User Interface Design for Two-Dimensional Polygonally Encoded Geological Survey Maps

Joann M. Ammann
Robert B. McGhee
Michael J. Zyda

July 1986

Approved for public release; distribution unlimited

Prepared for:
Chief of Naval Research
Arlington, VA 22217
The work reported herein was supported in part by the Foundation Research Program of the Naval Postgraduate School with funds provided by the Chief of Naval Research.

Reproduction of all or part of this report is authorized.

This report was prepared by:

MICHAEL J. ZYDA
Assistant Professor
Computer Science Department

Reviewed by:

KNEALE T. MARSHALL
Dean of Information and Policy Science

Released by:
User Interface Design for Two-Dimensional Polygonally Encoded Geological Survey Maps

Joann M. Ammann
Robert B. McGhee
Michael J. Zyda

Naval Postgraduate School
Monterey, CA 93943-5100

Chief of Naval Research
Arlington, VA 22217

Approved for public release; distribution unlimited

cartography, user interface design

We present in this study an overview of a cartographic processing pipeline for the generation and maintenance of polygonally encoded data bases from published U.S. Geological Survey maps. The focus of this research centers on the development of an interactive editing system. The editor, serving as the final step in the overall project, provides the user with the capability to correct and modify dated topographic characteristics. A variety of processing and digitizer induced errors introduced into the data base from previous utility...
steps can also be corrected. Included is a discussion on the internal indexing scheme used for managing revisions and the techniques and algorithms for updating the data bases.
User Interface Design for Two-Dimensional Polygonally Encoded Geological Survey Maps †

Joann M. Ammann, Robert B. McGhee and Michael J. Zyda *

Naval Postgraduate School,
Code 52, Dept. of Computer Science,
Monterey, California 93943

ABSTRACT

We present in this study an overview of a cartographic processing pipeline for the generation and maintenance of polygonally encoded data bases from published U.S. Geological Survey maps. The focus of this research centers on the development of an interactive editing system. The editor, serving as the final step in the overall project, provides the user with the capability to correct and modify dated topographic characteristics. A variety of processing and digitizer induced errors introduced into the data base from previous utility steps can also be corrected. Included is a discussion on the internal indexing scheme used for managing revisions and the techniques and algorithms for updating the data bases.

Categories and Subject Descriptors: 1.3.6 [Methodology and Techniques]: human interface design, interactive graphics dialogues, user interface; 1.3.m [Miscellaneous]: cartography.

General Terms: Techniques;

Additional Key Words and Phrases: cartography, user interface design;

† This work has been supported by the Naval Postgraduate School Foundation Research Program and a grant from the Naval Ocean Systems Center, San Diego (Ref. # N0001486WR4B123AC).
* Contact author.
# TABLE OF CONTENTS

I. INTRODUCTION ........................................................................................................ 7

A. DATA ENTRY ............................................................................................................. 7

B. MOTIVATION ............................................................................................................ 9

C. SCOPE OF THE STUDY .......................................................................................... 10

D. ORGANIZATION ..................................................................................................... 10

II. A CARTOGRAPHIC DATA PROCESSING SYSTEM ................................................. 12

A. OVERVIEW ............................................................................................................. 12

B. PRE-PRINTED MAP SELECTION .......................................................................... 12

C. DIGITIZATION ....................................................................................................... 14

D. IMAGE PROCESSING .............................................................................................. 15

1. General ................................................................................................................ 15

2. The Dichl Feature Extraction System .................................................................. 16

E. DATA COMPRESSION ............................................................................................. 18

F. THE EDITOR ........................................................................................................... 19

III. RESEARCH FACILITIES ....................................................................................... 20

A. IRIS WORKSTATION ............................................................................................. 20

B. IRIS GRAPHICS LIBRARY ..................................................................................... 20

IV. THE EDITING SYSTEM .......................................................................................... 23
A. GETTING STARTED ............................................................... 23
B. SCREEN LAYOUT ............................................................... 23
C. INPUT DEVICES ............................................................... 25
D. MENU STRUCTURE .............................................................. 25
   1. Area Edit ................................................................. 27
   2. Point Edit ............................................................... 30
   3. Change View ............................................................. 30
   4. Read in a File ............................................................. 31
   5. Write to a File ............................................................ 31
   6. Exit System ............................................................... 32
E. SYSTEM PROMPTS ........................................................... 32
F. FEATURES SUPPORTED .................................................. 33
G. FILE FORMAT ............................................................... 34
V. MANAGING USER REVISIONS ......................................... 36
   A. INTERNAL INDEXING SCHEME ....................................... 36
   B. UPDATING THE DATABASE .......................................... 37
      1. Update Database .................................................... 39
      2. Data Compression .................................................. 39
      3. Expand Area .......................................................... 42
      4. Subset Area .......................................................... 42
I. INTRODUCTION

Since the computer's inception, the cartographic industry has profited from repeated hardware and software advances. Computer-aided mapping systems effectively reduce labor costs associated with the production and maintenance of cartographic documents. Orbiting satellites widen land coverage to encompass the entire globe. Significantly improved remote-sensing systems extend the cartographer's field of sight beyond the visible spectrum. Yet despite such progress, the process of cartographic capture and encoding remains a continuing concern.

A. DATA ENTRY

Current mapping systems lack the capability to automatically enter and classify data from the source. Most employ electromechanical digitizers with which human operators manually trace lineations and boundaries, recording (X,Y) coordinates at defined intervals or pivot points. Collection of information in this format is commonly referred to as vector mode storage. Numeric keys, located on hand-held devices, assist feature labeling with specified category codes. The above procedure is often a time-consuming, fatiguing and error prone operation.

Alternatively, manufacturers offer automatic digitizing facilities promoting raster mode storage. In raster mode, images are divided into scan lines and
information is stored in a sequential fashion by horizontal scan line pattern arranged from the top of a picture downwards. If retention of additional area characteristics is desired, manual intervention, tiring by nature, is necessary to separate and identify individual components.

The market also supports automatic line-following digitizers capable of sensing and tracking linear features with servomechanisms. Aimed at alleviating the attentive and tedious handling common to other methods, these systems have limited use and require high quality maps as input. Uncertainties arising at intersecting features are resolved through the assistance of an operator.

The solution to automatic data input and feature identification from paper maps presents a most difficult challenge. The production of a fully automated system may possibly never be attained. Present day manufacturers have already acceptably confined their systems' capabilities to only certain sub-tasks. Two major reasons account for the limited progress seen in the area of data capture:

1. the market is characteristically small and undercapitalized, offering little growth potential

2. hardware facilities have lacked sufficient computing power and display resolution capabilities to handle the demanding load of processing and manipulating cartographic data.
B. MOTIVATION

Many government mapping agencies and commercial map companies have realized the potential benefits of maintaining digital cartographic data bases. The U. S. Geological Survey, for example, is in the process of converting all of its United States quadrangle series maps into digital form. Topographic data is being accumulated into separate categories such as digital elevation, surface hydrography, public land-survey network, geographic names and other classes. Specialized projects, requiring digitized map files, are newly emerging. For instance, work conducted by Richbourg, Rowe and Zyda [Ref. 1] on solution techniques for two-dimensional route planning problems for mobile autonomous vehicles identifies a need for topographic information grouped by terrain speed regions.

The inability of current data bases and mapping systems to fulfill the latter example's requirements served as a catalytic force in reopening the study for improving computer-assisted cartographic entry and encoding techniques. A discussion of the processing system under development is presented in Chapter Two. The major goal of the research is to offer an alternative approach from the popular interactive digitizing methods used in handling data entry from published maps. Although it is difficult to design an accurate, reliable and low cost scheme, the proposed approach seems promising.
SCOPE OF THE STUDY

This study develops and examines an interactive editing system for polygonally encoded data bases generated from the cartographic processing system described in [Ref. 2]. The editor, serving as the final step in the overall project, provides the user with the capability to update and modify topographic information. A variety of common processing and digitizer induced errors, including misencoded coordinates, line noise and superfluous points can also be corrected. The editor's algorithms were implemented using the C programming language with calls made to available Graphics Library routines. A Silicon Graphics, Inc. IRIS (Integrated Raster Imaging System) Turbo 2400 workstation was chosen for this study as it supports the high resolution and fast transmission rates needed for manipulating cartographic data.

ORGANIZATION

Chapter Two reviews the complete cartographic data processing pipeline under development at the Naval Postgraduate School. Each step within the system package is briefly discussed. Chapter Three reports on the available hardware and software research facilities utilized. Chapter Four introduces a prototype editing system, the topic of this study. Chapter Five addresses the strategies employed for managing user revisions. The algorithms used for converting an updated map image from the screen into stored data are also covered. Finally Chapter Six is an examination of the presented interface's
limitations and weaknesses with suggestions for future enhancements and improvements. Attached as appendices are before and after snapshots taken from sample editing sessions, a list of USGS map symbols supported by the editor and the algorithms utilized for updating a cartographic data base.
II. A CARTOGRAPHIC DATA PROCESSING SYSTEM

A. OVERVIEW

The integrated cartographic processing package, developed at the Naval Postgraduate School, is organized as a set of five stages, each performing a distinct function. A collection of methods and techniques, well-known in some cases and novel in others, assist in making this prototypical model operational. The component steps are shown schematically in Figure 2.1 in the order they are generally executed. Two of the stages (items 1 and 2) deal primarily with the selection and preparation of maps for use in subsequent steps; two (items 3 and 4) handle actual digital information processing and the creation of a computerized database, and the remaining one (item 5) conducts revision and editing of files stored on disk. Steps three and four are the subject of research conducted by Ref. 2, while step five is the focus of this study. Each stage of the project is briefly described below.

B. PREPRINTED MAP SELECTION

Prior to implementing any cartographic information processing system, selection of a standardized mapping scheme is necessary. For this research, primary input consisted of sectional pieces from U. S. Geological Survey (USGS) Quin-Qua (7.5-minute x 7.5-minute) series maps with conventional units at a
Figure 2.1 Processing Pipeline for Cartographic Information Capture and Revision
1:24,000 scale. Widely available and accepted, each USGS map complies with established specifications in regards to size, scale, symbolism and content [Refs. 3 and 4]. The use of color assists in discriminating the various cartographic features (see Table 2.1 below).

<table>
<thead>
<tr>
<th>Color</th>
<th>Application Area</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLACK</td>
<td>Cultural features such as roads and buildings.</td>
</tr>
<tr>
<td>BLUE</td>
<td>Hydrographic features such as lakes, rivers, wetlands and reservoirs.</td>
</tr>
<tr>
<td>BROWN</td>
<td>Hypsographic features shown by contour lines such as slopes and elevations.</td>
</tr>
<tr>
<td>GREEN</td>
<td>Surface cover including woodlands, scrub, orchards and vineyards.</td>
</tr>
<tr>
<td>PURPLE</td>
<td>Features added from aerial photographs during map revision. The changes are not field checked.</td>
</tr>
<tr>
<td>RED</td>
<td>Important roads and public land survey system.</td>
</tr>
<tr>
<td>WHITE</td>
<td>Nonvegetative features such as barren waste areas and piles.</td>
</tr>
</tbody>
</table>

**DIGITIZATION**

The raw pictorial representation of the paper map is converted into computer-readable form for subsequent processing through the use of a video digitizer camera. Digitization comprises the scanning of a document to resolve print, line or character information into small picture elements, or pixels. The optical
response at each pixel controls detection circuitry for the generation of positionally defined digital output.

An EYECOM\textsuperscript{1} Picture Digitizer and Display unit interfaced to a Digital Equipment Corporation (DEC) PDP-11 computer was utilized for this project. Additional auxiliary lighting and filters (red, green and blue) complement the setup. Files produced after digitization are downloaded onto a DEC V.X 11/780 computer. The File Transfer Protocol (FTP) program on the connecting ETHERNET allows file movement to the final destination host, a Silicon Graphics, Inc. IRIS Turbo 2400 workstation.

D. IMAGE PROCESSING

1. General

After transforming a map’s contents into digital form, ideally one applies an image processing system which promotes automatic component recognition. A component is any feature that is judged to be significant at a chosen scale and that exists within an area. As discussed above, present day systems fall short in reaching this \textit{optimum} capability. Most if not all commercially available systems require varying degrees of human intervention to demarcate cartographic information. Although the market supports some automatic line-following systems, these tools are useful only for certain types of features and only on maps with high quality line work. In general, human assistance is needed to bridge

\begin{footnotesize}
\begin{itemize}
\item 1 A trademark of Spatial Data Systems, Inc.
\end{itemize}
\end{footnotesize}
interrupts, to resolve uncertainties at intersections marked by unreliable
directional trends and to affix identifying codes for individual features.

A technique proposed by Diehl [Ref. 2] attempts to resolve this noted
deficiency. Although in a prototypical state, his method offers an alternative
avenue for further research exploration in solving this most difficult problem. An
overview of his research follows.

2 The Diehl Feature Extraction System

From a black box viewpoint (see Figure 2.2), Diehl's system analyzes
individual pixels of an input video image and through heuristic reasoning,
categorizes each pixel into a subset of colors synonymous to those comprising the
input paper map. In the digitization process, each pixel has the potential of
adopting one of over sixteen million color identities under the RGB color system
of 24 bits per pixel, with 8 bits each for red, blue and green intensity values.
Determination of actual pixel color assignment is derived by sample readings
taken over small contiguous map areas. This averaging procedure accounts for
small variations observed in pixels of seemingly uniform regions. Points lying in
the proximity to boundaries separating dissimilar colors often assume fuzzy values
while interior ones approximate original map colors to within a measurable
deviation. Preliminary tests show Diehl's work to be highly accurate in correctly
assigning pixels of internal feature sections with small error.

1 A color in the spectrum can be formed by blending the additive primitives (red, green and
Possible Pixel Values in a Digitized Image Form

Image Processing and Feature Recognition

Homogeneous Area Groupings

Figure 2.2  Conversion from video image to homogeneous area regions [Ref 2]
rate-occurring near obscure border lines. The resulting data base generated from this processing scheme can either be retained as is or be funneled through the Data Compression Stage to optimize storage space.

E. DATA COMPRESSION

Once the map has been segmented into one of seven equivalence classes, an appropriate storage format (raster versus vector) must be chosen. Several factors were considered during the selection process:

- Speed tolerance for terminal redisplay and screen refresh,
- Degree of resolution needed to maintain accuracy,
- Retrieval rate for information of interest,
- Ease in data manipulation during editing sessions,
- Storage resource requirements,
- Quantity of information to be retained per unit of measure, and
- Applicability of format to problem area.

Since our interest centers on the generation of data bases for use in autonomous vehicle route planning, vector storage proved to be most economical.

A triangular tessellation, formulated by Diehl [Ref. 2], serves as the fundamental data storage mechanism. In general, the process consists of two repeatedly called parts.
identification of the next homogeneous area, and
division of the identified region into polygons.

A detailed discussion of the algorithms involved is presented in Chapter Five. These algorithms were slightly modified and incorporated into the editing system's facility for updating a cartographic data base.

F. THE EDITOR

Finally, an editing system is introduced to ensure proper maintenance of the data base generated from the previous utility steps. Typically most printed maps on the market are several years old and are rarely updated. A computer-assisted map data base editor allows easy modification and correction of dated topographic characteristics. The digital map data can be altered either to eliminate or remedy errors and blunders or to revise established files when new updates arrive. Often proved to be time consuming for the user, this is an essential step in the overall process of assuring a map's usefulness and accuracy. Possible adverse effects could result based on outdated or erroneous information portrayed by an unkept data base. The contents of the remaining chapters focus on the editing system's design.
III. RESEARCH FACILITIES

A. IRIS WORKSTATION

The IRIS Turbo 2400 Graphics Workstation, manufactured by Silicon Graphics, Inc., served as the primary tool for this research. By incorporating custom-built VLSI circuits into its design, the IRIS offers an attractive alternative to the more conventional workstations [Ref. 5]. The special-purpose hardware, designed to replace less efficient software, yields high processing speeds and increased performance reliability necessary for manipulating cartographic data. The system notably combines real-time color graphics with Unix operating system software and Ethernet network communications. A very high resolution color monitor provides crisp displays and fine delineations for even the most complex work. Figure 3.1 outlines the distinctive features of the system.

B. IRIS GRAPHICS LIBRARY

The IRIS Graphics Library is an extensive collection of utility and graphics subroutines providing high level access to the hardware enabling graphical objects to be easily manipulated as geometrical objects (points, lines, polygons, etc.) rather than pixels. A series of coordinate systems and mapping instructions provides the user with the capability to define such objects in world space.
A subset of the Library's routines were used for this research. They can be generalized into the following nine categories:

- **Global state** commands initialize the hardware and control global state variables.

- **Primitive drawing** commands draw points, lines, polygons, circles, arcs, and text strings on the screen.

- **Drawing attribute** commands select characteristics for drawing lines, filling polygons, and writing text strings.

- **Coordinate transformation** commands perform manipulations on coordinate systems, including mapping user-defined coordinate systems to screen coordinate systems.

- **Display mode and color** commands determine how the bitplane image memory is used and how objects are colored on the screen.

- **Input/output** commands initialize and read input/output devices.

- **Object creation and editing** commands provide the means to create hierarchical structure of graphics commands.

- **Picking and selecting** commands identify the commands that draw to a specified area of the screen.

- **Textport** commands allocate an area of the screen for writing text. 

A detailed account on command usage for each of the above classes is covered in the IRIS User's Guide.

---

Ethernet connection between IRIS's and VAX's

- 32-bit Motorola 68020 Processor
- 1024 X 768 X 32-bit Display Memory
- 2 MB of CPU Memory
- 144 MB of Disk Storage
- Floating Point Accelerator
- Geometry Pipeline with Geometry Engines and Geometry Accelerators
- 40 Hz Non-interlaced Display
- 16-bit Z-buffer for Hidden Surface Elimination
- Hardware Smooth Shading
- Unix System V
- IRIS Graphics Library
- Cartridge Tape Unit
- Ethernet to VAX's
- Digitizer Tablet

Features of the IRIS Turbo 2400 Graphics Workstation

Figure 3.1
IV. THE EDITING SYSTEM

A. GETTING STARTED

The editing system is invoked by a single command to the IRIS workstation:

% MAPEDIT

Upon execution, an initial title page is displayed for a few seconds prior to the appearance of the main program. Prompts and instructions guide the user through the entire operation of the editor.

B. SCREEN LAYOUT

The screen is partitioned into five non-overlapping windows (see Figure 4.1). Each is reserved for a particular use. Window One serves as the primary workbench for area viewing and editing of a sectional map. Window Two exhibits a dynamic picture display of the current map image in miniature form. Window Three contains any menu options available during a particular point in time. The use of pop up menus, described below, assists in reducing screen clutter and user distraction. Window Four is restricted to point viewing and editing. A small region (20 pixels by 20 pixels) surrounding the desired point is magnified by a factor of ten. Finally, Window Five displays pertinent system status information and helpful instructions to the user.
Figure 4.1 Screen Layout for the Editor
The editor also employs temporary and overlapping windows to allow more surface area from a fixed size screen. Designed to preserve partially obscured terminal contents, the windows enable viewing of information on drawing selections, feature classifications and directory listings. These overlays are displayed only upon request.

C. INPUT DEVICES

User responses are entered either through the IRIS keyboard or three button mouse. The keyboard’s function is limited to the entry of input/output filenames, desired directories for display, and feature categorization codes. Remaining input is regulated by the mouse.

The three mouse buttons (Left, Middle and Right) offer flexible and easy control. By properly positioning the cursor on the monitor screen and simultaneously pressing the appropriate button, the user can effortlessly perform such actions as the selection of menu choices, areas to be updated and different drawing options. Movement of the mouse across a hard surface dictates the cursor’s actual location on the screen.

D. MENU STRUCTURE

The editing system consists of a series of pop up menus organized in a hierarchical structure (see Figure 4.2). Source code for these menus was derived by modifying a general purpose menu package provided by Silicon Graphics.
Figure 4.2  Menu Structure for Editing System
The *pop up* menu routine is very simple. When invoked, a menu is automatically drawn on the screen displaying the possible program selections. By moving the cursor with the mouse device across an option, that entry is highlighted. Upon the depression of any mouse button, three actions occur:

- the entry under the cursor at that time is selected,
- the menu disappears from the screen, and
- the program branches to the procedure specified by the option.

If the cursor lies completely outside the menu field when the button is pressed, the menu remains on the screen and the routine continues to wait until an appropriate response is entered.

A description of the operations and capabilities for each of the available commands follows.

1. **Area Edit**

   This procedure permits large region editing of the map by allowing the user to add, delete, move and change displayed components. Refer to the section "Point Edit" for modifying individual pixels.

   Features are designed by combining different drawing commands and attributes. Other modifications such as deleting, reproducing, moving and changing attribute settings require the desired figure be identified. Selection of an individual object is possible by positioning the cursor over it and then depressing the middle mouse button. The editor will flash the chosen figure and request
confirmation of its correctness. A rejection by the user causes the system to search for the next best solution. This process of searching and requesting approval continues until the user is satisfied with an object or when no further figures are found. If a selection is made, the current editing action is then performed. A warning is displayed when no figure is found.

Six choices are available in the Area Edit Menu:

1. **Drawing Commands** - allow feature sketching using one or more of the following shapes:
   
   Circle  
   Rectangle  
   Polygon  
   Horizontal Line  
   Vertical Line  
   Single Line  
   Continuous Line  

   The last produced shape can either be erased or reproduced without entering the "Delete Figure" or "Reproduce Figure" choices of the Area Edit Menu. The new shape constructed will be situated at the cursor's current location for further repositioning on the map.

2. **Attributes** - enable viewing or redesignation of the current setting or a specific figure's characteristics. This option comprises four subcommands:
   
   View Attribute Set  
   Change Attribute Set  
   View Figure Attributes  
   Change Figure Attributes  

   Selection of the first subcommand provides a display of the current attribute settings without modification.

   **CAUTION** The HRS system does not support concave polygons; drawing these figures may produce unpredictable results.

28
The second subcommand allows any of the current attributes to be changed. When the editor is first entered, they are initially set to the following values:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Color</td>
<td>White</td>
</tr>
<tr>
<td>Linestyle</td>
<td>Continuous</td>
</tr>
<tr>
<td>Linewidth</td>
<td>One pixel wide</td>
</tr>
<tr>
<td>Fill Pattern</td>
<td>Solid</td>
</tr>
<tr>
<td>Feature Code</td>
<td>001 (nonvegetative)</td>
</tr>
</tbody>
</table>

The first four settings are modified by choosing a desired alternative from among the visually displayed pop up pallets. There are twelve colors (black, light and dark blue, light and dark brown, light and dark green, light and dark purple, red, yellow (unknown) and white), fourteen linestyles, five linewidths and ten fill patterns. Feature codes are altered by entering in the new value after a system prompt. Depression of the PF1 key displays the entire listing of descriptions with corresponding codes for review.

The third subcommand allows one to view a particular figure's attributes without modification.

To change an individual figure's attributes, the fourth subcommand is employed. After the figure is located, current settings for the figure are displayed. Changes are entered in a similar manner as described in the process for altering current attributes.

(3) **Delete Figure** - removes the selected object from the display. The vacancy is filled with the color white and identified as nonvegetative.

(4) **Reproduce Figure** - produces a carbon copy of the selected object with identical attribute settings.

(5) **Move Figure** - repositions the selected object to any desired region on the map. The cancel option returns the object to its original location.

(6) **Change View** - refer to the section "Change View" discussed below.
Point Edit

Area Editing commands can become quite cumbersome when performing slight alterations to existing features or when identifying unknown regions around border lines. The Point Edit facility offers a viable alternative. With this procedure, one can easily modify individual pixels within a map display using the following two commands:

1. Point Zoom-In - refer to the section "Change View".

2. Attributes - allow viewing or redesignation of the color and/or feature description of the current setting or of any specified pixel within the map. The desired point to be updated is first located with the Point Zoom-In command. Selected pixels assume the currently specified color and feature categories. By default, these settings are initialized to nonvegetative color - white, feature code = 001). The editing color may be changed at any time from a pallet of map colors displayable upon request. Feature description codes are modified by entering in the new value after the system prompt. The entire listing of description codes can be viewed by depressing the PF1 key. Changes recorded on the magnified area are also reflected in the current map view.

Change View

Magnification of different map sections may be necessary to assist the human eye in correctly perceiving and locating small features requiring updating. Three sections are provided by the editor:

1. Area Zoom-In - 2x and 4x magnification of selected user-defined areas. The current map view is overwritten by the enlarged area. (screen location: Window One).

2. Point Zoom-In - 10x magnification on a 20 pixel x 20 pixel area surrounding the desired point. (screen location: Window Four).
(3) **Reset** - restores the screen to the original map view.

Exiting this procedure, when called from the Main Menu, invokes the Reset command.

4. **Read in a File**

This choice allows entry of previously stored map data for further editing. After display of the system prompt, the user types in the desired filename at the keyboard. If the file exists, the data is drawn on the screen overwriting any prior map images; otherwise, an appropriate error message appears with control returning to the Main Menu. The user may cancel any inadvertent calls made to this routine by using the Escape key on the keyboard.

A directory listing facility is also available. By depressing the PF1 key, a prompt is shown requesting the desired directory name: the user's current directory serves as default. This facility is capable of listing all files in any directory not read protected.

5. **Write to a File**

To save changes entered by the user, a method is necessary to capture the screen display as stored data within a file. This command selection provides such a capability. The projected map image is transformed into similarly colored areas which are then subdivided into polygons before being written to the created file. Specification of an existing filename results in the destruction and
overwriting of any old contents. The actual algorithm used for this procedure is presented in detail in Chapter Five.

The process of designating an output filename in addition to the Escape and PF1 key functions are identical to those previously described (refer to the section "Read in a File").

6. Exit System

Selection of this option terminates the editing session. Changes not previously saved are lost forever. The graphics hardware system is returned to its initial state. Input devices are unqueued. the viewport and textport are restored to full screen. and the communication buffers are flushed.

1. SYSTEM PROMPTS

System information and prompts appear on the screen at various times during the execution of the editor. The prompts serve as a guide through an editing session without reference to an operating manual. These helpful messages reduce the amount of information the user must memorize while keeping him informed of what is happening now and what to expect next. They include:


2. Key Function Instructions - display the active keyboard keys at any given time. The keys are identified by dark blue squares with an abbreviation of the key inside (for example. ESC signifies the Escape Key). Keys not listed are inactive; pressing them has no effect. (screen location: Window Five).
3. **Mouse Function Instructions** - list the outcome of depressing any of the three mouse buttons. The buttons are represented by dark blue squares with a letter (L for left M for middle and R for right) inside. Buttons not shown are inactive, pressing them has no effect. (screen location: Window Five)

4. **System Status** report on the program's reaction under the following conditions: reading in a file, writing out a file, delay due to processing, and no action taken. (screen location: Window Five).

5. **Error Messages** warn when erroneous or adverse conditions are present (for example, invalid input). Depending on the situation, the system tries to recover by returning to its last working state. (screen location: Window One).

6. **Other Prompts** request additional input information from the user such as a filename, directory name and/or feature category code. (screen location: Window One)

**Features Supported**

Cartographic components are designed and updated using the Area and Point Edit options described above. The drawing commands and attributes, provided by the editing system, enable the user to generate most of the symbology related to USGS maps (see Appendix B).

A three-digit description code assists in feature identification. During the preprocessing stages, all components are initially classified into one of the following broad categories based on color:

<table>
<thead>
<tr>
<th>Color</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yellow</td>
<td>000</td>
<td>Unknown</td>
</tr>
<tr>
<td>White</td>
<td>001</td>
<td>Nonvegetative</td>
</tr>
<tr>
<td>Black</td>
<td>002</td>
<td>Man made object</td>
</tr>
</tbody>
</table>
These initial settings can be redefined with either edit facility. The remaining possible feature descriptions with corresponding program defined codes are listed in Appendix B.

G. FILE FORMAT

The cartographic data base used for this study was generated from the processing scheme presented in Chapter Two. The data within each file represents a section from a USGS map stored in binary format.

Each map section is divided into a number of uniformly colored areas. An area is identified by a set of short integer codes representing its color, description and allowable speed zone. Each area is further subdivided into a number of polygons as determined by the Data Compression Algorithm. The polygons are defined by short integer codes representing their type (point, line, polygon, filled polygon) and vertex locations. An end of area marker separates each successively defined area region. Figure 4.3 depicts a typical file format.

The editor extends the seven map colors to twelve. Light and dark colors for green, blue, purple and brown support cartographic feature contrast in given regions and have no significance with respect to classification. Yellow identifies unknown areas.
Figure 4.3 Typical File Format with Key
V. MANAGING USER REVISIONS

A major concern in the design of any man-machine interface is the ability to ensure rapid response rates. Users by nature become very anxious when reactions to their requests do not occur within a few seconds of input. Even prompting messages, indicating processing is in progress, lose their effectiveness after a period of time. The user may begin to fidget and start pressing buttons and "try stuff", perhaps leading to disastrous results. In order to minimize such processing delays, an internal indexing scheme was incorporated into the editor's design. This plan, in addition to the method used for updating a database, are presented below.

A. INTERNAL INDEXING SCHEME

As mentioned in Chapter Four, each map is divided into several uniformly divided areas which are further subdivided into a number of polygons. Each area is defined as an object. Due to the feature complexity of most cartographic documents, it is not unusual to have in excess of 5000 objects defined at any point in time. This large number places a strain on the IRIS's memory management for locating a particular object quickly. Response time slowly decreases at an inverse

---

Note 1: the graphics packaging mechanism provided on the IRIS, is composed of a suite of graphics commands.

36
rate to the number of objects created. To help alleviate this problem, an artificial indexing scheme (see Figure 5.1) was added to the editing system.

The scheme consists of an indexing array subscripted from 0 through 3277. As an area of the map is entered into memory, it is assigned a short integer identification number. The area is then placed into a linked list, arranged in descending order, at an array location equal to the absolute value of the index number divided by ten.

To locate an object for updating or modifying, the pick mechanism from the IRIS graphics library is employed. The picking facility allows the user to identify an object by maneuvering the cursor as a pointing device. Objects intersecting a small rectangular area around the cursor are considered "hits" and their identification numbers, assigned during input, are placed into a names stack. The indexing scheme, which reduces search time to a maximum of 20 objects, can then be used to quickly locate each object by number for manipulation.

B. UPDATING THE DATABASE

Upon completion of an editing session, changes to a map are saved by using the "Write to a File" option from the Main Menu. The four algorithms, listed in Appendix C, comprising this command selection include: Update_Database, Data Compression, Expand_Area and Subset_Area. The latter three were written and used by Diehl in the Data Compression phase of the cartographic
rate to the number of objects created. To help alleviate this problem, an artificial indexing scheme (see Figure 5.1) was added to the editing system.

The scheme consists of an indexing array subscripted from 0 through 3277. As an area of the map is entered into memory, it is assigned a short integer identification number. The area is then placed into a linked list, arranged in descending order, at an array location equal to the absolute value of the index number divided by ten.

To locate an object for updating or modifying, the pick mechanism from the IRIS graphics library is employed. The picking facility allows the user to identify an object by maneuvering the cursor as a pointing device. Objects intersecting a small rectangular area around the cursor are considered "hits" and their identification numbers, assigned during input, are placed into a names stack. The indexing scheme, which reduces search time to a maximum of 20 objects, can then be used to quickly locate each object by number for manipulation.

B. UPDATING THE DATABASE

Upon completion of an editing session, changes to a map are saved by using the "Write to a File" option from the Main Menu. The four algorithms, listed in Appendix C, comprising this command selection include: Update Database, Data Compression, Expand Area and Subset Area. The latter three were written and used by Diehl in the Data Compression phase of the cartographic
processing package. Slight modifications were added to enable handling of a larger set of colors. Each of the algorithms are described below.

1. **Update Database**

   Since the primary use for these databases is in autonomous vehicle route planning, it is desirable for components to be uniquely represented in polygonal form. Duplicated areas, as seen in the case of overlying features, need to be eliminated. The Update Database Algorithm provides this capability.

   The program itself assigns a set of unique color shades, representing the twelve possible map colors, to each feature description. During an editing session, default colors are actually used for display and to reflect entered changes. This algorithm replaces the default color for each area of the map with an appropriate color shade as dictated by the area's description code (see Figure 5.2a). After reassignment, the map is redrawn. The new map image is subsequently read off the screen and funneled through the Data Compression Algorithm (see Figure 5.2b).

2. **Data Compression**

   The entire map surface is constructed as a grid, each cell being one pixel wide. A duplicate setup, hereafter known as the *scratch pad*, has all its cells initially set to "0". Working in a sweeping motion from left to right and top to bottom on the proposed map layout, the algorithm checks for unexamined pixels. When one is found, the process converts the pixel's color shade back to the default color (see Figure 5.2b) and writes the color, description code and speed
Figure 5.2a  Updating the Data Base

002 = Man Made Object
Figure 5.2b  Updating the Data Base
code to a 1. The Expand Area algorithm are invoked prior to outputting an end of area marker. This event persists until each pixel of the map has been surveyed.

3 Expand Area

This algorithm focuses on a three by three subgrid successively encompassing each cell of the map. Three actions occur when any of the surrounding eight neighbors have the same color as the center. For each similarly colored pixel:

- the pixel is tagged as used to avoid duplicate inspections,
- a flag is set to "1" in the appropriately corresponding position of the scratch pad, and
- the pixel's location is placed in a queue for later examination of its neighbors.

The interrogation process continues until the queue is empty, resulting in a uniformly colored region marked by set flags in the scratch pad.

4 Subset Area

The identified homogeneous area marked in the scratch pad is then subdivided using the ten possible geometric shapes shown in Figure 5.3. Items a through e represent right equilateral triangles, items f through i describe lines, and item j defines a point. [Ref. 2] addresses the actual selection criteria for these forms and the process involved in dividing intact area regions. Upon completion
Figure 5.3  Polygonal Encoding Scheme
of the subdivision, scratch pad flags are reset to "0" and the area identification phase is reexecuted.

The above method is a simple scheme allowing compact storage for processed and edited information. The resulting data base itself contains a cartographic image divided into homogeneous areas which are further subdivided into polygons. Each polygonal side and/or vertex is uniquely defined avoiding redundant representations. One and only one feature classification is attached to each polygon.
VI. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

This study examines an interactive editing system for maintaining data bases generated from a cartographic processing system currently under development at the Naval Postgraduate School. The amount of updating necessary on any stored map depends on the care taken in digitization, the accuracy of the image processing phase and the level of quality required in the finished product. In general, the time devoted towards this effort is proportional to the map's complexity. If each step involved in the generation of the data base has been carefully executed and there are no mistakes in content, the editing is mostly cosmetic.

The editor was written in the C programming language and contains approximately 10,000 lines of code. Driven by a set of pop up menus arranged in a hierarchical structure, the system has been designed to handle modifications to either large map areas or individual points. The latter capability is useful in correcting processing and digitizer induced errors around linear features. A mouse served as the primary input device and proved to be fairly accurate in manipulating and designing cartographic data. Display colors approximate those used on the paper map.
Preliminary tests indicate the editor's performance depends on the map's complexity and the hardware's internal memory size. The system performs best using an IRIS with six megabytes of memory and a floating-point accelerator.

B LIMITATIONS AND RECOMMENDED EXTENSIONS

The editor developed in this study merely scratches the surface of the problem in applying interactive techniques to update and modify cartographic databases. Several limitations prevent the present system from operating at an as-yet practical level in a real-time environment. A few of these limiting factors with possible research extensions are discussed below:

1. Each preprocessed map contained within the database defines a $280 \times 440$ pixel area. This self-imposed size constraint is addressed in [Ref. 2]. Many cartographic features commonly span across several adjoining map sections. Ideally, any portion of these features should be readily available for editing, reflecting all updates. The current editing system is unable to handle changes occurring over several neighboring regions. Modifications are instead restricted to within one defined map section at any given time.

   EXTENSION: Addition of a facility to merge adjacent sectional map pieces and modification of the editor to handle the larger areas.

2. USGS cartographic features served as a standard for this research. They are designed by combining different colors, linewidths, linestyles and fill patterns. The editing system, however, lacks appropriate checks for preventing entry of illegal combinations which may render a database as unusable.

   EXTENSION: Incorporation of proper error checking to the editor to minimize such mishaps.

3. Some USGS cartographic features are not supported by the editor due to their inherent complexity and the difficulty of approximating their display by setting bit maps such as with Tailings, Intricate Surface Areas and
Gravel Beaches. Others require several overlays in order to capture the proper design. The latter case, in particular, is very tiring to the user and inefficient for the computer. Additionally, the algorithm employed in updating the database (discussed in Chapter Five) fails to preserve the integrity of areas composed of complex fill patterns. Instead, these areas are needlessly splintered into smaller regions. Many of the standard cartographic symbols look nice on paper but significantly tax memory requirements in duplication efforts on the computer.

**EXTENSION**: Development of a more efficient graphical representation for mapping symbols. The proposed scheme should allow one to depict the various natural, man-made and aerial features. The result achieved should be equal to traditional cartographic techniques as far as accuracy, integrity of design and quality of reproduction.

4. The Data Compression Algorithm used for subdividing a map produces areas composed of many irregularly sized polygons having no common sides. Due to these variations, editing of sub-areas tends to be awkward.

**EXTENSION**: Modification of the Data Compression Algorithm to produce more equally sized subdivisions.

5. Curved features (contours, winding roads/rivers, etc) can only be approximated by using several straight line segments. The accuracy of these features is dependent on the user’s drawing skills.

**EXTENSION**: Development of a procedure capable of generating accurate curved lines without excessive memory or execution requirements.

6. No text labeling procedures are provided by the editor.

**EXTENSION**: This facility is not necessary unless the editor’s role is extended to uses requiring that capability.

Despite a number of limitations, the editing system illustrates the useful role interactive computer graphics plays in maintaining geographic data bases.

Additional studies are needed to expand and improve upon the current system's
capabilities of allowing new editions and revisions without expensive recomilation.
APPENDIX A  SAMPLE EDITING SESSION

BEFORE 'SNAP SHOT': prior to Edit Session

GREEN

BROWN

BLUE

BLACK
BEFORE "SNAP SHOT" : prior to Edit Session (cont'd)
AFTER 'SNAP SHOT': after an Edit Session
AFTER 'SNAP SHOT': after an Edit Session (cont'd)
### APPENDIX B – USGS TOPOGRAPHIC MAP SYMBOLS

[Ref. 6]

<table>
<thead>
<tr>
<th>CODE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>010</td>
<td>CONTROL DATA AND MONUMENTS</td>
</tr>
<tr>
<td>011</td>
<td>Aerial photograph roll and frame number †</td>
</tr>
<tr>
<td>012</td>
<td>Horizontal control:</td>
</tr>
<tr>
<td>013</td>
<td>Third order or better, permanent mark †</td>
</tr>
<tr>
<td>014</td>
<td>With third order or better elevation †</td>
</tr>
<tr>
<td>015</td>
<td>Checked spot elevation †</td>
</tr>
<tr>
<td>016</td>
<td>Coincident with section corner †</td>
</tr>
<tr>
<td>017</td>
<td>Unmonumented †</td>
</tr>
<tr>
<td>018</td>
<td>Vertical control:</td>
</tr>
<tr>
<td>019</td>
<td>Third order or better, with tablet †</td>
</tr>
<tr>
<td>020</td>
<td>Third order or better, recoverable mark †</td>
</tr>
<tr>
<td>021</td>
<td>Bench mark at found section corner †</td>
</tr>
<tr>
<td>022</td>
<td>Spot elevation ‡</td>
</tr>
<tr>
<td>023</td>
<td>Boundary monument:</td>
</tr>
<tr>
<td>024</td>
<td>With tablet ‡</td>
</tr>
<tr>
<td>025</td>
<td>Without tablet ‡</td>
</tr>
<tr>
<td>026</td>
<td>With number and elevation †</td>
</tr>
<tr>
<td>027</td>
<td>U. S. mineral or location monument</td>
</tr>
</tbody>
</table>

### BOUNDARIES

<table>
<thead>
<tr>
<th>CODE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>030</td>
<td>National</td>
</tr>
<tr>
<td>031</td>
<td>State or territorial</td>
</tr>
<tr>
<td>032</td>
<td>County or equivalent</td>
</tr>
<tr>
<td>033</td>
<td>Civil township or equivalent</td>
</tr>
<tr>
<td>034</td>
<td>Incorporated city or equivalent</td>
</tr>
<tr>
<td>035</td>
<td>Park, reservation, or monument</td>
</tr>
<tr>
<td>036</td>
<td>Small park</td>
</tr>
</tbody>
</table>

### LAND SURVEY SYSTEMS

<table>
<thead>
<tr>
<th>CODE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>040</td>
<td>U. S. Public Land Survey System:</td>
</tr>
<tr>
<td>041</td>
<td>Township or range line</td>
</tr>
<tr>
<td>042</td>
<td>Location doubtful</td>
</tr>
<tr>
<td>043</td>
<td></td>
</tr>
<tr>
<td>CODE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>044</td>
<td>Section line</td>
</tr>
<tr>
<td>045</td>
<td>Location doubtful</td>
</tr>
<tr>
<td>046</td>
<td>Found section corner</td>
</tr>
<tr>
<td>047</td>
<td>Found closing corner</td>
</tr>
<tr>
<td>048</td>
<td>Witness corner</td>
</tr>
<tr>
<td>049</td>
<td>Meander corner</td>
</tr>
<tr>
<td>050</td>
<td>Other land surveys:</td>
</tr>
<tr>
<td></td>
<td>051 Township or range line</td>
</tr>
<tr>
<td>052</td>
<td>Section line</td>
</tr>
<tr>
<td>053</td>
<td>Land grant or mining claim</td>
</tr>
<tr>
<td>054</td>
<td>Monument</td>
</tr>
<tr>
<td>055</td>
<td>Fence line</td>
</tr>
</tbody>
</table>

**ROADS AND RELATED FEATURES**

<table>
<thead>
<tr>
<th>CODE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>061</td>
<td>Primary highway</td>
</tr>
<tr>
<td>062</td>
<td>Secondary highway</td>
</tr>
<tr>
<td>063</td>
<td>Light duty road</td>
</tr>
<tr>
<td>064</td>
<td>Unimproved road</td>
</tr>
<tr>
<td>065</td>
<td>Trail</td>
</tr>
<tr>
<td>066</td>
<td>Dual highway</td>
</tr>
<tr>
<td>067</td>
<td>Dual highway with median strip</td>
</tr>
<tr>
<td>068</td>
<td>Road under construction</td>
</tr>
<tr>
<td>069</td>
<td>Underpass</td>
</tr>
<tr>
<td>070</td>
<td>Overpass</td>
</tr>
<tr>
<td>071</td>
<td>Bridge</td>
</tr>
<tr>
<td>072</td>
<td>Drawbridge</td>
</tr>
<tr>
<td>073</td>
<td>Tunnel</td>
</tr>
</tbody>
</table>

**BUILDINGS AND RELATED FEATURES**

<table>
<thead>
<tr>
<th>CODE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>081</td>
<td>Dwelling or place of employment - small</td>
</tr>
<tr>
<td>082</td>
<td>Dwelling or place of employment - large</td>
</tr>
<tr>
<td>083</td>
<td>School</td>
</tr>
<tr>
<td>084</td>
<td>Church</td>
</tr>
<tr>
<td>085</td>
<td>Barn, warehouse, etc - small</td>
</tr>
<tr>
<td>086</td>
<td>Barn, warehouse, etc - large</td>
</tr>
<tr>
<td>CODE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>------</td>
<td>---------------------------------------</td>
</tr>
<tr>
<td>087</td>
<td>House omission tint</td>
</tr>
<tr>
<td>088</td>
<td>Racetrack</td>
</tr>
<tr>
<td>089</td>
<td>Airport</td>
</tr>
<tr>
<td>090</td>
<td>Landing strip</td>
</tr>
<tr>
<td>091</td>
<td>Well (other than water)</td>
</tr>
<tr>
<td>092</td>
<td>Windmill</td>
</tr>
<tr>
<td>093</td>
<td>Water tank: small</td>
</tr>
<tr>
<td>094</td>
<td>Water tank: large</td>
</tr>
<tr>
<td>095</td>
<td>Other tank: small</td>
</tr>
<tr>
<td>096</td>
<td>Other tank: large</td>
</tr>
<tr>
<td>097</td>
<td>Covered reservoir</td>
</tr>
<tr>
<td>098</td>
<td>Gaging station</td>
</tr>
<tr>
<td>099</td>
<td>Landmark object</td>
</tr>
<tr>
<td>100</td>
<td>Campground</td>
</tr>
<tr>
<td>101</td>
<td>Picnic area</td>
</tr>
<tr>
<td>102</td>
<td>Cemetery: small</td>
</tr>
<tr>
<td>103</td>
<td>Cemetery: large</td>
</tr>
</tbody>
</table>

**RAILROADS AND RELATED FEATURES**

<table>
<thead>
<tr>
<th>CODE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>111</td>
<td>Standard gauge single track</td>
</tr>
<tr>
<td>112</td>
<td>Station</td>
</tr>
<tr>
<td>113</td>
<td>Standard gauge multiple track</td>
</tr>
<tr>
<td>114</td>
<td>Abandoned</td>
</tr>
<tr>
<td>115</td>
<td>Under construction</td>
</tr>
<tr>
<td>116</td>
<td>Narrow gauge single track</td>
</tr>
<tr>
<td>117</td>
<td>Narrow gauge multiple track</td>
</tr>
<tr>
<td>118</td>
<td>Railroad in street</td>
</tr>
<tr>
<td>119</td>
<td>Juxtaposition</td>
</tr>
<tr>
<td>120</td>
<td>Roundhouse and turntable</td>
</tr>
</tbody>
</table>

**TRANSMISSION LINES AND PIPELINES**

<table>
<thead>
<tr>
<th>CODE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>131</td>
<td>Power transmission line: pole</td>
</tr>
<tr>
<td>132</td>
<td>Power transmission line: tower</td>
</tr>
<tr>
<td>133</td>
<td>Telephone or telegraph line</td>
</tr>
<tr>
<td>134</td>
<td>Aboveground oil or gas pipeline</td>
</tr>
<tr>
<td>135</td>
<td>Underground oil or gas pipeline</td>
</tr>
<tr>
<td>CODE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>140</td>
<td>CONTOURS</td>
</tr>
<tr>
<td>141</td>
<td>Topographic:</td>
</tr>
<tr>
<td>142</td>
<td>Intermediate</td>
</tr>
<tr>
<td>143</td>
<td>Index</td>
</tr>
<tr>
<td>144</td>
<td>Supplementary</td>
</tr>
<tr>
<td>145</td>
<td>Depression</td>
</tr>
<tr>
<td>146</td>
<td>Cut</td>
</tr>
<tr>
<td>147</td>
<td>Fill</td>
</tr>
<tr>
<td>148</td>
<td>Bathymetric:</td>
</tr>
<tr>
<td>149</td>
<td>Intermediate</td>
</tr>
<tr>
<td>150</td>
<td>Index</td>
</tr>
<tr>
<td>151</td>
<td>Primary</td>
</tr>
<tr>
<td>152</td>
<td>Index Primary</td>
</tr>
<tr>
<td>153</td>
<td>Supplementary</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CODE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>160</td>
<td>MINES AND CAVES</td>
</tr>
<tr>
<td>161</td>
<td>Quarry or open pit mine</td>
</tr>
<tr>
<td>162</td>
<td>Gravel, sand, clay, or borrow pit</td>
</tr>
<tr>
<td>163</td>
<td>Mine tunnel or cave entrance</td>
</tr>
<tr>
<td>164</td>
<td>Prospect</td>
</tr>
<tr>
<td>165</td>
<td>Mine shaft</td>
</tr>
<tr>
<td>166</td>
<td>Mine dump</td>
</tr>
<tr>
<td>167</td>
<td>Tailings</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CODE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>170</td>
<td>SURFACE FEATURES</td>
</tr>
<tr>
<td>171</td>
<td>Levee</td>
</tr>
<tr>
<td>172</td>
<td>Sand or mud area, dunes, or shifting sand</td>
</tr>
<tr>
<td>173</td>
<td>Intricate surface area</td>
</tr>
<tr>
<td>174</td>
<td>Gravel beach or glacial moraine</td>
</tr>
<tr>
<td>175</td>
<td>Tailings pond</td>
</tr>
<tr>
<td>CODE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td>180</td>
<td><strong>VEGETATION</strong></td>
</tr>
<tr>
<td>181</td>
<td>Woods</td>
</tr>
<tr>
<td>182</td>
<td>Scrub</td>
</tr>
<tr>
<td>183</td>
<td>Orchard</td>
</tr>
<tr>
<td>184</td>
<td>Vineyard</td>
</tr>
<tr>
<td>185</td>
<td>Mangrove</td>
</tr>
<tr>
<td>190</td>
<td><strong>MARINE SHORELINE</strong></td>
</tr>
<tr>
<td>191</td>
<td>Topographic maps:</td>
</tr>
<tr>
<td>192</td>
<td>Approximate mean high water</td>
</tr>
<tr>
<td>193</td>
<td>Indefinite or unsurveyed</td>
</tr>
<tr>
<td>194</td>
<td>Topographic-bathymetric maps:</td>
</tr>
<tr>
<td>195</td>
<td>Mean high water</td>
</tr>
<tr>
<td>196</td>
<td>Apparent (edge of vegetation)</td>
</tr>
<tr>
<td>200</td>
<td><strong>COASTAL FEATURES</strong></td>
</tr>
<tr>
<td>201</td>
<td>Foreshore flat</td>
</tr>
<tr>
<td>202</td>
<td>Rock or coral reef</td>
</tr>
<tr>
<td>203</td>
<td>Rock bare or awash</td>
</tr>
<tr>
<td>204</td>
<td>Group of rocks bare or awash</td>
</tr>
<tr>
<td>205</td>
<td>Exposed wreck</td>
</tr>
<tr>
<td>206</td>
<td>Depth curve</td>
</tr>
<tr>
<td>207</td>
<td>Sounding</td>
</tr>
<tr>
<td>208</td>
<td>Breakwater, pier, jetty, or wharf</td>
</tr>
<tr>
<td>209</td>
<td>Seawall</td>
</tr>
<tr>
<td>210</td>
<td><strong>BATHYMETRIC FEATURES</strong></td>
</tr>
<tr>
<td>211</td>
<td>Area exposed at mean low tide</td>
</tr>
<tr>
<td>212</td>
<td>Sounding datum</td>
</tr>
<tr>
<td>213</td>
<td>Channel</td>
</tr>
<tr>
<td>214</td>
<td>Offshore oil or gas: well</td>
</tr>
<tr>
<td>215</td>
<td>Offshore oil or gas: platform</td>
</tr>
<tr>
<td>216</td>
<td>Sunken rock</td>
</tr>
</tbody>
</table>

57
<table>
<thead>
<tr>
<th>CODE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>220</td>
<td>RIVERS, LAKES, AND CANALS</td>
</tr>
<tr>
<td>221</td>
<td>Intermittent stream</td>
</tr>
<tr>
<td>222</td>
<td>Intermittent river</td>
</tr>
<tr>
<td>223</td>
<td>Disappearing stream</td>
</tr>
<tr>
<td>224</td>
<td>Perennial stream</td>
</tr>
<tr>
<td>225</td>
<td>Perennial river</td>
</tr>
<tr>
<td>226</td>
<td>Small falls</td>
</tr>
<tr>
<td>227</td>
<td>Small rapids</td>
</tr>
<tr>
<td>228</td>
<td>Large falls</td>
</tr>
<tr>
<td>229</td>
<td>Large rapids</td>
</tr>
<tr>
<td>230</td>
<td>Masonry dam</td>
</tr>
<tr>
<td>231</td>
<td>Dam with lock</td>
</tr>
<tr>
<td>232</td>
<td>Dam carrying road</td>
</tr>
<tr>
<td>233</td>
<td>Intermittent lake or pond</td>
</tr>
<tr>
<td>234</td>
<td>Dry lake</td>
</tr>
<tr>
<td>235</td>
<td>Narrow wash</td>
</tr>
<tr>
<td>236</td>
<td>Wide wash</td>
</tr>
<tr>
<td>237</td>
<td>Canal, flume, or aqueduct with lock</td>
</tr>
<tr>
<td>238</td>
<td>Elevated aqueduct, flue, or conduit</td>
</tr>
<tr>
<td>239</td>
<td>Aqueduct tunnel</td>
</tr>
<tr>
<td>240</td>
<td>Water well</td>
</tr>
<tr>
<td>241</td>
<td>Spring or seep</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CODE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>250</td>
<td>GLACIERS AND PERMANENT SNOWFIELDS</td>
</tr>
<tr>
<td>251</td>
<td>Contours and limits</td>
</tr>
<tr>
<td>252</td>
<td>Form lines</td>
</tr>
<tr>
<td>CODE</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>------</td>
<td>------------------------------</td>
</tr>
<tr>
<td>260</td>
<td>SUBMERGED AREAS AND BOGS</td>
</tr>
<tr>
<td>261</td>
<td>Marsh or swamp</td>
</tr>
<tr>
<td>262</td>
<td>Submerged marsh or swamp</td>
</tr>
<tr>
<td>263</td>
<td>Wooded marsh or swamp</td>
</tr>
<tr>
<td>264</td>
<td>Submerged wooded marsh or swamp</td>
</tr>
<tr>
<td>265</td>
<td>Rice field</td>
</tr>
<tr>
<td>266</td>
<td>Land subject to inundation</td>
</tr>
</tbody>
</table>

* Features not supported by the editor.

Note: The editor does not provide automatic symbol plotting. Features containing curved lines can only be approximated using small line segments.
begin UPDATE DATABASE algorithm
{
  initialize variables:
  initialize map region (TOTAL_ROWS x TOTAL_COLUMNS)
    pixel colors to unknown;
  initialize scratch pad values to 0;
  create a border around map region;

  /* index through each area of the map */
  for (i = 0; i < MAX_DIMENSION; ++i)
  {
    new_area = mapindex[i].next_area;

    /* examine all areas at the same index reference point */
    while (new_area = exist)
    {
      change new_area colortype to new_area description colorshade;

      /* increment to next area */
      new_area = new_area -> next_area;
    }
  }

  redraw map on screen;
  read color values within map off the screen;
  convert colorshades to appropriate processing color;

  call data compression algorithm;

end UPDATE DATABASE algorithm
begin DATA_COMPRESSION algorithm
{
    initialize variables:

    /* create file to hold compressed data representation */
    create output file;

    /* successively examine each pixel of the map region */
    for (i = 1; i <= TOTAL_ROWS; i++)
    {
        for (j = 1; j <= TOTAL_COLUMNS; j++)
        {
            /* if pixel has not been examined as yet */
            if (pixel(i, j) = not used)
            {
                convert pixel color back to description colorshade;
                determine true map color:
                determine description of map area:

                write true map color to output file;
                write description code to output file;
                write speed code to output file;

                /* expand then subdivide homogeneous areas */
                call expand_area algorithm;
                call subset_area algorithm;

                write end of area marker to output file;
            }
        }
    }
}
end DATA_COMPRESSION algorithm

Data Compression Algorithm

61
begin EXPAND AREA algorithm
{
    set pixel(x, y) = used;
    set corresponding (x, y) location in scratch pad to 1;
    insert pixel (x, y) location into the queue;

    /* examine only pixel values within the queue */
    while (queue = not empty)
    {
        remove (x, y) value off the queue;

        /* check the surrounding eight neighbors of the grid's center to see if any are similar in color */
        for (i = (x-1); i <= (x+1); i++)
            for (j = (y-1); j <= (y+1); j++)
            {
                /* tag map region and scratch pad location of similarly colored neighbor. Make queue entry of pixel's location */
                if ((pixel color(i, j) = pixel color(x, y)) and (pixel(i, j) = not used))
                {
                    set pixel(i, j) = used;
                    set corresponding (x, y) location in scratch pad to 1;
                    set maximum x value = maximum(previous x value, i);
                    insert pixel (x, y) location into the queue;
                }
            }
        }
    }
}
end EXPAND AREA algorithm

Expand Homogeneous Area Algorithm
begin SUBSET_AREA algorithm
{
  do
  {
    /* initialize temp variables to selected positions within the
       scratch pad */
    set temp0 = scratch pad area (x, y+1);
    set temp1 = scratch pad area (x+1, y+1);
    set temp2 = scratch pad area (x+1, y);
    set temp3 = scratch pad area (x+1, y-1);

    if (temp0 = 1)
    {
      if (temp1 = 1)
      {
        if (temp 2 = 1)
        {
          expand type two polygon; /* Figure 5.3, item b */
          write triangle verticies to output file;
        }
        else
        {
          expand type four polygon; /* Figure 5.3, item d */
          write triangle verticies to output file;
        }
      }
      else if (temp2 = 1)
      {
        if (temp3 = 1)
        {
          expand type five polygon; /* Figure 5.3, item e */
          write triangle verticies to output file;
        }
      }
    }
  }
}

Algorithm for Subdividing an Area into Polygons

63
else
{
    expand type six polygon: /* Figure 5.3, item f */
    write triangle vertices to output file;
}
}
else if (temp2 = 1)
{
    if (temp1 = 1)
    {
        if (temp 3 = 1)
        {
            expand type one polygon: /* Figure 5.3, item a */
            write triangle vertices to output file;
        }
        else
        {
            expand type three polygon: /* Figure 5.3, item c */
            write triangle vertices to output file;
        }
    }
    else if (temp3 = 1)
    {
        expand type five polygon: /* Figure 5.3, item e */
        write triangle vertices to output file;
    }
    else
    {
        expand type eight polygon: /* Figure 5.3, item h */
        write line endpoints to output file:
    }
}

Algorithm for Subdividing an Area into Polygons {cont'd}
else if (temp3 = 1)
{
    expand type_nine polygon; /* Figure 5.3, item i */
    write line endpoints to output file;
}
else if (temp1 = 1)
{
    expand type_seven polygon; /* Figure 5.3, item g */
    write line endpoints to output file;
}
else
{
    expand type_ten polygon; /* Figure 5.3, item j */
    write point location to output file;
}

reset scratch pad region flags;
}
while (x <= maximum x value);
}
end SUBSET_AREA algorithm

Algorithm for Subdividing an Area into Polygons (cont’d)

65
LIST OF REFERENCES


Distribution List

Defense Technical Information Center,
Cameron Station,
Alexandria, VA 22314

Library, Code 0142
Naval Postgraduate School,
Monterey, CA 93943

Center for Naval Analyses,
2000 N. Beauregard Street,
Alexandria, VA 22311

Director of Research Administration,
Code 012.
Naval Postgraduate School.
Monterey, CA 93943

Dr. Henry Fuchs.
208 New West Hall (035A).
University of North Carolina.
Chapel Hill, NC 27514

Dr. Kent R. Wilson.
University of California. San Diego
B-014.
Dept. of Chemistry.
La Jolla, CA 92038

Dr. Guy L. Tribble, III
Next, Inc.
3475 Deer Creek Road,
Palo Alto, California 94304

Bill Atkinson,
Apple Computer,
20525 Mariani Ave.
Cupertino, CA 95014

Dr. Victor Lesser.
University of Massachusetts, Amherst
Dept. of Computer and Information Science.
Amherst, MA 01003

Dr. Gunther Schrack,
Dept. of Electrical Engineering,
University of British Columbia,
Vancouver, B.C., Canada V6T 1W5
Professor Eihachiro Nakamae,  
Electric Machinery Laboratory,  
Hiroshima University,  
Higashihiroshima 724, Japan

Carl Machover,  
Machover Associates,  
199 Main Street,  
White Plains, New York 10601

Dr. Buddy Dean,  
Naval Postgraduate School,  
Code 52, Dept. of Computer Science,  
Monterey, California 93943

Earl Billingsley,  
43 Fort Hill Terrace,  
Northampton, MA 01060

Dr. Jan Cuny,  
University of Massachusetts, Amherst  
Dept. of Computer and Information Science, Amherst, MA 01003

Robert Lum,  
Silicon Graphics, Inc.  
2011 Stierlin Road,  
Mountain View, CA 94043

Jeff Hansch,  
Silicon Graphics, Inc.  
2011 Stierlin Road,  
Mountain View, CA 94043

Robert A. Walker,  
7657 Northern Oaks Court,  
Springfield, VA 22153

Dr. Barry L. Kalman,  
Washington University,  
Department of Computer Science,  
Box 1045,  
St. Louis, Missouri 63130

Dr. Wm. Randolph Franklin,  
Electrical, Computer, and Systems Engineering Department,  
Rensselaer Polytechnic Institute,  
Troy, New York 12180-3590
Dr. Gershon Kedem,
Microelectronics Center of North Carolina,
PO Box 12889,
3021 Cornwallis Road,
Research Triangle Park,
North Carolina 27709

Dr. Branko J. Gerovac,
Digital Equipment Corporation,
150 Locke Drive LMO4/H4, Box 1015
Marlboro, Massachusetts 01752-9115

Robert A. Schumacker,
Evans and Sutherland,
PO Box 8700,
580 Arapene Drive,
Salt Lake City, Utah 84108

R. A. Dammkoehler,
Washington University,
Department of Computer Science,
Box 1045,
St. Louis, Missouri 63130

Dr. Lynn Ten Eyck,
Interface Software,
79521 Highway 99N,
Cottage Grove, Oregon 97424

Kazy K. Yokota,
Japan Tech Services Corporation,
3F Ohkura Building,
1-4-10 Shiba-Daimon,
Minato-Ku, Tokyo 105, Japan

Toshiaki Yoshinaga,
Hitachi Works. Hitachi Ltd.
1-1. Saiwaicho 3 Chome,
Hitachi-shi, Ibaraki-ken,
317 Japan

Takatoshi Kodaira,
Omika Works, Hitachi Ltd.
2-1. Omika-cho 5-chome,
Hitachi-shi. Ibaraki-ken,
319-12 Japan

Atsushi Suzuki,
Hitachi Engineering. Co. Ltd.
Model Group,
2-1. Saiwaicho 3-Chome,
Hitachi-shi. Ibaraki-ken,
317 Japan
Toyoaki Nishimura,
Hitachi Engineering, Co. Ltd.
Motor Group,
2 1-Sawai-cho 3-Chome,
Hitachinaka, Ibaraki-ken,
317 Japan

Dr. John Standhammer,
Dept. of Electrical Engineering,
University of Florida,
Gainesville, Florida 32611

Dr. Lewis E. Hitchner,
Computer and Information Science Dept.
257 Applied Science Building,
University of California at Santa Cruz,
Santa Cruz, California 95064

Dr. Jane Wilhelms,
Computer and Information Science Dept.
257 Applied Science Building,
University of California at Santa Cruz,
Santa Cruz, California 95064

Dr. Pat Mantey,
Computer Engineering Department,
University of California at Santa Cruz,
Santa Cruz, California 95064

Dr. Walter A. Burkhardt,
University of California, San Diego
Dept. of Computer Science,
La Jolla, California 92093

P. K. Restaglia,
Silver Graphics, Inc.
2044 Scovill Road,
Mountain View, CA 94043

Peter Bodiswell,
Silver Graphics, Inc.
2044 Scovill Road,
Mountain View, CA 94043

N. L. Boer,
Silver Graphics, Inc.
2044 Scovill Road,
Mountain View, CA 94043
Dr. Tosiyasu L. Kunii,
Department of Information Science,
Faculty of Science,
The University of Tokyo,
7-3-1 Hongo, Bunkyo-ku, Tokyo 113,
Japan

Dr. Kazuhiro Fuchi,
Institute for New Generation Computer Technology,
Mita-Kokusai Building 21FL,
1-4-28 Mita, Minato-ku, Tokyo 108, Japan

Tony Loeb,
Silicon Graphics, Inc.
1901 Avenue of the Stars,
Suite 1774,
Los Angeles. CA 90067

Kevin Hammons,
NASA AMES-Dryden Flight Research Facility,
PO Box 273,
Mail Stop OFI,
Edwards. California 93523

Sherman Gee,
Code 221,
Office of Naval Technology,
800 N. Quincy St.
Arlington. VA 22217

Dr. J.A. Adams,
Department of Mechanical Engineering,
US Naval Academy,
Annapolis. MD 21402

Dr. David F. Rogers,
Dept. of Aerospace Engineering,
US Naval Academy,
Annapolis. MD 21402

Dr. Robert F. Franklin,
Environmental Research Institute of Michigan,
PO Box 8618,
Ann Arbor, MI 48107

LT Mark W. Hartong,
900 Cambridge Dr 17,
Benicia. CA 94510

Capt. Mike Gaddis,
DCA/JDSSC/C720,
1860 Wiehle Ave
Reston. VA 22090
Lt. Cdr. Patrick G. Hogan, USN
102 Borden Avenue,
Wilmington, North Carolina 28403

Dr. Edwin Catmull,
LucasFilm,
PO Box 2009,
San Rafael, CA 94912

Dr. John Beatty,
Computer Science Department,
University of Waterloo,
Waterloo, Ontario,
Canada N2L 3G1

Dr. James Foley,
George Washington University,
Dept. of Electrical Engineering and Computer Science,
Washington, D.C. 20052

Dr. Donald Greenberg,
Cornell University,
Program of Computer Graphics,
Ithaca, NY 14853

Dr. David Gries,
Cornell University,
Computer Science Department,
405 Upson Hall,
Ithaca, NY 14853

Dr. Leo J. Guibas,
Systems Research Center,
Digital Equipment Corporation,
130 Lytton Avenue,
Palo Alto, CA 94301

Dr. S. Ganapathy,
Ultrasonic Imaging Laboratory,
Dept. of Electrical and Computer Engineering,
University of Michigan,
Ann Arbor, MI 48109

Dr. Hank Christiansen,
Brigham Young University,
Dept. of Civil Engineering,
368 Clyde Bldg,
Provo, Utah 84602

Dr. Thomas A. DeFanti,
Dept. of Electrical Engineering & Computer Science,
University of Illinois at Chicago,
Box 4348,
Chicago, IL 60680
Dr. Lansing Hatfield,
Lawrence Livermore National Laboratory,
7000 East Avenue,
PO Box 5504, L-156.
Livermore, CA 94550

El Wells,
Naval Ocean Systems Center,
Code 443.
San Diego, California 92152

Dr. Al Zied,
Naval Ocean Systems Center,
Code 443.
San Diego, California 92152

Dr. Glen R. Allgaier,
Naval Ocean Systems Center,
Code 9302.
San Diego, California 92152

Richard L. desJardins,
Defense Advanced Research Projects Agency/IPTO,
1400 Wilson Boulevard,
Arlington, VA 22209

Zsuzsa Molnar,
Silicon Graphics, Inc.
2011 Stierlin Road,
Mountain View, CA 94043

Robert Comperini,
NASA ADFRF,
PO Box 273,
Datamax.
Edwards, California 93523

Shohei Tomita,
Hitachi Software Engineering Co., Ltd.
6-81. Onoe-Machi.
Naka-Ku, Yokohama 231, Japan

Tomo Yamada,
Digital Computer Limited.
No. 25 Kowa Building 8-7,
Sanbancho, Chiyoda-Ku,
Tokyo 102, Japan

Tohru Gotoh,
Digital Computer Limited.
No. 25 Kowa Building 8-7,
Sanbancho, Chiyoda-Ku.
Tokyo 102, Japan
Hiroyuki Kurihara,
Digital Computer Limited,
No. 25 Kowa Building 8-7,
Sannai-cho, Chiyoda-Ku,
Tokyo 102, Japan

Koichi Murakami,
Mitsubishi Heavy Industries, Ltd.
CAD/CAM Engineering Section,
Systems Engineering Department,
1-1 Yonemae, Wadasaki-cho,
Hyogo-ku, Kobe 652, Japan

Takashi Obata,
Mitsubishi Heavy Industries, Ltd.
CAD/CAM Engineering Section,
Systems Engineering Department,
1-1 Yonemae, Wadasaki-cho,
Hyogo-ku, Kobe 652, Japan

Dr. Arthur L. Kashmer,
Ph. D. CBE.
Computing Research Laboratory,
New Mexico State University,
Las Cruces, New Mexico 88003

Iwao Ohmachi,
R&D Development Laboratory,
Mitsubishi Corp.,
Osaka, Asao-ku,
Osaka Pref., 554 Japan

Kai Denson,
Computer Aided Engineering Program,
Naval Weapons Center,
China Lake,
China Lake, California 92555

Frank Ayers,
MicroGraphics, Inc.
701 Miralea Road,
Mountain View, CA 94043

Wong Sing,
2245 North Village,
Wong Amruam Rd.
Bangkok 10210 Thailand

Don Bucher
Fixtite Equipment Corporation,
21 Augustine Drive
New Cairo, California 95054
M. Creon Levit,  
NASA, Ames Research Center  
Mail Stop: 233-1  
Moffett Field, California 94035

Dr. Velvin R. Watson,  
NASA, Ames Research Center  
Mail Stop: 202A-14  
Moffett Field, California 94035

Phyllis F. Flynn,  
Trancept Systems, Inc.  
521F Uwharrie Ct.  
Raleigh, North Carolina 27606-1456

Mr. Zesheng Tang,  
Palo Alto Research Center,  
XEROX Corporation,  
3333 Coyote Hill Road.  
Palo Alto, California 94304

Larry Ledden,  
Hughes Aircraft,  
MS-604D216.  
PO Box 3310.  
Fullerton, California 92634

Dr. Robert Leighty,  
Research Institute (CUDE Bldg),  
U.S. Army Engineer Topographic Laboratory,  
Fort Belvoir, VA 22060-5546

Dr. Olin Mintzer,  
Research Institute (CUDE Bldg),  
U.S. Army Engineer Topographic Laboratory,  
Fort Belvoir, VA 22060-5546

Mr. Russell Davis,  
HQ. USACDEC,  
Attention: ATEC-IM.  
Fort Ord, California 93941

Capt. Roger K. Diehl,  
1105 Richmond Drive,  
Stafford, VA 22554

LT Joann M. Ammann,  
Naval Security Group Activity,  
Skaggs Island,  
Sonoma, California 95476-5000
Dr. Edward Riseman,
University of Massachusetts, Amherst
Dept. of Computer and Information Science.
Amherst, MA 01003
END

DTIC

9-86