TEXT/GRAPHICS SEPARATION IN RASTER-SCANNED COLOR CARTOGRAPHIC MAPS

Aurelio VELÁZQUEZ[1], Serguei LEVACHKINE[2]
[1]Instituto Mexicano del Petróleo
Eje Central Lázaro Cárdenas 152, México D. F. 07730, MÉXICO
avelaz@imp.mx
[2]Centro de Investigación en Computación-Instituto Politécnico Nacional
U.P. Adolfo López Mateos, Edif. CIC, México D.F. 07738, MÉXICO
palych@cic.ipn.mx

ABSTRACT

In this paper, we propose a method to separate and recognize to the fullest extent possible those characters that are touching or even overlapping each other. The characters are processed in raster-scanned color cartographic maps. The map is segmented first to extract all text strings including those that are touched other symbols and strokes. Second, OCR-based recognition with Artificial Neural Networks (ANN) is applied to define the coordinates, size and orientation of alphanumeric character strings in each case presented in the map. Third, four straight lines or a number of curves computed in function of primarily recognized by ANN characters are extrapolated to separate those symbols that are attached. Finally, the separated characters input into ANN again to be finally identified. Experimental results showed 95÷97% of successfully recognized alphanumeric in raster-scanned color maps.

KEY WORDS

Cartographic Color Map, Text Segmentation, Character Recognition, Artificial Neural Networks.

1. INTRODUCTION

The maps such are electrical schematics, cartographic color maps, topographic maps, cadastral maps, engineering drawings, technical illustrations, public service maps and other thematic maps contain a lot of text.

Cartographic color maps are plenty of punctual, linear and area objects. With points, lines and areas a map is able to model the real world and the man-made boundaries, using a map scale, which is a ratio of distance on the map to distance on the Earth. To describe those objects there can be used symbols (portrays) and labels (alphanumeric characters) presenting a great variety of features, some of them in equal shape but in different color. Different colors are used to represent different objects, including a number of character's fonts. These can be colored and following different paths in all kind of angles.

The process of development of a Geographical Information System (GIS) includes the selection of the paper and raster maps for vectorization Levachkine et al. [1]. To be included into GIS, the paper maps should be changed to a computer readable format, normally a raster format. After that, the raster maps can be converted into vector format that is most adequate to GIS-applications. In the context of raster-to-vector conversion of graphical documents, the problem of text recognition is of special interest, because textual information can be used for verification of vectorization results (post-processing).

The retrieval of all presented elements in a map can be made manually or by means of a computer system. In the former case, the map is scanned in a raster format and then converts to vector. Before a raster-to-vector conversion a map segmentation and recognition are usually employed.

General frameworks. The text segmentation and its subsequent recognition in raster images are very difficult problems because, in general, there is either text embedded in graphic components, or text touching graphics Doermann [14]. These challenging problems have been received numerous responses from the graphic recognition community Nagy [15]. However, there have not been developed efficient programs that solve the task automatically. Thus, the main idea of the most works is to put the operator in the loop. As proposed, for example, by Ganesan [16], the operator can draw line through the text, marking it as text and revealing its orientation all in one step. Fletcher et al. [17] and Tan et al. [18] developed the algorithms to extract text strings from text/graphics image. Both methods however assume that the text does not touch or overlap with graphics. For maps, the problem is much more complex since the touching or overlapping as well as many other character configurations are commonly presented in maps. Cao et al. [20] proposed a specific method of detecting and extracting characters that are touching graphics in raster-scanned color maps. It is based on observation that the constituent strokes of characters are usually short segments in comparison with those of graphics. It combines line continuation with the feature line width to decompose and reconstruct segments underlying the region of intersection. Experimental results showed that proposed method slightly improved the percentage of correctly detected text as well as the accuracy of character recognition with OCR.

Segmentation. Applying color and spatial attributes to segment thematic maps, Silva [2] used a 300-dpi resolution in a *RGB* color system to perform a Karhunen-Loeve transformation. Luo et al. [3] used the directional morphological operations. They coded images by run-length-encoded as an enchained list, deleting the text that is represented by lines, and finally subtracting the new image from the original one to obtain an image with text without lines. In [4], Li described the Comb algorithm based on the best common structure of local minima found at a moment to search for global minima. He used the concept of maximum a posteriori (MAP) and Markov random fields (MRF) as the frameworks. To segment text from engineering drawings Adam et al. [11] used Fourier-Mellin transform in a five-step process. Using a heuristics, they found broken chains. In [12], Hase et al. described a three-step algorithm of segmentation called "multi-stage relaxation". However, they do not recognize characters. In [5], Levachkine et al. used false colors in a *RGB* model. They applied different combinations of *R, G* and *B* basic colors to segment map objects, and then a neighborhood analysis to recover or eliminate pixels.

Extraction and recognition. Some proposals for character extraction and recognition to be mentioned are as follows. In [6], Myers et al. described the verification-based approach for automated text and feature extraction from raster-scanned maps. They used a gazetteer to propose a forecasting hypothesis, which characters are in labels and where is their position in the map, having the information from other map in a different scale. Character and text boxes are used in [7] by Wenyin et al. The authors considered only horizontal and vertical text in which a character box is a rectangle with rate sides are no larger than 10 pixels to join character boxes. Thus, they built the text box that can grow horizontally or vertically under a threshold to fit the letters. Using directional morphological operations Luo et al. [3] separated the text from lines but not from curves. Deseilligny et al. [8] proposed different knowledge levels to solve the task. They begun with an analysis of related components (semiologic), then built the character chains (syntactic), detected related characters (higher semiologic level) and, finally, following the natural language rules corrected the text (semantic). Using templates Friscknecht et al. [9] linked them with symbols and characters. The approach does not require the complete template. It is pondered and hierarchically built. To retrieve street names Nagy et al. [10] used one of the four black layers. Taking the hue component from a *HSV* model for segmentation, they subtracted the street layer from the black layer and then made a connected component analysis to distinguish text characters. An efficient system to recognize characters by means of adaptive ANN is described in [13] by Velázquez et al. To train ANN, they used characters from a word processor in different fonts, sizes and inclinations by applying them to identify a great variety of characters in cartographic maps.

The rest of paper is organized as follows. In Section 2, we describe an alphanumeric segmentation-recognition system. In Section 3, we consider a case of touching and overlapping characters presented in raster-scanned color cartographic maps. A method (*V-lines* and *V-curves*) to separate and further recognize the touching and overlapping characters is described in this section as well. Section 4 contains paper's conclusion.

2. SEGMENTATION OF OVERALL MAP CHARACTERS

A raster map has to be segmented first. All its elements should be retrieved with their coordinates and features, and then sent to the corresponding thematic layers. These layers could be symbols, rivers, landmarks, roads, railroads, pipelines, isoclines, natural and artificial surroundings, words and numbers, lakes and other punctual, linear and polygonal bodies.

Cartographic maps are the most complex graphical documents due to the high density of information that they contain. A typical example is shown in Figure 1 (*RGB* image). There are labels of different types, sizes, orientations and colors, roads, rivers, symbols and artificial surroundings. Some strokes are touching each other's.

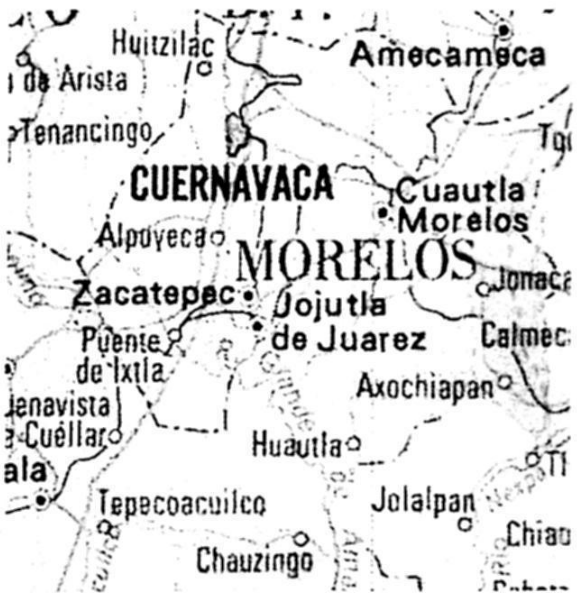

Figure 1. Example of color cartographic map with different types of characters.

To obtain a binary image from color image, the former must be changed to a gray-level. One way to make this change is to convert the *RGB* model to the *YIQ* model, where the luminance (*Y*) is a gray-level image. Another way is to average the *R, G* and *B* values. Figure 2 shows the gray-level image obtained from the image of Figure 1. In this work, we used both conversion procedures as well as their combination as described in [5].

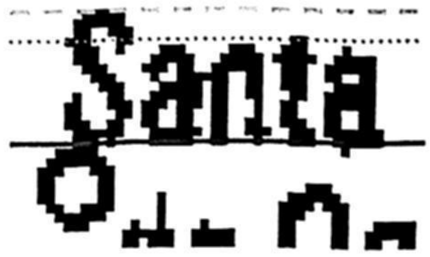

Figure 8. V-lines to separate the letter '*S*' from the symbol is bellow it.

A similar process to separate the letters '*n*' and '*o*' from the letter '*S*' is below them is applied to the image shown in Figure 7d to obtain the whole word "*Ochentaiuno*" (a special process is required to separate the '*n*' from the '*o*'). Used lines for this task are shown in Figure 9.

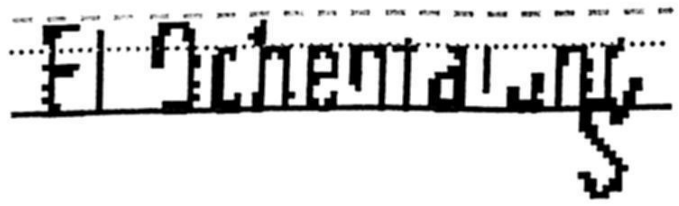

Figure 9. V-lines to separate the letters '*n*' and '*o*' from the letter '*S*' are below them.

If the upper case characters are not horizontal, we can trace a diagonal rectangle following the same angle that those letters have. In figure 10, the first '*S*' is touching the river's line which name is '*BALSAS*'.

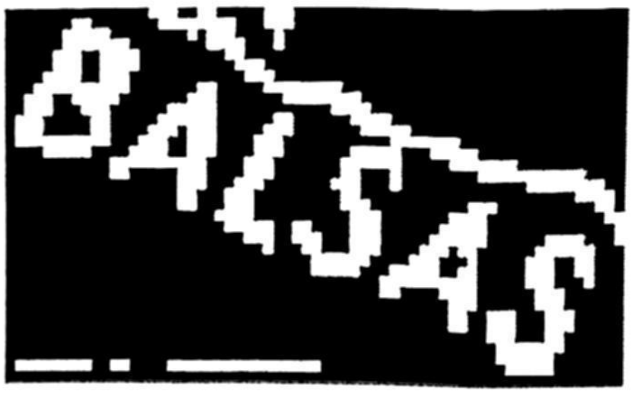

Figure 10. Label of '*BALSAS*' river is touched the line of the river by the first letter '*S*'.

With the other five letters, we can build a rectangle that covers all the word letters. Using the upper and lower pixels from the first and last characters of the chain, we compute their mean value obtaining two points to trace a line with them as shown in Figure 11.

Then the left or upper (if the chain is vertical) point is moved one pixel up and one left so that we have two new points, the right or lower point is moved one pixel down and one right so that there are four points for the first dynamic rectangle.

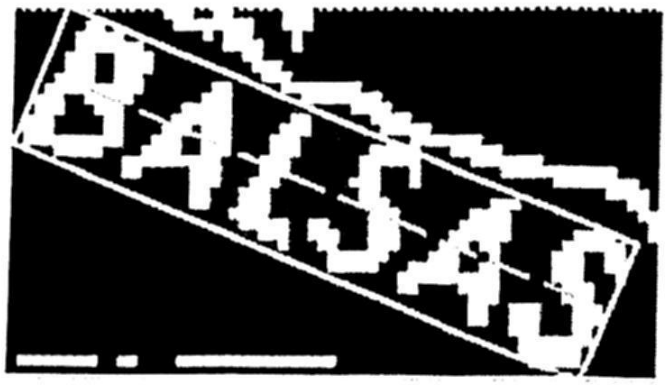

Figure 11. Starting with a line following the label's angle, a dynamic rectangle is built.

Computing the position of each character in the largest lines, we can find if there are outside pixels. The shortest lines are used for the same purpose with the first and last characters. If there are outside pixels in a line, it is moved one pixel in corresponding direction. If more than one line has outside pixels, all those lines are moved. The process is continued until no more outside pixel are found.

Now, it is possible to identify the missing letter using only the pixels inside of the rectangle box, sending it to ANN and testing the word '*BALSAS*' in the gazetteer.

On the other hand, for upper case letters, four lines should be computed. Figure 12 shows the word '*Chicayán*' touching a line of the river labeled for it.

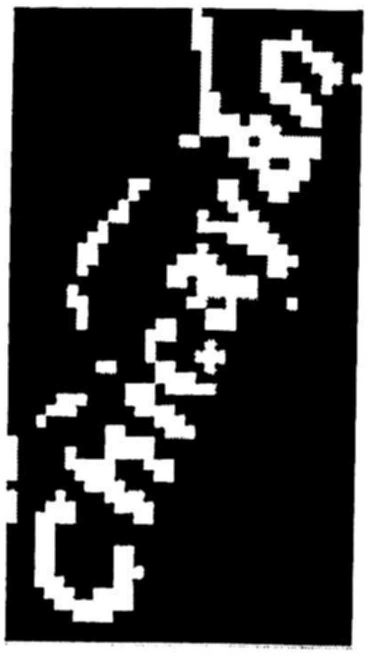

Figure 12. Label of '*Chicayán*' river is touched the line of the river with the letter '*á*'.

Using the procedure employed for capital letters, it is possible to construct a rectangle as shown in Figure 13. The largest lines will be used to find the four V-lines, if they are present. Normally, three of them are present.

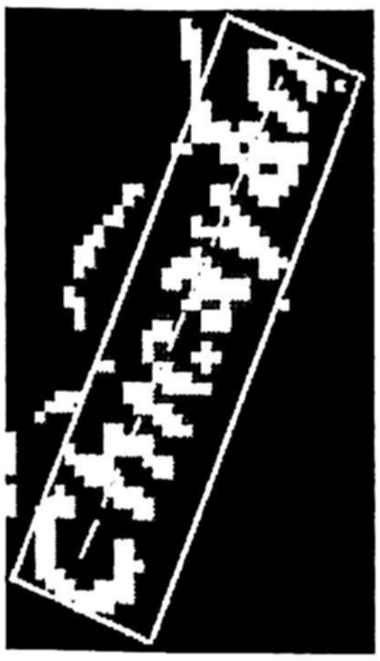

Figure 13. First, it is necessary to build a rectangle as it was made with capital letters.

Each line should be moved with pixel-by-pixel procedure until it is reached the next numbered level. An additional adjustment is made: each ending point is moved up three pixels and the line that better fits one of the inner lines is selected as the "leader". To the other three lines, if exist, we assign the same angle that the leader has. Figure 14 shows these lines.

Cutting the label with line one, the missing letter can be analyzed. It could be recognized as a letter '*d*', but the word '*Chicaydn*' does not exist in gazetteer. Thus, we attempt now with line two. Then, the letter can be interpreted as an '*a*' and the word '*Chicayan*' is already in the gazetteer.

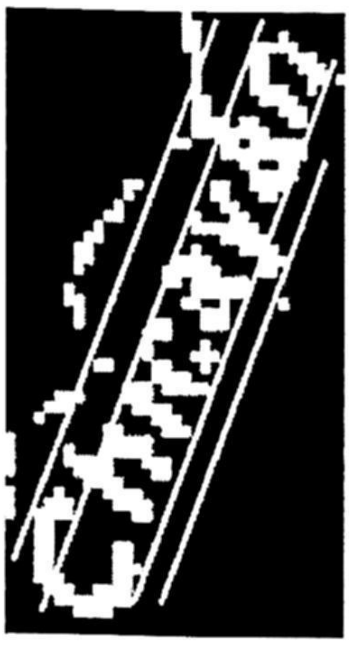

Figure 14. Four lines to unglue letter '*a*' from the riverbed.

There are other manners in which objects can touch an alphanumeric character: at its left or right, at its top or bottom as in example shown in Figure 15, where letter '*M*' is touched at its left by a state boundary.

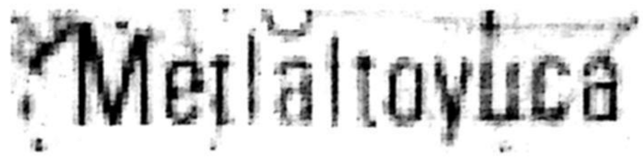

Figure 15. The word is touched at one extreme of the chain.

The V-lines are not helpful themselves to separate the characters. However, the V-lines are useful to build a "growing" rectangle that is fitted to the character's pixels, identifying the characters with the ANN and using the gazetteer until it matches with a correct word. The growing rectangle is shown in Figure 16.

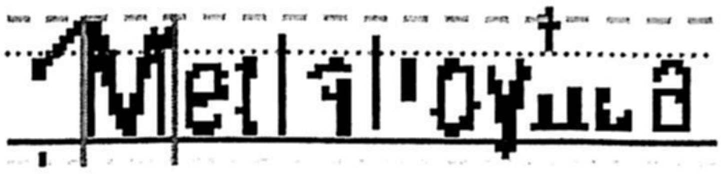

Figure 16. "Growing" rectangle to identify the letter 'M'.

To build the rectangle, we employ the following steps: *1) if the character is at left, we start at the beginning of the first letter. At this moment, we have lines two, three and four; there is another line, the second from the top to bottom. We use a tolerance of one third of the distance between lines two and three. A perpendicular line, that begins at line three plus tolerance and ends at line two minus tolerance, is moved left until an appropriated pixel is found, 2) the line moved is the first line of the rectangle. Other is formed by copying this line two pixels left. The other lines are two and three, unless there are pixels outside those lines, but inside of the tolerance. The first line found is the "anchor", all others can be removed, and 3) left line is moved pixel-by-pixel. There could be a motion of upper or lower lines always inside of the tolerance. Each time the line is moved, the pixels in the rectangle are analyzed and tested by ANN and the gazetteer. The process is continued until a correct word is found or the distance between the line and the anchor is more than one and half times of the distance between upper and lower lines.*

Unfortunately, there are some man-made errors on the maps and, even though our method outputs the complete chain of characters, the word could be misspelled. Figure 17a shows a word where letters '*l*' and '*n*' are touched by two lines. After the processing with V-lines shown in Figure 17b is applied the chain "*Tulacingo*" is built, but it is not in the gazetteer, because the right word is "*Tulancingo*" in which the letter '*n*' was missing. This error can be corrected by another procedure [22].

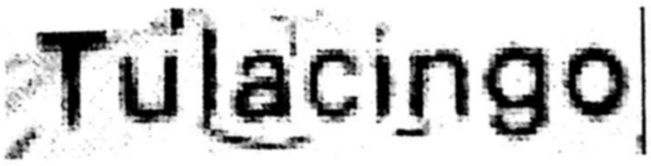

Figure 17a. Original chain touched two arcs of a boundary.

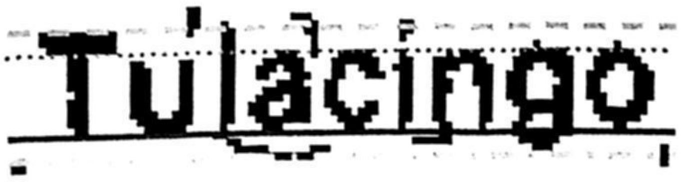

Figure 17b. Characters were unglued and the chain '*Tulacingo*' was build.

On the other hand, some labels are nearly impossible to detect because the background features are too close to their own features. Figure 18a shows such a label and Figure 18b shows its binary image with the damaged characters hard to identify.

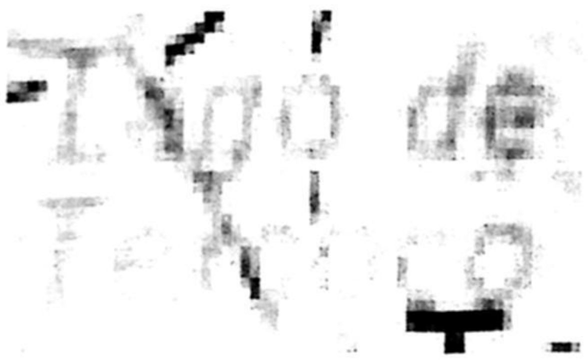

Figure 18a. Label overlapped by other objects with similar attributes.

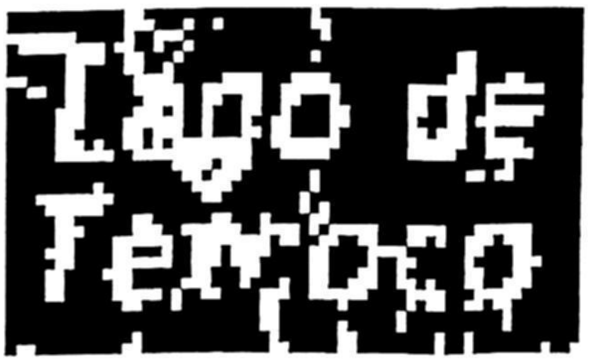

Figure 18b. Binary image of Figure 18a, showing distorted characters.

Another example is shown in Figure 19. It is impossible to detect, in automatic way, the chain of characters because all of them are touched other elements.

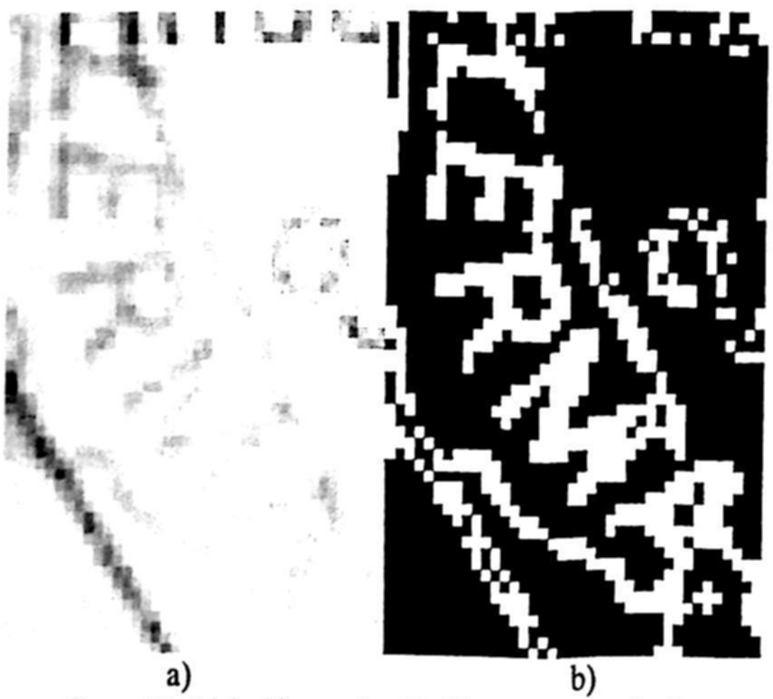

Figure 19. Original image a) and its binary representation b).

In last two cases the operator intervention is certainly required as of the philosophy by Bodansky [23] or (better) by Gelbukh et al. [22].

V-curves. Take a look to Figure 20a. It displays a curvilinear text associated to a riverbed with the letter '*g*' touching the line of the river (Figure 20b shows corresponding binary image).

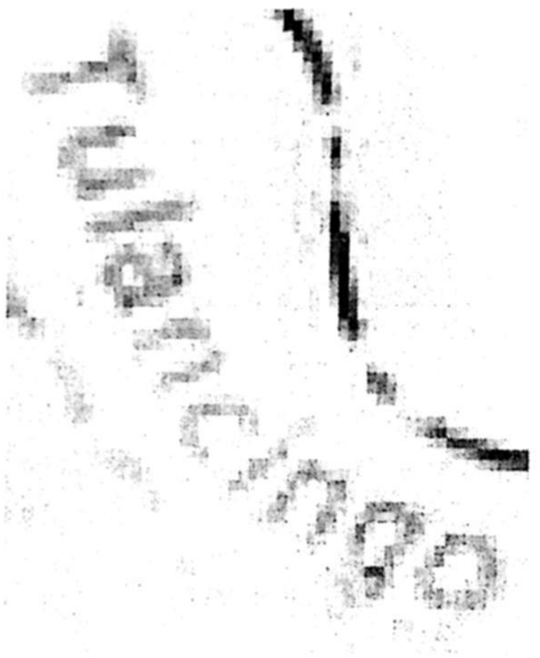

Figure 20a. Color image with a curvilinear text "*Tulancingo*" (name of the river).

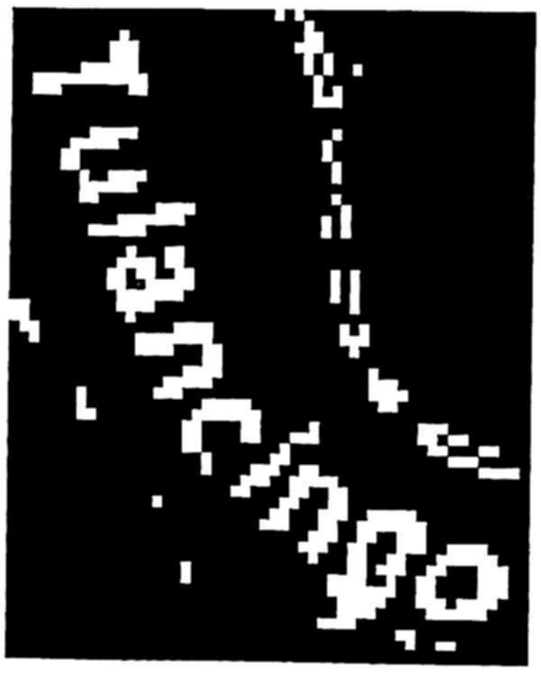

Figure 20b. Binary image of Figure 20a.

An application of V-lines method is difficult in such a case. However, we can use the following procedure that we call *V-curves* to solve the task. The text is divided into blocks of fixed (or nearly fixed) inclination. To each block the V-lines method is applied. Thus obtained lines are connected by a linear extrapolation, forming linear splines. These splines are V-curves. The following steps are similar to those that were used in V-lines method.

Because of the paper space limit we have to stop our explanations on V-curves method. Probably, it will be a topic of our subsequent paper.

4. CONCLUSION

In this paper, a method to separate and recognize touching and overlapping characters in raster-scanned color cartographic images has been developed. The method performs the process to segment the character layer and most valid (or "geographically meaningful") words are built. Though some words cannot be obtained complete, the system is able to suggest one word from a gazetteer to support the operator to resolve the ambiguous cases. OCR-based recognition procedure with ANN applied to the case of study possesses some peculiarities. ANN were tested first with synthetic characters. Among 18,432 synthetic samples, 23 were not recognized, giving 99.87% of successfully recognized characters. After that the same ANN were employed for the characters of cartographic maps on a set of 2,125 samples. The results gave 93.21% of success. These results were improved with the V-lines to some 96.73%.

REFERENCES

[1] S. Levachkine and E. Polchkov, Integrated technique for automated digitization of raster maps, *Revista Digital Universitaria*, 1(1), 2000, on-line: http://www.revista.unam.mx/vol.1/art4/ (ISSN: 1607-6079)

[2] C.J. Silva, Segmentation of thematic maps using color and spatial attributes", *Lecture Notes in Computer Science Vol. 1389*, 1998, 221-230.

[3] H. Luo and K. Rangachar, Improved directional morphological operations for separation of characters from maps/graphics, *Lecture Notes in Computer Science, Vol. 1389*, 1998, 35-47.

[4] S.Z. Li, Toward global solution to map image restoration and segmentation using common structure of local minima, *Pattern Recognition*, 33(1), 2000, 715-723.

[5] S. Levachkine, A. Velázquez, V. Alexandrov and M. Kharinov, Semantic analysis and recognition of raster-scanned color cartographic images, *Lecture Notes in Computer Science, Vol. 2390*, 2002, 178-189.

[6] G.K. Meyers and C.H. Chen, Verification–based approach for automated text and feature extraction from raster-scanned maps, *Lecture Notes in Computer Science, Vol. 1072*, 1996, 190-203.

[7] L. Wenyin, and D. Dori, Genericity in graphics recognition algorithms, *Lecture Notes in Computer Science Vol. 1389*, 1998, 9-20.

[8] M.P. Deseilligny, R. Mariani and J. Labiche, Topographic maps automatic interpretation: Some proposed strategies, *Lecture Notes in Computer Science, Vol. 1389*, 1998, 175-193.

[9] S. Frischknecht, E. Kanani, Automatic interpretation of scanned topographic maps: a raster-based approach, *Lecture Notes in Computer Science, Vol. 1389*, 1998, 207-220.

[10] G.A. Nagy, S. Seth Samal, T. Fisher, E. Guthmann, K. Kalafala, L. Li, P. Sarkar, S. Sivasubramanian and Y. Xu, A prototype for adaptive association of street names with streets maps, *Lecture Notes in Computer Science Vol. 1389*, 1998, 302-313.

[11] S. Adam, J. M. Ogier, C. Cariou, R. Mullot, J. Labiche and J. Gardes, Symbol and character recognition: application to engineering drawings, *International Journal on Document Analysis and Recognition (IJDAR)*, 2000, 89-101.

[12] H. Hase, T. Shinokawa, M. Yoneda and C. Y. Suen, Character String Extraction from Color Documents, *Pattern Recognition*, 34(1), 2001, 1349-1365.

[13] A. Velázquez, J. H. Sossa and S. Levachkine, Reconocimiento eficiente de caracteres alfanuméricos provenientes de mapas ráster por medio de clasificadores neuronales, *Computación y Sistemas. Revista Iberoamericana de Computación*, 6(1), 2002, 38-50.

[14] D. Doermann, An introduction to vectorization and segmentation, *Lecture Notes in Computer Science, Vol. 1389*, 1998, 1-8.

[15] G. Nagy, Twenty years of document image analysis in PAMI, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1), 2000, 38-62.

[16] A. Ganesan, Integration of surveying and cadastral GIS: From field-to-fabric & land records-to-fabric, *Proceedings of 22nd ESRI User Conference*, 2002, http://gis.esri.com/library/userconf/proc02/abstracts/a0868.html

[17] L.A. Fletcher and R. Kasturi, A robust algorithm for text string separation from mixed text/graphics images, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(6), 1988, 910-918.

[18] C.L. Tan and P.O. Ng, Text extraction using pyramid, *Pattern Recognition*, 31(1), 1998, 63-72.

[19] A. Velázquez, Localización, recuperación e identificación de la capa de caracteres contenida en los planos cartográficos. *Ph.D. Thesis*. Centre for Computing Research-IPN. Mexico City, Mexico, 2002 (in Spanish).

[20] R. Cao and C.L. Tam, Text/Graphics separation in maps, *Lecture Notes in Computer Science, Vol. 2390*, 2002, 168-177.

[21] V. Alexandrov, M. Kharinov, A. Velázquez and S. Levachkine, Object-oriented color image segmentation, *Proc. IASTED International Conference on Signal Processing, Pattern Recognition, and Applications (SPPRA 2002)*, Crete, Greece, June 25-28, 2002, 493-498.

[22] A. Gelbukh and S. Levachkine, Error detection and correction in toponym recognition in cartographic maps, In: Levachkine, S. Bodansky, E., Ruas, A. (eds.), *e-Proceedings of International Workshop on Semantic Processing of Spatial Data (GEOPRO 2002)*, 3-4 December 2002, Mexico City, Mexico (2002) (CD ISBN: 970-18-8521-X).

[23] E. Bodansky, System approach to a R2V conversion: From research to commercial system, In: Levachkine, S. Bodansky, E., Ruas, A. (eds.), *e-Proceedings of International Workshop on Semantic Processing of Spatial Data (GEOPRO 2002)*, 3-4 December 2002, Mexico City, Mexico (2002) (CD ISBN: 970-18-8521-X).