Logical Structure Detection for Heterogeneous Document Classes

Leon Todoran[a] Marco Aiello[a,b] Christof Monz[b]
Marcel Worring[a]

[a]Intelligent Sensory Information Systems, Univ. of Amsterdam, The Netherlands
[b]Institute for Logic, Language and Computation, Univ. of Amsterdam, The Netherlands

ABSTRACT

We present a fully implemented system based on generic document knowledge for detecting the logical structure of documents for which only general layout information is assumed. In particular, we focus on detecting the reading order. Our system integrates components based on computer vision, artificial intelligence, and natural language processing techniques. The prominent feature of our framework is its ability to handle documents from heterogeneous collections. The system has been evaluated on a standard collection of documents to measure the quality of the reading order detection. Experimental results for each component and the system as a whole are presented and discussed in detail. The performance of the system is promising, especially when considering the diversity of the document collection.

Keywords: Document Analysis, Logical Structure Detection, Reading Order Detection, Natural Language Processing, Spatial Reasoning.

1. INTRODUCTION

The goal of document analysis is to automatically process scanned documents and convert them into a digital format, which can for example be further processed for reproduction, digital libraries, information retrieval, and text-to-speech purposes. This process mainly consists of two steps: layout analysis and document understanding. During the layout analysis the constituents of the document image of a page are identified and classified as text or image objects and further font information, textual content, geometric features, and spatial relations are extracted. This information is captured in the layout structure. Document understanding takes the layout structure as input, further classifies its items into logical items (e.g., title, paragraph, etc.) and detects relations between them (e.g., the reading order). This information is captured in the logical structure.

Most of the document analysis systems developed so far make use of specific a priori document knowledge and are therefore domain-dependent. The systems described in the literature and implemented in commercial software (e.g., FINEREADER[1]) can successfully handle simple black-and-white documents with a layout structure which is known in advance. The treatment of colored documents, complex layouts, and the analysis of a heterogeneous class of documents is definitely a challenging task and open question.

In this paper we try to answer this question. Without using any document class-specific information, as the data set is composed by a large collection of document classes, we detect the logical structure. In particular, we focus on the reading order detection which is a fundamental part of the logical structure. The *reading order* is the sequence of textual document objects in which the user is going to (or is supposed to) read the document at hand. To detect the reading order in a scanned document of which the layout structure is available, we introduce two components that take full advantage of the layout information. The first component is based on formal methods. A spatial reasoner, using a set of document rules decides which reading orders are formally correct from the spatial point of view. The second component is based on natural language processing (NLP) and considers the text present in the textual document objects identifying the syntactically most plausible reading orders.

In the last decade, several systems for detecting logical structures from scanned text have been developed. One example of a domain specific system has been developed by Tsujimoto and Asada to process multi-column black-and-white scientific

Further author information:
E-mail: {todoran,aiellom,christof,worring}@science.uva.nl
URL: http://www.science.uva.nl/∼{todoran,aiellom,christof,worring}

papers.[2] Both in layout and logical structure detection, the domain knowledge is used to derive the classification rules. The main shortcoming of this system is that it cannot be adapted to other classes of documents.

Ishitani proposes to exchange information between layout and logical analysis, which are applied iteratively.[3] This improves both layout and logical detection. But as in the previous case, this system can be used only for documents falling in one specific class.

A step toward generality is made by Cesarini et al.[4] Two distinct categories of knowledge are identified: specific to a class of documents and generic or independent from the class of documents. A pitfall is the use of XY-trees[5] which reduces the generality of documents to which their system is applicable. For instance, color documents where overlapping is present cannot be processed.

There have been attempts to automatically generate rules to detect the logical structure for a general class of documents. Sainz and Dimitriadis use a fuzzy-neural system to learn from a given training set what the rules are for converting the layout into the logical structure.[6] Li and Ng propose a domain-independent document understanding system with learning ability.[7] They use a directed weighted graph to represent the layout structure, allowing for a more general class of documents to be considered than by using tree representations. Both Sainz and Dimitriadi and Li and Ng use only geometrical information of the layout, but in the extraction of the logical structure the content has a key role. For example, when no a priori document knowledge is given, detecting the reading order of the textual elements of a document can only be achieved by considering the textual content of the elements themselves, which in turn implies the use of natural language processing.

We have previously proposed a framework to extract the logical structure given the layout structure making some use of the content of the document. Some very preliminary experimental results were presented.[8] In this paper, we extend the framework in two ways. On the one hand, we provide some vertical integration, that is, we do not assume anymore the logical object classification is available a priori. On the other hand, we use more effective spatial reasoning and natural language processing techniques.

The remainder of this paper is organized as follows: In the next section we describe the adopted representation for the layout and logical structure of a document. In Section 3, we present the architecture of our system, and describe each of its components. Experimental results and evaluation are discussed in Section 4.

2. DOCUMENT REPRESENTATION

The document image analysis can be seen as the inverse process of document authoring. Therefore these two processes should use similar document models. Requiring the document class to be generic, the document model should be able to represent any complex document structure. Rather than tree-based representations, we consider a graph-based one to encode the relations among document objects. Our model is a flexible representation suitable for a broad class of documents. A document $\mathcal{D}$ is a set of layout $\mathcal{G}$ and logical $\mathcal{L}$ structures: $\mathcal{D} = \langle \mathcal{G}, \mathcal{L} \rangle$.

As for the *layout (or geometric) structure* $\mathcal{G}$ of a document $\mathcal{D}$. Let O_g be a set of layout document objects and $\mathcal{R}_g$ a set of geometric relations between the document objects, such that $\mathcal{G} = \langle O_g, \mathcal{R}_g \rangle$. In the current implementation of the system, the layout structure has three categories of layout objects: text, image and graphics. Because we have to deal with generic documents, we consider a flexible list of features rather than a fixed set. Besides the bounding boxes coordinates, the document object's features considered are: `font size ratio` defined as ratio between the font size of the current document object, and the most common font size of the entire page; `aspect ratio` defined as width divided by height of the document object; `area ratio` defined as the ratio between the object's area and the page area; `content size` defined as number of characters; `font style` with the possible values "Plain", "Bold" and "Italic". The same holds for relations: rather than keeping one single relation we have a list of relations. The spatial relations considered among document objects are `adjacency` and the product of the Allen's relations on the two document axes: `precedes`, `meets`, `overlaps`, `starts`, `during`, `finishes`, `equals` (and their inverses).[9] The adjacency relation is determined based on Voronoi diagrams.[10]

As for the *logical structure*, it is defined analogously to the layout structure as a set of logical document objects and a set of logic relations between them: $\mathcal{L} = \langle O_l, \mathcal{R}_l \rangle$.

3. FROM LAYOUT TO LOGICAL STRUCTURE

Assuming that the bounding boxes of textual document objects are given, our system extracts the logical structure from it, as shown in Figure 1. The first module (depicted at the bottom of the figure) assigns to the layout document objects O_g logical labels, thus creating the set of logical objects O_l. This process is described in detail in Section 3.1. For each type of O_l

the reading order is extracted independently, as presented in Section 3.2.1. These reading orders of each type of O_l are then combined together into a set of admissible reading orders. On the right of the figure, a zoom-in of the spatial reasoning module is shown. This set is further reduced using the natural language processing module described in Section 3.2.2.

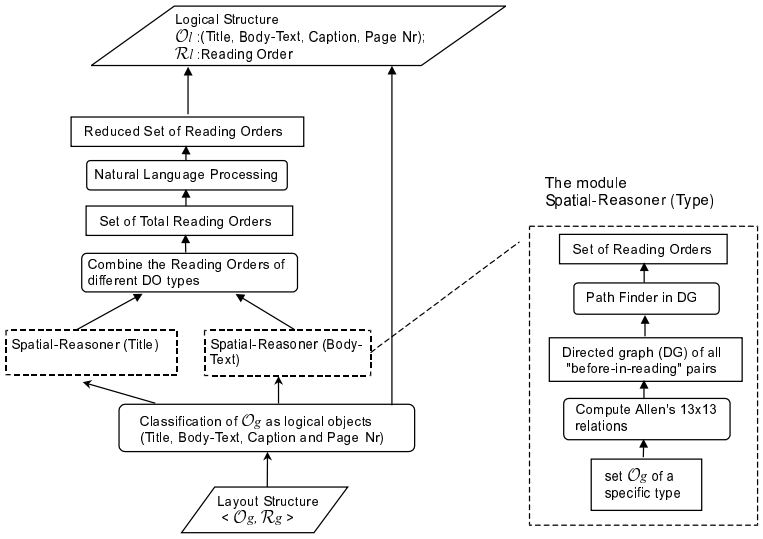
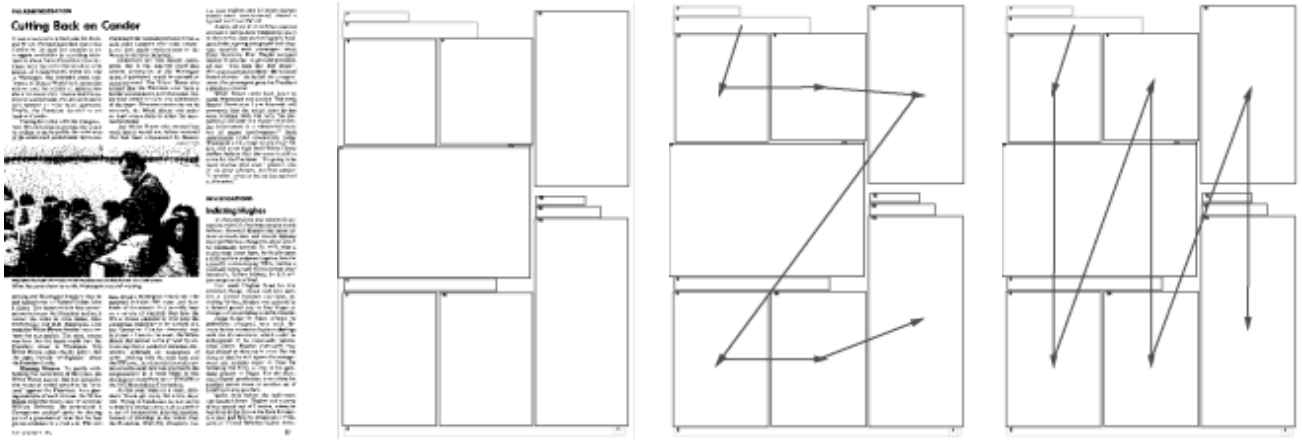


Figure 1. The system architecture.

We consider the layout detected at the granularity of text blocks. Using geometric features and content we classify the layout document objects and determine the reading order. In this paper, we focus on text document objects only.

The outcome of our system for a given image can be seen in Figure 2. The input image and the result of document object classification are depicted in (a) and (b), respectively. The two admissible reading orders detected by the spatial reasoning component are shown by (c) and (d). From these, the NLP component removes the row-wise reading order (c), giving as output the correct one, which is the column-wise (d).



(a) (b) (c) (d)

Figure 2. The processing of a document image.

3.1. Document Object Classification

When considering a broad class of documents there is a limited number of logical document object types common to all of them.[11] For text document objects, we are considering the following logical labels: `Title`, `Body Text`, `Caption` and `Page`

Number. After studying the ground truth, we have selected the five features defined in Section 2 as discriminative for the logical types presented above. Furthermore, we expressed the adjacency relation as the feature distance to the closest figure. This is the distance (obtained from the adjacency relation) between the current text document object and the closest figure document object among the neighbors. If there is no figure in the neighbors list the distance gets the maximum value.

Each feature from the set just presented is discriminative between two or more types of O_g: font size ratio is discriminative between Title and Body Text; aspect ratio differentiates Title and Caption from Body Text; area ratio is mostly discriminative for Page Number against Body Text; content size separates the Page Number, Title and Caption from Body Text; font style separates the Title and Caption from Body Text[12] and distance to the closest figure is discriminative between Title and Caption. None of these features alone is discriminative enough for a reliable selection. For instance, in Figure 3 the font size ratio feature is shown for Title, Body Text, Caption and Page Number document objects. The document objects are represented by their index on the X axis, and the actual value of the feature on the Y axis. One can see that for Body Text this is most of the time equal to 1 while for Title is larger than 1, as expected. The exceptions appear due to errors in font information detection, or due to peculiarities of specific documents. For the Caption feature, the values of the font size ratio feature are also most of the time equal or closer to 1. The exceptions here are due to some document styles where captions are written with larger fonts, or pages containing pictures and captions only. The exceptions from the rule produce overlap in values of each feature, and make selection difficult using a unique feature. Therefore, we use the six dimensional space for classification.

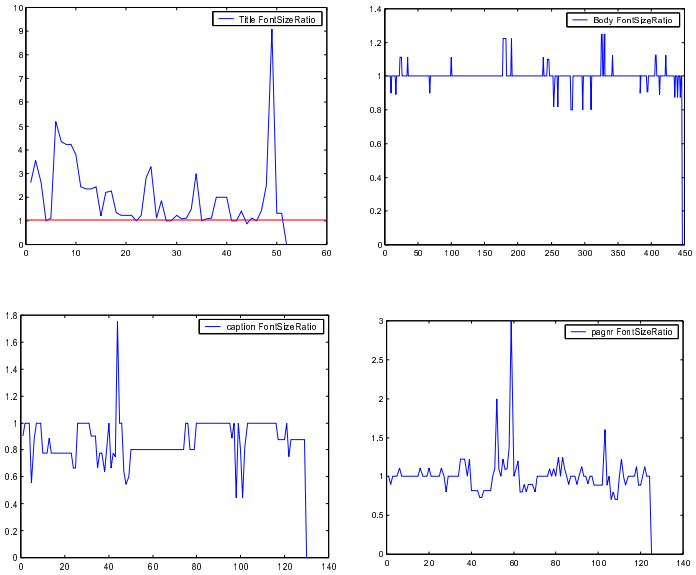


Figure 3. The font size ratio feature for Title, Body Text, Caption and Page Number document objects.

3.2. Reading Order Detection

In this section, we present the two components to detect the reading order in the scanned document assuming the layout structure is available. We focus on the extraction of a unique reading order.

3.2.1. Spatial reasoning on document objects

Usually, bounding boxes of the document objects are considered in the logical analysis step, and it appears natural to resort to a formal method based on rectangles. In 1983, Allen and van Benthem independently proposed a calculus for time intervals identifying 13 basic relations.[9,14] The calculus has proved to be quite successful ever since and is one of the most cited work in AI. Recently an extension to two dimensions has been proposed by Balbiani et al.: a calculus for rectangles.[15] The idea is to consider a rectangle model $\langle R, m_x, m_y \rangle$, that is a set of rectangles R and the set of Allen's 13×13 relations over the rectangles on the two document axes.

In order to compensate for imprecision in layout information, we slightly adapt the original rectangle model by considering 'thick boundaries' for rectangles. The thickness, the size of the sides, is defined according to the scale of the scanned document. This prompted for a rewriting of Allen's relations to take in account for such thick boundaries while preserving all fundamental properties of the calculus, most notably, the relations over intervals should be jointly exhaustive and pairwise disjoint. For example, `meets(B1,B2)` is *not* true if the end point of B1 coincides with the starting point of B2, as in Allen's original work. Rather, `meets(B1, B2)` is true if the difference between the the end point of B1 and the starting point of B2 is smaller than the identified 'thickness' of the boundaries of the rectangles.

The use of the rectangle model with thick boundaries for spatial reasoning consists of the following: first, we use geometric information (the layout structure) to build the rectangle model associated with the document. Second, we encode a set of document rules as constraints over document objects, i.e., rectangles in the rectangle model. Third, by using the efficient constraint satisfaction solver Eclipse,[16] we check the consistency of the document rules instantiated by the textual document objects.

To illustrate the kind of document rules that we are using, consider the following general and basic fact of the occidental culture: A textual component is read before another one if it is on top or to the left of the second one. This can be written as six constraints in the rectangle model (expressed in Eclipse code):

```
before_in_reading(B1, B2):-          before_in_reading(B1, B2):-
       precedes_X(B1, B2).                  precedes_Y(B1, B2).

before_in_reading(B1, B2):-          before_in_reading(B1, B2):-
       meets_X(B1, B2).                     meets_Y(B1, B2).

before_in_reading(B1, B2):-          before_in_reading(B1, B2):-
       overlaps_X(B1, B2).                  overlaps_Y(B1, B2).
```

where B1 and B2 are any two textual document objects. Given the formal properties of the rectangle model,[15] we know that the order in which the rules are considered is irrelevant and that checking the consistency of instantiated rules over a rectangle model is not only a decidable problem, but also its complexity has a polynomial bound.

Rules can further constraint the set of admissible relations. One may want to force the reading order to be column-wise (like for most newspapers), or prefer the horizontal direction over the vertical one (such as in Tsujimoto and Asada's work[2]).

In Section 2, we have introduced the layout structure as $\mathcal{G} = \langle O_g, \mathcal{R}_g \rangle$. It is immediate to see that the rectangle model $\langle R, m_x, m_y \rangle$ is a subset of it, in the sense that $R \subseteq O_g$ and that $m_x \subset \mathcal{R}_g$, $m_y \subset \mathcal{R}_g$. Then the spatial reasoner builds the elements of the ternary relation

$$\mathcal{R}_{\text{ord}} : O_l \times O_l \times \mathbb{N}$$

interpreted as: a document object belonging to the logical structure is immediately before in reading of a second document object of the logical structure in the n-th spatially admissible reading order. $\mathcal{R}_{\text{ord}}$ is a logical structure relation, i.e., $\mathcal{R}_{\text{ord}} \subset \mathcal{R}_l$. For every two textual objects o_i, o_j the consistency of reading order relation over them $\mathcal{R}_{\text{ord}}(o_i, o_j, \_)$ is checked against the rectangle model. As a point of notation, by an underscore $\_$, we intend a value which is not relevant, i.e., an unconstrained variable. Then, the n-th reading order is a sequence of objects such that:

1. $\mathcal{R}_{\text{ord}}(o_k, o_{k+1}, n)$ is consistent,
2. each document object o_k appears at most once as the first argument in a $\mathcal{R}_{\text{ord}}(\_, \_, n)$ triple and at most once as the second argument,
3. every $o_k \in O_l$ appears at least once in a $\mathcal{R}_{\text{ord}}(\_, \_, n)$ triple.

Referring to Figure 1, we note that the set of pairs o_k, o_{k+1} in the relation $\mathcal{R}_{\text{ord}}(o_k, o_{k+1}, n)$, independently from the n, forms a directed graph (DG) over document objects. The edges of the graph are interpreted as the node o_k is 'before in reading' of the node o_{k+1}. The conditions (2) and (3) above are the definition of a total strict ordering over all nodes of the graph (DG). The final output of the spatial reasoning module is a set of spatially admissible reading orders. This set is then passed on to the next component to check for linguistic admissibility.

3.2.2. Natural language processing

In a number of cases it is possible to find a single reading order just by applying spatial reasoning rules, but the more complex the layout of a document, the harder it is to identify a unique correct reading order.

When applying the spatial reasoning rules, as described above, only the relative positions of one text block with respect to another are considered. The content of a text block, i.e., the textual information itself, has not been considered as a means helping to decide which of the potential reading orders are correct. Of course, fully understanding the content of a document is beyond the current state of the art of natural language processing. On the other hand, our current results indicate that shallow NLP tools like taggers can contribute to the resolution of reading order ambiguities. A tagger assigns a part-of-speech tag, such as `DT` (determiner: *the, a*), `VBD` (past tense verb: *took, said*), `SENT` (sentence boundary: . ! ?), etc., to each word or punctuation sign. In our current implementation, we used TREETAGGER, a statistical tagger based on decision trees.[17] Assigning the correct part-of-speech tag to a word is often non-trivial. For example, the word *programming* is a gerund in contexts like *she is programming* and a noun in contexts like *they consider programming an important skill*. Most part-of-speech taggers rely on statistical information to keep tagging computationally feasible. It is inevitable that the statistical model fails in some situations and incorrect part-of-speech tags are assigned to the words. This misclassification is very likely to cause a decrease of the accuracy of our system, but the precise impact has not been evaluated yet.

In general, if one has two reading orders $o^1 \dots o^n$ and $o^{\pi(1)} \dots o^{\pi(n)}$, where the second sequence is a permutation of the first one, and both reading orders are admissible by the spatial reasoning component, to decide which of the reading orders is to be preferred, we focus on the transitions between each object of the respective reading orders. Let o and o' be two consecutive objects within a reading order, we want to compute the transition score $ts(o,o')$, as defined below.

First, both objects are tagged and the last two (tagged) words ending and the first (tagged) word beginning objects o and o' are identified. We refer to the last two tags of an object o by t_o^{-2} and t_o^{-1} and to the first by t_o^1. To refer to the words themselves, we use w_o^{-2}, w_o^{-1}, and w_o^1.

The restriction to sequences of length 3 is mainly due to the sparse data problem, where it can be expected that most training corpora are too small to assign reliable frequencies to seldomly occurring sequences. The transition score of a pair o,o' is computed as follows:

$$ts(o\ o') = \begin{cases} 1 & \text{if } t_o^{-1} = \texttt{SENT} \text{ and } w_{o'}^1 \text{ starts with a upper-case letter,} \\ 10^{-10} & \text{if } t_o^{-1} = \texttt{SENT} \text{ and } w_{o'}^1 \text{ starts with a lower-case letter,} \\ 1 & \text{if } w_o^{-1} \text{ ends with a hyphen and } w_o^{-1}w_{o'}^1 \text{ is in the lexicon,} \\ 10^{-10} & \text{if } w_o^{-1} \text{ ends with a hyphen and } w_o^{-1}w_{o'}^1 \text{ is not in the lexicon,} \\ P(t_{o'}^1 | t_o^{-2} t_o^{-1}) & \text{otherwise.} \end{cases}$$

The first four case distinctions implement simple heuristics dealing with sentence boundaries and hyphens. Transitions which are very unlikely are given a low, non-zero score. The last case computes the maximum likelihood of a tagged sequence in the remaining situations.

$$P(t_{o'}^1 | t_o^{-2} t_o^{-1}) = \frac{P(t_o^{-2} t_o^{-1} t_{o'}^1)}{P(t_o^{-2} t_o^{-1})}$$

where $P(t_o^{-2} t_o^{-1} t_{o'}^1)$ is computed by dividing the number of occurrences of the tag sequence $t_o^{-2} t_o^{-1} t_{o'}^1$ in a pre-tagged training corpus by the number of all trigrams occurrences. $P(t_o^{-2} t_o^{-1})$ is computed analogously. The reader is referred to Manning and Schütze's textbook[18] for a comprehensive introduction to statistical natural language modeling.

To decide the whole reading order of a document page with the textual objects $o^1 \dots o^n$, the transition score is computed as the product of the transition scores of all consecutive text objects:

$$ts(o^1 \dots o^n) = \prod_{i=1}^{n-1} ts(o^i\ o^{i+1})$$

By computing scores it is also possible to rank the different reading orders. During the experiments we considered only the reading order with the highest score.

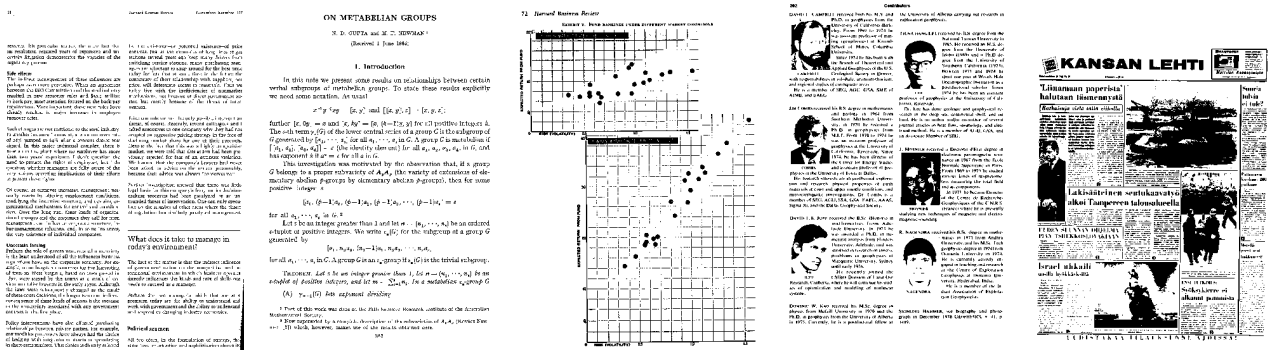
4. EXPERIMENTS AND EVALUATION

We have applied our method to 206 English document images from the Document Database II (MTDB).[19] Among these were newspapers, magazine articles, scientific articles from several journals (physics, math, chemistry). In the next three sections, we present the experimental results for each module and then give a general evaluation of the system.

At the current state, we are not able to identify independence between textual objects, and the generated ordering of the objects is always total. To compare the generated total reading order to a set of independent reading orders, we consider any permutation of the independent reading orders and define the total reading order as *correct* if it is identical to one of the permutations.

4.1. Document Object Classification

The documents of the MTDB data set are gray-level and color images scanned at the image resolution of 300dpi. In the dataset the ground truth for layout and logical structure is available. In the pages considered, 445 document objects are `Body Text`, 132 `Caption`, 125 `Page Number` and 52 `Title`.



(a) (b) (c) (d) (e)

Figure 4. Examples of heterogeneous documents from the data set.

In the evaluation of feature based classification, we have considered the ground truth defined in MTDB, using the following error rate formulas:

$$f_1 = \frac{FN}{GT} * 100\% \qquad f_2 = \frac{FP}{GT^c} * 100\%$$

where GT represents the ground truth, FN (false negative) the number of document objects of a class not detected and FP (false positive) the number of wrongly classified as being of the current type. For f_2 the complement of the ground truth GT^c is defined as the number of all document objects minus those of the current type. The classifiers considered here are linear, quadratic and respectively the Parzen classifier.[13]

From the total number of objects, half of each class were selected as training set and the rest as test set. The comparative results are presented in Table 1. The average error made in classification of all blocks is computed as the average of non-diagonal items from the confusion matrix. The quadratic classifier (QDC) gives the best classification.

Table 1. Comparative results of classification using linear, quadratic and Parzen Classifiers.

Type	*GT*	*LDC*		*QDC*		*Parzen*	
		$f_1(\%)$	$f_2(\%)$	$f_1(\%)$	$f_2(\%)$	$f_1(\%)$	$f_2(\%)$
Body	445	1.57	19.74	3.82	3.88	11.23	6.79
Caption	132	28.03	1.28	12.87	2.89	11.36	6.43
PageNumber	125	19.20	2.54	0.80	0.95	20.00	8.58
Title	52	34.61	0.28	11.53	0.71	48.07	0.14
Total	754	23.30		11.11		31.16	

In Table 2, one can see the confusion matrix for QDC classifier. The prominent mistakes made are the confusions of `Caption` with `Titles` and `Body Text` with both `Titles` and `Captions`.

Table 2. The confusion matrix of the quadratic classifier presented in Table 1.

%	*Body*	*Caption*	*PageNumber*	*Title*
Body	92.20	6.42	0	1.37
Caption	15.38	73.84	9.23	1.53
PageNumber	0	0	98.36	1.63
Title	8.00	16.00	0	76.00

Using the leave-one-out method[13] for selection of training and test set, which is more appropriate for this small data set, the average error made in classification, using the quadratic classifier, was 8.65%. The average confusion matrix is presented in Table 3.

Table 3. The confusion matrix of the quadratic classifier, using leave-one-out method.

%	*Body*	*Caption*	*PageNumber*	*Title*
Body	92.13	7.41	0	0.44
Caption	4.04	92.13	1.34	2.47
PageNumber	0	0	97.52	2.47
Title	5.39	11.01	0	83.59

Most of the errors made are due to overlap of the feature values because of variations in document style. Difficult examples are given in Figure 4.a and Figure 4.c. The bottom-right most object of the image (a) is a body-text. But its size and aspect ratio makes it similar to a title. In (c), one can see a page with large images and few text. There the font size ratio feature cannot be computed correctly.

4.2. Reading Order Detection

We have applied the spatial reasoner and the natural language components to detect the reading order of the subset of English documents of the MTDB dataset. We provide the experimental results for each component separately.

We use the precision measure,[20] a standard effectiveness measure in Information Retrieval, to evaluate the spatial reasoning component, the NLP component, and the system as a whole. For the spatial reasoning component, the set of spatially admissible reading orders (SARO) is compared to the ground truth, which defines the correct reading order. Analogously for the NLP component where the the most-likely reading order is compared to the ground truth. For a number of documents, the ground truth defines independent reading orders for a non-intersecting subsets of the textual objects within the same document; e.g., a page containing two different articles or independent text blocks such as information about the authors of an article, as exemplified by Figure 4.d. Since both components return total reading orders, we consider a reading correct if it is identical to at least one permutation of the independent reading orders as defined in the ground truth. We refer to the set of permutations of the ground truth as the set of correct reading orders (CRO). Then, the precision of the spatial reasoning component is defined as follows:

$$precision = \frac{|\,\mathrm{SARO} \cap \mathrm{CRO}\,|}{|\,\mathrm{SARO}\,|}$$

The precision value lies between 0 and 1 inclusive, where 0 indicates that the correct reading is not among the spatially admissible reading orders, 1 indicates that there is exactly one spatially admissible reading order and it is correct, and any other value indicates the degree of uncertainty of the spatial reasoning component.

Evaluating the NLP component is a bit simpler, because not a set of admissible reading orders is identified but one single reading order. Let *mts*(SARO) be the reading order with the maximal transition score among the spatially admissible reading orders. For evaluating the NLP component, precision is defined as:

$$precision = \mid \{mts(\mathrm{SARO})\} \cap \mathrm{CRO} \mid$$

Here, precision will always be either 1, if the statistically most likely reading order is correct, or 0 if it is incorrect.

4.2.1. Spatial reasoning

We have previously presented the result of applying the general rules of Section 3.2.1 considering rectangles with no boundaries.[8] In the current system, we consider thick boundaries and use two specific sets of spatial rules. The two sets of rules further constraint the spatial admissible reading orders by forcing 'coherence' throughout the document. The idea is that a document is organized either vertically or horizontally. In other words, it is not acceptable that within the same document, there is a subset of objects to be read column-wise and others row-wise.

Table 4. Experimental results of applying the spatial reasoning component to detect the reading order.

	no. of documents	*no. of relevant document objects*	*no. of possible reading orders*	*avg. size of SARO*	*avg. precision*	*difficulty*
	13	1	1	1	1	trivial
	22	2	2	1	1	easy
	81	3–8	6–40320	1.43	0.7778	medium
	6	> 8	> 40320	1.5	0.7222	hard
average	122	3.928	695,705,662	1.26	0.8452	
median	122	4	24	1	1	

In Table 4, the results of applying the spatial reasoner to 122 documents in English of the MTDB data set are summarized. Depending on the number of relevant document objects different subsets of the documents are considered. The higher the number of document objects, the harder the task of analyzing the document at hand. The rows are sorted in ascending order of difficulty. In the first column, the number of documents is shown. In the second column, the number of relevant document objects—titles and text bodies—is presented. In the third column, the number of possible reading orders given the number of relevant document objects is displayed. This number is computed as the factorial of the number of document objects. In the forth column, the number of identified spatially admissible reading orders (the output of the spatial reasoner) is shown. In the fifth column, the precision value provides an evaluation of the performance of the spatial reasoner. Except for the last row, the value of precision is given as the average value of each document of the class considered on that row.

In the last column of Table 4, there is an intuitive statement regarding the difficulty of processing documents of the class. Finally, the last two rows summarize the results for the whole collection by giving the average of all values and the median. The median is more significant in the current context as the distribution of the possible reading orders is extremely skewed.

4.2.2. Natural language processing experimental results

Among the 109 non-trivial documents of the English documents of the MTDB, are 35 documents which could only be partially disambiguated by the spatial reasoning component and therefore retain more than one spatially admissible reading order. The NLP component has been applied to those 35 documents only.

Table 5. Experimental results for the NLP component.

no. of documents	*SARO*	*avg. precision*
34	2	0.9091
1	4	0
35		0.8824

There is only one document with more than two spatially admissible reading orders. Although it is mentioned in Table 5, it can be discarded when evaluating the NLP component, because none of the four reading orders is correct and applying the

NLP component to these reading orders to choose the most likely one cannot change this fact. Focusing on the remaining 34 documents shows that the average precision is 0.9091, meaning that for approximately 91% of the pages the statistically most likely reading is indeed the correct one.

In a situation where the most-likely reading order is incorrect, this is mainly due to OCR errors. For instance, one of the text blocks starts with the word *We* which is recognized by the OCR software as *Ve*. This again causes the part-of-speech tagger to assign wrong tags to those words. Since the computation of the transition score is based on the part-of-speech tags, this results in a score deviating from the score using the correct part-of-speech tags.

4.2.3. General evaluation of the system

To give an overall evaluation of the system, we first explain how to combine the precision of both components. Evaluation focuses on the identification of a single reading order:

$$precision = \begin{cases} 1 & \text{if } |\,\text{SARO}| = 1 \text{ and SARO} \subseteq \text{CRO}, \\ 1 & \text{if } |\,\text{SARO}| \geq 2 \text{ and } mts(\text{SARO}) \in \text{CRO}, \\ 0 & \text{otherwise.} \end{cases}$$

The first case accounts for situations where the spatial reasoning component was able to uniquely identify the correct reading order. The second case accounts for situations where more than one spatially admissible reading order was identified and the statistically most-likely reading is correct. The third case accounts for situations where one of the two components excludes the correct reading order. Table 6 presents the average precision for identifying a unique reading order for 122 English documents from MTDB.

Table 6. Experimental results for the whole system.

no. of documents	*SARO*	*avg. precision*
87	1	0.9884
35	≥ 2	0.8824
122		0.9580

The presented results are encouraging as they are a clear improvement of our own previous experimentations.[8] This is mainly due to the use of the more specific constraints which impose 'coherence' and, in part, to the consideration of thick boundaries for the rectangles. The NLP component has benefited from using CELEX,[21] a rather comprehensive lexicon, which we used to recognize hyphenated words, and the implementation of a better language model allowing to compute more precise transition scores.

5. CONCLUSIONS

We have proposed a method based on generic document knowledge for detecting the logical structure of documents for which only general layout information is assumed. Given the layout of a document, simply by using geometric information, font features and textual content, we are able to identify the logical structure with reasonable accuracy.

The main contribution of the presented approach lies in its generality. Virtually nothing is assumed of the document handled by the system. The current implementation is able to work only for English documents, but the same techniques presented here could be applied to other languages. The restriction to English is imposed by the NLP component where at the moment only a part-of-speech tagger for English is considered. Part-of-speech taggers are available for a number of languages allowing our system to be applicable to other languages.

Furthermore, the spatial rules employed by the spatial reasoning component are general. They apply to documents prepared according to the common knowledge of the occidental culture: one proceeds from top-down and/or left-right. Again we remark the generality of the framework. If one wants to move to documents from different cultures, the left-right rule may be violated, but this prompts only for a rewriting of a few spatial reasoning rules, not for redesigning the whole document analysis framework.

Having set the achievement of generality as a goal, some key problems need to be addressed in our future investigations. Prominently, dealing with independent reading orders is fundamental. There are at least two reasons for this: On the one hand, many documents contain texts which are independent from another and need not be read in a strict sequential order. Newspapers are the most prominent example, but are not the only one. Magazines, scientific documents with frames or tables are all examples of this sort. On the other hand, within the same document, items of different nature cannot be constrained to be within a unique total reading order. A picture immersed in a surrounding of text is very likely to be related to that text, but at the same time it need not be placed exactly after one word rather than another one or after one block of text rather than another one. Partial orders, instead of strict total orders, represent the logical structure of documents in a better way and should be the final output of a system extracting the logical structure from a scanned document.

Of independent interest is the vertical integration of the presented framework. One moves from image processing for scanned documents, to natural language processing, passing through constraint satisfaction techniques and formal methods. The integration of layout, spatial and natural language information is innovative for document analysis and understanding. We have made a first attempt, but more integration is possible. For example, distinguishing a title from a quotation which are in-between blocks of text that form columns is not possible by resorting only to layout information, or by applying spatial reasoning constraints. It is only the textual content of the document object which can allow to distinguish a title from a citation. Thus, natural language processing which we have used to disambiguate among spatially admissible reading orders, could be of great help also to assign types to individual document objects; setting another step towards information integration for document analysis and understanding.

ACKNOWLEDGMENTS

Leon Todoran is supported by Senter den Haag and Océ Technologies BV, Venlo (IOP project IBV 96008). Marco Aiello is supported in part by the Italian National Research Council (CNR), grant 203.15.10. Christof Monz is supported by the Physical Sciences Council with financial support from the Netherlands Organization for Scientific Research (NWO), project 612-13-001.

REFERENCES

1. FineReader, "http://www.abbyy.com."
2. S. Tsujimoto and H. Asada, "Major Components of a Complete Text Reading System," *Proceedings of the IEEE* **80**(7), pp. 1133–1149, 1992.
3. Y. Ishitani, "Logical structure analysis of document images based on emergent computation," in *ICDAR'99*,[22] pp. 189–192.
4. F. Cesarini, E. Francesconi, M. Gori, and G. Soda, "A two level knowledge approach for understanding documents of a multi-class domain," in *ICDAR'99*,[22] pp. 135–138.
5. G. Nagy and S. Seth, "Hierarchical representation of optically scanned documents," pp. 350–369, 1984.
6. G. S. Palermo and Y. Dimitriadis, "Structured document labeling and rule extraction using a new recurent fuzzy-neural system," in *ICDAR'99*,[22] pp. 181–184.
7. X. Li and P. Ng, "A document classification an extraction system with learning ability," in *ICDAR'99*,[22] pp. 197–200.
8. M. Aiello, C. Monz, and L. Todoran, "Combining linguistic and spatial information for document analysis," in *Proceedings of RIAO'2000 Content-Based Multimedia Information Access*, J. Mariani and D. Harman, eds., pp. 266–275, CID, 2000.
9. J. Allen, "Maintaining knowledge about temporal intervals," *Communications of the ACM* **26**, pp. 832–843, 1983.
10. F. Aurenhammer, "Voronoi diagrams–a survey of a fundamental geometric data structure," *ACM Computing Surveys* **23**(3), pp. 345–405, 1991.
11. G. Nagy, "Twenty years of document image analysis in PAMI.," *IEEE Trans. Pattern Analysis and Machine Intelligence* **22**(1), pp. 38–62, 2000.
12. U. Garain and B. Chaudhuri, "Extraction of type style based meta-information from imaged documents," in *ICDAR99*,[22] pp. 341–344.
13. A. Jain, P. Duin, and J. Mao, "Statistical pattern recognition: A review," *IEEE Transactions on PAMI* **22**(1), pp. 4–37, 2000.
14. J. van Benthem, *The Logic of Time*, vol. 156 of *Synthese Library*, Reidel, Dordrecht, 1983. [Revised and expanded, Kluwer, 1991].

15. P. Balbiani, J. Condotta, and L. Fariñas del Cerro, “A model for reasoning about bidimensional temporal relations,” in *Proceedings of the 6th International Conference on Principles of Knowledge Representation and Reasoning (KR’98)*, A. Cohn, L. Schubert, and S. Shapiro, eds., pp. 124–130, Morgan Kaufmann, 1998.
16. Eclipse Constraint Logic Programming System, “http://www-icparc.doc.ic.ac.uk/eclipse.”
17. H. Schmid, “Probabilistic part-of-speech tagging using decision trees,” in *Proceedings of International Conference on New Methods in Language Processing*, 1994.
18. C. Manning and H. Schütze, *Foundations of Statistical Natural Language Processing*, MIT Press, 1999.
19. J. Sauvola and H. Kauniskangas, “MediaTeam Document Database II. CD-ROM collection of document images, University of Oulu, Finland.” http://www.mediateam.oulu.fi/MTDB/index.html.
20. R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*, Addison Wesley, 1999.
21. R. Baayen, R. Piepenbrock, and L. Gulikers, “The CELEX lexical database (release 2).” Distributed by the Linguistic Data Consortium, University of Pennsylvania, 1995.
22. *Proceedings of the 5th International Conference on Document Analysis and Recognition (ICDAR’99)*, IEEE, 1999.