Marçal Rusiñol[*], Rafael Roset[**], Josep Lladós[*], Carme Montaner[**]

Automatic index generation of digitized map series by coordinate extraction and interpretation

Keywords: Map Series; Index Sheets; Georeferencing; Computer Vision; Document Image Analysis

Summary
By means of computer vision algorithms scanned images of maps are processed in order to extract relevant geographic information from printed coordinate pairs. The meaningful information is then transformed into georeferencing information for each single map sheet, and the complete set is compiled to produce a graphical index sheet for the map series along with relevant metadata. The whole process is fully automated and trained to attain maximum effectivity and throughput.

Introduction

Map series or maps composed of multiple sheets are present in many map libraries, and are, in most cases, the greatest amount of collections. Series can have a very large number of sheets depending on the scale of the map and its territorial extension. Hence they are difficult to visualize and manage as a whole unless the index map of the distribution of the sheets is available. But many times the map series are not complete, and when they are so, map libraries might not have all the published sheets. Users need to locate the desired sheets on the chart and see if their map library of choice has them available. Often this query can not be answered due to the lack of the index sheet.

Usually that index map is provided by the map editor of the series or the map seller, and most times it is on paper. Lately the paper edition has been replaced by a PDF file, on which map libraries annotate the sheet availability. At present, however, the need for indices in vector format is becoming a basic need for digital catalogs. And that transformation is proving very difficult, because map libraries have and continue to receive paper maps and often lack the tools to draw vector graphics themselves.

There are currently several proposals put forward to address this situation but the final solution still seems far away. Getting the indices from the producers and sharing them among all map libraries seems a logical move but it has hit various objections, from economic interests in the sale of these indices, to the absence of vector indices for older series.

It is in this context that the Map Library of the Institut Cartogràfic de Catalunya has explored new means for obtaining these vector indices with the collaboration of the Computer Vision Center at the Universitat Autònoma de Barcelona. Relying on the project for mass digitization of its collection that the CTC started in 2007 we conducted a pilot test for the automatic extraction of the coordinates of the sheets in order to facilitate its automatic georeferencing. And by aggregation, the georeferencing of all sheets provides at the end the index map of those available and thus a graphic index can be obtained.

[*] Centre de Visió per Computador, Dept. Ciències de la Computació, Universitat Autònoma de Barcelona
[**] Institut Cartogràfic de Catalunya [rafael.roset@icc.cat]

For this purpose we selected the incomplete series "Ortofotomapa de Catalunya 1: 25 000" in B&W, published by the Institut Cartogràfic de Catalunya during 1985 to 1987, since it did not have an index map and the 59 published sheets were not contiguous. The original maps were archived at the Cartoteca de Catalunya as single flat paper sheets, and were digitized at 600 ppi using a Metis DRS2A0 scanner which delivered RGB TIFF files that were later downsampled to 300 ppi JPG files used in this demonstration.

At a time when most centers have begun a process to digitize their own collection, a tool that can be managed internally can become a very valuable resource. This article summarizes the methods employed for coordinate recognition, georeferencing and index map creation by means of computer vision techniques, as opposed to the usual GIS techniques.

The analysis and processing of map images has received a lot of interest from the early days of the Computer Vision and, in particular, from the Document Image Analysis and Recognition community. The existing literature applying computer vision techniques to map images is vast, and the researchers have focused in a number of different final applications. For instance, [7], [2] and [11] are examples of studies dealing with the interpretation of text elements (names of cities, rivers, etc.) appearing in maps. Other people focused their research on the understanding of graphic elements as opposed to textual ones. In [3] we can find a method to detect roads in maps and in [8] the authors propose a method to put into correspondence symbols from the legend and its appearances within the map image. Finally, another example is the analysis of Cadastral maps proposed in [5], [6] for the automatic segmentation of land properties. In the proposed work, we will just focus on specific textual information: the coordinate pairs that will help us to georeference a map sheet.

The remainder of this paper is organized as follows: section 2 is devoted to present an overview of the system. In section 3 the map segmentation step is presented, while in section 4 the coordinate pairs extraction and interpretation is introduced. Subsequently, in section 5, the final index sheet generation is presented. Section 6 provides the experimental results and finally section 7 is a summary and discussion of extensions and future work.

System Overview

In order to automatically generate an index sheet from a given set of digitized maps, as a byproduct of the georeferencing of each single map sheet composing the map series, our system is made of three main steps. Let us enumerate them and briefly overview their purpose.

The first step includes different pre-processing operations of the images and the segmentation between the map and the rest of background objects such as titles, legends, etc. By having the exact position within the image where the map is located, we can focus the next processing steps on each of the four corners of the map, where the coordinate pairs appear.

The subsequent step is in charge of coordinate pair identification, extraction and interpretation. Having defined a region of interest as a subpart of the image corresponding to a map corner, we need to identify which parts of these sub-images correspond to textual elements, then segment them and interpret them by an ad-hoc optical character recognizer (OCR).

The final step is devoted to the index sheet generation. For a complete map series the above steps have to be calculated for each of the images in the map series in order to build a single index sheet georeferencing the map images.

Segmentation between Map and Background

The digitalization of map series is done at high resolutions leading to huge file sizes. The fact of having such high quality images is a benefit for processing but is usually a burden in terms of computational time. Since the only information we have to extract from the map images in order to build an index sheet are the coordinates at the corners of the map, the first step of the presented method is devoted to performing a coarse segmentation of the map region.

For efficiency reasons, the coarse segmentation is done by using a low resolution version of the map images. Map images are resized by using a bicubic interpolation method, where the output pixel values are expressed as a weighted average of pixels in the nearest 4-by-4 neighborhoods. Then, these tiny color maps are converted to gray-level images and then thresholded in order to obtain a black and white version of the image. This thresholding operation is computed by using the adaptive binarization method presented by Niblack in [4].

The rationale behind the segmentation method is that the pixel density in the map area is usually much higher than the pixel density in the background zones. The proposed segmentation is a combination of three different methods that follow this rationale. Let us detail these three segmentation techniques.

Mathematical Morphology

The first technique for segmenting the map from the background is based on mathematical morphology operators [9]. The basic idea in mathematical morphology is to probe an image with a simple pre-defined shape, called the structuring element, drawing conclusions on how this shape fits or misses the shapes in the image. Given a simple structuring element (such as a 10x10 pixel square), we define a set of morphology operators that will filter the image by erasing the zones where the original image has less pixel density than the structuring element. In order to extract the map position from the filtered image, we perform an analysis of the connected components. Connected components are identified and labeled after scanning the image and grouping its pixels into components based on pixel connectivity. All pixels in a connected component share the same pixel intensity value and are in some way connected with each other. The map position corresponds to the bounding-box of the largest connected component of the filtered image. We can appreciate an example of the behavior of the whole segmentation process in Figure 1.

Figure 1: Map segmentation using mathematical morphology. a) Original image, b) binary image, c) filtered image after applying the morphology operators, d) original image with the bounding-box of the largest connected component.

Run-length Encoding

The second technique proposed to perform the map segmentation is based on a run-length encoding of pixels [12]. After obtaining the zones with high pixel densities by the application of some mathematical morphology operation to the binary image, the image is encoded by run-lengths. Traversing the image in the horizontal direction, for each image row, we count how many consecutive white pixels we have. These pixel sequences are called runs and their length will be thresholded. If the length of a given run is less than a certain threshold, the whole run is erased. The same operation is computed in the vertical direction for all the image columns. The resulting image is computed as the intersection of the horizontally and vertically filtered images. That is, the resulting image contains blocks that are greater than the established threshold in height and width. This simple encoding and filtering is used to get rid of all the small elements that are present in the image. The final map segmentation is determined by the bounding-box of the element with the biggest area in the filtered image. We can see an example of this technique in Figure 2.

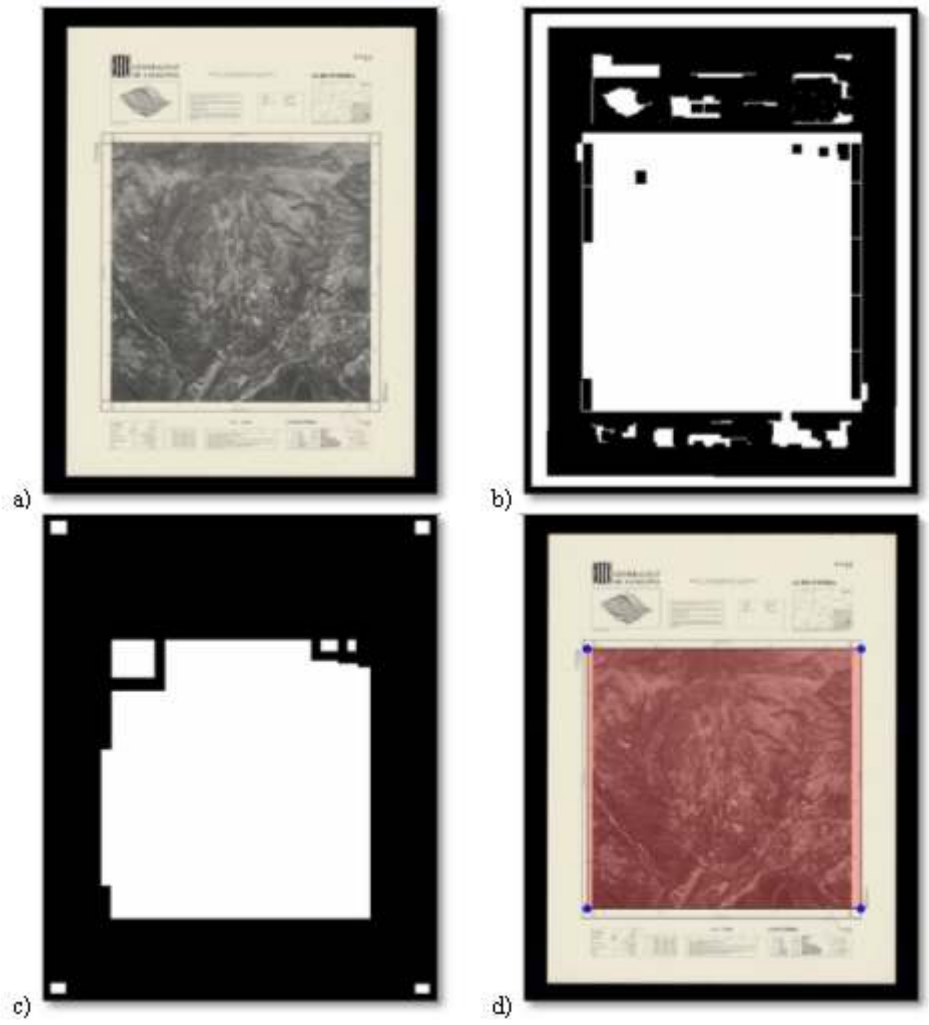

Figure 2: Map segmentation using run-lengths. a) Original image, b) image after applying morphology operators, c) filtered image after applying run-length encoding, d) original image with the bounding-box of the largest connected component.

Pixel Intensities Projections

The third segmentation technique is based on pixel intensities projections [1]. In the gray-level image, the pixels from the map are darker than the pixels in the background so they have a lower pixel intensity (in gray-level images, pure black has an intensity of 0 whereas pure white has an intensity of 255). The position of the map within the image can thus be determined by projecting the pixel values horizontally and vertically and looking for discontinuities in the pixel value sums. For the horizontal projection, given a row of the image, all their pixel values are added resulting in a signal with the same length as the image height. For the vertical projection the same procedure is applied column-wise. By looking for discontinuities (changing from a high intensity sum to a low one or vice versa) in the projection signals we can determine the coordinates of the corners of the bounding-box of the map within the image. We can see a representative example in Figure 3.

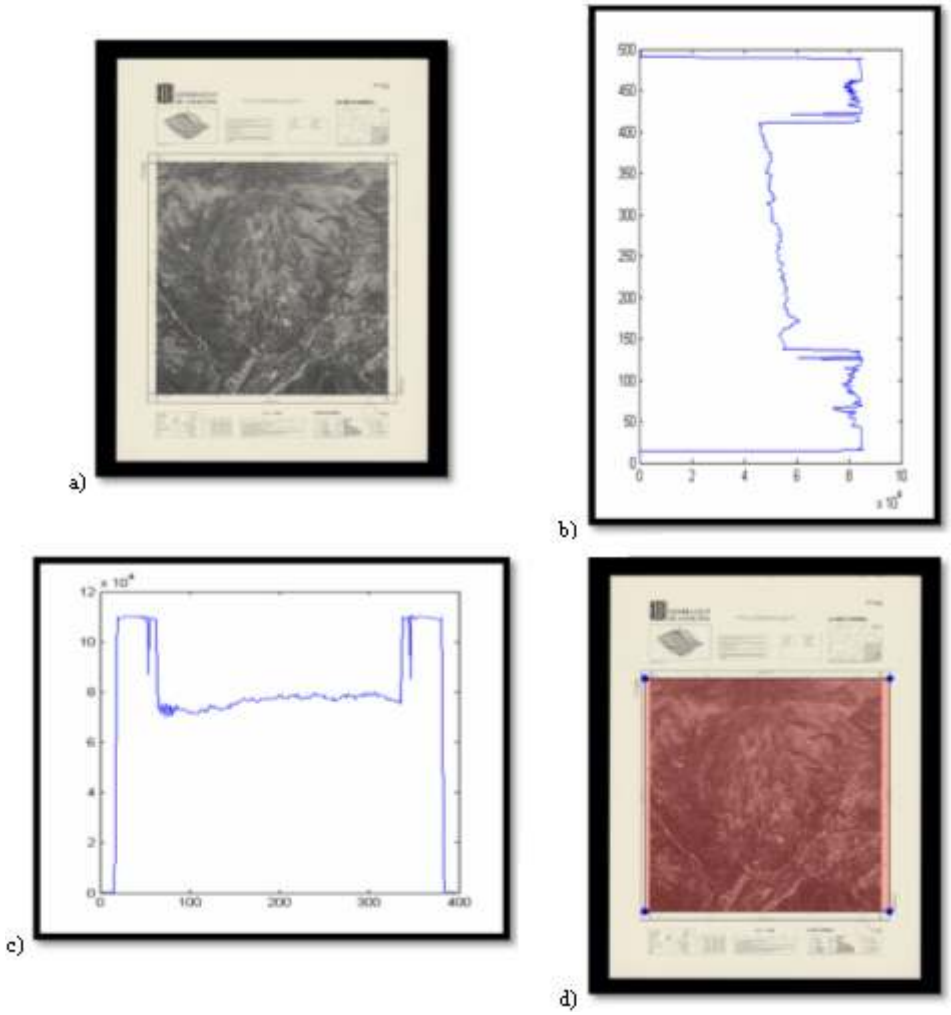

Figure 3: Map segmentation using pixel intensities projections. a) Original image, b) horizontal projection, c) vertical projection, d) original image with the bounding-box determined by projection discontinuities.

Segmentation Combination and Regions of Interest

Given an image from the map series and its three tentative segmentations, a validation procedure is applied in order to combine them. If all three methods propose similar segmentations, the final result is an average of the three different segmentations. If one of the segmentations differs a lot from the other two, this one is not taken into account, and the final one is the average among the two segmentations that are similar. If the three segmentations differ one from the other, the automatic process is stopped and we ask the user to validate which segmentation is the good one or to propose a new one.

Once we have a final segmentation of the map within the image, we can focus our further processing steps on the four regions of interest determined by the four corners of the map. Given the image coordinates of one of the corners, we can go back to the high-resolution image and crop a subimage centered in this coordinate. In order to be sure that the coordinates needed to georeference the image appear in the regions of interest, in our application scenario we defined these regions of interest as 600x600 pixel images. We can see an example in Figure 4.

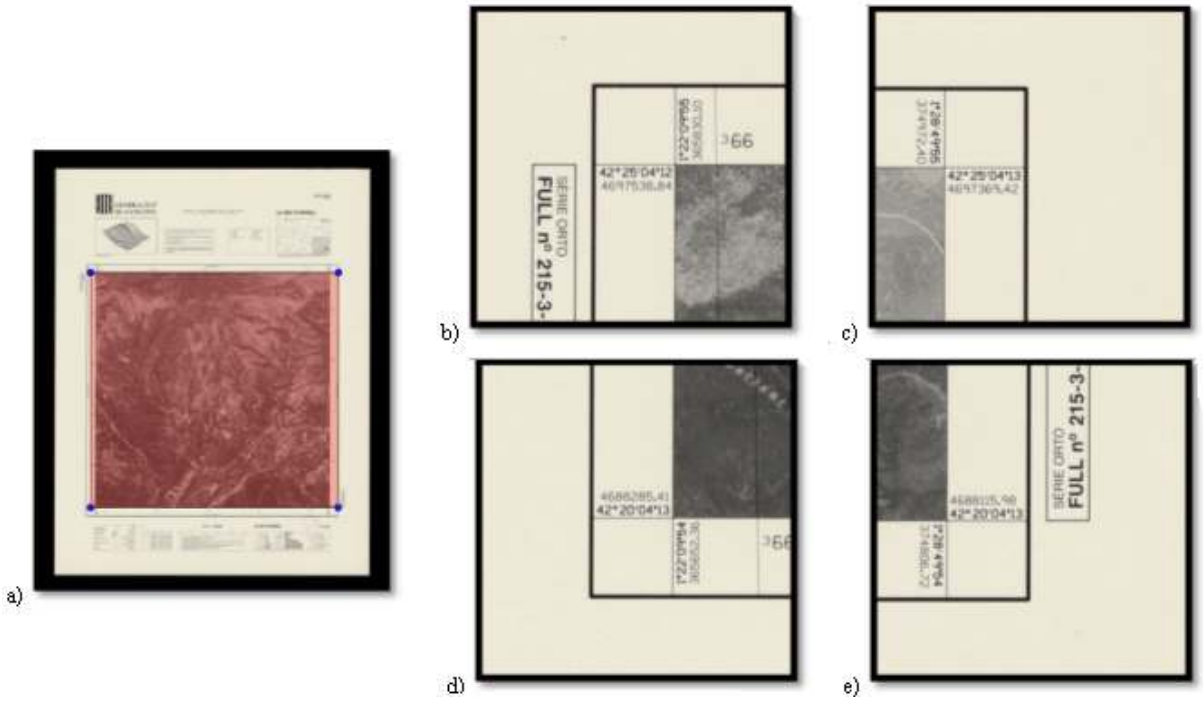

Figure 4: Regions of interest. a) Original image, b) North-West region, c) North-East region, d) South-West region, e) South-East region.

For each of these regions of interest, the next steps will be to extract and interpret the coordinate pairs in order to produce the four tuples needed to georeference the processed image.

Coordinate Pairs Extraction and Interpretation

This step is focused on reading the coordinate pairs in each of the four regions of interest from an image. We will separate this step into two different stages, the first one dealing with separating what is actually text content from graphical content. The second one dealing with reading the text content corresponding to coordinates.

Text/Graphics Separation

In order to separate what is text and what is graphics from the regions of interest, we have used the state-of-the-art method presented by Tombre et al. in [10]. The method aims at segmenting the document into two layers: a layer assumed to contain text and a layer containing graphical objects. Starting from the binarized image resulting after applying Niblack's algorithm [4], we run a connected component analysis. Several features of connected components (such as aspect-ratio, elongation factors, pixel densities, etc.) are analyzed in order to decide whether the connected component has to appear in the text layer or in the graphical layer. An example of text/graphics separation is shown in Figure 5.

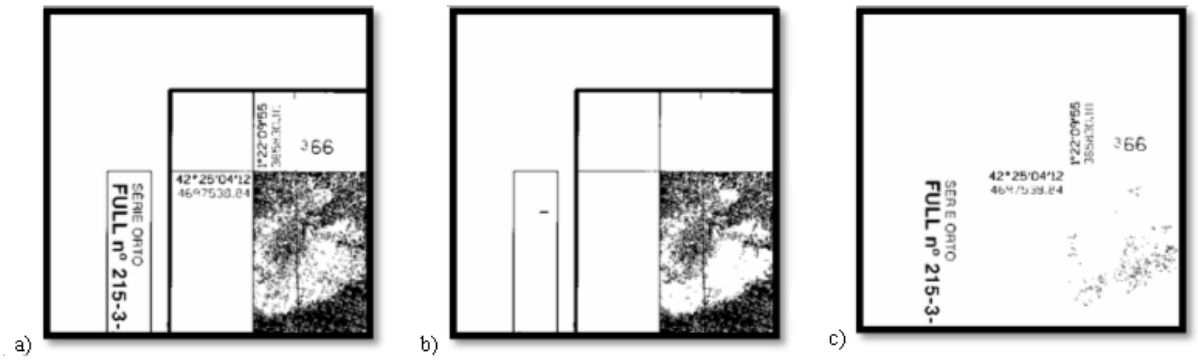

Figure 5: Text/Graphics separation. a) Binary image, b) graphical layer, c) textual layer.

After separating the image in layers, we have group connected components that are aligned and sharing the same properties in order to extract text blocks. This grouping of connected components can be easily achieved by applying mathematical morphology operations using horizontal or vertical rectangles as structuring elements. Figure 6 shows a text block identification example. For the text image, two blocks of horizontal and four blocks of vertical text are identified. These blocks are then fed to the OCR engine.

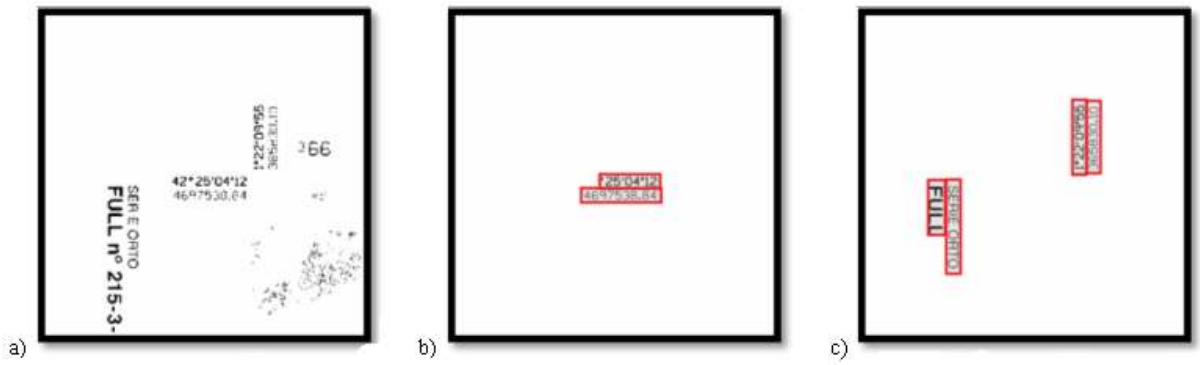

Figure 6: Horizontal and vertical text grouping. a) Textual layer of the region of interest, b) horizontal text blocks, c) vertical text blocks.

Optical Character Recognition Applied to Coordinate Pairs

In order to interpret the coordinate pairs we have used the commercial OCR engine from ABBYY[1]. Although we use a commercial OCR, we have applied some modifications in order to obtain the best performance when reading coordinate pairs. The first adjustment is to reduce the alphabet to be considered as a smaller set formed by just the digits and some symbols such as degree, minute, second, etc.

The second modification corresponds to the specification of regular expressions that will help us validate coordinate pairs. A regular expression defines a set of patterns that are valid in the language model one wants to consider. In our case, since we just want to consider coordinate pairs, we have built a set of regular expressions that model the coordinate syntax. Considering that the symbol # represents any digit from 0 to 9, some examples of regular expressions might be:

- the regular expression ##° ##’ ##” ## that validates strings as
  42°25’ 04” 12.

[1] http://finereader.abbyy.com/

- The regular expression [###### | #######].## that validates strings as 4697538.84.

The first example validates text blocks that are formed by two digits followed by the degree symbol plus two digits followed by the minute symbol, plus two digits followed by the seconds symbol, plus two final digits. The second example validates groups of six or seven digits followed by a period and two final digits.

With these two modifications we have built an ad-hoc OCR to be used for coordinate pair interpretation in maps. Since we have already identified which text blocks are horizontal and which are vertical, for a given region of interest, we know which coordinate corresponds to the longitude and which to the latitude. The text/graphics separation and the OCRing of coordinate pairs is computed for all the four corners of the map obtaining for each map sheet the eight coordinates needed to georeference the image. At this point the georeferencing information for each image is available and thus the image can be georeferenced using many known methods, and a world file for each image can also be created. The last step is devoted to using all the presented methods for a complete map series in order to generate an index sheet.

Index Sheet Generation

Given a set of images belonging to a unique map series, we can automatically extract georeferencing information for each single map sheet, and the complete set is compiled to produce a single index sheet for the map series in KML format.

Besides linking the coordinate pairs of the map with the image filename, the same method can be applied to enhance this index sheet with other metadata extracted from the original image files. In our application scenario, we also segmented and read the image title as well. The title segmentation is straightforward once the map zone is segmented. The title location in the image is quite stable regarding the map zone which is used as an anchor point to determine where the title might be. The same OCR engine has been used to read the map titles. Other metadata (as legends, scales, etc.) could be extracted as well by using the same techniques.

We can see in Figure 7 the generated index sheet viewed in Google Maps for the proposed map series. We can appreciate that only one of the 59 map images in the collection has been erroneously indexed. This error is provoked because the OCR interpreted the UTM coordinate 4.576.780,7 as 4.676.780,7. Looking at the index sheet in more detail we can observe some other small errors that provoke that the indexed regions are not completely squared. We can see however that these errors should be easy to correct manually by the operator.

Figure 7: Generated index sheet KML file viewed in Google Maps for the test map series.

Conclusions and Future Work

The process described in this document has proven to be valid for series of modern maps, but it is only a prototype and therefore needs to be improved to adapt it to other writings and other formats of maps. It is a however a very convenient option to automatically create index maps, while providing a very visual and accurate quality control of the georeferencing of each map sheet of a map series. Further development of this tool will allow adding the georeferencing and index map creation process to the digitization workflow with little effort.

The precision and accuracy of the process should improve when more metadata, the scale of the map and the scan resolution, will be added to validate the automatically extracted coordinate pairs. The development of an interface that allows the operator to fine tune segmentation, or to solve OCR recognition errors or to verify the final result will improve the whole process and accommodate other map formats. The automatic acquisition of the index of the sheets of a map series available at the map library will also ease the creation of the index for the whole map series by means of manually adding the remaining sheets with little effort and great accuracy.

Acknowledgements

This work has been partially supported by the Spanish projects TIN2006-15694-C02-02, TIN2008-04998, TIN2009-14633-C03-03 and CONSOLIDER – INGENIO 2010 (CSD2007-00018).

References

[1] T. Akiyama and I. Masuda. *A method of document-image segmentation based on projection profiles, stroke densities and circumscribed rectangles*. Systems and Computers in Japan 18(4): 101–111, 1987

[2] R. Cao and C. Tan. *Text/graphics separation in maps*. In Graphics Recognition Algorithms and Applications, volume 2390 of Lecture Notes on Computer Science, pages 167–177. 2002.

[3] Y.Y. Chiang and C.A. Knoblock. *A Method for Automatically Extracting Road Layers from Raster Maps*. In Proceedings of the 10th International Conference on Document Analysis and Recognition, pages 838–842, 2009.

[4] W. Niblack. *An Introduction to Digital Image Processing*, pages 115–116. Prentice Hall, 1986.

[5] R. Raveaux, J.C. Burie, and J.M. Ogier. *Object Extraction from Colour Cadastral Maps*. In Proceedings of the 8th IAPR International Workshop on Document Analysis Systems, pages 506–514, 2008.

[6] R. Raveaux, J.C. Burie, and J.M. Ogier. *A Segmentation Scheme Based on a Multi-graph Representation: Application to Colour Cadastral Maps*. In Graphics Recognition. Recent Advances and New Opportunities, volume 5046 of Lecture Notes on Computer Science, pages 202–212. 2008.

[7] P. P. Roy, U. Pal and J. Lladós. *Query Driven Word Retrieval in Graphical Documents*. In Proceedings of the 9th IAPR International Workshop on Document Analysis Systems, pages 191–198, 2010.

[8] H. Samet and A. Soffer. *A Legend-Driven Geographic Symbol Recognition System*. In Proceedings of the 12th International Conference on Pattern Recognition, pages 350–352, 1994.

[9] J. Serra. *Image Analysis and Mathematical Morphology*. Academic Press, 1982.

[10] K. Tombre, S. Tabbone, L. Pélissier, B. Lamiroy and P. Dosch. *Text/Graphics Separation Revisited*. In Proceedings of the 5th International Conference on Document Analysis and Recognition, pages 200–211, 2002.

[11] A. Velázquez and S. Levachkine. *Text/Graphics Separation and Recognition in Raster-Scanned Color Cartographic Maps*. In Graphics Recognition, volume 3088 of Lecture Notes on Computer Science, pages 63–74. 2004.

[12] F.M. Wahl, K.Y. Wong and R.G. Casey. *Block segmentation and text extraction in mixed text/image documents*. Computer Graphics and Image Processing 20: 375–390, 1982.