Extraction of Tabular Data from Document Images

Manolis Vasileiadis, Nikolaos Kaklanis, Konstantinos Votis and Dimitrios Tzovaras
Information Technologies Institute, Centre for Research and Technology Hellas
6th km Charilaou-Thermi Road, 57001, Thermi-Thessaloniki, Greece
{mavasile, nkak, kvotis, tzovaras}@iti.gr

ABSTRACT

In this paper, we propose a heuristics-based method for automatic detection and extraction of tabular data from document images. The proposed approach utilizes page segmentation techniques, along with an OCR engine, in order to acquire the text data and bounding boxes of each word in the document. These elements are then grouped in a bottom-up fashion, based on a series of rules, in order to identify and reconstruct tabular arrangements of data. Based on this methodology, an open source cross-platform tool capable of recognizing the semantic structure of documents containing tabular data has been implemented, thus widening the range of document types than can be successfully converted into alternative accessible formats, suitable for users with visual impairments.

CCS Concepts

• Information systems ~ Structured text search • Information system ~ Document structure • Applied computing ~ Document management and text processing • *Human-centered computing ~ Accessibility technologies.*

Keywords

Table recognition; OCR; document image

1. INTRODUCTION

Tables are strictly formatted text structures, used to efficiently represent data and information in a variety of documents and reports. In order to make this data accessible to users with visual impairments, simply acquiring the textual content of the words is not sufficient; it is critical to identify their basic building blocks (table cells) and group them together in rows and columns. Thus, in order to extract and understand the information stored in a table, the raw data generated by an OCR engine must be further processed in order to recreate the table's layout and structure.

Various methods have been proposed for automatic table recognition, including bottom-up approaches performing grouping of words in segments by also utilizing line-by-line analysis for the selection of initial table areas [1]. Heuristics-based methods for detection of figures and tables in strictly structured scientific papers have been also presented [2][3]; these methods, however, are limited by their reliance on document layout-specific features (i.e. using keywords such as "Table", grid lines etc.). Moreover, machine learning techniques have also been utilized [4] for text classification.

W4A 2017, April 02-04, 2017, Perth, Western Australia, Australia
ACM 978-1-4503-4900-0/17/04.
http://dx.doi.org/10.1145/3058555.3058581

Towards this end, we introduce a heuristics-based method for automatic table detection, which builds on [1], while proposing a novel method for the grouping of the word segments in table columns and multiple-line table rows. The method does not require any prior training, and can be applied to a variety of documents as it does not rely on document layout-specific features, but treats tables as strictly formatted text segments.

2. PROPOSED METHODOLOGY

According to the proposed methodology, document images (either scanned or digitally created) are exported to simple HTML format, while tabular structures present in the document are also reconstructed. Detailed descriptions of these steps are provided in the following paragraphs. A corresponding open-source software tool[1] has also been developed to support the automation of this process.

2.1 Data Preprocessing

2.1.1 Image Processing

Initially, the image is binarized using Wolf-Jolion binarization [5]. Then, it is resized and resampled in order to approximate a 300dpi A4 page (optimal for the Tesseract OCR engine), while a simple line detector is used to remove any visible grid lines. Next, page analysis is performed using the open source Leptonica library[2] which creates masks that correspond to blobs of discrete text and non-text areas (Figure 1a), with the non-text areas being discarded.

The text-only image is recursively searched for continuous horizontal empty spaces, which can indicate multi-column areas. The algorithm selects subspaces between horizontal empty areas and searches within them for vertical empty spaces that pass through the whole height of the subspace. Any areas found are considered as potential document column separators, and if the standard deviation of their width is smaller than a thresholdD_{thresh}, the subspace is registered as a multicolumn text area. All the detected multicolumn text areas are sequentially vertically reordered, in top-to-bottom, left-to-right order (Figure 1b).

2.1.2 Optical Character Recognition

The detected text areas are transformed into readable, editable text, using Google's Tesseract v3.04 OCR[3] engine, which exports for each recognized word: a) the text data, b) the bounding box, c) the recognition confidence and d) font information.

2.2 Table Reconstruction

The table detection and reconstruction algorithm includes four steps, each following a set of heuristic-based rules in order to

[1] https://github.com/P4ALLcerthiti/P4ALL_OCR-TABLES

[2] http://www.leptonica.org

[3] https://github.com/tesseract-ocr/tesseract

recognize potential tabular structures, based on the assumption that text tables are areas of strictly formatted text.

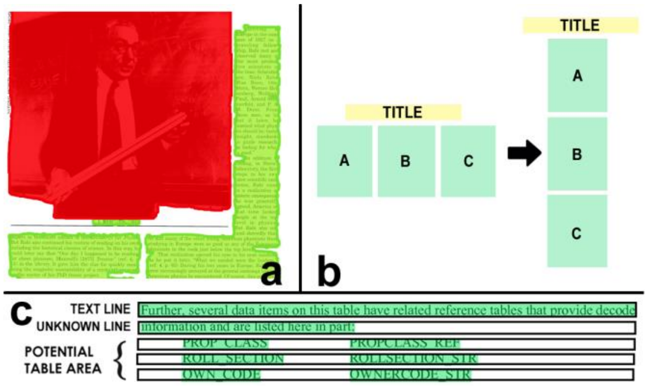
a
b
TITLE
A
B
C
TITLE
A
B
C
c
TEXT LINE
UNKNOWN LINE
POTENTIAL TABLE AREA
Further, several data items on this table have related reference tables that provide decode
information and are listed here in part:
PROP_CLASS
PROPCLASS_REF
ROLL_SECTION
ROLLSECTION_STR
OWN_CODE
OWNERCODE_STR

Figure 1. a) Page Segmentation: text/non-text, b) multi-column text reordering, c) word segments & text lines

2.2.1 Text Lines and Word Segments

Text lines are created from words that overlap vertically, using the Cartesian coordinates of each word. For each line, word segments are defined, by merging together words whose horizontal distance is below a threshold T_{seg} (dense text). Finally, each line is assigned to one of three types (Figure 1c): 1) Text – Lines with a single long segment, 2) Table – Lines with more than one segments, and 3) Unknown – Lines with a single short segment

For multiple-page documents, the top and bottom areas of each page are also checked for similar repetitive word segments which are registered as footers/headers and, if such segments are identified, they are removed.

2.2.2 Initial Table Areas

Initial table areas are created by grouping adjacent lines of type 2 and 3 together. However, for a type 3 line to be assigned to a table area, it must have at least one type 2 line above it.

2.2.3 Table Column Generation

For each table area, word segments are selected as column generators. The corresponding algorithm can be described as follows:

1. Select the left-most segment not assigned to a column.
2. Find all the segments that horizontally align with it.
3. Estimate the average length of these segments.
4. Select the segment that is closest to the average length found on step 4 as a column generator.
5. Assign all the segments that horizontally overlap with the column generator to a new column.
6. Go back to step 1, until all the segments in a table area are assigned to a column.

The newly generated columns are then further customized:

- If a column has more multiple-column than single-column segments, it is merged to the column on its left.
- If a column has only one segment, which is on the 1st table row, while the column on its left does not have a segment in the same row, these columns are merged.
- Tables with a single column are treated as simple text.
- Type 3 lines at the bottom of a table, with a multiple-column segment, are removed from the table.
- If each column of a table has more empty cells than cells with data, the table is discarded.

2.2.4 Multiple-line Table Rows

Some of the table rows are merged into multiple-line rows:

- If a table row does not have a segment in the first column, and there is one-to-one correspondence with the segments of the row above, these two rows are merged.
- If a type 3 table row has its single segment assigned to the first column, and the row below it has an indented segment in the first column, these two rows are merged.
- Tables ending up with a single row or two single-line rows are discarded and treated as simple text.

3. EXPERIMENTAL RESULTS

The developed implementation was experimentally evaluated on a set of 45 random document images retrieved from the internet, including various table layouts (e.g. with/without gridlines, multi-line/column text, with/without headers etc.) and achieved high precision and recall rates: 88.30% and 97.22% for tables, and 89.45% and 93.70% for cells respectively. False table detections were mainly observed in cases of formatted text (i.e. right text alignment), while the cell accuracy was decreased when the table data were badly aligned (i.e. misaligned table headers). Some sample experimental document images can be found at: http://160.40.50.183/table_recognition/OCR_Tables_Samples/.

4. CONCLUSIONS & FUTURE WORK

This paper presented a heuristics-based approach for automatic extraction of tabular data from document images. Potential improvements may include further refinement of the page analysis algorithms, which seem to have the largest effect on the recognition accuracy, as well as refinements and additions to the table-generation rules. Moreover, further user testing will be conducted to evaluate the algorithm's efficiency, while the tool's higher-level interface will be updated so as to make it accessible to its target audience of people with visual impairments.

5. ACKNOWLEDGMENTS

This work is partially funded by the EU FP7 project Prosperity4All, contract no. 610510.

6. REFERENCES

[1] Oro, E. and Ruffolo, M., 2009, July. PDF-TREX: An approach for recognizing and extracting tables from PDF documents. In *Document Analysis and Recognition, 2009. ICDAR'09. 10th International Conference on* (pp. 906-910). IEEE. DOI= http://doi.org/10.1109/ICDAR.2009.12.

[2] Clark, C. and Divvala, S., 2015, April. Looking beyond text: Extracting figures, tables, and captions from computer science paper. In *Workshops at the Twenty-Ninth AAAI Conference on Artificial Intelligence*.

[3] Klampfl, S., Jack, K. and Kern, R., 2014. A comparison of two unsupervised table recognition methods from digital scientific articles. *D-Lib Magazine*, *20*(11), p.7.

[4] Bansal, A., Harit, G. and Roy, S.D., 2014, December. Table Extraction from Document Images using Fixed Point Model. In *Proceedings of the 2014 Indian Conference on Computer Vision Graphics and Image Processing* (p. 67). ACM. DOI= https://doi.org/10.1145/2683483.2683550.

[5] Wolf, C., Jolion, J.M. and Chassaing, F., 2002. Text localization, enhancement and binarization in multimedia documents. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on* (Vol. 2, pp. 1037-1040). IEEE. DOI= http://doi.org/10.1109/ICPR.2002.1048482