Contour Line Recognition From Scanned Topographic Maps

Spinello Salvatore
LaBRI - INRIA Futurs
Université Bordeaux 1
351 Cours de la Libération
33405 Talence Cedex, France
spinello@labri.fr

Pascal Guitton
LaBRI - INRIA Futurs
Université Bordeaux 1
351 Cours de la Libération
33405 Talence Cedex, France
guitton@labri.fr

ABSTRACT

In this paper we present a new method for contour lines recognition starting from scanned topographical maps. This is a difficult problem due to the presence of complex textured backgrounds and information layers overlaid on the elevation lines (e.g. grid lines, rivers, roads, buildings, etc.). Our approach uses local geometric properties to recognize the contour lines but it is substantially based on the global topology of a generic topographic map (i.e. a set of non-intersecting closed lines). Starting from a scanned map the image is first segmented by a color classification process. The resulting binary image is thinned and then vectorized using a Delaunay Triangulation where the Delaunay edges are filtered using local and global rules (i.e. *complete contours and weight matrix*).

Keywords

Single color extraction, curve reconstruction, topographic map.

1. INTRODUCTION

A topographic map is a representation of the Earth, or part of it. Traditionally, maps have been printed on paper. When a printed map is scanned, the computer file that is created may be called a digital raster graphic map. The distinctive characteristic of a topographic map is that the shape of the Earth's surface is represented by contour lines. Contours are imaginary lines that join points of equal elevation on the surface of the land above or below a reference surface such as mean sea level. Therefore, contours make it possible to measure the height of mountains, depths of the ocean bottom, and steepness of slopes. A traditional topographic map shows much more than contours. It includes symbols that represent different features such as streets, buildings, streams, and woods. These symbols, unfortunately, overlay the elevation lines and make difficult their automatic recognition and modeling. In effect, after filtering the digital raster map, it remains a merely set of points on a plane that could be considered as a sample of an existing set of curves.

If the sample is large enough and is well distributed, it is an easy task for a human being to perceive the shape of the curves. Human perception of curve shape includes not only topological aspects, such as the identification of connected components and the differentiation between open and closed pieces, but also geometrical aspects, such as qualitative measures of curvature and winding.

A computer, in contrast, has no such natural perception and in the general case of curve reconstruction the sample has no a priori structure that can be exploited to provide a computational description of the curve. To the machine the sample is merely a set of coordinates. A fundamental problem, which we would expect to solve using such computational descriptions, is exactly how to sort the points in an order compatible with the natural trace of the contour line, as perceived by a human. This order may be used then to structure the sample into a polygonal approximation of the curve, thus reconstructing the curve from the sample.

In the case of contour lines reconstruction from topographic maps the topology of the curves is well defined (i.e. a set of non-intersecting closed lines). That makes the general problem of curve reconstruction simpler.

Journal of WSCG, Vol.12, No.1-3, ISSN 1213-6972
WSCG'2004, February 2-6, 2003, Plzen, Czech Republic.

Our approach starts from a scanned topographic map and in Section 3 we describe a method to segment that image by a color classification in contour and background pixels. In Section 4 and 5 we describe the vectorization procedure. This is the core of our algorithm that starts considering some local geometrical information contained in the Constrained Delaunay Triangulation and continues filtering the Delaunay edges satisfying the global topology of the contour lines. This filter will leave at the end only the edges belonging to the contour lines. Finally, in Section 6 we give some discussions of general problems.

2. PREVIOUS WORK

General Procedure

Research on automated topographical map recognition has been going on for many years resulting in a huge amount of publications. Early reports about the vectorization of line drawings already introduced the main necessary steps of any automatic procedure: (1) digitalization of the original paper document using a scanner; (2) filtering; (3) thresholding; (4) thinning and pruning the binary image; (5) raster to vector conversion.

These steps can be found in [Leb82] for the automatic vectorization of clean contour and drainage/ridge sheets, in [Gre87] for an early attempt to extract elevation contour lines on topographical maps, [Ami87] for the recognition of lines and symbols, and [Mus88] for the processing of land record maps.

For the first 4 steps different algorithms could be applied all with good results. Task 5, on the contrary, gives different results depending on the applied methods, therefore, in this section we point out only references to the raster to vector problem.

Image Based Approach

Usually approaches for the raster to vector conversion are based on perceptual principles, i.e. to decide for closing or grouping two different segments/pixels, the main two criteria that have been used are proximity and continuity.

Soille et al. [Soi99] used the 5 steps described above to reconstruct contour lines from color topographical maps using a procedure based on mathematical morphology. They include some new ideas for extracting extreme points of a set, then, a combination of Euclidean distances between extremities and differences between their directions is used to join the disconnected lines.

Line Tracing is another technique used in contour line reconstruction [Eik95]. In this approach when gaps occur it is assumed that there is only one possible continuation, and that the continuation can be found along the current direction of the line. The gap is crossed by searching from the point at the end of the line within a sector around the current direction.

Almost all the existing closing algorithm based on perception criteria fail at discontinuity points.

Geometric Based Approach

It is clear that the raster to vector step is the most difficult task of the whole procedure. Luckily it can be analyzed like an instance of the more general problem of curve reconstruction: Given a finite sample V of an unknown curve γ, the task is to construct a graph $G=(V, E)$ such that two points in V are connected by an edge of G iff the points are adjacent on γ. The graph G is called a polygonal reconstruction of γ.

The curve reconstruction problem has received a lot of attention in the graphics and the computational geometry community and a great amount of work has been written.

The first algorithms for curve reconstruction [Att97, Ber97, Ede83, Kir85] imposed a *uniform* sampling condition, as they basically demanded that the distance between any two adjacent samples must be less than some constant. This is not satisfactory as it may require a dense sampling in areas where a sparse sampling is sufficient.

Amenta et al. [Ame98] introduced the concept of the *local feature size* (distance of a point to the *medial axis* of his curve). Using this concept they define a non-uniform sampling condition that allows for sampling of variable density. Then they give an algorithm that, from a sample set of a collection of smooth closed curves, which satisfies this sampling condition, computes the correct reconstruction. This algorithm works by computing the Delaunay Triangulation of the point set and then filtering it to obtain the reconstruction. A survey of algorithms based on *Delaunay filtering* can be found in [Ede98].

Subsequently, several variations that still only handle smooth closed curves were presented [Dey99, Gol99]. Later, Dey et al. [Dey99a] extended this work to handle a collection of open and closed smooth curves. Their algorithm is also based on Delaunay filtering.

Giesen [Gie99] uses a different sampling condition for corner areas and shows that for a sufficiently dense sampling, the TSP (traveling salesman problem) is the correct reconstruction for a single closed curve (possibly with corners). Althaus and Mehlhorn [Alt00] have extended this result by showing that in this case the TSP can be computed in polynomial time. The problem with this approach is that so far it can only handle *single*, *closed* curves. Recently, Dey and Wenger [Dey00] gave an algorithm that allegedly handles well corners and endpoints. The algorithm has no guarantee and, in

fact, it is not difficult to find counterexamples where it fails.
In this paper, we present an algorithm also based on Delaunay filtering, however, unlike all the previous algorithms, our filtering does not use only local rules: the algorithm first detects 'smooth' edges reliably with Amenta's properties and then, starting from the endpoints of the resulting smooth chains, it connects the segments using a combination of distance and direction considering always the global topology of a topographic map.

3. SINGLE COLOR EXTRACTION

In studying complex maps, the use of the color information is essential for recognizing its features. Color scanners being increasingly cheaper, more recent papers deal with maps directly scanned in colors. For example, Ansoult et al. [Ans90] use the mean and variance of the hue channel for discriminating soil types on a digitized soil map. Ebi et al. [Ebi94] transform the input RGB color space into another color space taking the chromaticity into account. Classification-clustering techniques are then applied to the bivariate histograms constructed from the resulting two-chromaticity channels.

Color Space Selection

The RGB color format is the most common color format for digital images. The primary reason for this is because it retains compatibility with computer displays. However, the RGB space has the major drawback in that it is not perceptually uniform. Because of this, uniform quantization of RGB space gives perceptually redundant bins and perceptual holes in the color space. Furthermore, ordinary distance functions defined in RGB space will be unsatisfactory because perceptual distance is a function of position in RGB space.
Other color spaces, such as CIE-LAB, CIE-LUV and Munsell offer improved perceptual uniformity. In general they represent with equal emphasis the three-color variants that characterize color: *hue*, *lightness*, and *saturation*. This separation is attractive because color image processing performed independently on the color channels does not introduce false colors. Furthermore, it is easier to compensate for many artifacts and color distortions. For example, lighting and shading artifacts will typically be isolated to the lightness channel. In general, these color spaces are often inconvenient due to the basic non-linearity in forward and reverse transformations with RGB space. For color extraction we utilize the more tractable HSV color space because it has the above-mentioned characteristics and the transformation from RGB space is non-linear but easily invertible [Hun89].

Quantization

The next issue after color space selection is quantization. The HSV color space can be visualized as a cone. The center represents *value*: black to white. Distance from the axis represents *saturation*: amount of color present. The angle around the axis is the *hue*: tint or tone of the color. Quantization of hue requires the most attention. The hue circle contains the primaries red, green and blue separated by 120 degrees. A circular quantization at one-degree steps sufficiently separates the hues such that the three primaries yellow, cyan and magenta are represented each with sixty sub-divisions. Saturation and value are each quantized in our approach to five levels yielding greater perceptual tolerance along these dimensions. A simple example of quantized HSV space appears in Fig.1.

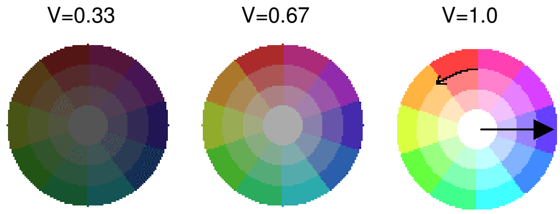

Figure 1. Quantized HSV color space, 10 hues, 4 saturations and 3 values

Segmentation

In topographic maps we have to handle with a standard set of colors:
Black – grid lines and man-made features such as roads, buildings, etc.
Blue - water, lakes, rivers, streams, etc.
Brown - contour lines
Green - areas with substantial vegetation (could be forest, scrub, etc.)
White - areas with little or no vegetation; white is also used to depict permanent snow fields and glaciers
Red - major highways; boundaries of public land areas
Purple - features added to the map since the original survey. These features are based on aerial photographs but have not been checked on land.
We have found, in the literature [Her98, Gon95, Smi95] and experimentally, that colors with $value<0.25$ can be classified as black and that colors with ($saturation<0.20$ and $value>0.60$) can be classified as white. In a scanned topographic map, usually, 80% of pixels are black or white. The remaining pixels all fall in the chromatic region of the HSV cone. We build the hue histogram of these remaining pixels (Fig.2). The resulting peak near the brown (10<hue<30) is referred to the contour lines.

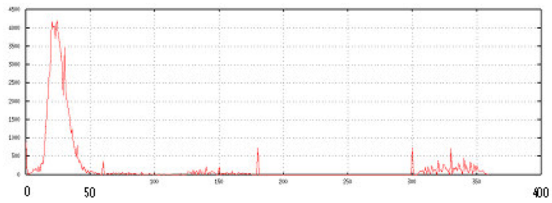

Figure 2. Hue histogram of a scanned map without black pixels and white pixels

In Fig.4 are shown the pixels classified belonging to a contour line of the raster map in Fig.3. That image is the result obtained after step 2 (filtering) and 3 (thresholding) of the general procedure.

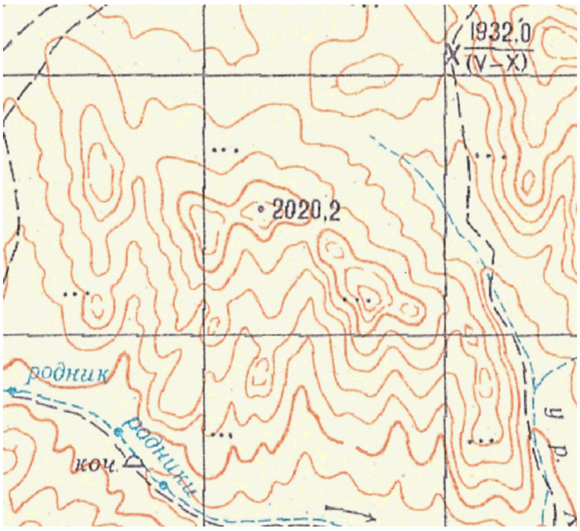

Figure 3. Original scanned map

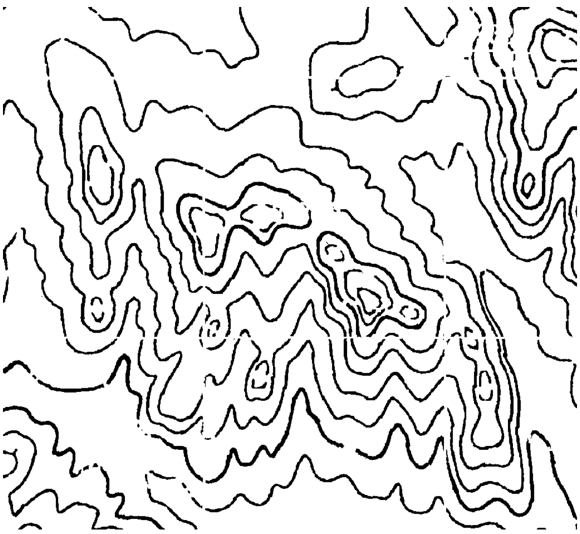

Figure 4. Contour's pixels extraction from image in Fig. 3

Thinning

Once obtained the binary image, a thinning procedure is needed (step 4) to reduce the thresholded output to lines of a single pixel thickness, while preserving the full length of those lines (*i.e.* pixels at the extreme ends of lines should not be affected) [Gon92]. In Fig.5 it is shown a zoom of a thinned area. The thinning procedure used creates lines that are 4-connected. Therefore, it is straightforward to extract a set of non-intersecting segments and smooth those segments, for example with a Laplacian Operator. A smoothing process is needed to avoid excessive branching of the Voronoi Diagram used for the raster to vector step described in the next section. Fig. 6 shows the resulting segments that will be used as input for the Constrained Delaunay Triangulation.

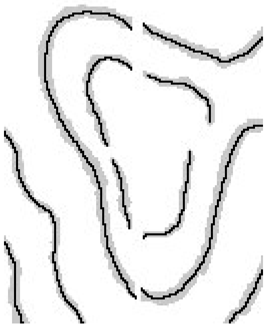

Figure 5. Zoom of the Thinning pre-processing

Figure 6. Smoothing result of thinned segments

4. CRUST EXTRACTION

Amenta et al. [Ame98] showed that the "crust" of a curve or polygon boundary can be extracted from unstructured (and unlabelled) input data points if the original curve is sufficiently well sampled. Their intuition was that, as the vertices of the Voronoi diagram approximate the medial axis of a set of sample points from a smooth curve then by inserting the original vertices plus the Voronoi vertices into a

Constrained Delaunay Triangulation, the circumcircles of this new triangulation approximate empty circles between the original smooth curve and its medial axis. Thus any Delaunay edge connecting a pair of the original sample points forms a portion of the sampled curve called the "crust". Using this approach for our application we have obtained good results. In Fig.7 contours *c* and *d* are recognized correctly, but contour *b* has still 3 gaps not recognized from Amenta's algorithm because the Voronoi vertices inserted into the original Delaunay Triangulation avoid the direct connection between original vertices. This simple example shows that an algorithm that uses just local geometric properties is not enough.

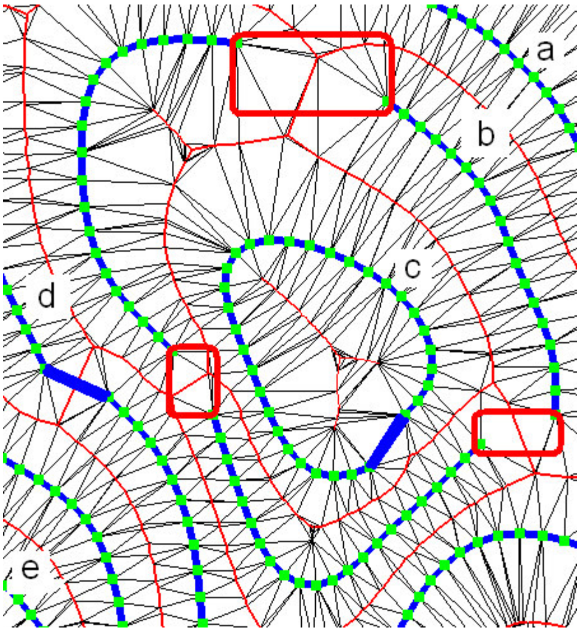

Figure 7. Simple example of edge classified as crust

Thus we decided to include already at this stage of recognition some global information, called *complete contours,* coming from our particular goal: reconstruction of contour lines.

After having processed the original vertices with Amenta's algorithm, we check if any contour is *complete*. We define a contour *complete* iff it is a closed curve (Fig.7 contour *c*) or if it is a curve with both extremes touching the border (Fig.7 contour *d*). We erase all the vertices belonging to a complete contour from our original input data.

We repeat this procedure until we recognize a complete contour.

We noticed by experience that for simple topographic map this method is enough for a total recognition of the contour lines (for example gaps in contour *b* in Fig.7 are all recognized after two iterations).

In fact, at each iteration, the Constrained Delaunay Triangulation is simplified; therefore, Voronoi diagram changes too. This makes possible to classify more edges as *crust* (see Fig.8).

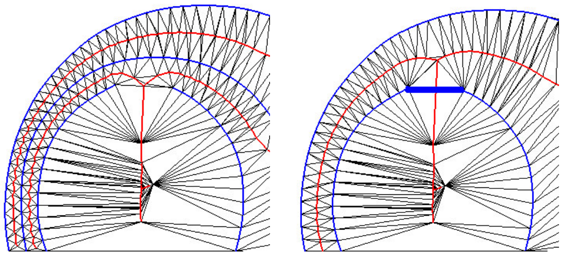

Figure 8. After having eliminating the middle contour the gap is recognized

5. GRAPH REDUCTION

Although the procedure described in Section 5 simplify considerably the set of the original input vertices, usually for complex maps it is not enough to reconstruct the whole set of contour lines. In Fig.9 it is represented what still remains after the performance of previous algorithm with input the pixels coming from the thinning pre-processing of Fig.4. Many contours have been already erased because they have been found *complete*, however, many other segments are still *incomplete.* What remains in Fig.9 are all the edges classified *crust* belonging to a contour still with some gap. In the same figure is shown the Voronoi Diagram (Medial Axis) of the remaining segments to justify why the remaining contours are still open.

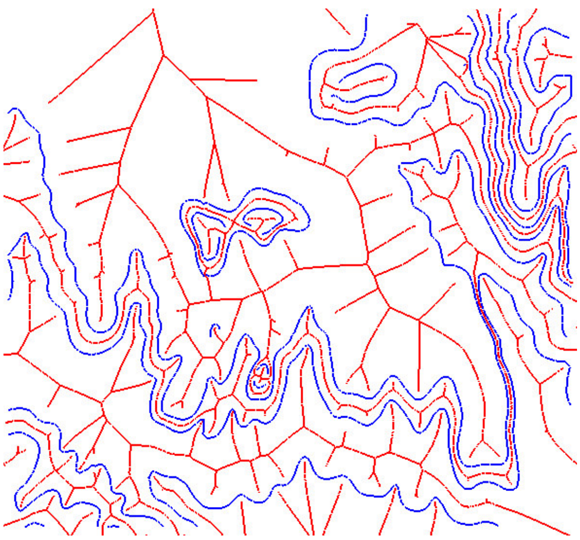

Figure 9. Crust and Voronoi Diagram after three iteration of Amenta's algorithm

To perform the closure of the remaining segments we change structure from Delaunay Triangulation to an indirect Graph $G=(V, E)$.
To construct the graph we take as vertices only the extremities of the remained segments and all their adjacent Voronoi vertices (Fig.10). The adjacency set E is made of all and only the Delaunay Edges e_i that are connected with an extreme and a Voronoi vertex. For the Amenta's algorithm any couple of extremes are not adjacent, thus in G all the extremes have only Voronoi vertices as adjacent. In Fig.10 is an example of that graph.
Following the idea described in [Spi02] for each edge e_i we associate a weight w_i depending on his length and on the direction related to the existing chain (tangent of the segment at the extreme).
We compute the weight of any possible connection-path between two extremes. This procedure is easy and fast. In fact, to check if two extremes are neighbors, a *Breadth First Visit* of extreme's adjacent with depth less than three is enough.
We compute then a symmetric square matrix where the entries are infinite if two extremes are not neighbors and are the weight of the path otherwise. This matrix is the second global structure, after the *complete* contour, which we use to solve the problem.
We start connecting the couple of neighbors with minimal weight. If a contour is finally found *complete* we erase it from the input vertices and restart with Amenta's algorithm.
We perform these steps until any gap still remains.
The structures utilized, good implementation and fast computers make all the procedure really fast. It takes about 4 seconds to obtain the result in Fig.11 (5767 vertices) starting from the map in Fig.3 (1031x938) on a 2.6GHz Pentium 4 WorkStation.

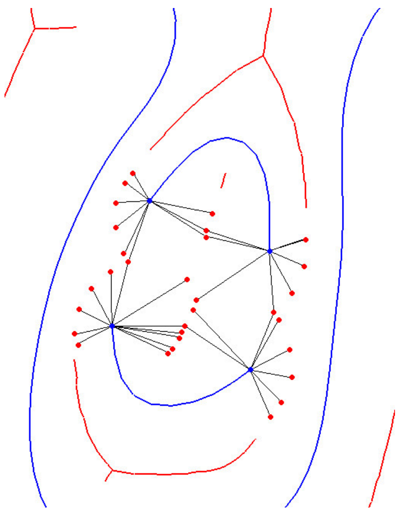

Figure 10. Four End-Points and their adjacent Voronoi vertices

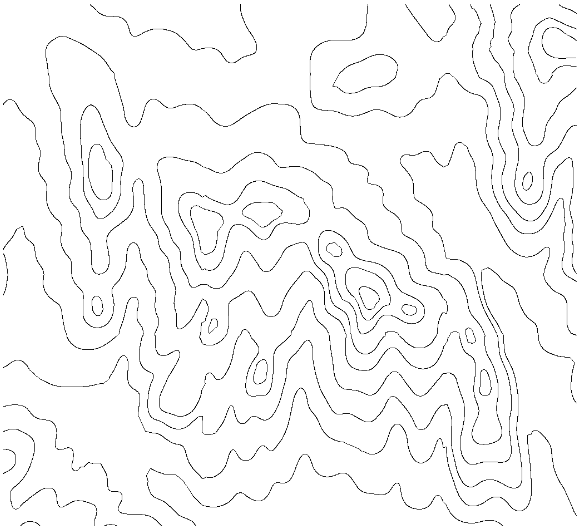

Figure 11. Raster to Vector result of map in Fig.3

In Fig.12 there is a visualization of a 3D reconstruction starting from the contours using the modeling method described in [Hor03].

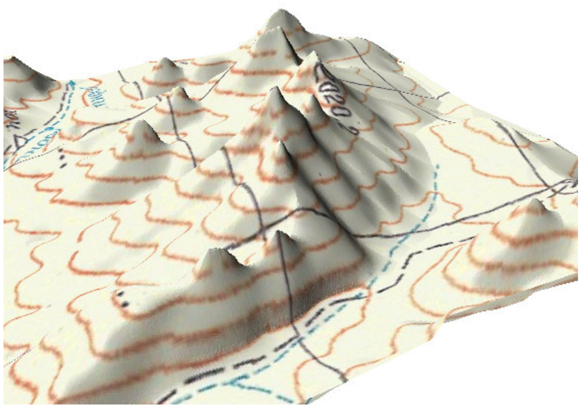

Figure 12. Terrain Reconstruction from Contours

6. PROBLEMS AND DISCUSSION

Strictly automatic processing is not always a possible solution in topographic map recognition. There are several problems that must be considered with real cartographic maps: poor conditions and topological errors are two great opponents for the raster to vector process. A semi-interactive approach using user expertise could be an alternative.
If the input image is poor it will be difficult to find a proper classification method. A trade-off between many gaps and thick lines must be taken.
Thick lines are the first problem discussed in this section. When thick lines happen (Fig.13) it is really difficult to recover the right topology (Fig.14). Therefore more gaps must be preferred to thick lines.

Figure 13. Single color extraction

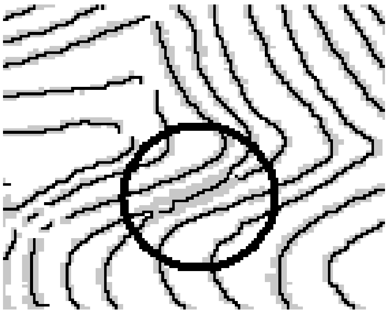

Figure 14. Error in the segmentation of the binary image can cause topological problems

Another problem is that the same color is used to represent contour lines and elevation numbers (Fig.15). This characteristic makes difficult any automatic approach. Adding an OCR pre-processing will probably solve this problem. The elevation of the contour lines could be also automatically inserted and in the raster map, in place of the string, could be inserted a contour segment.

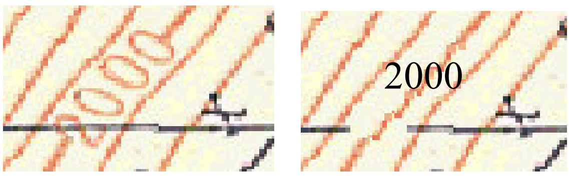

Figure 15. Problems occurred in case of elevation number

Last problem is a Topological problem (Fig.16). Very difficult to detect automatically, this case is easy to treat with our approach interactively. After Amenta's procedure has recognized the crust, a digital rubber could be used to delete all the edges classified *crust* starting from a T-point (vertex with 3 adjacent) to another T-point. For this purpose the Delaunay Triangulation is really powerful and fast.

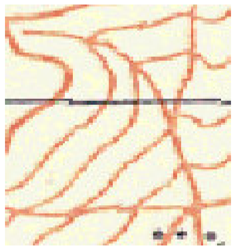

Figure 16. Elements in the topographic map with same color and characteristics of contour lines.

7. CONCLUSIONS

The efficiency of our algorithm is based on a combination of local and global criteria. We have shown that reconstruction algorithms based only on local criteria fail even in simple cases.

The purpose of our algorithm is to identify significant structural relationships. The perceptual organization starts with primitives coming from the image. The starting point of these primitives is the edge. Local geometric properties are used to construct reliable chains of edges. Using global information, contours found complete are removed from the input data. This step simplifies the problem and makes possible the recognition of further complete contours.

Another global information (the matrix of weights) is then used to fill the remaining gaps grouping different segments. Grouping is done by processing iteratively the gaps in a bottom-up fashion. Firstly are processed gaps with almost certain match and lastly those with more uncertainty.

Although the algorithm finds automatically all the contours in a well-conditioned map, some problems still remain in general maps. To overcome these problems user interaction is actually necessary. Future work is in progress to automatically resolve these problems.

8. ACKNOWLEDGMENTS

Thanks are given to Department of Geography of University of Erlangen-Nürnberg for the data. Thanks are also given to Kai Hormann for his interesting suggestions.

9. REFERENCES

[Alt00] E.Althaus and K.Mehlhorn. TSP-based curve reconstruction in polynomial time. In Proceedings of 11th Sympos. Discrete Algorithms, ACM and SIAM, pp 686-695, 2000.

[Ame98] N.Amenta, M.Bern, and D.Eppstein. The crust and the β-skeleton: Combinatorial curve reconstruction. Graphical Models and Image Processing, pp 125-135, 1998.

[Ami87] T.Amin, R.Kasturi. Map data processing: Recognition of lines and symbols. Optical Engineering, 26(4), pp 354-358, 1987.

[Ans90] M.Ansoult, P.Soille, J.Loodts. Mathematical morphology: a tool for automated GIS data acquisition from scanned thematic maps. Photogrammetric Engineering and Remote Sensing, 56(9), pp 1263-1271, 1990.

[Att97] D.Attali. r-regular shape reconstruction from unorganized points. In Proceedings of the 13th Annual ACM Symposium on Computational Geometry (SCG'97), pp 248-253, 1997.

[Ber97] F.Bernardini and C.L.Bajaj. Sampling and reconstructing manifolds using α-shapes. In Cambridge Proceedings of the 9th Canadian Conference on Computational Geometry (CCCG'97), pp 193-198, 1997.

[Dey99] T.K.Dey and P.Kumar. A simple provable algorithm for curve reconstruction. In Proceedings of the 10th ACM-SIAM Symposium on Discrete Algorithms (SODA`99), pp 893-894, 1999.

[Dey99a] T.K.Dey, K.Mehlhorn, and E.A.Ramos. Curve reconstruction: Connecting dots with good reason. In Proceedings of the 15th Annual ACM Symposium on Computational Geometry (SCG'99), pp 197-206, 1999.

[Dey00] T.K.Dey and R.Wenger. Reconstructing curves with sharp corners. Proc. 15th Ann. ACM Sympos. Comput. Geom., (2000), 233–241.

[Ebi94] N.Ebi, B.Lauterbach, W.Anheier. An image analysis system for automatic data acquisition from colored scanned maps. Machine Vision and Applications, 7(3), 148-164, 1994.

[Ede83] H.Edelsbrunner, D.G.Kirkpatrick, and R.Seidel. On the shape of a set of points in the plane. IEEE Transactions on Information Theory, 29(4): pp 71-78, 1983.

[Ede98] H.Edelsbrunner. Shape reconstruction with the Delaunay complex. In LATIN'98: Theoretical Informatics, volume 1380 of Lecture Notes in Computer Science, pp 119-132, 1998.

[Eik95] L.Eikvil, K.Aas, H.Koren. Tools for Interactive Map Conversion and Vectorization. International Conference on Document Analysis and Recognition, Montreal, August 14-16, 1995.

[Fig94] L.H.Figueiredo and J.M.Gomes. Computational morphology of curves. Visual Computer, 11: pp 105-112, 1994.

[Gie99] J.Giesen. Curve reconstruction, the TSP, and Menger's theorem on length. In Proceedings of the 15th Annual ACM Symposium on Computational Geometry (SCG'99), pp 207-216, 1999.

[Gol99] C.Gold. Crust and anti-crust: A one-step boundary and skeleton extraction algorithm. In Proceedings of the 15th Annual ACM Symposium on Computational Geometry (SCG'99), pp 189-196, 1999.

[Gon95] Y.Gong, M.Sakauchi. Detection of regions matching specified chromatic features. Computer Vision and Image Understanding 61, 263-269, 1995.

[Gon92] R. Gonzalez and R. Woods. *Digital Image Processing*, Addison-Wesley Publishing Company, 1992, pp 518 – 548.

[Gre87] D.Greenlee. Raster and Vector processing for scanned line work. Photogrammetric Engineering and Remote Sensing, 53(10), pp 1383-1387, 1987.

[Her98] N.Herodotou, K.Plataniotis. A.Venetsanopoulos, A content-based storage and retrieval scheme for image and video databases. Visual Communications and Image Processing 98.

[Hor03] K.Hormann, S.Spinello, P.Schröder. C1-continuous Terrain Reconstruction from Sparse Contours. To appear in Visual Modelling and Visualization 2003, Munich.

[Hun89] R.W.G.Hunt. *Measuring Color*. John Wiley and Sons, 1989.

[Kir85] D.G.Kirkpatrick and J.D.Radke. A framework for computational morphology. Computational Geometry, pp 217-248, 1985.

[Leb82] F.Leberl, D.Olson. Raster scanning for operational digitizing of graphical data. Photogrammetric Engineering and Remote Sensing, 48(4), pp 615-627, 1982.

[Mus88] M.Musavi, M.shirvaikar, E.Ramanathan, A.Nekovei. A vision based method to automate map processing. Pattern Recognition, 21(4), 319-326, 1988.

[Smi95] J.R.Smith and S.F.Chang. Single Color Extraction and Image Query. In Proceedings, IEEE International Conference on Image Processing (ICIP-95), Washington, DC, October 1995.

[Soi99] P.Soille, P.Arrighi. From scanned topographic maps to digital elevation models, In Proceedings Geovision 99.

[Spi02] S.Spinello, G.Greiner. Automatic Contour Line Recognition From Scanned Topographic Maps. Poster section, Fifth International Conference on Curves and Surfaces, Saint-Malo, France, June 27-July 3, 2002