the subroutine is a 3x3 matrix of move 
indicator weights IA(I,J) shown at the top of Fig. 7.16. This matrix 
has the property that the sum of any combination of its elements, 
other than the central element AI(2,2) is a unique number in decimal 
or binary representation. This unique sum is termed a move indicator 
number M and is obtained as indicated in the top of the flowchart of 
Fig. 7.15 by assigning the elements of IA(I,J) corresponding to the 
similarly addressed elements of MFLD(I,J) which are not '8's or MLFs 
being mapped. Several examples of M for MFLD(I,J) inputs when MLF=2
Fig. 7.15 Generalized flowchart for SUBROUTINE MOVE.
\[
\begin{bmatrix}
7 & 15 & 31 \\
3 & 0 & 63 \\
1 & 255 & 127
\end{bmatrix}
\begin{bmatrix}
2^{3}-1 \\
2^{2}-1 \\
2^{1}-1
\end{bmatrix}
\begin{bmatrix}
2^{4}-1 \\
2^{0}-1 \\
2^{7}-1
\end{bmatrix}
\begin{bmatrix}
2^{5}-1 \\
2^{6}-1
\end{bmatrix}
\]

<table>
<thead>
<tr>
<th>MFLD(I,J) Input</th>
<th>Movement Indicator Number</th>
<th>Move Direction Indicated</th>
<th>MFLD(2,2) Coordinate Changes</th>
<th>Moved MFLD(I,J) Changes</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>71212</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>01010</td>
</tr>
<tr>
<td>71222</td>
<td>M=11</td>
<td></td>
<td>IX=IX+1</td>
<td>IY=IY</td>
<td>7122</td>
</tr>
<tr>
<td>71212</td>
<td></td>
<td></td>
<td>01010</td>
<td></td>
<td>7122</td>
</tr>
<tr>
<td>71712</td>
<td></td>
<td></td>
<td>IX=IX+1</td>
<td>IY=IY+1</td>
<td>2210</td>
</tr>
<tr>
<td>71212</td>
<td>M=26</td>
<td></td>
<td>01010</td>
<td></td>
<td>7120</td>
</tr>
<tr>
<td>71717</td>
<td></td>
<td></td>
<td>IX=IX</td>
<td>IY=IY+1</td>
<td>2210</td>
</tr>
<tr>
<td>71212</td>
<td>M=57</td>
<td></td>
<td>01212</td>
<td></td>
<td>2210</td>
</tr>
<tr>
<td>81818</td>
<td>M=94</td>
<td></td>
<td>IX=IX</td>
<td>IY=IY-1</td>
<td>01212</td>
</tr>
<tr>
<td>81818</td>
<td></td>
<td></td>
<td>0122</td>
<td></td>
<td>0122</td>
</tr>
<tr>
<td>81818</td>
<td>M=0</td>
<td></td>
<td>IX=IX</td>
<td>IY=IY-1</td>
<td>0122</td>
</tr>
</tbody>
</table>

Fig. 7.16 Movement indicator weights IA(I,J) and examples of SUBROUTINE MOVE applied to selected MFLD(I,J) inputs.
are given in the lower part of Fig. 7.16.

Each movement indicator number indicates one of eight move directions for the movement of MFLD(2,2) to one of its eight neighbors. For each move direction there will be a finite set of acceptable M's e.g., for M= 10, 11, 137, 232, 266, 393, 456, or 487 the MFLD(2,2) coordinates change such that IX = IX + 1 and IY = IY, as shown in the first example of Fig. 7.16. When MFLD(2,2) is located on the study area boundary, i.e., when IX = 1 or 32 and/or IY = 1 or 32, then a row or column of MFLD(I,J) will lie outside of MAP(IX,IY) and MFLD(I,J) will contain 8's in the overlap row or column and a separate set of M's exist for each move direction. Another special case exists when M = 247, 375, 499, 501, or 502; which indicates a thin boundary exists and the boundary is terminated.

After determining the coordinate changes for MFLD(2,2) elements of the previous position are transferred to the new MFLD(I,J) as shown in the right-most column of Fig. 7.16. When this operation is completed control returns to SUBROUTINE LFCMP.

7.4.5 Subroutine SMSRCH

The purpose of SUBROUTINE SMSRCH, which is called by the main program LPRP, is to locate a point from which a new MLF boundary is to be mapped. This is accomplished by detecting a '7' in IBMAP(IX,IY) using a simple row scan as indicated in the generalized flowchart of Fig. 7.17 and as discussed in Section 7.3.2.3. If no '7' exists in IBMAP(IX,IY) control returns to LPRP after setting NM7=1. When a '7' is found, set NM7=0 and begin the process of classifying the MLF at this point (IX,IY).

To identify the new boundary, the neighboring points (IX+1,IY) to (IX,IY) in IBMAP(IX,IY) are checked to locate the MLF class of the previously mapped boundary. Given this MLF, we determine from IC(I,J) a tentative MLF inference for (IX,IY). This method of branching to
Fig. 7.17 Generalized flowchart for SUBROUTINE SMSRCH.
The MLF relational definition to be checked is given in Fig. 7.18. The lower matrix in this illustration represents the matrix IC(I,J) when the row and column representing the start state have been removed.

The MLF relational definitions determine if the elevation difference between (IX,IY) and (IIIX,IIY) are within the allowable range. If not, the next relational definition inferred by IC(I,J) is checked. If the definition is verified the MLF definition is verified for another neighbor point (IIIX,IIY) adjacent to (IX,IY). This is required to assure that more than a single point of the same MLF class will be on a boundary. SUBROUTINE LFCLAS is called to test the MLF class and associated MCL for the inference. If the definitions in LFCLAS is false the next relational definition in IC(I,J) is checked; if true then IBMAP(IX,IY) and MAP(IX,IY) are given the verified classification at (IX,IY) and control is returned to LPRP.

7.4.6 Subroutine LFCLAS

This subroutine is called by SUBROUTINES LFCMP and SMSRCH. It has the purpose of verifying or denying the MLF definitions of a point in the data matrices. The MLF definitions are as outlined in Section 7.3.3.3 and the manner in which they are addressed is given in the generalized flowchart of Fig. 7.19. SUBROUTINE TCLASS is called to determine the material classification MCL at (IX,IY). The material implication matrix MTLI(MCL,MLF) is then checked. If this is true, then logical variable DEF is given the value TRUE and return is made to the calling subroutine; otherwise DEF has the value FALSE upon return.

7.4.7 Subroutine TCLASS

This subroutine is called by SUBROUTINES STRTS and LFCLAS. Its purpose is to determine the material classification MCL associated with the two-dimensional spectral vector data IT2(IX,IY) and IT3(IX,IY) as outlined in Section 7.3.3.1 for the discriminant relations defined in
Fig. 7.18 Program branching in SUBROUTINE SMSRCH to verify or deny MLF relations suggested by the contiguity matrix IC(I,J) in the process of identifying a new MLF boundary to be mapped. Numbers refer to statements in SMSRCH source listing.
Fig. 7.19 Generalized flowchart for SUBROUTINE LFCLAS.
Fig. 7.8 which partition MCL classification space. Fig. 7.20 presents the generalized flowchart for TCLASS.

The process is to check the definitions sequentially. If a true definition is found, the appropriate MCL class is assigned and control is returned to the calling program; otherwise, if no definition is satisfied an MCL class of '0' is assigned as an error class prior to return.

7.4.8 **Print Subroutines**

The purpose of the print routines, SUBROUTINES PRINT3 and PRINT4, is to annotate and format the output data from LPRP.

The difference between PRINT3 and PRINT4 is only in the output data format; '13' and 'I4' print formats are used in PRINT3 and PRINT4, respectively.

7.5 **LPRP Results**

The LPRP program output results for the given input data in IZ (IX, IY), IT2(IX, IY), and IT3(IX, IY) given in Figs. 7.2, 7.3, and 7.4, respectively, is MAP(IX, IY) as formatted by PRINT3 is presented in Fig. 7.21. The boundary codes are the LFC codes given in Fig. 7.6.

To illustrate the sequential production of the data in Fig. 7.21, the stages of boundary mapping are given in Fig. 7.22. The start state occurs at MAP(1,15) and has associated MFLD(I,J) as shown in Fig. 7.16 for the movement indicator M=94. Stage j and Stage m of Fig. 7.22 represent the thin boundary stopping conditions at MAP(23,21) and MAP (25,26), respectively.

The final IBMAP(IX, IY), with all intermediate '7's removed, is given in Fig. 7.23. With the sequential development of MAP(IX, IY) given in Fig. 7.22, the movement indicator numbers in SUBROUTINE MOVE, and the final IBMAP(IX, IY), one can reconstruct all or any portion of MAP(IX, IY) by hand methods to test the algorithms.
Fig. 7.20 Generalized flowchart of SUBROUTINE TCLASS according to the material classifications given in Fig. 7.8.
Fig. 7.21 MAP(IX, IY) output of LPRP for input data given in Figs. 7.2, 7.3, and 7.4.
Fig. 7.22 Stages of boundary mapping of MAP(IX, IY). Black squares in each boundary indicate starting point.
Fig. 7.23 Final IBMAP(IX, IY).
In the following chapter real data will be used in essentially
the same software to demonstrate the applicability of this approach as
an operational technique for engineering analysis of aerial imagery.
This chapter will illustrate the application of logical terrain pattern recognition programming with real data, i.e., data from an existing geographic area as opposed to artificial or simulated data. The example given is from a geographic area selected to be similar to the simulated model of Chapter 7 and the engineering problem also retains the general nature of the previous chapter. Our purpose is to demonstrate that the approach of the previous chapter can be used in a practical problem of a similar nature. We will not be interested in defining all details of the problem organization, but rather our concern will be in indicating the changes required to implement the results of the previous chapter to real data. The chapter ends with a discussion of the example results.

8.1 Example Terrain Pattern Recognition Problem

This section presents a brief discussion of the real study area and its associated data, and the changes required to develop and implement the approach of Chapter 7 will be outlined. Our major result will be a coded grey tone image similar to MAP(IIX,IIY) in Fig. 7.21 showing the distribution of major landforms in the study area.

8.1.1 Study Area Description

The study area occupies about 7 sq. mi. along the Whitewater River in the southeastern corner of the State of Indiana, at the boundary between Franklin and Dearborn Counties and about 20 mi. from the Ohio
River. We will refer to this as the 'Whitewater' study area. Fenneman (1938) includes this area within the Till Plain Physiographic Section of the Central Lowland Province of the Interior Plains. It can be assumed that the major rivers of this area carried the runoff melt waters of the last glacial period and as such extensive deposits of glacio-fluvial materials can be expected in the alluvial valleys. The uplands of this area will be mantled with glacial till materials of Wisconsin and Illinoian ages; with the Wisconsin till overlying the Illinoian till at the higher elevations in the study area. Our interest focuses upon the alluvial valley of the Whitewater River containing locally floodplain and terrace landforms.

8.1.2 Study Area Data

Data for our pattern recognition problem is of two types: elevation data sampled from a topographic map sheet and tone data samples from an aerial photograph.

8.1.2.1 Elevation Data

Elevation data for the problem consists of a matrix IZ(IX, IY), IX=1, 64, IY=1, 64 obtained from the Cedar Grove Quadrangle, Indiana, 7.5 minute quadrangle, scale 1:24,000 contour interval 10 ft. the map sample spacing interval corresponds to about a 200 ft. ground sample spacing. Fig. 8.1 illustrates IZ(IX, IY) spatially expanded to 256x256 by duplicating each element of the 64x64 matrix in a local 4x4 matrix. The maximum and minimum elevations in IZ(IX, IY) are 1,000 and 540 ft., respectively, and the tones in Fig. 8.1 represent a linear scaling of each element of IZ(IX, IY) into 64 grey shades representative of the elevations in this range and the darkest tone representing the lowest elevation. Fig. 8.1 then represents a photograph of an image tube display having grey tones representative of the spatial organization of quantized terrain surface elevations in the study area.
Fig. 8.1 Photograph of DICOMED display with IZ(IX,IY) expanded from 64x64 to 256x256 and quantized to 64 shades of grey. Image is degraded due to poor response of the display and the photographic process used to record the display. The darkest tones represent the lowest elevations.
8.1.2.2 Tone Data

Tone data for the problem consists of a matrix IT1(IIX,IIY), IIX=1,256, IIY=1,256 obtained from sampling and quantizing a 70mm transparency of U.S. Department of Agriculture, photograph RM-1P-43, dated September 13, 1955, scale 1:20,000. Fig. 8.2 represents the 256x256 matrix of the resulting transmission values quantized in 64 grey shades and photographed from the above mentioned display tube. The elevation data sample points were matched to tone data matrices manually.

8.1.3 Development of Pattern Recognition and Mapping Strategies

Development of the pattern recognition and mapping strategy for the present task parallels that of Chapter 7. Fig. 8.3 represents a generalized flowchart of PROGRAM LPRPWR and its similarity with the generalized flowchart of PROGRAM LPRP and shown in Fig. 7.11 is readily apparent. Except for minor changes, the approach used in PROGRAM LPRPWR follows the outline in Table 7.1 which was used in the development and implementation of PROGRAM LPRP. These changes will now be discussed in the following section.

8.1.4 Implementation of Pattern Recognition and Mapping Strategy

To implement PROGRAM LPRP with large data matrices on a stored program computer required several program changes with the result being PROGRAM LPRPWR. The software was prepared in FORTRAN IV to operate on the CDC 6600 computer system having 247,000 word memory. No attempts were made to compact data into the sixty bit word length offered by this system, thus data storage occupied 69,632 words for IZ(IX, IY) and IT1(IIX, IIY) and an additional 65,536 words were required for MAP(IIX, IIY). Without a change in programming the temporary matrix IBMAP(IIX, IIY) could not be accommodated, thus MAP(IIX, IIY) is used to store intermediate results which IBMAP(IIX, IIY) stored in PROGRAM LPRP.
Fig. 8.2 Photograph of DICOMED display with \( IT_1(IX, IIY) \) as sampled in a 256x256 matrix and quantized to 64 shades of grey. Image is degraded due to poor response of the display and the photographic process used to record the display.
Fig. 8.3 indicates SUBROUTINE IMBRDR follows the data input in PROGRAM LPRPWR. This subroutine is required to frame the IT1(IIX,IIY) data with a discrete boundary where the frame of the 70mm photographic transparency was scanned in the optical sampler. For each column and row around the outer edges of IT1(IIX,IIY) containing a data value greater than or equal to '60', a '60' is used for each element of that column or row in MAP(IIX,IIY) and the interior elements are given the value '0'. This subroutine also provides the capability to place a predetermined rectangular aperture over any portion of MAP(IIX,IIY).

SUBROUTINE STRTWR is similar to that of SUBROUTINE STRTS in Fig. 7.12 when the SUBROUTINE TCLASS statements are included. Next SUBROUTINE LFCMWR accomplished the boundary mapping is similar to SUBROUTINE LFCMP in Fig. 7.13 except for the three called subroutines. SUBROUTINE LFCLWR accomplished the classification of elements of the mapping field of view MFLD(I,J) as did SUBROUTINE LFCLAS with landform and materials definitions appropriate for the 'White River' data, i.e., IZ(IX,IY) and IT1(IIX,IIY). Upon classification of the elements of MFLD(I,J) from IZ(IX,IY), SUBROUTINE RESLV utilizes data, mainly from IT1(IIX,IIY), to extrapolate boundary points. Next SUBROUTINE MOVWR translates MFLD(I,J) around the MLF boundary similar to SUBROUTINE MOVE in Fig. 7.15. A major modification was required in this subroutine to follow the boundary of the river, which is only one element in width at some locations. The resulting changes make the subroutines more general boundary in mapping capability.

SUBROUTINE SMSRWR is almost identical to SUBROUTINE SMSRCH in Fig. 7.17 and its result in the variable NM7 is still the stop state which controls the cycling of PROGRAM LPRPWR as in PROGRAM LPRP, i.e., when NM7=1, the data from MAP(IIX,IIY) is written on magnetic tape and the program terminates. The data tape is then read into the DICOMED image
Fig. 8.3 Generalized flowchart of PROGRAM LPRPWR.
display and photographed for permanent storage. Fig. 8.4 shows the results of the PROGRAM LPRPWR when the 'White River' data was used as input. In this illustration four grey shades code the mapped landforms.

8.2 Discussion of Example Results

We have mainly been interested in definition of an approach and the techniques required for our logical approach to terrain pattern recognition in Chapters 7 and 8. The examples and their respective results serve to demonstrate the nature of this approach, i.e., the manner in which the program is developed and implemented. In this respect the examples, especially the simulation problem which was discussed in detail, have more generality than the example products produced. This increased generality and the potential limitations of the given examples require additional clarification.

It should be understood that the example engineering problem and the input data have been carefully selected to illustrate the logical pattern recognition approach as simply as possible. Though this does not detract from the generality of the approach, it does indicate that the approach has not been severely tested. Thus the same engineering problem, when applied to other types of terrain data can be expected to yield results appropriate to the landform definitions and structure which are products of apriori knowledge of the terrain and the input data. There will be cases when the data does not allow preparation of the desired discriminant functions, especially from apriori knowledge, e.g., when the data does not have the required spatial of quantitization resolution, the spectral data is not optimum for discrimination, or the terrain features have transitional rather than discrete boundaries. Additionally we know that our solutions, and hence our programs, are different for each problem-terrain situation, but this will in general only involve changes in the definition of the start state, the MLF's
Fig. 8.4 Photograph of DICOMED display with MAP(IIX,IIY) indicating the results of PROGRAM LPRPWR. Here the darkest tones represent RIVER, while FLOOD PLAIN, TERRACE, and UPLAND, respectively, are indicated with lighter tones. The image is degraded due to the poor response of the display and the photographic process used to record the display.
and the stop state, while the remainder of the programming approach should require only minor modifications for different problem-terrain situations. This is then the generality of the approach and programming.

No attempt was made to optimize the programming for memory requirements or processing time. At the expense of increased operational time, data compression procedures could be employed to reduce memory requirements. Though the programming will not be shown here, PROGRAM LPRP was rewritten to allow mapping from segments of the data. Basically this involved operation to SUBROUTINE LFCMP in the normal manner, i.e., translating MFLD(I,J) around the boundary of the MLFs, but with the added provision of detecting boundaries of data segments in MAP (II,IX,IY) where II represents the submatrix segment of the total data and IX and IY represent the address within the submatrix. Realization of this capability required considerable additional programming, but the result, i.e., the processing output, was the same as Fig. 7.21 of Fig. 7.23. This would indicate that the programming of large real data matrices can be accomplished on smaller computers if the temporary storage is adequate.
CHAPTER IX
CONCLUSIONS AND RECOMMENDATIONS

This study has developed a logical approach towards pattern recognition for engineering purposes. The approach is modeled after that used by the engineering image analyst in obtaining terrain information for engineering purposes from stereoscopic aerial imagery. To facilitate the development and communication of concepts used by the engineer­ing analyst in matters related to terrain-related engineering problems and terrain organization, a formalism in the language of predicate calculus is employed. These concepts form the base for development of the logical approach to terrain pattern recognition and mapping which involves problem organization, terrain pattern description, strategy selection, and boundary mapping. A simulation terrain pattern recognition and mapping problem is discussed in detail to illustrate the logical approach on a digital computer system. To demonstrate the applicability of this approach to practical problems, real terrain elevation and photographic tone data was used with essentially the same program to produce a coded map indicative of that used by the engineer­
terrain analyst as a part of his solution to terrain-related engineer­ing problems.

9.1 Conclusions

9.1.1 Formalism of Concepts Used By Engineering Image Analysts

1. It has been demonstrated that the qualitative concepts used by the engineering image analyst to obtain information from aerial imagery for solution of his engineering problems can be formalized in the language of predicate calculus. This method of characterization requires that general concepts be defined to the level where important
parameters are explicitly related to data from terrain attributes. Such formalisms are of value for organization of the problem solving effort and provides the background for selection of parameters for the pattern recognition program development.

2. For terrain-related engineering problems, the formalism begins with general ideas and then, as implicit terrain and problem parameters are defined for terrain description and terrain partitioning, a level is reached where all necessary problem criteria are related to the terrain and the problem solutions are obtained from terrain data in the form of coded boundary matrices. Having obtained this level of formalism, several additional concepts are relatively simple to formalize. Coded matrices for multiple attributes can be combined by a recoding equivalence function to produce a composit matrix or map. Selected areas of this map can be sequentially partitioned to provide the required degree of refinement for the problem. The concept of terrain information can be related to the composit map matrices. Also an expected cost matrix can be determined and this can be of value in selecting the "best" of a set of competitive solutions based upon minimum cost criteria.

3. The occurrence and physical characteristics of terrain materials can be related formally to causative natural and geomorphic processes. This genetic link allows terrain partitioning based upon observable sets of attributes that form terrain descriptions useful in terrain recognition and classification. Three approaches to terrain classification, i.e., the pragmatic approach, the parametric approach, and the genetic approach, can be defined by the manner in which terrain descriptions are formulated.

9.1.2 Logical Pattern Recognition

1. The logical approach to terrain pattern recognition for en-
engineering purposes requires that each terrain-problem situation have a unique solution. The problem solving effort begins with an analysis of the problem statement which contains the problem goals, the geographic area of interest, and the problem constraints. The approach to problem organization and solution involves a development phase and an implementation phase. In the development phase, the target landforms which may serve as solutions to the goals are selected and the gross terrain pattern structure and terrain pattern attributes are defined. A sequential strategy is selected for locating and delineating the boundary of each target landform and then the pattern attributes and their relations are defined in terms of the data. The implementation phase involves the preparation of a modular software program and its testing and modification until accepted.

2. The simulated logical terrain pattern recognition and mapping problem illustrates in detail the development and implementation phases of the logical terrain pattern recognition and mapping approach. This model is an analytical representation of strategies which the terrain analyst might employ to recognize and delimit selected terrain features from stereoscopic aerial imagery. The operation of the program, in all or any of its various stages of landform boundary location and mapping, can be followed with the data provided and the software listing in the appendix.

3. Application of real data to essentially the same software which was developed for the simulation model indicated the generality of the software and the logical pattern recognition approach. If the problem-terrain situation is changed, it is expected that the programming changes in definition of the start state, the MLFs, and the stop state, while the remainder of the programming will require only minor modifications. No attempt was made to optimize the programming for
memory requirements or processing time, but it was determined that the programming could be modified to allow mapping from segments of the data.

9.2 Recommendations

9.2.1 Formalism of Concepts Used by Engineering Image Analysts

1. An additional study is suggested to extend the formalism of concepts of terrain-related problems and terrain organization to include the formalism of the pattern recognition model. The grammatical and syntactical inference techniques in pattern recognition (Evans, 1971; Fu and Swain, 1971) may then be employed to formalize the complete process of logical terrain pattern recognition.

2. Nilsson (1971) indicates applications of predicate calculus in problem solving which involve answer extraction by automated theorem proving and indicates that this process can be used to produce automatically some simple computer programs. These efforts are yet in a very primitive stage but are suggested as an approach to automated logical terrain pattern recognition which would include much of the problem organization tasks which are presently accomplished manually prior to software preparation.

3. A study is suggested which formalizes all aspects of qualitative terrain pattern recognition with the objective of presenting a high level teaching aid and introduction of engineering terrain analysis and pattern recognition techniques to other elements of the multi-disciplined pattern recognition community.

9.2.2 Logical Terrain Pattern Recognition

1. The logical terrain pattern recognition approach developed in this study should be applied to additional terrain-problem situations to demonstrate its generality and correct unforeseen limitations. One recognized limitation involves the collection of terrain elevation data
from topographic maps. For this study the elevations were manually
read from a 1:50,000 scale map, but the process is very time consuming.
Several photogrammetric techniques could be modified to rapidly obtain
coincident photographic tone and elevation data for the preparation of
on-line topographic maps while the tone and elevation data could be
used in a logical pattern recognition program to produce off-line
thematic maps. Another source of elevation data, termed the Digital
Topographic Data (DTD), is available through the Topographic Center,
Defense Mapping Agency, Washington, D.C. This data is available for
more than 75% of the United States and consists of digitally encoded
elevations in a matrix format which have been obtained by sampling and
interpolating 1:250,000 scale maps at 0.001 inch centers, i.e., sample
spacing of approximately 21 feet. The advantages of this data source
are its availability and extensive coverage, while it has limitations
of poor resolution associated with medium scale maps and the require­
ments for registering the elevation and tone data in the pattern recog­
nition process.

2. A logical extension of the present effort, which involved
stored program computer operation, would be the application of inter­
active computer techniques to the problem organization and programming
phases. When using large data matrices with stored program computers,
the program is usually given a low service priority which results in
long turn around periods. This becomes significant when one has im­
perfect apriori knowledge of terrain data for preparation of terrain
descriptions for discrimination and many correction and submission
cycles are required to achieve the desired pattern recognition results.
Interactive computer techniques, particularly involving interactive
graphics, should significantly increase the programming efficiency and
significantly reduce the iterations required to achieve desired results.
3. To achieve general acceptance in the remote sensing community for the approach to pattern recognition developed in this study, the programs must be capable of operating on minicomputers. Since minicomputers are available in a variety of configurations and capabilities, we can only indicate that data compaction techniques coupled with the division of data into manageable spatial segments is generally indicated. It is therefore recommended that a future study might consider the efficiency of programming the logical pattern recognition approach on minicomputers.
REFERENCES


211


APPENDIX

COMPUTER SOURCE LISTING

FOR PROGRAM LPRP
LPRP

01C :: LPRP--LOGICAL PATTERN RECOGNITION PROBLEM :::
02C ******************************************************************************************
03C *****PURPOSE:
04C  »**TO ILLUSTRATE LOGICAL PATTERN RECOGNITION PROGRAMMING
05C  *******USING SIMULATED DATA OF TERRAIN ELEVATIONS AND TWO
06C  *******PHOTOGRAPHIC SPECTRAL BANDS****************************************************************
10 DIMENSION IZ(32,32),IT2(32,32),IT3(32,32),MAP(32,32),IBMAP(32,32)
20 DIMENSION IC(5,5),MTLI(4,5),NUMB(32)
25 COMMON IA(3,3),IMFLD(3,3),MFLD(3,3)
30 LOGICAL IC,MTLI,DEF
40C
41C
44C
45C  ******OPEN DATA FILES*****
46C
49C ***TERRAIN ELEVATION MATRIX***
50 CALL OPENF(1,"ZXY")
59C ***SPECTRAL MATRIX T2***
60 CALL OPENF(2,"IT2XY")
69C ***SPECTRAL MATRIX T3***
70 CALL OPENF(3,"IT3XY")
79C ***CONTIGUOUS LANDFORM MATRIX (LOGICAL)***
80 CALL OPENF(4,"CONLF")
89C ***MATERIAL-LANDFORM IMPLICATION MATRIX (LOGICAL)***
90 CALL OPENF(5,"MTRLI")
99C
99C ***READ DATA FILES***
100 DO 100 I=1,32
120 READ(1,),(IZ(I,J),J=1,16)
130 READ(1,),(IZ(I,J),J=17,32)
140 READ(2,),(IT2(I,J),J=1,16)
150 READ(2,),(IT2(I,J),J=17,32)
160 READ(3,),(IT3(I,J),J=1,16)
170 READ(3,),(IT3(I,J),J=17,32)
180 100 CONTINUE
190 DO 120 I=1,5
200 READ(4,),(IC(I,J),J=1,5)
210 120 CONTINUE
220 DO 130 I=1,4
230 READ(5,),(MTLI(I,J),J=1,5)
240 130 CONTINUE
241C
242C ***ZERO MAP MATRIX--MAP(32,32)--AND TEMPORARY BOUNDARY
243C *****MAPPING MATRIX--IBMAP(32,32)***
250 DO 150 I=1,32
260 DO 150 J=1,32
270 MAP(I,J)=0
280 150 IBMAP(I,J)=0
281C
LPRP CONTINUED

282C *****VERIFY START STATE DEFINITION AND LOCATE START POINT*****
285 CALL STRTS(I2, IT3, MLTI, IC, MAP, IBMAP, IZMIN, IZMINX, IZMINY, MLF)
286 IX=IZMINX
287 IY=IZMINY
290C
294C, NB=0
299C *****MAP BOUNDARY OF MLF*****
300 50 CALL LFCMP(I2, IT3, MLTI, IC, MAP, IBMAP, IZMIN, IX, IY, MLF, NB)
302C
309C *****SEARCH AND CLASSIFICATION OF NEW BOUNDARY*****
310 CALL SGMRCH(I2, IT3, MLTI, IC, IBMAP, MAP, IX, IY, MLF, NM7)
312 IF(NM7.EQ.1)GOTO 185
314 NB=NB+1
315 GOTO 50
320C
360C *****OUTPUT PRINT ROUTINES*****
370 185 PRINT 190
380 190 FORMAT(I1, 15X, I0HMAP(32,32)/)
390 CALL PRINT3(MAP, NB)
400 PRINT 200
410 200 FORMAT(I1, 15X, 12HIBMAP(32,32)/)
420 CALL PRINT3(IBMAP, NB)
430 PRINT 210
440 210 FORMAT(I1, 15X, 9HIZ(IX, IY)/)
450 CALL PRINT3(IZ, NB)
460 PRINT 220
470 220 FORMAT(I1, 15X, 10HIT2(IX, IY)/)
480 CALL PRINT3(IT2, NB)
490 PRINT 230
500 230 FORMAT(I1, 15X, 10HIT3(IX, IY)/)
510 CALL PRINT3(IT3, NB)
520C
530 STOP
540 END

STRTS

1000 SUBROUTINE STRTS(I2, IT3, MLTI, IC, MAP, IBMAP, IZMIN,
1001 & IZMINX, IZMINY, MLF)
1005 DIMENSION IZ(32,32), IT2(32,32), IT3(32,32), MAP(32,32),
1006 & IC(5,5), MLTI(4,5), IBMAP(32,32)
1010 LOGICAL IC, MLTI
1015C
**ORIGINAL TEXT**

**SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:

**PURPOSE:

SUBROUTINE VALIDATES START STATE AND SELECTS AND CLASSIFIES INITIAL MAPPING POINT.

**APPROACH:
SUBROUTINE LFCMP(I2, I1, MT, IC, MAP, IBMAP! IZMIN, IX, IY, MLF, NB)

DIMENSION IZ(32,32), IT(32,32), IT2(32,32), IT3(32,32), MAP(32,32),
        IBMAP(32,32), IC(5,5), MT(4,5)

COMMON IA(3,3), IMF, MFLD(3,3), MFLD(3,3)

LOGICAL DEF, IC, MT

I40C ******** ****** ****** ******************************** *********
I40C *****PURPOSE: *******T0 MAP A BOUNDARY IN MAP(32,32) FOR A GIVEN LFC=MLF AND
I40C *******INDICATE ADJACENT MLF IN IBMAP(32,32)*****
I40C *****APPROACH:
I40C *******A 3BY3 MAPPING FIELD OF VIEW MFLD(3,3) IS MOVED ALONG
I40C *******A GIVEN BOUNDARY BEGINNING FROM A GIVEN POINT, EACH
I40C *******ELEMENT OF MFLD(3,3) IS CLASSIFIED ACCORDING TO LFC
I40C ******DEFINITIONS, AND THIS INFORMATION IS USED TO MOVE
I40C *******THE BOUNDARY IS TERMINATED WHEN THE FIRST
I40C *******POINT IS REOCCUPIED OR WHEN OTHER STOP CONDITIONS
I40C *******FOR LINES OR POINTS ARE REALIZED***********************
I40C ****************************
I40C
I40C ******INITIAL BOUNDARY COORDINATE (XX,KY) AND ITS FLAG KF***
I40C XX=IX
I40C KY=IY
I40C KF=0
I40C
I40C ******IS ENTRY FROM SUB STRTS OR SUB SMSRCH?*****
I40C IF(NB,GT,0)GOTO 2
I40C
I40C ******ZERO FIELD OF VIEW MATRIX UPON ENTRY FROM SUB STRTS****
I40C DO 1 I=1,3
I40C DO 1 J=1,3
I40C MFLD(I,J)=0
I40C MFLD(2,2)=MLF
I40C GOTO 10
I40C
I40C ******UPON ENTRY FROM SUB SMSRCH USE ITS CLASSIFIED DATA*****
I40C 2 DO 3 I=1,3
I40C DO 3 J=1,3
I40C II=IX+I-2
I40C IIY=IY+J-2
I40C IF(II<1.OR.II>32)GOTO 4
I40C IF(IY<1.OR.IY>32)GOTO 4
I40C MFLD(I,J)=IBMAP(II,IIY)
I40C GOTO 3
I40C 4 MFLD(I,J)=0
I40C 3 CONTINUE
I40C
I40C ******IS BOUNDARY COMPLETE?*****
I40C 10 IF(IX.EQ.XX.AND.IY.EQ.KY.AND.KF.EQ.1)GOTO 1000
LFCMP CONTINUED

1455 IBF=0
1456C
1459C *****IS (IX,IY) ON STUDY AREA BOUNDARY?*****
1460 IF(IX.EQ.1.OR.IX.EQ.32.OR.IY.EQ.1.OR.IY.EQ.32)GOTO 6
1461 GOTO 70
1464C *****PUT 8'S IN MFLD(,) ELEMENTS THAT OVERLAP STUDY AREA BOUND*
1465 6 IF(IX.EQ.1)GOTO 20
1466 IF(IY.EQ.32)GOTO 30
1467 GOTO 40
1470 20 DO 25 J=1,3
1471 25 MFLD(1,J)=8
1472 DO 40 J=1,3
1473 GOTO 40
1475 30 DO 35 J=1,3
1476 35 MFLD(3,J)=8
1477 DO 45 J=1,3
1478 45 IF(IY.EQ.1)GOTO 40
1479 IF(IY.EQ.32)GOTO 50
1480 GOTO 65
1483 50 IF(IX.EQ.1)GOTO 55
1484 IF(IY.EQ.32)GOTO 60
1485 GOTO 65
1488 55 DO 60 J=1,3
1489 60 MFLD(I,3)=8
1490C ***SET BOUNDARY FLAG IBF INDICATING BOUNDARY OVERLAP EXISTS***
1491 65 IBF=1
1492C
1501C *****FILL MFLD(,) WITH MAPPING BOUNDARY CODES*****
1502 70 DO 500 I=1,3
1503 DO 500 J=1,3
1504 IF(I.EQ.2.AND.J.EQ.2)GOTO 500
1505 K=MFLD(I,J)
1508 IF(K.NE.0.AND.K.NE.7.AND.K.NE.9)GOTO 500
1511 IIX=IX+I-2
1512 IIX=IX+1-2
1513 IIY=IY+J-2
1514 IMP=IBMAP(IIX,IIY)
1515 IF(IMP.EQ.0.OR.IMP.EQ.7.OR.IMP.EQ.9)GOTO 100
1516 GOTO 200
1522C *****DEFINITION OF MFL MUST BE VERIFIED OR DENIED BY SUB
1523C *****LFCLAS AT (IIIX,IIY)*************************************************************************
1525 80 CALL LFCLAS(I2,I3,MFL,IIX,IY,IIY,MFL,DEF)
1527C ***DEFINITION IS VERIFIED IF DEF IS TRUE***
1528 IF(DEF)GOTO 100
1532C ***DEFINITION IS DENIED IF DEF IS FALSE--SET MFLD(,)=7 AS CODE
1532 MFLD(I,J)=7
1533 IBMAP(IIX,IIY)=7
1534 GOTO 500
1535C ***WHEN DEFINITION IS TRUE FOR ELEMENT OF MFLD(,)***
1537 100 MFLD(I,J)=MFL
1538 IBMAP(IIX,IIY)=MFL
1539 GOTO 500
LFCMP CONTINUED

1540C ***USE PREVIOUS CLASSED DATA IF INFERENCE IS TRUE, OTHERWISE 
1541C *****SET MFLD(,)=9****
1542 200 IF (IC(MLF,IMP)) GOTO 205
1543 MFLD(I,J)=9
1544 IBMAP(IIX,IIY)=9
1545 GOTO 500
1546 205 MFLD(I,J)=IBMAP(IIX,IIY)
1547 500 CONTINUE
1548C
1550C *****THE FIELD OF VIEW MFLD(3,3) IS NOW FILLED AND MUST BE 
1551C *****MOVED TO A NEW CENTER ALONG THE BOUNDARY*****
1555 600 CALL MOVE(MFLD,IMFLD,MAP,IX,IY,IBF,IBSTOP,MLF)
1558 MAP(IX,IIY)=MFLD(2,2)
1559 IBMAP(IX,IIY)=MFLD(2,2)
1560 KF=1
1561 IF(IBSTOP.EQ.1) GOTO 1000
1569C *****REPEAT ALGORITHM FOR A NEW MFLD(,) LOCATION*****
1570 GOTO 10
1575C
1580 RETURN
1581 STOP
1582 END

MOVE

2000 SUBROUTINE MOVE(MFLD,IMFLD,MAP,IX,IY,IBF,IBSTOP,MLF)
2001 COMMON IA(3,3)
2002 DIMENSION MFLD(3,3),IMFLD(3,3),MAP(32,32)
2003C *****PURPOSE: TO MOVE THE BOUNDARY MAPPING FIELD OF VIEW 
2004C *****MFLD(3,3) TO THE NEXT POSITION AROUND THE BOUNDARY OF 
2005C *****MLF AND FILL MOVED MFLD(,) WITH MLF'S CLASSED AT 
2006C *****PREVIOUS POSITION. SEE TEXT FOR APPROACH**************
2007 DATA IA(1,1),IA(1,2),IA(1,3),IA(2,1),IA(2,2),IA(2,3),
2008& IA(3,1),IA(3,2),IA(3,3)/1,255,127,3,0,63,7,15,31/
2009 M=0
2010 IBSTOP=0
2011C *****DETERMINE MOVE INDICATOR NUMBER M ASSOCIATED WITH SUM OF 
2012C *****APPLICABLE WEIGHTS FOR MFLD(,) ELEMENTS**************
2013 DO 50 I=1,3
2014 DO 50 J=1,3
2015 IF(MFLD(1,J).EQ.8.OR.MFLD(1,J).EQ.MLF) GOTO 50
2016 M=M+IA(1,J)
2017 50 CONTINUE
MOVE CONTINUED

2018C *****DOES MFLD(3,3) OVERLAP STUDY AREA BOUNDARY?***************
2019C *****STMTS 51-52 MAP MOVE DIFFICULTIES; 56-57 DETERMINE MOVE
2020C *****DIRECTION; 60-130 MOVE MFLD(,) CENTER**********************
2021 IF(IBF.EQ.1) GOTO 55
2022 51 IXX=IX+1
2023 IYY=IY+1
2024 IF(M.EQ.183.OR.M.EQ.435) MAP(IX,IYY)=MFLD(2,3)
2025 52 IF(M.EQ.42) MAP(IXX,IYY)=MFLD(3,2)
2026 53 IF(M.EQ.12.OR.M.EQ.11.OR.M.EQ.137.OR.M.EQ.232.OR.M.EQ.266.OR...
2027 54 IF(M.EQ.393.OR.M.EQ.456.OR.M.EQ.487) GOTO 60
2028 IF(M.EQ.22.OR.M.EQ.25.OR.M.EQ.26.OR.M.EQ.149.OR.M.EQ.281.OR...
2029 55 IF(M.EQ.408.OR.M.EQ.470.OR.M.EQ.471) GOTO 70
2030 IF(M.EQ.243.OR.M.EQ.244.OR.M.EQ.246) GOTO 100
2031 IF(M.EQ.183.OR.M.EQ.190.OR.M.EQ.197.OR.M.EQ.221.OR.M.EQ.236.OR...
2032 56 IF(M.EQ.243.OR.M.EQ.436.OR.M.EQ.439) GOTO 80
2033 IF(M.EQ.94.OR.M.EQ.109.OR.M.EQ.116.OR.M.EQ.120) GOTO 90
2034 IF(M.EQ.183.OR.M.EQ.190.OR.M.EQ.197.OR.M.EQ.221.OR.M.EQ.236.OR...
2035 IF(M.EQ.436.OR.M.EQ.439) GOTO 110
2036 IF(M.EQ.382.OR.M.EQ.389.OR.M.EQ.435.OR.M.EQ.445.OR.M.EQ.476
2037 57 IF(M.EQ.498) GOTO 110
2038 IF(M.EQ.256.OR.M.EQ.383.OR.M.EQ.446.OR.M.EQ.477.OR.M.EQ.
2039 58 GOTO 120
2040 IF(M.EQ.4.OR.M.EQ.259.OR.M.EQ.368.OR.M.EQ.386.OR.M.EQ.480
2041 GOTO 130
2042 IF(M.EQ.247.OR.M.EQ.375.OR.M.EQ.499.OR.M.EQ.501.OR.M.EQ.
2043 59 GOTO 210
2044 IF(M.EQ.502) GOTO 210
2045 PRINT 59
2046 59 FORMAT(1H ,12XMOVE B ERROR)
2047 59 IXX=IX+1
2048 K=1
2049 L=0
2050 GOTO 1000
2051 70 IY=IY+1
2052 K=1
2053 L=1
2054 GOTO 1020
MOVE CONTINUED

2083 80 IY=IY+1
2085 K=0
2087 L=1
2089 GOTO 1000
2091 90 IX=IX-1
2093 IY=IY+1
2095 K=-1
2097 L=1
2099 GOTO 1000
2101 100 IX=IX-1
2103 K=-1
2105 L=0
2107 GOTO 1000
2109 110 IX=IX-1
2111 IY=IY-1
2113 K=-1
2115 L=-1
2117 GOTO 1000
2119 120 IY=IY-1
2121 K=0
2123 L=-1
2125 GOTO 1000
2127 130 IX=IX+1
2129 IY=IY-1
2131 K=1
2133 L=-1
2134 C
2135 C *****FILL MOVED MFLD(,) WITH MLF CLASSES DETERMINED FROM
2136 C *****PREVIOUS POSITION****************************
2140 1000 DO 1020 I=1,3
2142 DO 1020 J=1,3
2144 II=I+K
2146 JJ=J+L
2147 IF((II.LT.1.OR.II.GT.3.OR.JJ.LT.1.OR.JJ.GT.3))GOTO 1010
2148 IMFLD(I,J)=MFLD(II, JJ)
2149 GOTO 1020
2151 1010 IMFLD(I,J)=0
2153 1020 CONTINUE
2154 DO 150 I=1,3
2155 DO 150 J=1,3
2156 150 MFLD(I,J)=IMFLD(I,J)
2160 GOTO 300
2163 C *****WHEN M=0 ON BOUNDARY*****
2165 200 IF((IX.EQ.1.AND.IY.NE.1))GOTO 120
2170 IF((IX.NE.32.AND.IY.EQ.1))GOTO 60
2175 IF((IX.EQ.32.AND.IY.NE.32))GOTO 80
2180 IF((IX.NE.1.AND.IY.EQ.32))GOTO 100
2185 GOTO 120
2186 C ****STOP INDICATOR FOR CLASSIFIED POINTS AND LINES****
2187 210 IBSTOP=1
SMRCH

SUBROUTINE SMSRCH(IZ, IT2, IT3, MTLI, IC, IBMAP, MAP, IX, IY, MLF, NM7)
DIMENSION IZ(32,32), IT2(32,32), IT3(32,32), IBMAP(32,32), IC(5,5), MTLI(4,5), MAP(32,32)
LOGICAL IC, MTLI, DEF

C **************************************************************
C *****PURPOSE:
C
C ****** TO LOCATE A POINT FROM WHICH A NEW MLF BOUNDARY IS TO
C ****** BE MAPPED, TO IDENTIFY THE BOUNDARY MLF AT THIS POINT
C ****** AND AN ADJACENT POINT, TO SELECT A TENTATIVE INFERENCE
C ****** TO VERIFY OR DENY CONTIGUOUS MLF'S WITH SAMPLE DATA***
C ******APPROACH:
C ****** TO ACCOMPLISH A SCAN SEARCH FOR A 7 IN IBMAP(); WHEN
C ******FOUND DETERMINE ADJACENT MLF FOR TESTING INFERENCES WITH
C ******DATA APPLIED TO DEFINITIONS*******************************
C ******NO MORE 7'S FOUND--MAPPING COMPLETED*****
C
MLF=0
THE SEARCH FOR THE FIRST 7*****
5 DO 10 I=1,32
10 CONTINUE
I=I
J=J
IF(IBMAP(I,J).EQ.7)GOTO 15
10 CONTINUE

NO MORE 7'S FOUND--MAPPING COMPLETED*****
NM7=1
GOTO 1000

A 7 IS DETECTED*****
15 NM7=0

WHAT MLF IS ADJACENT TO THE 7?*****
DO 500 I=1,3
500 J=1,3
IF(I.EQ.2.AND.J.EQ.2)GOTO 500
2245 IIX=IX+1-2
2246 IIY=IY+J-2
2248 IF(IIX.LT.1.OR.IIX.GT.32.OR.IIY.LT.1.OR.IIY.GT.32)GOTO 500
2250 IF(IBMAP(IIX,IIY).EQ.0.OR.IBMAP(IIX,IIY).EQ.7)GOTO 500
2252 IF(IBMAP(IIX,IIY).EQ.9)GOTO 500
225C
2256C *****WHICH MLF CAN BE INFERRED TENTATIVELY?*****
2259 IJ=IBMAP(IIX,IIY)
2260 DO 300 K=2,5
2261 IF(IK(IJ,K))GOTO 16
2262 PRINT 6,IJ,K
2263 6 FORMAT(1H,3H1C<,1R,1H,,1R,,1H))
2265 GOTO 300
2266C
2267C *****VERIFY OR DENY TENTATIVE INFERENCE BY TESTING DEFINITION
2268C JANUARY SAMPLE DATA FOR MLF=K*****
2270 16 NNN=K-1
2275 IADJ=1
2279 IK=IJ-1
2280 IF(IJ.EQ.2)GOTO 20
2281 IF(IJ.EQ.3)GOTO 30
2282 IF(IJ.EQ.4)GOTO 40
2283 IF(IJ.EQ.5)GOTO 50
2285C **ILLEGAL MLF INDICATED--SET TO UNKNOWN**
2290 IBMAP(IIX,IIY)=9
2300 GOTO 500
2305C
2306C *****TRANSFER TO PROPER MLF RELATION FOR TESTING*****
2310 20 GOTO(22,24,26,28),NNN
2315 30 GOTO(32,34,36,38),NNN
2320 40 GOTO(42,44,46,48),NNN
2325 50 GOTO(52,54,56,58),NNN
2330C
2335C **RIVER TENTATIVELY INFERRED FROM CONTIGUOUS RIVER**
2340 22 IIX=IIX
2341 IIY=IIY
2342 GOTO 125
2345C *****FLOOD PLAIN TENTATIVELY INFERRED FROM CONTIGUOUS RIVER*****
2350 24 IF(IIZ(IIX,IIY)-IZ(IIX,IIY).GE.4.AND.IZ(IIX,IIY)
2351 & IZ(IIX,IIY).LE.7)GOTO 100
2355C **RIVER-FLOOD PLAIN RELATION DENIED**
2360 GOTO 360
2365C *****TERRACE TENTATIVELY INFERRED FROM CONTIGUOUS RIVER*****
2370 26 IF(IIZ(IIX,IIY)-IZ(IIX,IIY).GE.14.AND.IZ(IIX,IIY)
2371 & IZ(IIX,IIY).LE.18)GOTO 100
2375C **RIVER-TERRACE RELATION DENIED**
2380 GOTO 360
2385C *****UPLAND TENTATIVELY INFERRED FROM CONTIGUOUS RIVER*****
2390 28 IF(IIZ(IIX,IIY)-IZ(IIX,IIY).GE.20.AND.IZ(IIX,IIY)
2391 & IZ(IIX,IIY).LE.31)GOTO 160
2395C **RIVER-UPLAND RELATION DENIED**
2400 GOTO 300
2405C ****RIVER TENTATIVELY INFERRED FROM CONTIGUOUS FLOOD PLAIN****
2410 32 IF(IZ(IIX,IIY) - IZ(IX,IIY) .GE. 4 .AND. IZ(IIX,IIY) -
2411 & IZ(IX,IIY) .LE. 7 ) GOTO 100
2415C **FLOOD PLAIN-RIVER RELATION DENIED**
2420 GOTO 300
2425C ****FLOOD PLAIN TENTATIVELY INFERRED FROM CONTIGUOUS FLOOD PLAIN****
2430 34 IIIX = IIX
2431 111Y = I1Y
2432 GOTO 135
2435C ****TERRACE TENTATIVELY INFERRED FROM CONTIGUOUS FLOOD PLAIN****
2440 36 IF(IZ(IX,IIY) - IZ(IIX,IIY) .GE. 10 .AND. IZ(IX,IIY) -
2441 & IZ(IIX,IIY) .LE. 14 ) GOTO 100
2445C **FLOOD PLAIN-TERRACE RELATION DENIED**
2450 GOTO 300
2455C *+**RIVER TENTATIVELY INFERRED FROM CONTIGUOUS TERRACE****
2460 42 IF(IZ(IIX,IIY) - IZ(IX,IIY) .GE. 14 .AND. IZ(IIX,IIY) -
2461 & IZ(IX,IIY) .LE. 14 ) GOTO 100
2465C **TERRACE-RIVER RELATION DENIED**
2470 GOTO 300
2475C **♦* FLOOD PLAIN TENTATIVELY INFERRED FROM CONTIGUOUS TERRACE***
2480 44 IF(IZ(IX,IIY) - IZ(IIX,IIY) .GE. 10 # AND * IZ(IIX,IIY) -
2481 & IZ(IX,IIY) .LE. 14 ) GOTO 100
2485C **TERRACE-FLOOD PLAIN RELATION DENIED**
2490 GOTO 300
2495C ****TERRACE TENTATIVELY INFERRED FROM CONTIGUOUS TERRACE****
2500 46 IIIX = IIX
2501 111Y = I1Y
2502 GOTO 135
2505C ****UPLAND TENTATIVELY INFERRED FROM CONTIGUOUS TERRACE****
2510 48 IF(IZ(IIX,IIY) - IZ(IX,IIY) .GE. 2 .AND. IZ(IIX,IIY) -
2511 & IZ(IX,IIY) .LE. 11 ) GOTO 100
2515C **TERRACE-UPLAND RELATION DENIED**
2520 GOTO 300
2525C ****RIVER TENTATIVELY INFERRED FROM CONTIGUOUS UPLAND****
2530 52 IF(IZ(IIX,IIY) - IZ(IX,IIY) .GE. 20 .AND. IZ(IIX,IIY) -
2531 & IZ(IX,IIY) .LE. 31 ) GOTO 100
2535C **UPLAND-RIVER RELATION DENIED**
2540 GOTO. 300
2545C ****TERRACE TENTATIVELY INFERRED FROM CONTIGUOUS UPLAND****
2550 56 IF(IZ(IIX,IIY) - IZ(IX,IIY) .GE. 2 .AND. IZ(IIX,IIY) -
2551 & IZ(IX,IIY) .LE. 11 ) GOTO 100
2555C **UPLAND-TERRACE RELATION DENIED**
2560 GOTO 300
2565C ****UPLAND TENTATIVELY INFERRED FROM CONTIGUOUS UPLAND****
2570 58 IIIX = IIX
2571 111Y = I1Y
2572 GOTO 140
2575C
**SMSRCH CONTINUED**

2580C *****VERIFICATION OF MLF DEFINITIONS*****

2585C
2590 DO 205 L=1,3
2591 DO 205 M=1,3
2595 IADJ=0
2600 IF(L.EQ.2.AND.M.EQ.2)GOTO 205
2625 IF(L.EQ.1.OR.M.EQ.1)GOTO 205
2650 IF(L.EQ.1.AND.M.EQ.1)GOTO 205
2670 CONTINUE

2680C ****#TO TEST MLF CLASS AND ASSOCIATED MCL FOR INFERENCE****

2690C
2695 CALL LFCLASSIZE,IT2,IT3,MTL1,IIIX,IIIX,IIIIY,IIIIY,K,DEF)
2700C ***INFERENCE IS VERIFIED IF DEF IS TRUE***

2705 IF(DEF)GOTO 510

2710C ***INFERENCE IS DENIED IF DEF IS FALSE***

2715 CONTINUE

2720 500 CONTINUE

2721C

2722C *****ERROR MESSAGE*****

2725 PRINT 476

2730 476 FORMAT(1H0,15HNO MLF ADJ TO 7)

2735 GOTO 1000

2740C ***REPLACE 7 BY CLASSIFIED MLF***

2745 512 MLF=K

2750 IBMAP(IIX,IIY)=MLF

2755 MAP(IIX,IIY)=MLF

2760C

2765 1000 RETURN

2770 STOP

2775 END
SUBROUTINE LFCLAS, IZ, IT2, IT3, MTLI, IX, IY, IIX, IIIY, MLF, DEF

DIMENSION IZ(32,32), IT2(32,32), IT3(32,32), MTLI(4,5)

LOGICAL DEF, MTLI

* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

******PURPOSE:

******TO VERIFY OR REFUTE DEFINITION OF MLF BEING BOUNDARY

******MAPPED IN SUB LFCLMP******

******APPROACH:

******CHECK DEFINITION OF MLF AND IMPLIED MCL; IF VERIFIED

******BY DATA, SET DEF TO TRUE, OTHERWISE SET DEF TO FALSE**

******Branch to applicable MLF definition*****

DEF=.FALSE.

******Branch to applicable MLF definition*****

GOTO(100,200,300,400,500),MLF

******Start state definition indicated*****

110 PRINT 110, MLF, IX, IY

110 FORMAT(1H MLF = , I2, 16H IN ERROR AT X = , 9H AND Y = , I2)

115 GOTO 1000

******River definition indicated*****

200 IF(IZ(IIX, IYY) .GE. 12) GOTO 250

******Sub TCLASS determines Material Classification MCL

240 CALL TCLASS(IT2, IT3, IIX, IYY)

******Associated with IT2(IIX, IYY) and IT3(IIX, IYY)*****

250 IF(CMTLI(MCL, MLF)) DEF=.TRUE.

******Flood Plain definition indicated*****

300 IF(IZ(IIX, IYY) .LT. 6 OR. IZ(IIX, IYY) .GT. 15) GOTO 250

******Check Material Implication Matrix for MLF inferred by MCL

500 IF(MTLI(MCL, MLF)) DEF=.TRUE.

500 IF(MTLI(MCL, MLF)) DEF=.TRUE.

GOTO 250

******Terrace definition indicated*****

400 IF(IZ(IIX, IYY) .LT. 21 OR. IZ(IIX, IYY) .GT. 28) GOTO 250

******Upland definition indicated*****

500 IF(IZ(IIX, IYY) .LT. 25 OR. IZ(IIX, IYY) .GT. 60) GOTO 250

800 GOTO 250

900 GOTO 250

800 GOTO 250

900 GOTO 250

500 IF(IZ(IIX, IYY) .LT. 25 OR. IZ(IIX, IYY) .GT. 60) GOTO 250

250 RETURN

1000 STOP

END
SUBROUTINE TCLASS(IT2, IT3, IX, IY, MCL)
DIMENSION IT2(32,32), IT3(32,32)

C *********************************************************
C *****PURPOSE:
C *****TO DETERMINE MATERIAL CLASSIFICATION MCL ASSOCIATED
C *****WITH TWO-DIMENSIONAL SPECTRAL VECTOR DATA IT2(IX, IY)
C *****AND IT3(IX, IY)
C *****APPROACH:
C *****LINEAR DISCRIMINANT FUNCTIONS PARTITION THE TWO-
C *****DIMENSIONAL CLASSIFICATION SPACE INTO FOUR REGIONS
C *****EACH REPRESENTING A SEPERATE MCL. A SPECTRAL VECTOR
C *****DETERMINED FROM IT2(IX, IY) AND IT3(IX, IY) SAMPLE DATA
C *****IS PLOTTED IN CLASSIFICATION SPACE WITH MCL RESULTING
C
C *********************************************************

IF(10-IT2(IX, IY)-IT3(IX, IY)).GT.0 GOTO 10
IF(IT2(IX, IY)+IT3(IX, IY)-10.GT.0.AND.25-IT2(IX, IY)
.AND.5.GT.0) GOTO 20
IF(IT2(IX, IY)+IT3(IX, IY)-25.GT.0.AND.IT3(IX, IY)
+5.GT.0) GOTO 30
IF(IT2(IX, IY)+IT3(IX, IY)-25.GT.0.AND.IT2(IX, IY)
+5.GT.0) GOTO 40

MCL=0 IS AN ERROR CLASS*****
MCL=1
MCL=2
MCL=3
MCL=4
RETURN
STOP
END
10 SUBROUTINE PRINT3(DATAMP, NUMB)
20 DIMENSION DATAMP(32,32), NUMB(32)
25 INTEGER DATAMP
30 DATA NUMB(1), NUMB(2), NUMB(3), NUMB(4), NUMB(5), NUMB(6), NUMB(7),
31 NUMB(8), NUMB(9), NUMB(10), NUMB(11), NUMB(12), NUMB(13), NUMB(14),
32 NUMB(15), NUMB(16), NUMB(17), NUMB(18), NUMB(19), NUMB(20), NUMB(21),
33 NUMB(22), NUMB(23), NUMB(24), NUMB(25), NUMB(26), NUMB(27), NUMB(28),
34 NUMB(29), NUMB(30), NUMB(31), NUMB(32)/01,02,03,04,05,06,07,08,09,
35 10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,
36 31,32/
40 PRINT 10,(NUMB(N),N=1,16)
50 10 FORMAT(1H1,15X,16I3)
60 PRINT 20
70 20 FORMAT(1H,14X,40H******************************)
71 & 10H**************************)
80 DO 30 I=1,32
90 K=33-I
100 30 PRINT 40,NUMB(K),(DATAMP(K,J),J=1,16)
110 40 FORMA(T1H,11X,13,1H*,16I3)
120 PRINT 20
130 PRINT 10,(NUMB(N),N=1,16)
140 PRINT 45
150 45 FORMAT(1H,1H*/")
160 PRINT 60,(NUMB(N),N=17,32)
170 PRINT 21
180 DO 50 I=1,32
190 K=33-I
200 50 PRINT 55,(DATAMP(K,J),J=17,32),NUMB(K)
210 55 FORMAT(1H,11X,16I3,2H*,12)
220 PRINT 21
225 21 FORMAT(1H,11X,40H******************************)
226 & 10H**************************)
230 PRINT 60,(NUMB(N),N=17,32)
240 60 FORMAT(1H,11X,16I3)
250 PRINT 45
260 RETURN
270 STOP
280 END