SIMPLE TOPOLOGY GENERATION FROM SCANNED MAPS.

Dr. Christopher Gold, Geomatics Research Centre,
Laval University, Quebec City, Qc, Canada G1K 7P4
(418)656-3308, Christopher.Gold@scg.ulaval.ca

ABSTRACT

For many GIS applications the data entry component is the most expensive, and frequently makes the difference between success and failure of the project, in terms of both time and money. The most time-consuming part of manual digitizing is usually the generation of correct topology, which usually involves many error correction steps. The automatic processing of scanned maps appears to have many advantages, but in practice the line extraction process does not always produce good topology automatically.

The situation with scanned maps may be improved with a conceptual change of emphasis. Instead of concentrating on the black pixels (linework) one may emphasize the white pixels (polygonal areas) and build the relationships from these. The process has three steps. Firstly a set of white pixels are selected, at a user-specified distance from the black linework, and these are given a polygon label on the basis of a flood-fill algorithm that scans all connected white pixels. Secondly these selected pixels are used as data points to generate the standard Euclidean point Voronoi diagram. Thirdly, this structure is scanned to extract only those Voronoi boundaries between pixels having different polygon labels. The result of this operation is a set of vector chains or arcs that are guaranteed to separate regions of white space, and are guaranteed to connect precisely at nodes.

Experiments were made on two types of map: a simple urban cadastral map, with lot boundaries and building outlines; and a typical Quebec forest map sheet that had been retraced by hand to preserve the forest stands and eliminate all the other superimposed data types. The cadastral map showed excellent automatic topology generation, and accurate linework. Black linework or symbols that are unclosed are not preserved with this method. Unclosed polygons are also lost, and may need pretreatment. In the case of the full forest map, containing about 3000 polygons, manual editing might be required if stand boundaries are too close together - for example along river valleys. Processing time on a small Sparcstation 10 was less than 30 minutes at maximum resolution. The most difficult part of the process was georeferencing the scanned image to the paper map or ground coordinates. A byproduct of the process was the automatic detection of polygon centroids - in this case defined as the centre of the largest circle falling in the polygon. The results were imported directly into the vector GIS without any cleanup or intersection- testing operations. In many cases the approach described here may make significant savings in the map entry process.

INTRODUCTION

The use of maps for planning and analysis depends heavily on their availability in digital form within the computer. This is usually a major bottleneck. In particular, the demands of a GIS include the topological structuring of map linework to form networks and polygons. This requires careful manual digitizing in order to produce a suitably structured map which is then usable for analysis. This is a long and error-prone operation, and is usually the major cost. This bottleneck in map input holds true for any application using a vector GIS. The underlying problem is the difficulty of taking more-or-less precise coordinates and converting them into a structured graph representation within the computer. This is difficult for a variety of reasons - the imprecision of coordinates and the traditional use of a line-intersection model of space, among others - but one of the basic issues is the formal definition (in advance) of what are the classes of objects being detected during the input process. Our experience has shown that, if this is clearly understood, the combination of appropriate object labelling with a general-purpose implementation of a spatial model may permit relatively simple data input. In particular, one should be clear as to whether one is attempting to define line (arc) or area (polygon) features.

Some work has been undertaken to attempt to reduce this bottleneck. Gold et al. (1996) worked on the problem of improving the speed of forest map digitizing by emphasizing the specification of the polygons themselves, rather than the bounding arcs or chains. This has been shown to improve map preparation times significantly at the operational level. It was, nevertheless, a manual method.

An alternative approach is to scan the map, and then to process the resulting (black and white) image. Various commercial products exist for the "skeletization" or thinning of the pixels forming a line. These approaches have, however, run into difficulties with the extraction of good topology - it is difficult to produce a satisfactory vector skeleton that forms a complete set of polygon boundaries. Nevertheless, an automated technique would be a great help in map input.

Much experience has shown that the traditional data input model for GIS is cumbersome and error-prone. This approach could be called the "line-intersection" model of space, since the manual digitizing method involves entering individual chains of x-y coordinates, and the computer program first searches for intersections between these chains. Once intersections are found the system attempts to construct a graph model of the desired map - most often a polygon map. Many errors are possible at this stage, usually related to missing or duplicate intersections, causing difficulties in identifying nodes and completed polygons. Thus all spatial relationships are based on detecting intersections of lines or chains.

ALTERNATIVES TO THE LINE-INTERSECTION MODEL

There are alternatives to the line-intersection model of space. The most obvious is the raster or grid model. Here there are no particular map objects, but only an attribute associated with a particular square tile. This has the advantage that neighbour relationships are implicit in the whole structure (the tiles to the north, south, east and west), but it is rather awkward for the identification of specific map objects, such as roads or polygons. A third approach, which has been used with some success in recent years, is to combine the advantages of both systems: a set of map objects (as in a vector GIS) with a single associated tile (rather like a raster cell). This would give a set of spatial adjacency relationships between adjacent tiles - and hence between each tile's generating object. While various definitions of these tiles may be possible, the most obvious is the proximal definition used to generate the Voronoi diagram. Here each tile or cell contains all spatial locations closer to the generating object (traditionally a data point) than to any other object.

Various algorithms exist for generating this spatial data structure (or its dual, the Delaunay triangulation) for static sets of data points. Examples include Green and Sibson (1978), Guibas and Stolfi (1985), Sugihara and Iri(1989) and Lawson (1977). Extensions exist for constrained triangulations (Lee and Lin, 1986), generators that may be points or line segments (Lee and Drysdale, 1981, Fortune, 1987) and dynamic systems where objects may be added, deleted, or moved (Roos, 1993, Gold, 1991). What is of particular interest here, though, is that the algorithms give a form of automatic topology (Gold, 1994). Aurenhammer, (1991) gives a good review. Even with the algorithms for the static Voronoi diagrams of points, which will be adequate for this paper, the "topological" structure is built automatically, with reasonable levels of robustness. It is an attractive idea to attempt to take this property and to apply it to various GIS data entry problems.

AUTOMATIC TOPOLOGY GENERATION - MANUAL DIGITIZING

In Gold et al. (1996) the forest map problem was attacked by focusing on the primary objects of interest - the forest stands, rather than the boundaries. One or more generating points were digitized within each stand, and given the stand label. It was hoped that the cells would approximate the extent of each stand boundary. Further experimentation showed that the best approximation to the stand occurred when points were digitized closely around the interior of each polygon (Fig. 1a), the Voronoi cells generated (Fig. 1b) and then boundaries between those cells having the same label were suppressed. This gave a good approximation to the boundaries between stands.

The dashed lines are the original boundaries in Fig. 1c and the solid lines are the boundaries extracted from the Voronoi diagram. This digitizing was done

Fig. 1. a) Points digitized around the interior of each polygon;
b) the Voronoi cells generated;
c) the polygon boundaries generated (solid lines), compared with the original map (dashed lines).

manually, as the operator was able to distinguish between stand boundaries and the other information on the paper map. The approach was much more intuitive, as the operator focused on the objects of interest - the sequentially-labelled polygons - and not on the boundaries themselves. Operators could be trained rapidly, and digitizing time greatly reduced. The boundaries thus produced had errors well within the limits of the photo-interpretation used to generate the original paper maps.

The algorithm used was based on an early visibility-ordering approach for triangulations (Gold and Maydell, 1978). This guaranteed that triangles would be processed in a front-to-back order. This was modified to draw the Voronoi cell boundaries, suppressing those boundaries between vertices having the same label. Linked-lists were maintained, as in Gold and Cormack(1987), to preserve all complete arcs between triple-junctions (nodes). A single pass through the triangulation, with no searching, was all that was required to extract the complete polygon-arc-node topology for direct entry into a traditional vector GIS. Valid polygon centroids were generated automatically. It was guaranteed that the coordinates of arc end-points matched each other to form nodes, and that the result would always be a valid topological structure. Operator errors were confined to mis-labelling items or forgetting to digitize the interior of some polygon, and these could easily be detected, corrected, and the map re-built. For further details see Gold et al. (1996).

AUTOMATIC TOPOLOGY GENERATION - SCANNED MAPS

The next question concerned the possibility of eliminating the manual digitizing process entirely while preserving the automatic topology generation. The manual process was simulated by the use of functions similar to mathematical morphology (Serra, 1982). In mathematical morphology, binary images are processed using two operators: erosion (shrinkage) and dilation (expansion). The objective of the experiment was to take scanned polygon maps (either forest or urban) and use image processing techniques to generate a fringe of points around the black pixels. These points would then be entered into the Voronoi diagram and have the relevant boundaries extracted as in the above manual digitizing case.

URBAN MAPPING - SYMBOL EXTRACTION

In the case of urban mapping, perhaps the most interesting work is that done by Burge and Monagan (1995), as it is similar to the forest mapping project mentioned above in that it is based on Voronoi diagrams. It differs in that they have been developing a method for extracting features from scanned cadastral maps, with emphasis on the extraction of symbols, dashed lines and character strings. In summary, their procedure generates labelled points associated with connected sets of "black" pixels. These are inserted into the Voronoi diagram, and edges are removed between points with the same label, giving an "area Voronoi diagram" with a cell around each image element. Based on this, sets of dots, dashes or symbols are grouped together. While not discussed much in their papers, the pixels forming the linework of the cadastral maps are also grouped to form line image elements, but no attempt was made to structure the lot boundaries, for example, in the sense of GIS topology - indeed, all connected polygon boundaries will have the same label. The authors suggest that the identification and removal of symbol groups will facilitate the vectorization of the linework by other algorithms.

SCANNED MAPS - TOPOLOGY EXTRACTION

The work described in this paper attempts to concentrate on linework topology, based on the approach previously used on forest maps. Like the work of Burge and Monagan, labelled points are inserted into the simple Euclidean point Voronoi diagram. Again, boundaries are extracted from this diagram and saved only if they are between points with different labels (using the algorithms developed in Gold and Maydell, (1978), Gold and Cormack, (1987) and Gold et al., (1996)). In both projects the Voronoi diagram (or Delaunay triangulation) construction may be performed on O(n log n) time, and boundary extraction in O(n) time - although the boundary extraction algorithm of Gold et al. appears to be simpler. Unlike Burge and Monagan, however, the symbols themselves were not of interest. The primary bottleneck in GIS is the extraction of "topology" from manually digitized or scanned maps, and the emphasis on collecting all individual arcs forming the polygon boundaries and then connecting them together imposes a heavy workload on computer and operator. Thus our main interest was to extract complete polygons rather than symbols.

The forest mapping project, with its processing of digitized points inside each polygon, led to an emphasis on polygon detection rather than the identification of connected "black" pixels once we started processing scanned maps. In the manual digitizing project, points at the edge of each polygon were given the polygon label, and the Voronoi and boundary extraction functions selected arcs between differently labelled polygon points. An obvious extension was to attempt to process scanned maps in the same fashion, thus removing the need for manual digitizing. A set of "fringe" points were generated at a distance D from any black pixels in the scanned image, and then thinned to be a distance S apart. In practice we were able to set D to one or two pixels, and S to the same value. A standard flood-fill algorithm was then used to assign the same polygon label to all fringe points within the same connected white-space region.

These label points are then entered into the Voronoi diagram and boundary extraction modules, which generate the Voronoi diagram in Euclidean (as opposed to raster) metric. By the nature of the Voronoi diagram, the polygon topology is always complete, because all points with the same label whose Voronoi cells are connected will have a boundary around them. Any black pixels that do not connect to enclose a white region will have a fringe of label points generated, but the flood-fill algorithm, operating within each connected white region, will give them all the same label. Consequently none of their Voronoi boundaries will be extracted. Thus, using the Voronoi boundary extraction procedure in this case identifies polygonal regions, while the approach of Burge and Monagan based on connected black pixels identifies isolated symbols. In addition, the system can estimate a good centroid position for each polygon. These are calculated as the centre of the largest circumcircle of any triangle having all three vertices within the same polygon.

Fig. 2a shows a small scanned urban map, containing lot boundaries, buildings, labels, road boundaries and point symbols. The fringe label points were added to the image as described above. Fig. 2c shows the fringe label points in the southernmost portion of Fig. 2a. The Voronoi diagram was then generated, and the boundaries between differently-labelled points were extracted and exported as complete arcs. The results were imported directly into Arc/Info, and as no "clean" operation was required (because all arc ends matched precisely), the map could be viewed directly. The processing is fairly rapid, and the method is simple to implement. Fig. 2b is the final result of simple clean-up operations to remove small closed loops, (causing an error at the corner of one house which had text superimposed), and the use of the Douglas-Peucker algorithm to reduce the number of line segments for straight line boundaries. Notice that the northeastern-most lot boundary is lost, due to a small gap in the drawn boundary. (Nodes are represented by small dots.)

FOREST MAPPING

Fig. 3 shows a detail of the Quebec forest stand map generated by the same method, at 25% of the original scale. The original input was retraced to eliminate roads, contour lines, etc. that had been superimposed on the same map. Processing time for the full map was under 30 minutes on a small Sparcstation, approximately equally split between the image analysis step and the Voronoi plus edge extraction step. The map consisted of approximately 3000 polygons, and the resulting Voronoi diagram had about 500,000 points. Precision of the generated centreline is estimated to be within about one pixel. The algorithm gives a slight displacement of nodes when one arc meets another at right angles, and some editing is required where two input boundaries are extremely close or touching, causing the extracted boundaries to merge in places. The most difficult part of the process was georeferencing the scanned image to the original map or ground coordinates. The arcs and the automatically generated polygon centroids may be directly input int Arc/Info or any equivalent GIS without any need for further processing. Indeed, most forms of topological structure could be extracted directly from the Voronoi diagram as required. Polygon labels may be added interactively to the centroids (although industrial experience shows that this is a slow operation, and the manual data entry described above deserves serious consideration).

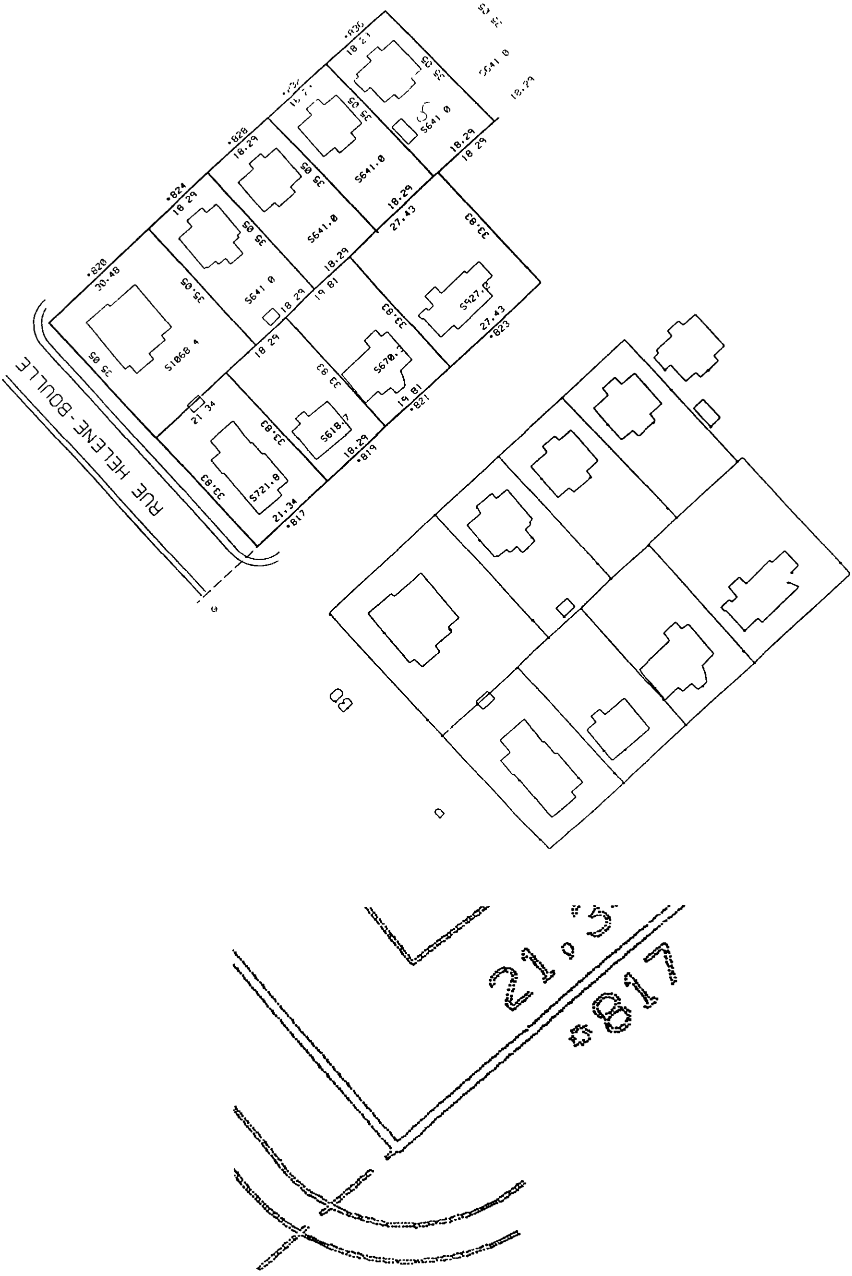

Fig. 2. a) A small cadastral map;
b) the final cleaned-up map.
c) a few of the fringe points generated in the south of the map.

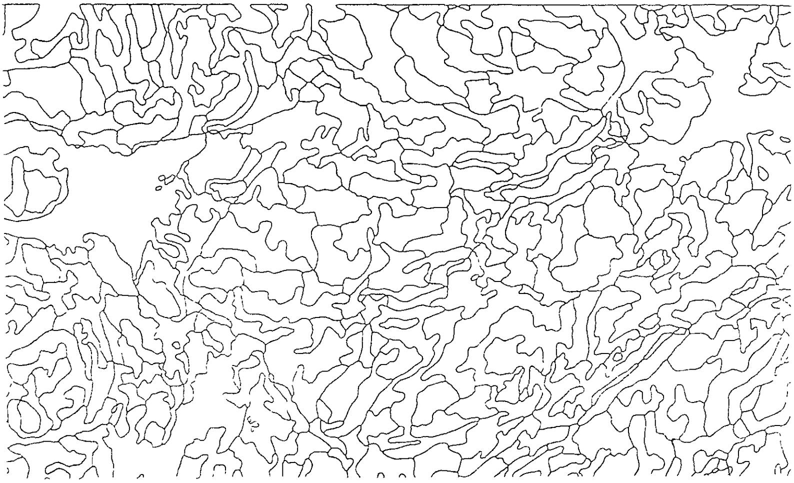

Fig. 3 Results for a portion of a Quebec forest map (at 25% scale), showing the complete polygon topology.

FUTURE WORK AND CONCLUSIONS

The automatic detection of topology within the Voronoi module opens up a variety of other developments. Further information can be extracted from the Voronoi diagram itself, for example which polygons are adjacent to, or enclosed within, other polygons. In addition, Gold et al. (1996) show that almost any GIS topological structure may be extracted from the Voronoi diagram as required. The links between mathematical morphology and the Voronoi diagram are close - dilation functions to form raster buffer zones have their equivalent in the Euclidean Voronoi diagram (Gold, 1991). Vincent (1988) has shown that mathematical morphology operations may be performed directly on the Voronoi cells, not just on raster images. These relationships need to be explored further. Even at this preliminary stage, it is clear that the Voronoi approach to the processing of scanned maps has definite advantages.

ACKNOWLEDGMENTS

This research was made possible by the foundation of an Industrial Research Chair in Geomatics at Laval University, jointly funded by the Natural Sciences and Engineering Research Council of Canada and the Association de l'Industrie Forestiere du Quebec. The author would like to gratefully acknowledge the assistance of Luc Dubois, Weiping Yang, Francois Anton and Darka Mioc with various aspects of the programming and testing.

REFERENCES

Aurenhammer, F., 1991. Voronoi diagrams: a survey of a fundamental geometric data structure. *ACM Computing Surveys* v. 23, pp. 345-405.

Burge, M. and G. Monagan, 1995. Extracting words and multi-part symbols in graphics-rich documents. In: *Eighth International Conference on Image Analysis and Processing: ICIAP. Lecture Notes in Computing Science, IAPR,* Springer Verlag, 1995.

Fortune, S., 1987. A sweepline algorithm for Voronoi diagrams. *Algorithmica,* v. 2, pp. 153-174.

Gold, C.M., 1991. Problems with handling spatial data - the Voronoi approach. *CISM Journal* v. 45, pp. 65-80.

Gold, C.M., 1994. Three approaches to automated topology, and how computational geometry helps. *Proceedings, Sixth International Symposium on Spatial Data Handling,* Edinburgh, pp. 145-158.

Gold, C.M. and S. Cormack, 1987. Spatially ordered networks and topographic reconstructions. *International Journal of Geographical Information Systems,* v. 1, no. 2, pp. 137-148.

Gold, C.M. and U.M. Maydell, 1978. Triangulation and spatial ordering in computer cartography. *Proceedings, Canadian Cartographic Association Third Annual Meeting,* Vancouver, pp. 69-81.

Gold, C.M., J. Nantel and W. Yang, 1996. Outside-in: an alternative approach to forest map digitizing. *International Journal of Geographical Information Systems* v. 10, pp. 291-310.

Green, P.J. and R. Sibson, 1978. Computing Dirichlet tessellations in the plane. *Computer Journal,* v. 21, pp. 168-173.

Guibas, L. and J. Stolfi, 1985. Primitives for the manipulation of general subdivisions and the computation of Voronoi diagrams. *ACM Transactions on Graphics,* v. 4, pp. 74-123.

Lawson, C.L., 1977. Software for C-1 surface interpolation. In: J. Rice (ed.), *Mathematical Software III.* Academic Press, New York, pp. 161-194.

Lee, D.T. and R.L. Drysdale III, 1981. Generalizations of Voronoi diagrams in the plane. *SIAM Journal of Computing,* v. 10, pp. 73-87.

Lee, D.T. and A.K. Lin, 1986. Generalized Delaunay triangulations for planar graphs. *Discrete and Computational Geometry,* v. 1, pp. 201-217.

Roos, T., 1993. Voronoi diagrams over dynamic scenes. *Discrete Applied Mathematics,* v. 43, pp. 243-259.

Serra, J., 1982. *Image analysis and Mathematical Morphology.* Academic Press, London.

Sugihara, K. and M. Iri, 1989. Two design principles of geometric algorithms in finite-precision arithmetic. *Applied Mathematical Letters,* v. 2, pp.203-206.

Vincent, L., 1988. Mathematical Morphology on graphs. *Proceedings of SPIE -The International Society for Optical Engineering,* 1988, pp. 95-105.