APPLICATION OF INTERACTIVE SCENE ANALYSIS
TECHNIQUES TO CARTOGRAPHY

by

Thomas D. Garvey
Jay M. Tenenbaum

Artificial Intelligence Center

APPLICATION OF INTERACTIVE SCENE ANALYSIS
TECHNIQUES TO CARTOGRAPHY[1]

by

Thomas D. Garvey
Jay M. Tenenbaum
Artificial Intelligence Center
Stanford Research Institute
Menlo Park, California 94025

Summary

One of the most time-consuming and labor-intensive
steps in map production involves the delineation of
cartographic and cultural features such as lakes,
rivers, roads, and drainages in aerial photographs.
These features are usually traced manually on a digi-
tizing table in painstaking detail. This paper in-
vestigates an alternative approach, an interactive sys-
tem that eliminates the need for detailed tracing. A
human operator graphically designates a feature of in-
terest by pointing at or crudely tracing it with a dis-
play cursor. Using this input as a guide, the system
employs context-dependent, scene-analysis techniques to
extract a detailed outline of the feature. The results
are displayed so that errors can be corrected by fur-
ther interaction, for example, by tracing small sec-
tions of the boundary in detail. This interactive ap-
proach appears applicable to many other problem domains
involving large quantities of graphic or pictorial
data, which are difficult to extract in digital form
by either strictly manual or strictly automatic means.

Index Terms: Automated Cartography, Scene Analysis,
             Interactive Systems

Introduction

The production of maps from aerial photographic
data is a process that, despite a large body of mechan-
ical techniques, is primarily labor-intensive. One of
the most time-consuming steps in this process is the
delineation of topographic, cultural, and land-use fea-
tures, such as lakes, rivers, roads, and drainages.
Currently, a trained operator must manually trace the
detailed boundaries of features, a lengthy process.
Similar problems also occur in the process of digitiz-
ing maps for later updating.

In such a labor-intensive craft, it is reasonable
to look toward computers as a possible means for elim-
inating much of the routine work. The idea of a fully
automatic, aerial photograph-to-map computer system,
while appealing, is not only unfeasible at present but
is likely to remain so for the foreseeable future. A
more promising approach would be to develop an inter-
active system whereby an operator could quickly indi-
cate gross features and have the system fill in fine
details. The system would relieve the operator of the

need to attend continuously to many details and allow
him, instead, to simply guide the processing. Any
errors made by the system could then be corrected in-
teractively. The feasibility of such an interactive
approach has been successfully demonstrated at the
Stanford Research Institute (SRI) using the Interactive
Scene Interpretation System (ISIS).[1]

Example

The following scenario illustrates how a user and
interactive system might work together on a typical
cartographic task, extracting an outline of the large
lake in Figure 1.[*] Human input will be shown by thick
white lines and the computer's response by thin ones.
In Figure 2, the user has designated an area of interest
that is then displayed at a magnified scale. In Fig-
ure 3, a crude triangular region is drawn by the user
to indicate roughly the center of the lake. The com-
puter's initial guess, shown in Figure 4, contains both
errors of omission (samples excluded along the periph-
ery), and of commission (unwanted tail in lower left-
hand corner). The operator crudely encircles the tail
(Figure 5) and tells the computer to omit all points in
the enclosed region. He also points at several omis-
sions (the crosses in Figure 5). The computer responds
with the boundary shown in Figure 6.

Method of Approach

The examples and counterexamples of lake were used
to develop and debug interactively a computer procedure
for distinguishing between pixels (picture elements)
from the lake and those from the shore. The resulting
procedure was then used by a conventional boundary-
following algorithm to extract the lake outline.

This algorithm first detects the lake boundary by
scanning outwards from the center of the designated
triangle until the discrimination procedure classifies
a pixel as "nonlake." It then follows the boundary in
a counterclockwise direction. The next boundary point
is determined by applying the discrimination procedure
to the pixel immediately to the right of the present
boundary element and then testing pixels in a counter-
clockwise arc about the present element until a "lake"
classification is encountered.

The interesting part of this work concerns the
methodology used to develop the discrimination procedure.

[*] Figure 1 is an orthophoto of Fort Sill, Oklahoma,
coarsely digitized at 256 × 256 resolution. A coarse
digitization was used to speed processing for this
example.

1

The objective is to construct the simplest procedure, using all available feature extraction operators, for distinguishing example points from counterexample points. Table 1 lists typical feature extraction operators, ordered by computational complexity. Details of these and other operators can be found in standard texts on scene analysis.[2],[3] There is a hierarchy of graphic interaction (pointing) modes, as Table 2 indicates, by which the machine can be shown examples. From a single sample pixel, it is possible

Table 1

TYPICAL OPERATORS

- Point Operators (applied to individual pixels)
    Brightness
    Color (hue and saturation)
    Elevation

- Local Area Operators (applied to sets of contiguous pixels in small circular or oblong areas)
    Average of Attribute Values
    Distribution of Attribute Values
    Weighted Averages (templates)

- Region Operators (applied to clusters of contiguous pixels)
    Texture Over Regions
    Shape of Regions
    Size of Regions

Table 2

GRAPHICAL INTERACTION (POINTING) MODES FOR
DESIGNATING EXAMPLES AND COUNTEREXAMPLES

- Single Points

- Small Regions
    Inside
    Outside

- Crude Outline
    Inscribed
    Circumscribed

- Detailed Outline
    Segments
    Complete

to construct a program that accepts contiguous pixels whose point attributes (i.e., brightness, hue, or elevation, if available) differ from the indicated pixel by less than a threshold. An implicit inference is being made here that the rest of the pixels on the feature resemble this single pixel. Given an example region, the thresholds can be widened to encompass the range of attributes measured on that region. Counterexamples can then be used to narrow these limits. In general, the more complete the example, the less iteration will be required to develop a good specification. If an example and counterexample cannot be distinguished on the basis of thresholded point attributes, averages or distributions of attribute value over local areas can be used. If this still is not sufficient, an attempt can be made to distinguish between the two on

the basis of the size and shape of the regions delineated by outlining. The final procedure will be composed of conjunctions and disjunctions of these criteria.

Now, let us examine in detail the interactive process by which the lake-outlining procedure illustrated above was developed. The sampled, digitized image (Figure 1) was read into ISIS and displayed on a RAMTEK self-refreshing CRT. With the cursor, the user then created a small region in the interior of the lake and asked the system for a distribution of brightness values for pixels in this area. From these data he composed a one-line program that tested whether a pixel belonged to the lake based on thresholded brightness (the only available point attribute). The edge follower used the program to produce the outline shown in Figure 4.

The user next drew a crude boundary around the "bad" pixels in the tail and again requested a brightness distribution. A significant overlap with the previous distribution of example pixels was observed. Adequate discrimination was achieved empirically by increasing the operator size so that brightness of a point was computed as the average brightness over a circular area centered on the point. This crude spatial filtering acted to exclude dark areas of the image with insufficient width to qualify as lakes. Finally, the brightness threshold was widened to include the brightnesses of the missed points that the user had indicated with the cursor. Using the updated program, the edge follower was able to obtain an outline that tracked fairly accurately the actual lake.

The final procedure for distinguishing lake points from nonlake points is, in fact, a "model" for what pixels from a lake look like to the computer. The program was written on-line in an interactive language (LISP) and then debugged interactively as contingencies arose. Interactive refinement is a powerful concept for a scene-analysis programmer. It frees him from the necessity of formulating programs in a language that is understood by the machine but that is cumbersome for people. Instead, it allows direct communication with the program via a common language of images. Debugging is simplified in this system. Instead of predicting the problems that the system is likely to encounter, the program is executed on exemplary images and debugged when errors arise.

Further Examples

Automatic Extraction of Previously Learned Features

The procedure developed in tracing the first lake can serve as the initial basis for extracting other lakes in similar terrain. Even if the outline is not exact, it provides a good starting point for further interaction. Figure 7 shows a boundary extracted for the small lake using the same discrimination procedure developed for the large lake. In this example, the user manually designated a single pixel in the center of the second lake to initiate the boundary follower. Alternatively, an initial sample could have been acquired automatically, by scanning systematically through the image for a reasonably sized set of contiguous pixels satisfying the criteria for "lake." Note that any

subsequent interaction required to refine a boundary could be used to further improve or generalize the original discrimination procedure.

Linear Features

Linear features, such as rivers and roads, may also be outlined using similar interactively generated procedures. In Figure 8, the upper branch of the river connecting the two lakes is shown. Here the user pointed at a single river point just above the fork. Starting from this point and using a threshold based on its brightness, the boundary follower tracked the river until it intersected the road. The trainer next indicated additional starting points on each river branch below the road, and, using the same threshold, the river boundary was completed. The final river boundary is shown in Figure 9. These crude boundaries could be improved by applying a thinning algorithm.[5] Figures 10 and 11 show the final results of tracing the designated features and then projecting them back onto the original, high-resolution image.

Possible Extensions

Automatic Generation of Discrimination Procedures

The above examples required that the user supply discrimination procedures for distinguishing between the brightness distributions of designated regions. These procedures were interactively formulated using data provided by the system. An obvious next step would be to have these procedures formulated automatically by the system based on the user-designated examples and counterexamples. In this mode of operation, a user might crudely sketch a feature of interest. The system would use this to formulate a discrimination procedure and then attempt to trace a detailed boundary. Errors would be handled interactively, either by manually editing part of the boundary or by providing the system with additional examples and counterexamples with which to refine the discrimination procedure.

For simple discrimination procedures of the type described in this paper, automatic generation appears straightforward. Existing ISIS subroutines could be used, for example, to select the appropriate threshold and operator size for distinguishing the brightness distributions of example and counterexample points.[4] The same approach should be applicable with the other operators in Table 1 when additional discrimination is required.

Elevation Data

The availability of elevation data would make many of the tasks described above much simpler. The constant elevation of a lake, combined with local brightness values, would provide a powerful discriminating test. And, in many cases where the brightness contrast between two features is poor, a difference in slope or elevation may be sufficient to distinguish them. Similarly, features in mountainous terrain would prove more tractable with elevation data.

Digitization of Existing Maps

The same techniques used to trace features on aerial photos would also be useful for tracing features on existing maps to reduce them to digital form. In many cases, the processing should, in fact, be easier, because of the better contrast available in maps. These digitized maps could then be updated interactively using recent aerial photographs. Ultimately, information in existing digitized maps could be used in lieu of pointing to indicate preexistent features on the photograph. This would allow the program to use the digitized map to guide the subsequent analysis of the aerial photograph, in the same way as would a person.

Elimination of Map Editing

The process described should eliminate the need for an independent editing step after the map features have been extracted. The editing is an inherent part of the process of incremental refinement of the outline and, therefore, should not be normally needed as a post-processing step.

Conclusions

We believe that the examples described above demonstrate the technical feasibility of applying interactive, scene-analysis techniques to cartography. Whether or not the techniques developed will prove practical in actual cartographic use is, of course, a matter for further study. The simple feature extraction operators used (essentially a threshold applied to the average brightness computed over a bar-shaped operator) almost certainly will not suffice in more complex aerial scenes. Moreover, processing times may become a key factor at the image resolutions required for cartographic accuracy. An appealing aspect of the interactive approach is that, when necessary, the user can always revert to detailed manual tracing. Thus, the approach would be useful even if it applied in only some of the cases encountered in practice.

In the future, we plan to apply interactive techniques in a variety of other problem domains involving large volumes of graphic and pictorial data that are difficult to extract in digital form by either strictly manual or automatic means.

3

References

1. J. M. Tenenbaum, T. D. Garvey, S. A. Weyl, and
H. C. Wolf, "ISIS: An Interactive Facility for Scene
Analysis Research," Proceedings Second International
Joint Conference on Pattern Recognition, Copenhagen,
Denmark (August 1974).

2. Azriel Rosenfeld, Picture Processing by Computer
(Academic Press, New York, New York, 1969).

3. R. O. Duda and P. E. Hart, Pattern Classification
and Scene Analysis (Wiley-Interscience, New York,
New York, 1973).

4. T. D. Garvey, "Perceptual Strategies for Purposive
Vision," SRI Artificial Intelligence Center
Technical Note 117, Stanford Research Institute, Menlo
Park, California (February 1976).

5. J.M.S. Prewitt, "Object Enhancement and Extrac-
tion," Picture Processing and Psychopictorics, Lipkin
and Rosenfeld ed. (p. 120) (Academic Press, New York,
New York, 1970).

FIGURE 1   DIGITIZED AERIAL VIEW OF
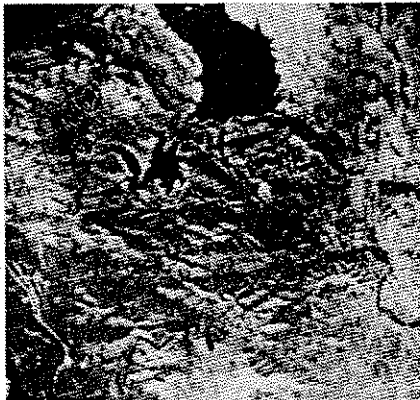   FORT SILL, OKLAHOMA
            (8 BITS AT 256 x 256
            RESOLUTION)

FIGURE 2   WINDOWING TO OBTAIN
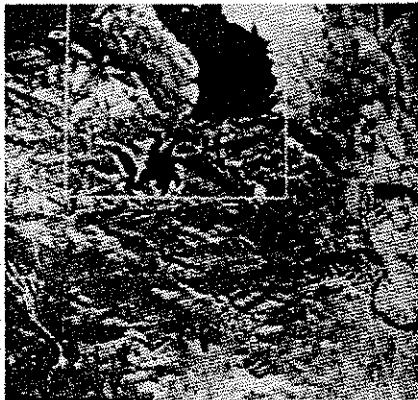   MAGNIFIED DISPLAY
            OF WORK AREA
                        SA-4683-28

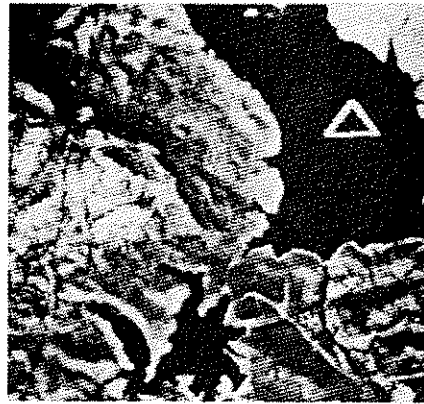FIGURE 3   USER MANUALLY DESIGNATES
            A FEW IMAGE POINTS
            CONTAINED IN LARGE LAKE

FIGURE 4   INITIAL BOUNDARY
            WITH DEFECTS
                        SA-4683-29

FIGURE 5   USER INDICATES ERRORS

4

FIGURE 6   FINAL BOUNDARY OF LARGE
           LAKE AFTER UPDATING MODEL
                              SA-4683-30

FIGURE 7   FINAL BOUNDARY OF
           SMALL LAKE

FIGURE 8   OUTLINE OF RIVER AFTER
           DESIGNATING ONE POINT IN
           THE UPPER BRANCH
                              SA-4683-31

FIGURE 9   OUTLINE OF RIVER AFTER
           DESIGNATING AN ADDITIONAL
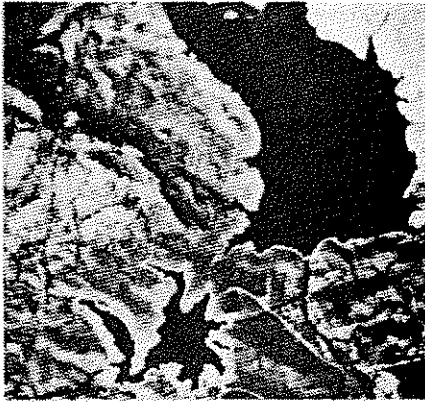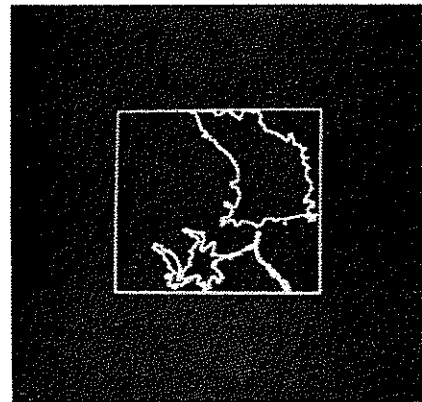   POINT IN EACH LOWER
           BRANCH

FIGURE 10 COMPLETED MAP OF MAJOR
           WATERWAYS WITHIN WINDOW
                              SA-4683-32
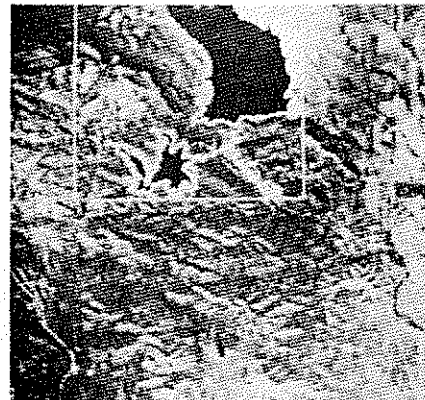

FIGURE 11 COMPLETED MAP
           SUPERIMPOSED ON
           ORIGINAL IMAGE
                              SA-4683-33

5