A Study on the Classification of Layout Components for Newspapers

Stefano Ferilli[1(✉)], Floriana Esposito[1], and Domenico Redavid[2]

[1] University of Bari, Bari, Italy
{stefano.ferilli,floriana.esposito}@uniba.it
[2] Artificial Brain S.r.l., Bari, Italy
redavid@abrain.it

Abstract. While nowadays most newspapers are born-digital (typeset directly in PDF), up to a few years ago they were only available in printed form. Digitizing the paper artifact to make it available in digital libraries yields a sequence of raster images of the pages that make up the documents. Such images consist of just matrices of pixels, and carry no explicit information about their organization into meaningful higher-level components. So, in the perspective of automatically extracting useful information from the newspapers and indexing them for future retrieval, a necessary preliminary task is to identify the layout components that are meaningful from a human interpretation viewpoint.

Unfortunately, approaches proposed in the literature for automatic layout analysis are often ineffective on newspapers, because of the much more complex layout of this kind of documents compared, e.g., to books and scientific papers. This work specifically focuses on the classification of layout blocks according to their content type. It investigates on the adaptation of an existing approach, that has been successfully applied to documents having standard layout, to the case of newspapers, working on the description features and set of classes. The modified approach was implemented and embedded in the DoMInUS system for document processing and management. Experimental results aimed at its evaluation are reported and commented.

Keywords: Layout analysis · Document representation · Document rendering

1 Introduction

In addition to book libraries, important information concerning our culture and history is preserved in newspaper and periodical libraries. As for the former, the current digital age is strongly interested in building digital versions of the latter, as well. This would significantly improve not just the availability and spread of the collected items, but first and most important might provide dramatic advantages in the retrieval of useful information, using suitably adapted versions of the search engine algorithms that have been developed in the recent decades.

M. Agosti et al. (Eds.): IRCDL 2016, CCIS 701, pp. 166–178, 2017.
DOI: 10.1007/978-3-319-56300-8_15

Nowadays, most newspapers provide for a digital edition that typically consists in the PDF version of the paper artifact that is sold in newsstands. Actually, these documents are born-digital, and the printed version is just a consequence of the original PDF file that was typeset by the editors. For several aspects, this provides a very desirable input for the existing automatic indexing procedures. Up to a few decades ago, however, typesetting was not digital, and the only available source for legacy newspapers is their printed version, that is to be digitized. Of course, digitization returns a sequence of images of the pages that make up the documents, where the basic bricks are just pixels, and no explicit information is provided about their organization into meaningful higher-level components.

Hence, the strong interest of the community in effective and efficient ways to extract such components and then for classifying them, so that they can undergo different processing depending on their type. The former step is the task of *segmentation* techniques, while the latter requires the availability of suitable models that the system may automatically apply. Manually building such models is significantly complex and sometimes impossible, due to the huge semantic gap separating the pixel level from the human perception level. So, there is a need for automatic approaches that can learn predictive classification models.

While interesting results were obtained by researchers in the past decades for documents having a more standard layout, such as books and scientific papers, solving this problem in newspapers poses new and significant challenges, due to the very complex layout and kind of layout components that they involve. First of all, they often do not use Manhattan layout. Also, they use extremely different character sizes. They are made up of several 'patches', each made up of related content blocks, but completely unrelated to each other. Particularly some kinds of newspapers, such as sports newspapers, provide additional difficulties, such as titles or articles in reverse (white characters on colored background), images with irregular contours that overlap text, and the like. Figure 1 shows a sample newspaper page, where many of these challenging peculiarities are evident.

Existing solutions available in the literature, that proved effective in handling documents having standard layout, cannot be straightforwardly applied to this kind of documents, and require suitable adaptations. The aim of this work is investigating which adaptations to the these solutions may help in handling the following aspects:

1. use of colors;
2. text blocks written on background different than the main background;
3. frequent interleaving of very different text font sizes.

Specifically, extensions of both the description features and the set of classes are studied. The performance of the adapted approach on newspapers is checked for determining its strengths and weaknesses, and for drawing conclusions about how to further improve it. For this evaluation, the proposed approach was embedded in a wider system for document processing and management, DoMInUS, that provided tools to carry out several preliminary and subsequent layout analysis tasks.


www.gazzetta.it | giovedì 14 giugno 2012 | 1,20 € anno 116 - numero 140

La Gazzetta dello Sport

Tutto il rosa della vita

CLAMOROSO SU THIAGO IL MILAN CI STA RIPENSANDO

STOPPATO!

La trattativa col Psg sembrava chiusa per 42 milioni, ma Galliani spiega: «Berlusconi vuole trattenerlo»

MERCATO Offerta Chelsea 25 per Jovetic

CONTESI Destro e Longo fra Genoa e Inter

EURO 2012 A POZNAN (ORE 18, RAI 1) SECONDA PARTITA DEGLI AZZURRI

SOLO VINCERE

Contro la Croazia l'Italia si gioca mezza Europa

Un pari complicherebbe la strada per i quarti di finale. Schillaci spinge l'altro Totò. Sammer e Beckenbauer: applausi a De Rossi difensore

DUBBIO ATTACCO BALO IN VANTAGGIO

l'Analisi

UN CONSIGLIO AL C.T.

di ARRIGO SACCHI

STAFFETTA Totò Di Natale, 34 anni, ha segnato con la Spagna. E' entrato per Mario Balotelli, 21 anni (FP)

LA SUPERSFIDA

Doppietta di Gomez La Germania mette nei guai l'Olanda

DANIMARCA K.O.

Colpaccio Portogallo Nonostante Ronaldo

Il Programma

CASO INTERCETTAZIONI LA DEPOSIZIONE DELL'EX CAPO DELLA SICUREZZA TELECOM

Tavaroli: «Spiavo anche Moggi»

E conferma l'incontro organizzato con Moratti e Facchetti: «Mi chiesero di indagare su frodi sportive del 2002-03»

IL ROMPI PALLONE DI GENE GNOCCHI

Cecchi Paone non si ferma più: «Nella nazionale degli stilisti ci sono due a cui piace la gnocca».

BASKET LA MPS SUL 3-0

Ancora Siena! Vince a Milano Ora può chiudere

IL COMPLEANNO

Agostini, i 70 di una leggenda della moto

Fig. 1. Sample newspaper page

The remainder of this document is organized as follows. In the next section, the background and relevant related work for this paper is presented. Then, Sects. 3 and 4 describe how the original approach to layout analysis and component classification was modified to deal with newspapers. Section 5 provides an evaluation of the proposed approach. Finally, Sect. 6 concludes the paper and outlines future work issues.

2 Background and Related Work

The full range of steps involved in document processing and management can be partitioned into two broad groups, yielding two macro-steps aimed at the following objectives: *Document Image Understanding* and *Document Understanding*. The following taxonomy reports the macro- or micro-steps that are specifically relevant to this work.

Document Image Understanding is concerned with determining the physical structure of the document, from both a syntactic and a semantic viewpoint (*layout structure* and *logical structure*, respectively). Among other tasks, it is

in charge of identifying the document class (e.g., book, scientific paper, bill, newspaper, etc.) and the role of its components (e.g., title, author, abstract, etc. in a scientific paper). It involves the task of

Layout Analysis. Starting from the basic components that are present in the source document, it identifies the high-level geometrical structure of the document, made up of frames that may be semantically relevant to the reader. Among others, it includes the following two sub-tasks.

Segmentation: Starting from the basic components that are present in the source document, it identifies the blocks having homogeneous and (hopefully) strictly related content.

Component Classification: Labels each block returned by segmentation with the type of content it includes.

Document Understanding. Aims at understanding and managing the information content of the document. This includes identifying its topic, extracting relevant information from it, and indexing it for future retrieval.

Of course, Document Image Understanding is very relevant to Document Understanding, in that it provides the ground on which the latter works. So, the quality of the outcome of the former is extremely important, since it may determine the quality of the outcome of the latter, or even its feasibility. In turn, a fundamental task in layout analysis is *segmentation*, that is specifically concerned with document pages represented as images. Given the source (raster or vector) document page representation, it returns a partition of its area into portions of content representing significant pieces of the layout (*blocks*) that should be consistent and significant (and possibly meaningful enough to deserve further and specialized processing).

Several segmentation methods have been proposed in the literature. Here we are interested in algorithms that work on digitized images directly at the pixel-level, ignoring any possible structure of pixel aggregates. These strategies aim at obtaining directly high-level components, without first identifying intermediate pixel aggregates that can play a role in the document (e.g., single characters or straight lines). Some methods are based on *Run Length Smoothing*. Given a sequence of black and white pixels, a *run* is defined as a sequence of adjacent pixels of the same kind (usually foreground), delimited by pixels of the opposite kind (usually background). The run length is the number of pixels in a run, and 'smoothing' a run means changing the color of its pixels so that they become of the same color as the pixels delimiting the run. A classical and efficient segmentation technique of this kind is the *Run Length Smoothing Algorithm* (RLSA) [12]. RLSA identifies runs of background pixels in the document image and fills them with foreground pixels whenever they are shorter than a given threshold. Much work in the literature is based on RLSA, exploiting it or trying to improve its performance by modifying it [2,9,10]. RLSA has some shortcomings. First, the presence of thin black lines produced on the border of the image by scanning or photocopying, may cause the horizontal smoothing to cover most of the margin of the page. Another shortcoming of this technique lays in its inability to handle documents having non-Manhattan layout (i.e., a layout in which blocks are not

always separated by perpendicular background rectangles). The assessment of suitable thresholds is a hot problem, directly affecting the overall effectiveness of the technique.

RLSO [5] is a variant of the RLSA, that works as follows:

1. horizontal smoothing of the image, carried out by rows with threshold t_h;
2. vertical smoothing of the image, carried out by columns with threshold t_v;
3. logical OR of the images obtained in steps 1 and 2.

Each connected component in the resulting image is considered a frame, and exploited as a mask to filter the original image through a logical AND operation in order to obtain the frame content. Compared to RLSA, RLSO is able to handle also non-Manhattan layouts. It involves one step less, and requires shorter thresholds (and hence fills less runs) to merge original connected components (e.g., characters) into larger ones (e.g., frames). Thus, it is more efficient than RLSA, and can be further sped up by avoiding the third step and applying vertical smoothing directly on the horizontally smoothed image obtained from the first step. This does not significantly affect, and may even improve, the quality of the result. However, the OR causes every merge of components to be irreversible, which can be a problem when logically different components are very close to each other and might be erroneously merged if the threshold is too high. Conversely, too low thresholds might result in an excessively fragmented layout. Thus, as for RLSA, the choice of proper horizontal/vertical thresholds is a very important issue for effectiveness.

The blocks singled out by segmentation may contain graphical or textual information. To properly submit them to further processing (e.g., text might be acquired using an Optical Character Recognition system, while graphical components could be input to an image processing system), their kind of content must be identified. Interesting results for this task, on A4 document images whose resolution was scaled down from 300 dpi to 75 dpi [1], were obtained by applying supervised Machine Learning techniques to distinguish text, horizontal or vertical lines, raster images and vector graphics based on several numeric features extracted from each block as suggested by [11].

Specifically, decision tree learning [8] was exploited. A *decision tree* is a branching structure in which the root is the starting point, each internal node corresponds to a test over an attribute, each branch represents a possible outcome (or set of outcomes) for the test, and the leaves bear class information. Given an observation (in this case, a content block) described according to the same attributes as the tree, starting from the root and routing the tree by repeatedly carrying out the tests in the nodes and following the branch labelled with the corresponding results, one gets the class to be assigned to the observation at hand (in this case, the type of content in the block). Machine Learning techniques for automatically building decision trees starting from a set of observations already tagged with the correct classes (called the *training set*) usually place in the nodes that are closer to the root the attributes that turn out to be more discriminative for partitioning the training set instances into different

classes, according to the outcome of class-driven information measures applied to the training set.

DoMInUS (acronym for Document Management Intelligent Universal System) [3,4] is a framework for document processing and management that embeds several Artificial Intelligence techniques to automatize the whole document lifecycle spanning from its submission to a digital library up to its retrieval and fruition by end users. It provides a variety of tools for the various steps involved in these tasks. Here, we will focus on the Pre-processing and Layout Analysis steps, that are in charge of identifying the high-level geometrical structure of the document.

3 Layout Analysis

Given a color raster image representing a newspaper page, we devised the following procedure:

1. pre-processing:
   (a) binarization, used to filter out noise from the image (iterative global thresholding);
   (b) **chromatic component separation**, used to divide the image in its relevant color components;
   (c) skew correction, used to compensate for acquisition problems.
2. *classification of layout components* **in each color layer**:
   (a) text
   (b) lines
   (c) **non-standard background**
   (d) images
3. **text blocks identification**:
   (a) **removal of non-textual components**
   (b) **extraction of text from non-standard background**
   (c) *text blocks aggregation* using RLSO

Compared to the standard procedure provided by DoMInUS, steps in bold are those specifically introduced for dealing with newspapers, and steps in italics are those that were already present but were changed for dealing with newspapers.

Step 1b allows to deal with peculiarity #1 in the Introduction. The chromatic components of interest for our purposes are typically artificially colored parts of the page, where halftones are not relevant. For this reason, the existing procedure available in DoMInUS was modified so as to ignore the saturation component of colors. The result is a sequence of filtered versions of the page, such that: the first one contains the background (which in the following will be considered white); the second one contains the graylevel pixels; and the other contain portions with other colors. The reverse of the background layer corresponds to a color-independent binarization of the document page.

Step 2c allows to deal with peculiarity #2 in the Introduction. This is obtained by taking all connected components in a layer that were classified

Fig. 2. Partial processing steps of the sample newspaper page

as Images, reversing them and running again the classifier to see whether the reversed block is classified as Text.

Step 3a modifies the overall binarized image by removing all components classified as non-text in the various color layers. Then, step 3b adds the text found on non-standard background, obtained by turning the original non-standard background into standard background, and representing the text as standard foreground. At this point, the binarized image includes only textual components on standard background (see Fig. 2 on the left). Now, step 3c performs segmentation on this input to obtain aggregate text blocks. Due to the non-Manhattan layout used by newspapers, the RLSO approach was used for this purpose. Note that the segmentation step is exploited here in a very different way than on other documents: first, it is applied as a last step, while on standard documents it is applied before classifying the type of layout blocks; second, it is applied on a filtered image containing only text, instead of the overall binarized image; third, it is applied iteratively to obtain block aggregations that are compliant to peculiarity #3 in the Introduction (the detailed procedure for this step is outside the scope of this paper). The outcome for the sample document is shown in Fig. 2 on the right.

4 Component Type Classification

The decision trees learned in the approach to layout components classification proposed in [11] are based on the following features:

1. block height (h);
2. block width (w);

3. block area ($a = w \times h$);
4. block eccentricity (w/h);
5. number of black pixels in the block (b);
6. number of black-white transitions in the block rows (t);
7. percentage of black pixels in the block (b/a);
8. average number of black pixels per black-white transition (b/t);
9. short run emphasis (*F1*);
10. long run emphasis (*F2*);
11. extra long run emphasis (*F3*).

Measures F1, F2 and F3, in particular, are to be interpreted as follows: F1 gets large values for blocks containing many short runs, which happens when the text is made up of small-sized characters (e.g., in newspaper articles); F2 gets large values for blocks containing many runs having medium length, which happens when quite large characters are used (e.g., newspaper subtitles); finally, F3 gets large values for blocks containing few runs, all of which very long, which means that the characters used for writing the text have a very large size (this is typical in the main titles of newspaper pages). F3 requires to properly set two parameters, T_1 and T_2.

These features were used in both [1,11] to learn decision trees for classifying the kind of layout blocks found in documents, with the following set of classes:

Text a group of alphanumeric characters or symbols (even just one character or symbol).
Horizontal Line
Vertical Line
Graphic an artificial image (e.g., one that might have been produced using vector graphics tools).
Image a (possibly halftone) raster image.
Mixed a combination of text and image(s), but clearly disjoint (text within images would fall in the Image class).
Undefined none of the above (e.g., a portion of an image, or a particularly eroded line).

[1] worked on scientific papers, while [11] specifically addressed newspapers. However, the sample newspapers shown in [11] seem not to show the complexities that this work aims at addressing.

A first consequence of these challenging peculiarities reported in the Introduction is that it is quite difficult to set the segmentation algorithm so that the resulting blocks correspond to semantically relevant components from a human perspective. Indeed, setting too high a threshold would identify titles as single blocks, but would also merge pieces of several different articles. Setting too low a threshold, on the other hand, would return several separate blocks for a single semantic component (e.g., a block for each letter in a title). Also, they do not work well with reversed text, and are not always able to handle non-Manhattan layouts and images that overlap text or are interleaved with text.

In such cases, a cautious approach is advisable, that prefers returning an over-segmented set of blocks (i.e., one in which a single semantic component is split into several blocks) rather than returning and under-segmented one (i.e., one in which semantically unrelated components have been merged), leaving to a subsequent post-processing step the task of merging different related blocks.

This landscape suggested to extend the set of features, adding the following:

Spread measures the spatial distribution of black pixels in a pattern, according to the following formula [7]:

$$s = \frac{n}{b} \cdot \min(w, h)^2$$

which is inverse to the number of black pixels b (because raising the density reduces the distance among pixels), and proportional to the number of black runs n (because the more the runs, the more fragmented the black zones) and to the area of square sections, obtained as follows:

$$a \cdot sq = w \cdot h \cdot \frac{\min(w, h)}{\max(w, h)} = \min(w, h)^2$$

(where $sq = \frac{\min(w,h)}{\max(w,h)}$ expresses how 'square' the block is).

Number of components useful because we expect that blocks having large area and many components are of type text, while blocks having small area and 1 component are of type character.

Number of black-white transitions in the block columns that provides a complementary perspective with respect to feature #6.

Extra long run emphasis (*F3*) with parameters $T_1 = 30$ and $T_2 = 5$

Extra long run emphasis (*F3*) with parameters $T_1 = 5$ and $T_2 = 5$

The parameters for *F3* were determined as the most appropriate for the peculiarities of newspapers, based on both the meaning of the parameter in the feature and the results of several tests with a range of different values.

We also tried to extend the set of classes of interest, by splitting the class Text into **Text**, **Character**, **Reverse Text** and **Reverse Character**. Indeed, it seemed likely that single characters are characterized by very different features than compound texts, and that the values characterizing reversed items are in some way complementary to those characterizing normal items.

5 Evaluation

To evaluate the modified approach, a baseline performance was obtained on a dataset including 30 images of newspapers' first pages, some in color and some in black and white, yielding 789 connected components of various kind, as reported in Table 1. There were no instances of graphic or diagonal line, but we think that these classes are meaningful and thus should be still taken into account in future investigations. A 10-fold cross-validation run on this dataset

using the decision tree learner J48 provided by the WEKA suite [6] returned the results reported in Table 1 (the last row reports the weighted average for performance columns, and the total for the number of components). The figures show that the worst class for accuracy is Mixed, possibly because instances of this class have very subtle (and mostly semantic) differences compared to instances of class Image, especially when they include text. Indeed, some newspapers use text superimposed to images. Based on this classification performance, the layout analysis task on an additional set of 45 newspapers reached the following results:

Precision	Recall	F-measure	Accuracy
0.885	0.909	0.897	0.784

Then, we ran additional experiments aimed at investigating the effect of adding new features and classes to the learning problem, as discussed in the previous section. Due to unavailability of the previous dataset, we ran these experiments on a different set made up of 10 newspapers. Statistics on the number of connected components in the dataset, and experimental results, are reported in Tables 2 and 3. All experiments were run using the extended set of features, but changing the set of classes. We tried the same set of classes as the baseline (see Table 3), where class Text included both text and single characters, both normal and reversed. Then, we tried to add a separate class for reversed text only (see Table 2 bottom). Finally, we added specific classes for text and characters, either normal or reversed (see Table 2 top). Looking at the figures, we can see that the new settings are all much better than the baseline, and that different settings yield mixed performances for the different classes, in that some are better on some classes and some are better on others. However, the overall results in terms of weighted averaged F-measure clearly show that the original setting, with no specific classes for characters and reverse text, is significantly better than the others. This suggests that the really relevant change in setting was the extension to the set of features. Looking at the learned models, it is interesting to note that attribute 'extra long run emphasis' ($F3$) with thresholds $T_1 = 30$ and $T_2 = 5$ is never considered by the models.

Table 1. Baseline experimental results for component type classification

Class	TP rate	FP rate	Precision	Recall	F-measure	Instances
Text	0.757	0.172	0.748	0.757	0.752	317
Horizontal line	0.916	0.013	0.906	0.916	0.911	95
Vertical line	0.857	0.004	0.923	0.857	0.889	42
Image	0.655	0.112	0.607	0.655	0.63	165
Mixed	0.368	0.04	0.42	0.368	0.393	57
Undefined	0.646	0.047	0.695	0.646	0.67	113
Overall	0.716	0.104	0.715	0.716	0.715	789

Table 2. Experimental results with additional features and classes

Class	TP rate	FP rate	Precision	Recall	F-measure	Instances
Text	0.875	0.103	0.848	0.875	0.861	376
Horizontal line	0.958	0.004	0.968	0.958	0.963	96
Vertical line	0.974	0.001	0.974	0.974	0.974	39
Image	0.845	0.056	0.801	0.845	0.822	200
Mixed	0.238	0.014	0.278	0.238	0.256	21
Undefined	0.741	0.033	0.748	0.741	0.744	112
Reverse Text	0.432	0.022	0.487	0.432	0.458	44
Character	0.680	0.011	0.773	0.680	0.723	50
Reverse character	0.143	0.002	0.333	0.143	0.200	7
Overall	0.812	0.059	0.804	0.812	0.807	945
Class	TP rate	FP rate	Precision	Recall	F-measure	Instances
Text	0.862	0.130	0.844	0.862	0.852	426
Horizontal line	0.958	0.004	0.968	0.958	0.963	96
Vertical line	0.949	0.002	0.949	0.949	0.949	39
Image	0.850	0.066	0.776	0.850	0.811	200
Mixed	0.238	0.011	0.333	0.238	0.278	21
Undefined	0.714	0.024	0.800	0.714	0.755	112
Reverse text	0.333	0.031	0.387	0.333	0.354	51
Overall	0.810	0.078	0.802	0.810	0.805	945

Table 3. Experimental results with additional features only

Class	TP rate	FP rate	Precision	Recall	F-measure	Instances
Text	0.876	0.121	0.880	0.876	0.878	477
Horizontal line	0.948	0.004	0.968	0.948	0.958	96
Vertical line	0.974	0.006	0.884	0.974	0.927	39
Image	0.830	0.051	0.814	0.830	0.822	200
Mixed	0.286	0.015	0.300	0.286	0.293	21
Undefined	0.768	0.031	0.768	0.768	0.768	112
Overall	0.849	0.076	0.846	0.849	0.848	945

6 Discussion and Conclusions

While nowadays most newspapers are born-digital (typeset directly in PDF), up to a few decades ago they were only available in printed form. Digitizing the paper artifact to make it available in digital libraries yields a sequence of raster images of the pages that make up the documents. Such images consist of just matrices of pixels, and carry no explicit information about their organization

into meaningful higher-level components. So, in the perspective of automatically extracting useful information from the newspapers and indexing them for future retrieval, a necessary preliminary task is to identify the layout components that are meaningful from a human interpretation viewpoint.

Unfortunately, even approaches specifically proposed in the literature for automatic layout analysis of newspapers, are often unable to handle particular features such as use of colors, text written on background different than the main background, and frequent interleaving of very different text font sizes. This work specifically focused on the classification of layout blocks according to their content type. It investigated on the adaptation of an existing approach, that was successfully applied to documents having standard layout, to the case of newspapers, working on the description features and set of classes. The modified approach was implemented and embedded in the DoMInUS system for document processing and management. Experimental results aimed at its evaluation were reported and commented.

Future work includes experimenting on a larger dataset, and testing the final effect that the improved block type classification approach has on the final layout analysis performance.

Acknowledgments. The authors would like to thank Vincenzo Raimondi for his help in implementing the prototype. This work was partially funded by the Italian PON 2007-2013 project PON02_00563_3489339 'Puglia@Service'.

References

1. Altamura, O., Esposito, F., Malerba, D.: Transforming paper documents into XML format with WISDOM++. Int. J. Doc. Anal. Recogn. **4**, 2–17 (2001)
2. Cao, H., Prasad, R., Natarajan, P., MacRostie, E.: Robust page segmentation based on smearing and error correction unifying top-down and bottom-up approaches. In: Proceedings of the 9th International Conference on Document Analysis and Recognition (ICDAR), vol. 1, pp. 392–396. IEEE Computer Society (2007)
3. Esposito, F., Ferilli, S., Basile, T.M.A., Di Mauro, N.: Machine learning for digital document processing: from layout analysis to metadata extraction. In: Marinai, S., Fujisawa, H. (eds.) Machine Learning in Document Analysis and Recognition. Studies in Computational Intelligence, vol. 90, pp. 105–138. Springer, Heidelberg (2008)
4. Ferilli, S.: Automatic Digital Document Processing and Management - Problems, Algorithms and Techniques. Springer, London (2011)
5. Ferilli, S., Biba, M., Esposito, F., Basile, T.M.A.: A distance-based technique for non-manhattan layout analysis. In: Proceedings of the 10th International Conference on Document Analysis Recognition (ICDAR), pp. 231–235 (2009)
6. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: an update. SIGKDD Explor. Newsl. **11**(1), 10–18 (2009)
7. Mitchell, P.E., Yan, H.: Newspaper layout analysis incorporating connected component separation. Image Vis. Comput. **22**(4), 307–317 (2004)
8. Mitchell, T.M.: Machine Learning. McGraw-Hill, New York (1997)

9. Shih, F.Y., Chen, S.-S.: Adaptive document block segmentation and classification. IEEE Trans. Syst. Man Cybern. - Part B **26**(5), 797–802 (1996)
10. Sun, H.-M.: Page segmentation for Manhattan and non-manhattan layout documents via selective CRLA. In: Proceedings of the 8th International Conference on Document Analysis and Recognition (ICDAR), pp. 116–120. IEEE Computer Society (2005)
11. Wang, D., Srihari, S.N.: Classification of newspaper image blocks using texture analysis. Comput. Vis. Graph. Image Process. **47**, 327–352 (1989)
12. Wong, K.Y., Casey, R., Wahl, F.M.: Document analysis system. IBM J. Res. Dev. **26**, 647–656 (1982)