"WYSIWYG" MAP DIGITIZING: REAL TIME GEOMETRIC CORRECTION AND TOPOLOGICAL ENCODING

Denis White and Jonathan Corson-Rikert
Laboratory for Computer Graphics and Spatial Analysis
Graduate School of Design, Harvard University
48 Quincy St., Cambridge, MA 02138
(617) 495-2526

Margaret Maizel
American Farmland Trust
1920 N St., Suite 400, N.W.
Washington, D.C. 20036
(202) 659-5170

ABSTRACT

Map input by manual digitizing no longer needs to be a multi-step process in which line gaps, overshoots, and topological coding errors are iteratively and painstakingly corrected. We have developed a "what you see is what you get" (WYSIWYG) approach to map digitizing that continuously displays on the computer screen a geometrically corrected and topologically structured representation of a map. This approach is analogous to the WYSIWYG style of word processing where insertions and deletions automatically cause lines, paragraphs, and pages to be adjusted such that a document is always displayed in its final form.

INTRODUCTION

Instruction in landscape architecture at Harvard includes courses and studios in regional scale landscape planning and design. For many years this instruction has included computer analysis of landscape suitability and environmental impacts. A major component of these projects is the building of the geographic data base. In the past, students have laboriously hand encoded data for soil types, vegetation types, elevation values, water features, cultural features, and other kinds of information into a grid cell format for use by the computer analysis programs.

In recent years the Laboratory for Computer Graphics and Spatial Analysis has assisted these courses and studios by developing software and related procedures for digitizing, verifying, and grid encoding large area data bases. These procedures and programs have been used in studies of Yosemite National Park, Minute Man National Historical Park, White Mountain National Forest, and Acadia National Park.

Our experiences have refined our thinking about a number of issues in map input to computers (Corson-Rikert and White, 1985a). With the support of the American Farmland Trust we are continuing to develop software to further improve this process. There are a number of key features of this software.

"WHAT YOU SEE IS WHAT YOU GET"

When our Lab's Odyssey system was designed ten or more years ago (White, 1979; Chrisman, 1979; Morehouse and Broekhuysen, 1982), there was virtually no moderately priced hardware that supported rapidly refreshed medium resolution interactive graphic display. A typical digitizing configuration was a digitizing table and a storage tube connected to a mainframe, or a storage tube micro standing alone with a digitizing table. In 1986, however, there is now no hardware limitation to immediately displaying points digitized, or even displaying points about to be digitized with so-called rubberbanding. CAD programs have been doing this for years; map digitizing programs should do likewise where appropriate.

With high speed refreshing, edited changes to a computer map data base can also be displayed immediately. Lines can be deleted, points can be inserted into a line or deleted from a line, nodal points where lines meet can be moved with their connected segments, and even entire lines can be rubberbanded into a translated position. Rapid display speed also allows arbitrarily positioned rectangular windows into the data base to be selected and displayed immediately.

GEOMETRIC CORRECTION

The speed of modern processors also allows for rubberbanding of coordinates registered to a map base with a global linear transformation. Piecewise linear rubbersheeting could certainly be accomodated as well for rubberbanding. Editing subsequent to initial map input from a digitizing table can often be done better with a mouse, but it should be easy to switch between mouse and table. Mouse editing normally occurs in map space without control point registration.

Geometric snapping of incoming points to existing points is another feature of the CAD environment quite suitable for map digitizing. In particular, this capability helps avoid the perennial plague of line overshoots and undershoots. Snapping in CAD is often enhanced by a grid of markers showing the points to which all input will be snapped. In map digitizing it is more appropriate to accept an input point exactly as digitized unless it falls within the tolerance range of a previously entered point, in which case it is merged with the older point. As a visual aid, points can be displayed with tolerance circles such that it is immediately clear whether a rubberbanded point will merge or not.

In fact, the snapping tolerance applies to a line as well. If an incoming point lies within the tolerance distance of a line segment (between the endpoints), an intersection should be formed and the old segment broken into two collinear segments. It is also possible, though potentially a strain on computational resources, to compute all intersections of a newly entered line with a number of older lines it crosses and display the new structure relatively quickly.

The rank ordering of the positional accuracy of features commonly found on maps is well supported in the computer data base by this method of map input. Features are entered in the order of most accurate to least accurate such that the latter are always snapped to the former when they are in close proximity.

TOPOLOGICAL ENCODING

Topological structuring of line and polygon networks on maps is now the standardly accepted method for insuring consistency and completeness in a computer map data base. The details of the dual incidence technique of encoding topology can be computed immediately and transparently while digitizing is taking place. In practice the construction of 2-cell topology in real time requires considerable extra work as new polygons are created and old ones destroyed by the entering of new lines. However, no additional steps need be required if polygons are to be labeled or tagged, since the 2-cell topology can be created during that process.

Successful implementation of instantaneous node topology maintenance is dependent on the correct operation of snapping within a tolerance and intersection finding. In this way, the need for an analysis of proximate points before creating final nodes of intersection is eliminated. (The overlay analysis of two or more polygon networks will require this analysis, however.)

FEATURE LABELING

The "spaghetti and meatballs" method of polygon labeling implemented in Odyssey was a major improvement over the method of labeling the left and right side polygons of each line. In interactive feature labeling, entering "meatballs" (or centroids or label points) is still useful since the points can be saved as text or symbol locations. A method requiring less input, however, is to highlight each feature in succession and prompt for its label (or tag). This process can be driven automatically, with the order in which features are presented determined by input order or perhaps more intelligently by a spatially sorted order. Either this semi-automatic naming method or the pick and name method can be used with point, line, or areal features.

CONTEXT

Often one layer or theme of a multi-layered map data base can be useful when digitizing another, particularly when features on separate layers partially or fully coincide. Displaying existing layers while entering a new one, another technique borrowed from CAD, is straightforward. Feature snapping or alignment from one layer to another is also desirable but less easy to implement.

When the eventual use of a computer map data base is for raster-based analysis (terrain analysis, viewsheds, path finding), visualizing the raster structure in the context of the vector data base helps determine the appropriate density of information to capture in vector form.

SEARCH OPTIMIZATION

Searching for the proximity of a new point in a very large data base can take a very long time unless some kind of spatial indexing or hierarchical structuring of the data base is used. One simple method is to limit searches to features within the current window. This method is consistent with the WYSIWYG philosophy; windows will tend to have a relatively constant density of information and be centered about the work at hand. Of course, computing the active list of features for a small window in a very large data base will be time consuming in itself.

IMPLEMENTATION

The digitizing strategies argued here have been implemented on three computers: initially and only partially in Pascal on an IBM PC/AT with a PGA graphics board, then translated to C on an Apple Macintosh, and finally in C on a Harris MCX/3 running Unix. The program, called Roots, is being used to create a data base for Clarke County, Virginia, under the sponsorship of the American Farmland Trust. The data base will include soils and topography (input of which will be commercially scanned), transportation features, hydrographic features, wells and sinkholes, and property boundaries and derivatives thereof such as zoning and farms. Most analysis of the data base will be done by the Geographical Resources Analysis and Support System (GRASS) developed by the Construction Engineering Research Laboratory of the U.S. Army Corps of Engineers.

CONCLUSION

For many years a major impediment to mapping and spatial analysis using computerized geographic data bases has been the exorbitant time and cost of entering map data into the computer. With the development of lower cost graphics hardware, integrated graphics operating systems on microcomputers, and improved algorithms for geometrical and topological processing of map data, this impediment is gradually being removed.

ACKNOWLEDGEMENTS

Portions of this work have been supported by the National Park Service and by IBM Corporation. Our work is continuing under the sponsorship of the American Farmland Trust, for whom we are helping to develop software and data bases to assist local government decision makers in natural resource conservation policies.

REFERENCES

Chrisman, N. 1979. "A Many Dimensional Projection of Odyssey", Laboratory for Computer Graphics and Spatial Analysis, Graduate School of Design, Harvard University.

Corson-Rikert, J. and White, D. 1986. "Comprehensive Map Input Software", *Proceedings, Urban and Regional Information Systems Association, 1986 Annual Conference.*

Corson-Rikert, J. and White, D. 1985a. "Issues in Map Digitizing", LCGSA Report 85-18, Laboratory for Computer Graphics and Spatial Analysis, Graduate School of Design, Harvard University.

Corson-Rikert, J. and White, D. 1985b. "The TRACE Program for Map Input", LCGSA Report 85-17, Laboratory for Computer Graphics and Spatial Analysis, Graduate School of Design, Harvard University.

Morehouse, S. and Broekhuysen, M. 1982. *Odyssey User's Manual,* Laboratory for Computer Graphics and Spatial Analysis, Graduate School of Design, Harvard University.

White, D. 1979. "Odyssey Design Structure", *Harvard Library of Computer Graphics, 1979 Mapping Collection,* Vol. 2, pp. 207-215.