

VOTING-BASED LAYOUT ANALYSIS

Costin-Anton Boiangiu¹

Paul Boglis²

Georgiana Simion³

Radu Ioanitu⁴

ABSTRACT

As a part of the Computer Vision domain, layout analysis is the process through which the regions of interest from a document available as an image are being classified. A scanned file could be an example of such a document. The components of the layout analysis process are: the geometrical analysis and the logical layout.

The geometrical analysis involves the detection and labeling of differing regions or blocks of the image as being text, illustration, mathematic symbols, tables etc.

The logical layout refers to the detecting of the logical role that various regions have in the document (titles, footnotes, etc.).

The layout analysis process is intended to be performed before the document is sent or the OCR engine but it can also be used to identify copies of the same document or for indexing documents by structure. This paper presents an approach to analyze the layout based on a voting scheme, thus combining a series of algorithms and using weighted, majority and unanimous votes, with the purpose of increasing the accuracy of the results.

KEYWORDS: OCR, layout analysis, voting system, X-Y cut, whitespace cover, smearing, Voronoi diagrams, image segmentation

1. INTRODUCTION

Although there are many cases in which most of the information is still stored on paper documents, in the current age of technology, the electronic manner of storing and processing the information is used usually over printed files.

In order to convert printed documents in an electronic format that can be indexed by their structure, there is a need for a system capable of this action. The first thing to do when aiming to convert a printed document in electronic format is to use Optical Character Recognition processing (OCR). This step needs information regarding the layout of the document, using some software solutions which either try to detect the layout and make conventions depending on page characteristics or require through an user interface detailed information about the page layout (for example: image regions, the number of columns in

¹ Associate Professor PhD Eng., costin.boiangiu@cs.pub.ro , "Politehnica" University of Bucharest, 060042 Bucharest, Romania

² Engineer, Intel Romania, 0203 Bucharest, Romania, boglis.paul@intel.com

³ Engineer, Telmap Romania, 030039 Bucharest, Romania, georgiana.simion@telmap.com

⁴ Engineer, European Patent Office (EPO), Bayerstrasse 34, Munich, Germany, rioanitu@epo.org

a given table, or any other information which the software is not able to automatically detect).

Layout analysis is also used for very large archives, where it is needed to compare documents based on their layout. The process of layout analysis makes possible the identification of key document sections which can be used to classify the document and perform significant indexing.

There are some methods for solving the problem of layout analysis. The most popular approach is based on geometrical methods [1], [4], but there are others which try to combine geometrical methods with semantic knowledge [2] as statistical approaches [3].

Each and every method has its drawbacks therefore a perfect method that will gain perfect results for any type of document doesn't exist.

This document presents a layout analysis method based on voting. This is a solution which combines a number of algorithms in the purpose of solving the layout analysis problem and obtains a single solution for each algorithm.

A voting system uses these solutions in order to obtain the optimal aggregated solution that would gain better overall performance. This solution may be used on a larger set of documents types than any individually considered algorithm. The essential fact behind the voting system is that the specifics of each algorithm can help in balancing the aggregated results and that usually the judgment of a group goes beyond to the one of the individuals.

The voting system proposed in this document allows the usage of weighted, majority and unanimous votes.

2. RELATED WORK

In other studies regarding layout analysis, top-down [6] and bottom-up [5],[6] solutions have been both used. To work with bottom-up solutions it is necessary to extract the information from a bitmap image and then group them by the number of common features like the density of neighbors. In Hybrid methods [7] the assembling of base block into larger components is realized by combining top-down and bottom-up approaches.

In [2] is presented an approach which combines semantic knowledge with geometrical methods. Known methods are used like: the transformation of the image into binary, the horizon angle of the content of the image is aligned and the page segmentation. After this, the result is divided and some of the components can be combined, based on prior numerical knowledge. The general analysis made by the method covers section, paragraph identification, columns, figures and tables and even local analysis that makes the connection between blocks of the same type.

Voronoi Area Diagrams [8] are meant for layout identification. Through this approach, non-rectangular blocks can be handled, which is usually the standard unit for most algorithms; however this is not always correct, so its applicability for layout analysis algorithm is therefore limited.

The other method presented in [1] performs the extraction of specific information from images and also makes some measurements. Then the components are grouped based on the obtained measurements, creating a cluster of components. From this group, measurements are extracted, for example: the distance between characters, their skew angle, the distance between lines and so on. In order to reconstruct the lines, paragraphs and obtain the segmented image, the extracted distances are used.

3. SUMMARY OF THE USED METHOD

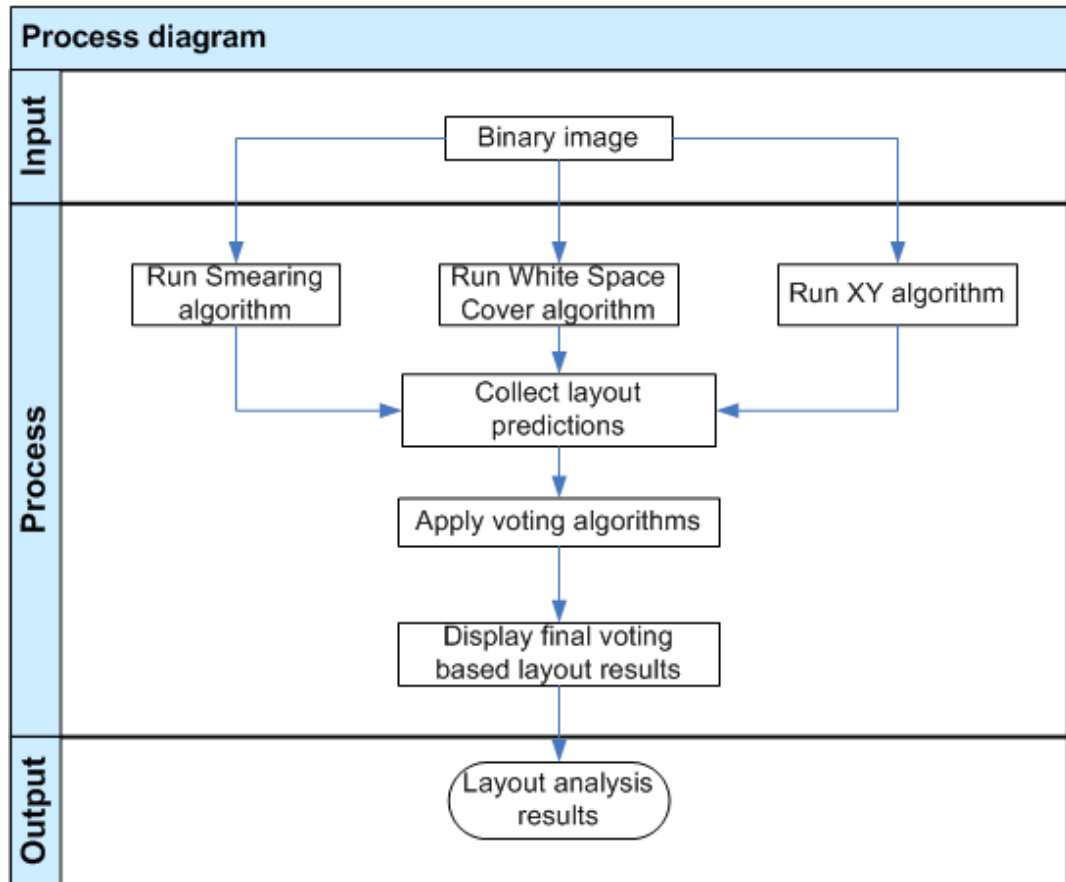


Figure 1. Block schema of the application

This paper is aiming to show that using combined simple segmentation techniques for layout analysis can lead to very good results. A very reliable combination of such methods can be implemented by using voting systems.

In order to gain the optimal results, some voting systems have been used: majority voting, unanimous voting and weighted voting.

The simplest voting system is represented by majority voting. It is very efficient, previous knowledge about the results that are being decided is not needed and does not require a large set of data to be analyzed. This system has very simple rules: every vote has a fixed

weight and a fixed probability of occurrence, being independent of the other types of voting. Its advantage is that it is almost impossible that more analyzed results can produce the same segmentation result, so almost all errors are reduced.

The other voting system, weighted voting, can gain output good results if it is properly used. It needs knowledge about previous data inputs and although the votes are independent, the weights differ. Establishing the weight can be done based on various criteria, like computing the efficiency of each algorithm on some test data and correlating the weights with these results. Another method would be dependent on the layout of the analyzed page, so if an algorithm produced good results first for rectangular components and then for pages of this kind, that algorithm would have a higher weight on documents of this type.

Unanimous voting system is also very simple. The votes are independent and have the same weight. Because of the unanimity of the voting process, the errors are eliminated through voting. As a drawback: there is the possibility that in this system, valid components will be eliminated too.

4. ALGORITHMS USED FOR OBTAINING THE INPUT DATA

X-Y cut

X-Y cut is a top-down algorithm which partitions a document into rectangular components.

The bits of the binary transformed image are summed by this algorithm. In this manner a density graph is obtained. The low ends of the graph stand for empty spaces that are lines in the segmented document. If the values reach a higher point than it, the segmentation process is stopped and the layout component is identified. This algorithm is usable for both horizontal and vertical projections. The process is continued until an empty line threshold is reached. At this point the segmentation algorithm ends.

A positive aspect of the algorithm is that the threshold controls the size of the found component. This way, the configuration setting makes the algorithm suitable for finding paragraphs, lines or words and many other elements of the page by simply performing an adjustment to one parameter.

More than that, the threshold controls the size of the segmented clusters which makes the algorithm scalable. The detection of the rows, paragraphs, section and so on is therefore possible.

Whitespace cover

The whitespace cover algorithm considers a collection of rectangular components as well as another component that represents the entire page, and is a container for all the other components. The main idea behind this algorithm is the maximization of the white spaces in order to obtain the optimal page. The algorithm principle is similar to quicksort.

The starting point of the algorithm is represented by a rectangular component that bounds the whole page and a group of white rectangles which are considered obstacles. One of

these obstacles is chosen as a pivot. Usually it is chosen one that is as central as possible. With this pivot the space is split in 4 components (right, left, top, down) which become candidates for being processed in the same way recursively. Every component is tested with a quality function in order to evaluate if there is a white space or not.

Smearing

This algorithm starts from the idea of extending black pixels from a binary image. This process aims to make the reconstruction of lost structures resulting from the digital conversion process by having groups of pixels turned into black where the number of white pixels is in minority based on a predetermined threshold.

In the traversing process of the binary image, if the number reached by 0 pixels (in binary image this value stands for white color) is below a limit that have been given, then the neighbor pixels are transformed into 1 (value for black). If the lower limit is not reached or exceeded, the pixels are not changed. The traversal is made both horizontally and vertically and the results are then combined to gain the final solution.

The algorithm is effective and simple and it depends on how the threshold value is set, value that depends on the type of document, namely the information density which is reflected in the number of black pixels. An important drawback of this algorithm is that it can only work with rectangular structures, as the traversal is done linearly.

5. TESTS AND RESULTS

There is one aspect that deserves special attention: the calibration process influences the individual algorithms chosen, this leading to variation of results.

The OpenCV library is the base of the current implementation. It uses the OpenCV layout algorithms in order to obtain voting input data. The application is built with extensibility in mind, taking a framework approach where adding a new algorithm is as simple as implementing an interface. All configured algorithms are run in parallel which inflicts a good efficiency of the framework.

The weights assigned for the weighted system have been chosen based on the previously known information about the algorithms.

In Figure 2 are presented the start input image and the results of the voting algorithm after running and comparing the results of the X-Y cut, whitespace cover and smearing algorithm. It is easy to observe that the results are good and they can easily be transformed into bigger cluster if more complex page hierarchies are necessary. As expected, the weight voting obtained the best results. This fact successfully balances the strong and weak points of the chosen input algorithms.

In Figure 3 are presented the start input image and the results of the same voting strategies based on the same input layout analysis algorithms that were described above. This time we have a more complicated input layout even if we consider the standard of the modern

German newspapers. In this case, more columns are separated by different height paragraphs and then are joined in the same article by the upper title which runs across them.

In such cases, methods like X-Y cut almost always fail as a successful order of cuts (or not from unassisted heuristics) cannot be found for the correct detachment of all articles.

The whitespace cover method also fails frequently because there are not enough long columns that can be used in order to limit the computational explosion. From time to time most of the columns that run across all document are traversed by titles therefore they are getting fragmented and the whitespace cover quality function may reject them.

The smearing approach isn't very helpful either; the method is too simple to endure on complex input with very different font sizes on a single page and as a result narrow columns may be ignored and blocks could be grouped in a wrong way.

The voting, however, produced encouraging results: some of the errors are corrected and extremely few are introduced by the voting. Again, as expected, the weighted voting is the most efficient method; it combines the algorithms advantages very well. As a result it may prove in time to be the most robust.

A much bigger project, a complete, modular, fully automatic content conversion system developed for educational purposes contains the method discussed in this paper as an important component. When ready, this system will be very helpful in fully automatically processing of large batches of documents. As a result, other problems may occur and the current input algorithms may take advantage of better overall weighting as well as by implementing another strong and general layout algorithms similar to the one based on area Voronoi diagrams.

There is the need of another method to be furthered investigated: the performing of internal tuning into the layout algorithms in order to tend toward over-segmentation and to apply unanimous voting to make the final decision.

By making the weighting mechanism more flexible at each image page run, the weight voting may obtain substantial benefits. In other words, if after a series of processed documents algorithm A is observed to execute over-segmentation then his weight heuristic will encourage not accepting splitting decisions of other algorithms on his generated clusters. On the other hand if algorithm B is observed to perform sub-segmentation then his adapted weight heuristic will not encourage join operations applied on his generated clusters requested by other algorithms.

down the money. Nicuesa gazed at him with astonishment, and could scarce believe his senses, but when he saw him actually pay off the debt and found himself suddenly released from this dreadful embarrassment, he embraced his deliverer with tears of gratitude, and hastened with all speed to embark, lest some other legal spell should be laid upon his person.

CHAPTER III.

Exploits and Disasters of Ofeda on the Coast of Cartagena—Fate of the veteran Juan de la Cruz.—(1500.)

It was on the 10th of November, 1500, that Alonso de Ofeda set sail from San Domingo with two ships, two brigantines, and three hundred men. He took with him also twelve beed mares. Among the remarkable adventurers who embarked with him was Francisco Pizarro, who was afterwards renowned as the conqueror of Peru.* Her-

* Francisco Pizarro was a native of Trastara in Extremadura. He was the illegitimate fruit of an union between Gonzalo Pizarro, a veteran captain of infantry, and a dancer in low life. His childhood was passed in gossamer occupations incident to the humble condition of his mother, and he is said to have been a swindler. When he had sufficiently increased in years and stature he enlisted as a soldier. His first campaign may have been against the Moors in the war of Granada. He certainly served in Italy under the banner of the Great Captain, Gonzalo of Cordova. His roving spirit then induced him to join the bands of adventurers to the New

Initial image



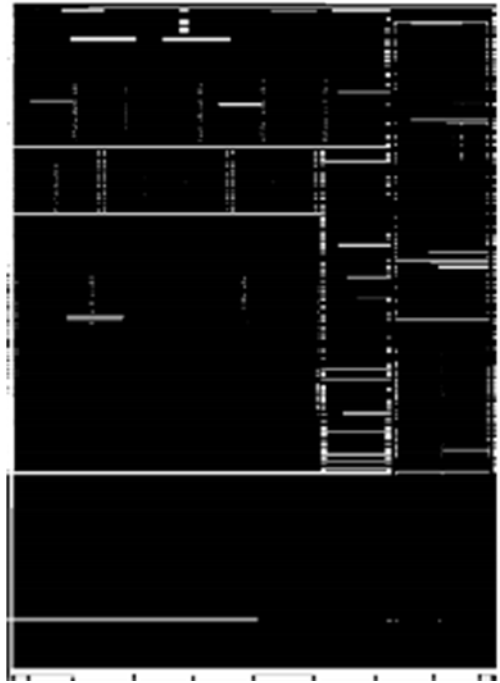
Majority voting



Weighted voting

Unanimous voting

Figure 2. Simple layout input results



Initial image

Majority voting



Weighted voting

Unanimous voting

Figure 3. Complex layout input results

6. CONCLUSIONS

This work analyses some possible methods of solving the document layout analysis problem and show a new solution which is based on the combination of various algorithms. A voting system is used to perform the combination of individual algorithm results.

Three voting systems are employed: unanimous voting, majority voting and weighted voted.

In the case of simple layouts the results are very promising. In the same time, for more complex layouts, they lead to lower improvements when compared with individual algorithm results.

The improvement of framework algorithms can be easily done because of how the solution is designed and new algorithms can be implemented for use.

7. FUTURE WORK

Adding new algorithms based on Voronoi Areas and also the adding of statistical algorithms that would influence semantic information can be considered as one possible improvement.

One way of improving the weighted voting process would be to apply heuristics to analyze each document and adjust the weights as needed. Precise weights assignments provide the promise of best results.

In the case of unanimous voting, a combination with the majority voting system can be realized or the least promising result can be simply discarded before voting.

This application can be used in an OCR system providing better accuracy for the layout analysis processes.

8. REFERENCES

- [1] T. M. Breuel, „Document Layout Analysis”, Image Understanding and Pattern Recognition Group, Diploma paper.
- [2] F. Esposito, D. Malerba , G. Semeraro, “A Knowledge-Based Approach to the Layout Analysis”, Document Analysis and Recognition, pp. 466-471, 1995.
- [3] T. A. Tokuyasu, P. A. Chou, “Turbo recognition: a statistical approach to layout analysis”, Proceedings of SPIE, 2000.
- [4] T. M. Breuel, “Two Geometric Algorithms for Layout Analysis”, Lecture Notes in Computer Science, vol. 2423/2002, pp. 687-692, 2002.
- [5] A. Simon, “A fast algorithm for bottom-up document layout analysis”, Pattern Analysis and Machine Intelligence, vol. 19, no.3, pp. 273-277, 1997.
- [6] L. O’Gorman, “The document spectrum for page layout analysis”, Pattern Analysis and Machine Intelligence, vol. 15, no. 11, pp. 1162-1173, 1993.
- [7] A. Dengel, G. Barth, “ANASTASIL: hybrid knowledge-based system for document layout analysis”, Proceedings of the 11th international joint conference on Artificial intelligence, vol. 2, pp. 1249-1254.
- [8] K. Kise, M. Iwata and K. Matsumoto, “On the Application of Voronoi Diagrams to Page Segmentation”, Department of Computer and Systems Sciences, College of Engineering, Osaka Prefecture University, 1999.