Document Representation Refinement for Precise Region Description

Christian Clausner, Stefan Pletschacher and Apostolos Antonacopoulos
PRImA Lab, School of Computing, Science and Engineering, University of Salford,
Greater Manchester, M5 4WT, United Kingdom
http://www.primaresearch.org

ABSTRACT
Precise description of layout entities (content regions on a page) is crucial for all but the most trivial document analysis and recognition applications. The output of layout analysis methods and state-of-the-art OCR systems varies significantly, from bounding boxes (e.g. Tesseract) to stacks of text line rectangles (e.g. ABBYY FineReader). There is a clear need for a consistent and accurate representation of regions (e.g. text paragraphs, graphics entities etc.) for further processing, correction and performance evaluation (comparison of segmentation results with ground truth regions). This paper describes a method for refinement of document representations by fitting polygons around lower-level layout objects (such as text lines, words and glyphs) in a systematic way that reconstructs region outlines and preserves the fine details of complex layouts. Experimental results on a standard dataset demonstrate the validity and usefulness of the proposed approach.

Categories and Subject Descriptors
I.4.9 [**Image Processing and Computer Vision**]: Applications.

General Terms
Algorithms, Performance, Experimentation.

Keywords
Document Image Analysis, Polygonal Fitting, Segmentation.

1. INTRODUCTION
Page reading systems (OCR engines) have advanced considerably since their advent in the 1980s. It is now possible to process documents that are significantly more complex than a simple book page or a form with fixed layout. All modern recognition systems use – and usually make accessible – a detailed document model including layout, structure and text content. The physical and logical elements of a document page, defined by layout and structural descriptions, are not only crucial for applications such as document reconstruction, visualisation, and repurposing but also for pre-production of ground truth data and performance analysis of the recognition methods involved.

DATeCH 2014, May 19 - 20 2014, Madrid, Spain

http://dx.doi.org/10.1145/2595188.2595198

Results of state-of-the-art layout analysis and OCR systems such as Tesseract [1] and ABBYY FineReader Engine are a valuable asset for many research and production applications in the field of Document Analysis. Apart from the option to output the results in file formats supported by the systems (for instance hOCR [2] or ALTO XML [3]) the recognised document model can be directly accessed through application programming interfaces (APIs) provided by all major OCR system vendors.

In almost all cases, where document layout data is to be processed, it is of considerable advantage to have access to geometrical region descriptions that are as precise as possible. The precision of the aforementioned layout analysis systems is limited to bounding boxes (Tesseract) or stacks of rectangles (FineReader). Especially for more complex layouts this is potentially insufficient.

Figure 1 shows an example with regions whose descriptions severely overlap. Some regions are almost completely covered by the bounding boxes of adjacent regions. Further processing based on such a description using boxes will suffer major problems.

This paper describes a systematic approach to refine region outlines by fitting polygons around lower-level layout objects (such as text lines, words and glyphs). The aim is to consistently reconstruct and accurately represent higher-level regions, preserving the fine-details of complex layouts for further analysis and recognition tasks.

The refined results are exported to PAGE (Page Analysis and Ground Truth Elements, [4]) XML which uses (arbitrary) polygons for the outlines of layout objects. The format is supported by a range of ground-truthing and performance analysis tools and has been used in major digitisation projects [5][6].

The refinement approach is described in Section II, followed by its experimental validation and evaluation in Section III, and concluding remarks in Section IV.

2. POLYGONAL FITTING
The polygonal fitting can be applied to text regions that have child objects in the document model. A typical object hierarchy contains regions, text lines, words and glyphs (characters). Outlines are re-calculated bottom-up, starting at the lowest available level and finishing at region level. The level of detail (supported levels of child objects) is predetermined by the given API or layout description format. The current ALTO XML format, for instance, does not define geometric attributes (position, shape or dimension) for characters, making words (ALTO: String) the lowest level.

For each layout object a polygon is fitted around its direct child objects using the following processing steps (see Fig. 2):

- Create an empty (white) bitmap and fill in the child objects as foreground (black)
- Fill the gaps between the child objects by using horizontal, vertical and diagonal run-length smearing
- Optional: Exclude neighbour regions by removing the foreground (filling in as white)
- Trace the contour of the foreground and create a polygon

The following subsections describe the steps in detail.

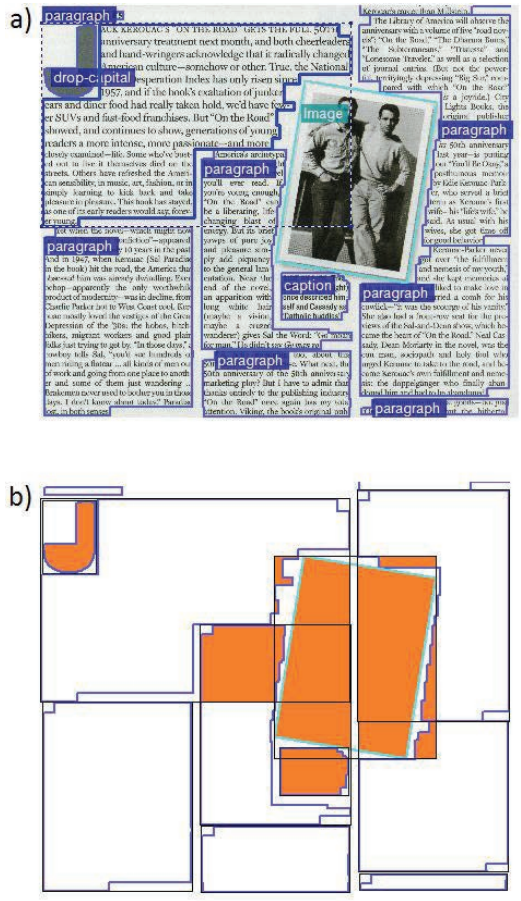

Figure 1. Example with overlap; a) Ground truth with polygonal outlines; b) Polygons (blue), bounding boxes (black) and conflicting overlap areas (orange).

2.1 Transferring a Polygonal (Child) Object to a Bitmap

First, as a prerequisite for the following step, the polygonal outline of each child object is converted to isothetic format by closely approximating diagonal polygon segments with a sequence of horizontal and vertical segments (see Fig. 3b). Then, rectangle-based interval representations [7] are calculated (Fig. 3c). All rectangles are finally transferred to the bitmap as foreground (black).

2.2 Run-Length Smearing Approach

The goal of this step is to connect all foreground objects in the bitmap by filling the gaps in-between using run-length smearing [8]. In a first stage, alternatingly horizontal and vertical gaps are filled if they are smaller than a dynamic threshold (Fig. 2c). The threshold is increased after each iteration until all foreground objects are connected or a maximum is reached. The number of foreground objects is determined using a one-pass connected components analysis approach [9]. By increasing the threshold gradually the resulting shape tends to be more compact because only the 'necessary' gaps are filled.

If there are still multiple foreground objects after the first stage, the remaining objects are connected using diagonal smearing in a second stage (Fig. 2d). The smearing direction is thereby determined from the position of smaller objects in relation to the largest object (using the centres of mass).

If both run-length smearing stages fail to merge all connected components (due to a separating obstacle), only the largest component is selected for the subsequent outline tracing.

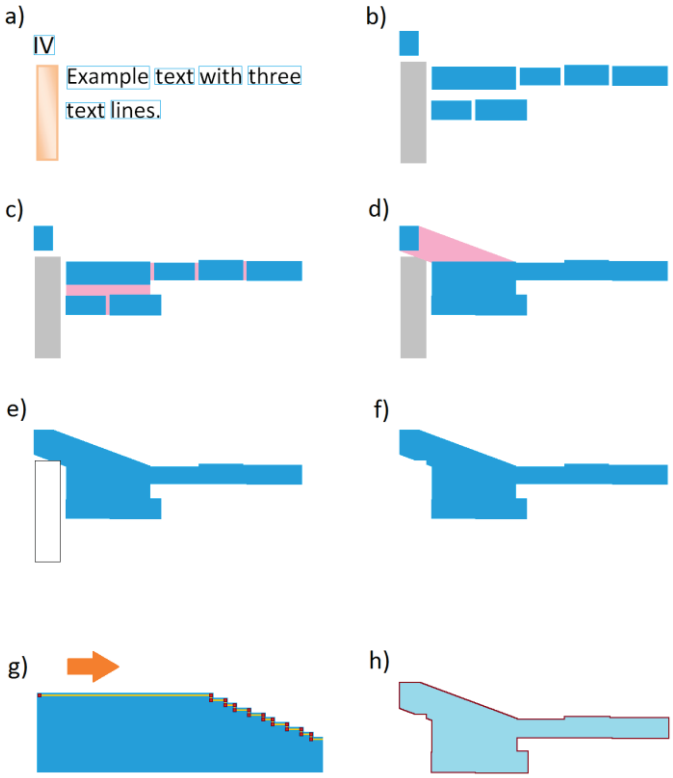

Figure 2. Process of polygonal fitting: a) Original text objects and a graphic; b) Objects transferred to bitmap as foreground; c) Horizontal and vertical filling of gaps; d) Diagonal filling of gaps; e) Excluding neighbours; f) Final foreground object; g) Outline tracing; h) Refined outline after polygonal fitting.

2.3 Subtraction of Neighbours

In this optional step adjacent layout objects are subtracted from the connected component that resulted from the run-length smearing method. Therefore, candidates for the subtraction are

determined by checking if their bounding boxes overlap the bounding box of the foreground component. The neighbour objects are then transferred to the working bitmap as white pixels, using the method described in section A (Fig. 2e).

2.4 Outline Tracing

In this final step the contour of the single foreground object is traced using a standard algorithm [10]. The outline polygon is created on-the-fly by adding one polygon point for each change of direction (Fig. 2g). The result is an isothetic polygon enclosing all child objects (Fig. 2h).

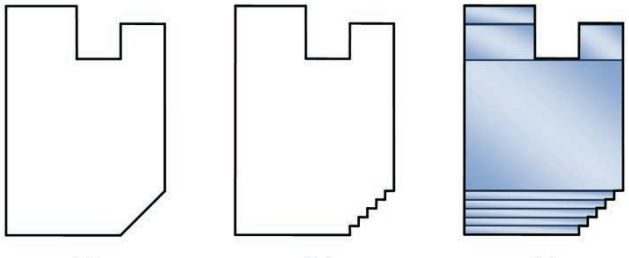

Figure 3. Interval decomposition: (a) Original polygon; (b) Isothetic polygon; (c) Interval Representation.

3. EXPERIMENTS AND RESULTS

Experiments to validate the proposed approach and evaluate its effectiveness have been carried out on a dataset of 55 contemporary documents consisting of scanned magazine and technical article pages. This standard and freely available set has also been used in the ICDAR2009 Page Segmentation Competition [11] and is part of the PRImA Contemporary Layout Analysis Dataset [12]. Fig. 3 shows three example pages.

Figure 4. Example images form the evaluation set.

3.1 Measurement of Refinement Success

All documents have been processed with Tesseract (version 3.02), a state-of-the-art open source document image analysis system. Tesseract provides an API to analyse an image and access the results. Layout and text content were exported to the PAGE XML format in two ways:

- Without refinement (original rectangular outlines as produced by Tesseract) (Fig. 5)
- With refinement through polygonal fitting (Fig. 6)

Figure 5. Example result of layout analysis by Tesseract 3.02.

Figure 6. Example of refined page layout after polygonal fitting.

In total, Tesseract produced 1354 layout regions, of which 1104 are text regions containing further child elements. The region area (original outlines without refinement) amounts to 267 megapixels (at 300 dpi image resolution).

For the evaluation of the results the absolute count of region overlaps and the overlap area have been calculated (Table I). It can be observed that the proposed refinement step reduces the occurrence of unwanted overlaps considerably (reduction of 54%). In terms of overlap area the improvement is even more pronounced (reduction of 87%). The number of whole documents

with one or more overlaps has been decreased from 41 for the original outlines to 35 for the refined outlines.

The remaining region overlap is mostly caused by segmentation errors and non-text content such as graphics and images which do not contain lower level objects for refinement.

Table 1. Results of Overlap Evaluation

	Overlapping Regions	Overlap Area (Megapixel)
Original Outlines	621 (45.8%)	19.9
Refined Outlines	286 (21.1%)	2.5

3.2 Impact on Performance Evaluation

To validate the approach using a real-world scenario, a second experiment has been carried out. Both the above result sets (original and refined) of layout data have been used to measure the performance of Tesseract's layout analysis method. In order to provide comparable results, the evaluation metrics of previous ICDAR page segmentation competitions were used (see for instance [13]). The applied performance evaluation approach is based on region correspondence analysis, identifying six conditions: Miss, partial miss, split, merge, misclassification and false detection (see also [14]). For the quantification of errors, either the whole area determined by the region polygons can be used or, as done for this experiment, only the foreground area (black pixels). It should be noted, that if the whole polygon area was used the results would have been even more pronounced.

Table II shows the results of the experiment. There is an average performance improvement of 3.4% over the 55 documents. This can be considered significant since both datasets represent the same actual quality of layout analysis. The value of 3.4% can also be interpreted as representation errors that have been filtered out, shifting the focus on real segmentation and classification errors.

Table 2. Impact on Performance Evaluation

Average success rate using original outlines	81.1%
Average success rate using refined outlines	84.5%
Average improvement	3.4%
Maximum improvement	22.9%

4. CONCLUDING REMARKS

It has been presented how existing geometric region data can be refined by fitting polygons around child objects. Validity and impact on real-world scenarios of the described method have been proven by carrying out two experiments using layout data produced by the open source OCR engine Tesseract (version 3.02).

The application of the proposed region description refinement approach in performance analysis workflows helps to eliminate problems that arise from insufficient geometric descriptions. This allows concentrating on the real issues of the layout analysis system under investigation.

The method has been made available as part of the freely available ground-truthing tool Aletheia [5] where it serves the purpose of refining results of the integrated Tesseract OCR engine which can be used for pre-production of layout data and text content (which is ultimately aiming at increasing the overall productivity).

Future work will comprise processing objects other than plain text regions (tables, charts etc.). Furthermore, layout objects that have no child objects could be refined by excluding adjacent objects, leading to a fully overlap-free layout description.

5. ACKNOWLEDGMENTS

This work has been supported in part through the EU 7th Framework Programme grants Europeana Newspapers (Ref: 297380) and SUCCEED (Ref. 600555).

6. REFERENCES

[1] Tesseract OCR Engine, http://code.google.com/p/tesseract-ocr/

[2] Breuel, T. 2007. The hOCR Microformat for OCR Workflow and Results. In *Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR '07) - Volume 02* (Curitiba, Paraná, Brazil , September 23-26, 2007, Pages 1063-1067).

[3] ALTO (Analyzed Layout and Text Object) XML Schema, http://www.loc.gov/standards/alto/techcenter/structure.php

[4] Pletschacher, S., Antonacopoulos, A. 2010. The PAGE (Page Analysis and Ground-Truth Elements) Format Framework. In *Proceedings of the 20th International Conference on Pattern Recognition (ICPR2010)* (Istanbul, Turkey, August 23-26, 2010, IEEE-CS Press, pp. 257-260).

[5] Clausner, C., Pletschacher, S., Antonacopoulos, A. 2011. Aletheia - An Advanced Document Layout and Text Ground-Truthing System for Production Environments. In *Proceedings of the 11th International Conference on Document Analysis and Recognition (ICDAR2011)* (Beijing, China, September 2011, pp. 48-52).

[6] Papadopoulos, C., Pletschacher, S., Clausner, C., Antonacopoulos, A. 2013. The IMPACT Dataset of Historical Document Images. In *Proceedings of the 2013 Workshop on Historical Document Imaging and Processing (HIP2013)* (Washington DC, USA, August 2013, pp. 123-130).

[7] Antonacopoulos, A., Ritchings, R.T. 1995. Representation and Classification of Complex-Shaped Printed Regions Using White Tiles. In *Proceedings of the 3rd International Conference on Document Analysis and Recognition (ICDAR1995)* (Montreal, Canada, August 1995, pp. 1132-1135).

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

[9] Shapiro, L., Stockman, G. 2001. *Computer Vision*. Prentice Hall, 2001, pp. 69 – 75.

[10] Vernon, D. 1991. *Machine Vision, Automated Visual Inspection and Robot Vision*. Prentice Hall, London, 1991, ISBN 0-13-543398-3, pp. 110 – 114.

[11] Antonacopoulos, A., Pletschacher, S., Bridson, D., Papadopoulos, C. 2009. ICDAR2009 Page Segmentation Competition. In *Proceedings of the 10th International Conference on Document Analysis and Recognition (ICDAR2009)* (Barcelona, Spain, July 2009, pp. 1370-1374).

[12] Antonacopoulos, A., Bridson, D., Papadopoulos C., Pletschacher, S. 2009. A Realistic Dataset for Performance Evaluation of Document Layout Analysis. In *Proceedings of the 10th International Conference on Document Analysis and Recognition (ICDAR2009)* (Barcelona, Spain, July 2009, pp. 296-300).

[13] Antonacopoulos, A., Clausner, C., Papadopoulos, C., Pletschacher, S. 2013. ICDAR2013 Competition on Historical Newspaper Layout Analysis - HNLA2013. In *Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR2013)* (Washington DC, USA, August 2013, pp. 1486-1490).

[14] Clausner, C., Pletschacher, S., Antonacopoulos, A. 2011. Scenario Driven In-Depth Performance Evaluation of Document Layout Analysis Methods. In *Proceedings of the 11th International Conference on Document Analysis and Recognition (ICDAR2011)* (Beijing, China, September 2011, pp. 1404-1408).