Document Image Analysis via Model Checking

Marco Aiello

Institute for Logic, Language and Computation, and
Intelligent Sensory and Information Systems
University of Amsterdam
Plantage Muidergracht 24 1018 TV Amsterdam, The Netherlands
aiellom@ieee.org

1 Introduction

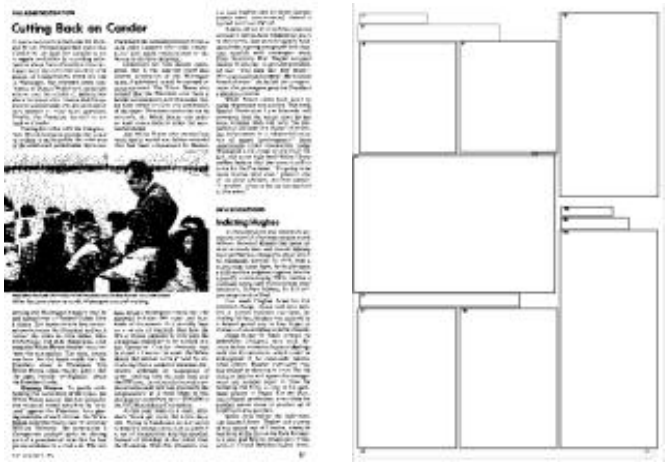
When Dave placed his own drawing in front of the 'eye' of HAL—in 2001: A Space Odyssey—HAL showed to have correctly comprehended and interpreted the sketch. "That's Dr. Hunter, isn't it?" [9]. But what would have happened if Dave used the first page of a newspaper in front of the eye and started discussing its contents? Considering HAL a system capable of AI, we expect HAL to recognize the document as a newspaper, to understand how to extract information and to understand its contents. Finally, we expect Dave and HAL to begin a conversation on the contents of the document.

Here we present a methodology based on model checking, which has been successfully experimented on an heterogeneous collection of documents [1, 11], to extract the content from images of documents. We focus on mechanically generated documents, in contrast with hand-writing and sketches. Using terms better-known to the image processing community, we are interested in logical structure detection in the context of document image analysis.

Document image analysis is the set of techniques involved in recovering syntactic and semantic information from images of documents, prominently scanned versions of paper documents. An excellent survey of document image analysis is provided in [8] where, by going through 99 articles appeared in the IEEE's Transactions on Pattern Analysis and Machine Intelligence, Nagy reconstructs the history and state of the art of document image analysis. Research in document images analysis is useful and studied in connection with document reproduction, digital libraries, information retrieval, office automation, and text-to-speech.

There are two distinct tasks in document image analysis. The first has a syntactical goal consisting of the identification of basic components of the document, the so-called *document objects*. The second has a semantic goal consisting of the identification of the role and meaning of the document objects in order to achieve an interpretation of the whole original document. The syntactic information is synthesized in the *layout structure* of the document, while the semantic information goes under the name of *logical structure*. In the latter task, two subtasks are usually identified: logical labeling, and reading order detection. *Logical labeling* consists of the assignment to document objects of labels indicating their

role (page number, title, sub-title, etc.). *Reading order detection* aims at reconstructing the sequence of textual document objects in which the user is going to (or is supposed to) read the document at hand.



(a) (b)

Fig. 1. A document image (a) and its layout information (b).

The pattern matching and, more generally, the computer vision communities have been very active in the field, especially in tasks tied to the layout structure detection. On the other hand, logical structure extraction has only been dealt with in very restricted domains, like the interpretation of addresses on envelops. A fundamental step toward generality is to be found in [12]. Tsujimoto and Asada present an algorithm to extract the logical structure from two-column black and white images of documents with a simple known a priori layout. On the negative side, no properties of the algorithm are known, the framework can not be extended to any other type of document (especially if the layouts get intricate) and no use is made of the textual information available.

The technique we use for reading order detection is based on model checking. Model checking is one of the most successful applications of logic techniques in computer science due to a number of factors, including effectiveness and robustness [6, 5, 4]. All model checking systems known in the literature involve a temporal logic and a model with a finite number of states. The system which is being modeled is represented as a finite state transition system, and specifications are expressed in a propositional temporal logic. The model checker works by exploring all the states of the model, in this way it is possible to automatically check if the specifications are satisfied. The termination of model checking is guaranteed by the finiteness of the model.

In the next section, we illustrate how to transform document information into a formal spatial model, overviewing the methodology we propose. In Section 3, we illustrate a test case for the methodology in which we focus on the detection of the reading order. In Section 4, we give some concluding remarks and open directions for future research.

2 Document Images as Formal Models

Given a document image, we assume available its layout and logical labeling information, that is, we assume identified the basic entities of the document, the document objects, their location and their logical type. Considering, for example, the document image in Figure 1.a, we assume given the segmentation of the document as represented in Figure 1.b together with its labeling information.

The core of the methodology we propose lies in viewing the layout together with the logical labeling information, as a spatial model, while considering logical document rules as formulas of a specific logic. The process of extracting logical information is then defined as an instance of model checking. The formulas encoding document logical rules are checked against the model of a given document image. The states (document objects) and paths (totally ordered collections of document objects) satisfying these formulas are the logical structure extracted from the document image. In Figure 2, we summarize the methodology. On the top left, a document image is represented. Via image processing techniques one gets the layout and logical labeling information, [11]. Here we assume that it is given. This information is transformed into a spatial model, as we shall see next. The model is then used in `SpaRe`, our model checker. In our current setup, the formulas used for model checking are written by an expert, but it is easy to imagine that these could be directly written by the document author, or they could be learned automatically. For the first case, imagine the designer of a magazine to write down which are the formal rules he follows in editing his magazine. While for the second, think of having a set of journals to analyze and to provide the correct logical structure for a number of issues. The learning system attempts to mine the formal rules behind the journal. These rules could then be used to analyze the whole collection of the journal.

Let us analyze more precisely the transition from layout to a formal model. In the present context, the layout is a set of document objects together with geometrical information

$$DO = \{do \mid do = \langle id, x_1, y_1, x_2, y_2 \rangle\}$$

where id is an identifier of the document object and (x_1, y_1) (x_2, y_2) represent the uppermost-leftmost corner and the lowermost-rightmost corner of the bounding box of the document object.[1] In addition, we consider the logical labeling

[1] Superimpose a coordinate system to the document image. The leftmost uppermost corner has coordinates (0,0). The x axis spans horizontally increasing to the right, while the y axis spans vertically towards the bottom.

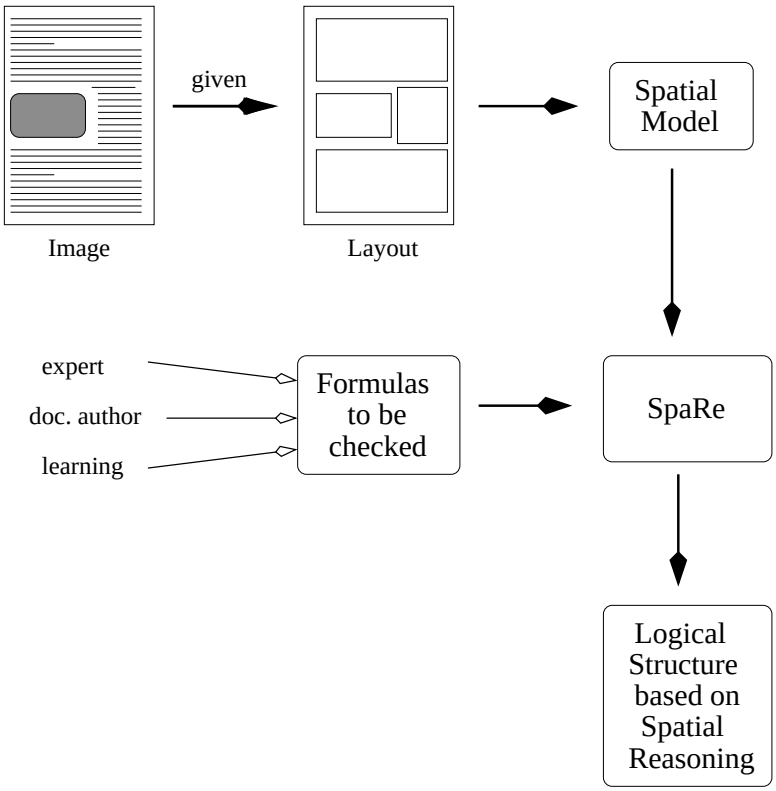


Fig. 2. The flow of information in document analysis as model checking.

information. Logical labels are associated with each document object and describe their function. Common examples are: title, subtitle, body, page number. Given a set of labels L, logical labeling is a function *lab*, typically injective, from document objects to labels:

$$lab: \ DO \rightarrow L$$

As for the formal model, we consider a special spatial kind: a *spatial model* is a tuple $\langle S, R, \nu, \rangle$, where S is a set of states, R is a set of bidimensional Allen's relations and $\nu : S \rightarrow L$ is a valuation function mapping states to proposition letters (the labels L in this case). The set of relations R consists of 13×13 relations, that is, the product of Allen's 13 interval relations [2] (precedes, meets, overlaps, starts, during, finishes, equals, and their inverses) on two orthogonal axes. The definition closely resembles the one of a rectangle model of Balbiani *et al.* in [3]. It is now easy to translate the layout and logical labeling information available for a document image into a spatial model, as we show next.

Definition 1 (spatial translation). A document image, i.e., a set of document objects DO and a labeling function lab, is *translated* into a spatial model $\langle S, R, \nu, \rangle$ by $\cdot^t$ in the following way:

1. For all $do \in DO$:

$$do^t \in S$$

2. For all $do^t \in S$:

$$\nu(do^t) = lab(do)$$

3. For all $do_i^t, do_j^t \in S$:

$$(do_i^t, do_j^t) \in R_k \text{ for some } R_k \in R$$

where R_k is given by Allen's relation on the x axis between the intervals $[x_{i1}, x_{i2}]$ and $[x_{j1}, x_{j2}]$, and by Allen's relation on the y axis between the intervals $[y_{i1}, y_{i2}]$ and $[y_{j1}, y_{j2}]$.

It is immediate to notice that $|S| = |DO|$ and $|R| = 13 \times 13$. Less obvious may be the fact that there is always a relation $R_k \in R$ among any two document objects. This follows from the fact that the Allen's relations are jointly exhaustive and pairwise disjoint, that is, given any two rectangles there is always one and only one bidimensional Allen relation holding among them. For example, a rectangle is equal on the x and equal on the y with itself, while the leftmost-uppermost and the rightmost-lowermost bounding boxes in Figure 1.b are precedes on the x and precedes on the y relation (in this order). If we look at the spatial model as being a directed graph, we notice that document object are nodes, labeled by the valuation function, that the graph is fully connected (there is a directed edge from any node to any node), and that for every edge connecting two nodes and denoting a bidimensional Allen relation, there is a converse edge denoting the inverse of the bidimensional Allen relation.

3 A test case

We have applied the proposed methodology to extract the reading order from an heterogeneous collection of documents. (For the experimental results and implementation choices see [1, 11], here we present the relation of the implementation with the methodology based on model checking.) First, we consider a sub-model of the spatial one defined in the previous section. We prune the model of many of its relations in R following the rule to keep only the relations that represent a "before in reading transition." Intuitively, we consider a document object to be before in reading of another one if it precedes or meets it on either axis, if it contains it or if it overlaps with it. For the full set of Allen's bidimensional relations that we consider to identify a before in reading relation we refer to [1]. Second, we regard the set of pruned relations R as a unique transition relation

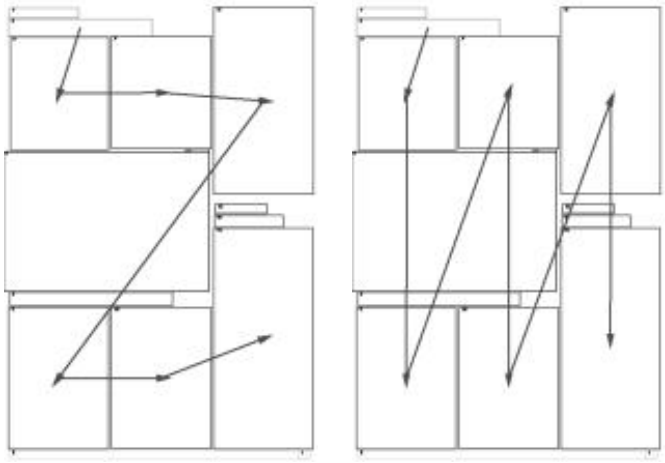
"before in reading." The new model just identified is a directed transitive graph (i.e., a directed graph, with the edge relation transitively closed). The modal logician may recognize in it the typical structure of an S4 model. Third, we consider the logic CTL to express rules to extract the reading order. In particular, the reading order for a page is given by the formula

$$EG((\texttt{title} \vee \texttt{body of text}) \wedge \neg\texttt{page no} \wedge \neg\texttt{caption})$$

for the complete syntax of CTL we refer to [5], here we just give an informal explanation. The formula states that there exists a path such that all elements of the paths are either labeled as title or body of text and are not captions and not page numbers.

In general, various paths in the model will satisfy a given CTL formula like the one above. Of all paths, we consider only those of maximal length. The driving assumption is that all textual document objects must contribute to the reading order. We further rank the remaining paths by considering the natural language probability of each given transition. By using shallow natural language processing techniques and retrieving the textual content of each document object associated with a state of the model, we assign a probability to each reading order found.

Experimental results and details of the natural language tools used are presented in [11]. For example, the result of the model checking for the document in Figure 1 are presented in Figure 3. Two reading orders are identified. The natural language processing then assigns a higher probability to the reading order on the right in Figure 3.



(a) (b)

Fig. 3. The result of model checking the document in Figure 1.

4 Concluding Remarks and Future Work

We have presented a new approach to the problem of logical structure detection in document image analysis. The prominent feature of the approach is its modularity. To extract different sorts of logical information from a document, one only needs to rewrite the modal formula to be checked and leave the whole architecture of the system untouched.

In the context of document image analysis, the presented proposal is the first to use Allen relations at the semantic level. Up to now Allen's relations have been only used as a feature descriptor (thus at the syntactic level of a layout property) [7, 10].

For future research there are various questions and directions open for investigation. The first question regards which is the appropriate logic to use for model checking with the most general spatial model as defined in Section 2. Given the sort of transitions in the model, bidimensional Allen relations, the appropriate starting point looks like a bidimensional generalization of Venema's logic of chopping intervals [13], but the issue is still open. The next challenge for us consists of identifying appropriate formulas to deal with independent reading orders. Independent reading orders arise in complex documents with independent portions of text. A typical example is the first page of a newspaper, where different unrelated news all appear together in the same page. A final important issue resides in automatically finding formulas describing logical rules of a document. Data mining and learning techniques seem the most appropriate to achieve this goal; the road to the solution is completely open for investigation.

References

1. M. Aiello, C. Monz, and L. Todoran. Combining Linguistic and Spatial Information for Document Analysis. In J. Mariani and D. Harman, editors, *Proceedings of RIAO'2000 Content-Based Multimedia Information Access*, pages 266–275, Paris, 2000. CID.
2. J. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26:832–843, 1983.
3. P. Balbiani, J. Condotta, and L. Fariñas del Cerro. A model for reasoning about bidimensional temporal relations. In A. G. Cohn, L. Schubert, and S. Shapiro, editors, *Proceedings of the 6th International Conference on Principles of Knowledge Representation and Reasoning (KR'98)*, pages 124–130. Morgan Kaufmann, 1998.
4. A. Cimatti, E. M. Clarke, F. Giunchiglia, and M. Roveri. NUSMV: A New Symbolic Model Checker. *International Journal on Software Tools for Technology Transfer*, 2(4):410–425, 2000.
5. E. Clarke, O. Grumberg, and D. Peled. *Model Chekcing*. MIT Press, 1999.
6. E. M. Clarke, E. A. Emerson, and A. P. Sistla. Automatic verification of finite-state concurrent systems using temporal logic specifications. *ACM Transactions on Programming Languages and Systems*, 8(2):244–263, 1986.
7. S. Klink, A. Dengel, and T Kieninger. Document Structure Analysis Based on Layout and Textual Features. In *Fourth International Workshop on Document Analysis Systems*. IAPR, 2000.

8. G. Nagy. Twenty Years of Document Image Analysis in PAMI. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(1):38–62, 2000.
9. A. Rosenfeld. Eyes for Computers: How HAL Could "See". In D. Stork, editor, *HAL's Legacy*, pages 210–235. MIT Press, 1997.
10. R. Singh, A. Lahoti, and A. Mukerjee. Interval-algebra based block layout analysis and document template generation. In *"Workshop on Document Layout Interpretation and its Applications (DLIA99)"*, Bangalore, India, September 1999. http://www.wins.uva.nl/events/dlia99/final_papers/singh.pdf.
11. L. Todoran, M. Aiello, C. Monz, and M. Worring. Logical structure detection for heterogeneous document classes. In *Document Recognition and Retrieval VIII*, pages 99–110. SPIE, 2001.
12. S. Tsujimoto and H. Asada. Major Components of a Complete Text Reading System. *Proceedings of the IEEE*, 80(7):1133–1149, 1992.
13. Y. Venema. A Modal Logic for Chopping Intervals. *Journal of Logic and Computation*, 1(4):453–476, 1991.