KNOWLEDGE-BASED INTERPRETATION OF SCANNED BUSINESS LETTERS

J. Kreich, A. Luhn, G. Maderlechner
Siemens AG, Corporate Research and Development

Otto-Hahn-Ring 6
D-8000 München 83, F. R. G.

ABSTRACT

Office Automation by electronic text processing has not reduced the amount of paper used for communication and storage. The present boom of FAX-Systems proves this tendency. With this growing degree of office automation the paper-computer interface becomes increasingly important. To be useful, this interface must be able to handle documents containing text as well as graphics, and convert them into an electronic representation that not only captures content (like in current OCR readers), but also the layout and logic structure.

We describe a system for the analysis of business letters which is able to extract the key elements of a letter like its sender, the date, etc. The letter can thus for instance be stored in electronic archival systems, edited by structure editors, or forwarded via electronic mail services.

This system was implemented on a Symbolics Lisp machine for the high level part of the analysis and on a VAX for the low and medium level processing stages. Some practical results are presented and discussed. Apart from this application our system is a useful testbed to implement and test sophisticated control structures and model representations for image understanding.

INTRODUCTION

Despite the much talked-about paperless office, paper remains an important medium for display, storage, and transmission of information for human beings. Also, personal computers and laser printers are effecting an increase in the use of paper in the office environment. Therefore the transformation of paper documents into a form that enables the electronic processing of the original information becomes increasingly important. The computer representation not only has to capture the plain ASCII content of a document, but also the logic and layout strucure. A useful system should be able to deal not only with forms having a fixed layout, but also with documents that show a considerable variation in their appearance and include graphics and halftone images.

SYSTEM OVERVIEW

The main processing steps in our approach to document analysis are the following: A document is scanned and converted to a binary raster with a typical resolution of 200 dpi. The connected components (sets of topologically connected pixels) are found and classified according to their compliance with one of the classes text, graphics (i.e. black and white line drawings) and halftone images [9,10]. The layout structure, that is the entities text-block, line, and word, their sizes and positions, are found from the geometrical arrangement of the text components. The recognition of the actual characters (OCR, for our methods see Bernhardt-84) is based on an analysis of the raster image of a whole line rather than on the isolated character components. This allows the seperation of fused character images, a common occurrence in the scanned document image or even the original. Up to now all processing steps are performed without any specific knowledge regarding the documents to be analysed, apart from the general knowledge of what comprises a word, line, or text-block, or the equally general (albeit very sophisticated) methods for OCR. The further analysis can only proceed with a model of the documents that are to be recognized.

Our model uses concepts of the layout and logical structure of documents similar to the standardized Office Document Architecture (ODA) [8]. We use an initially heuristic search procedure to compare key parts of document models with the document that is to be analysed. As the search (and match) progresses, successfully analysed parts guide the further selection of model and document parts that are to be analysed. Backtracking is thus dependent on the current context of the analysis and might also trigger a resegmentation of parts of the layout structure, if there is sufficient evidence for a faulty layout segmentation. When a sufficient number of parts have been identified, the document is considered to be recognized, and its logical components are instantiated with the corresponding objects in the original document. From this a text file can be generated with tags that designate the logical meaning of the text content (for instance in SGML or TeX format). An overview of the components of our document analysis system SODA (Sys-

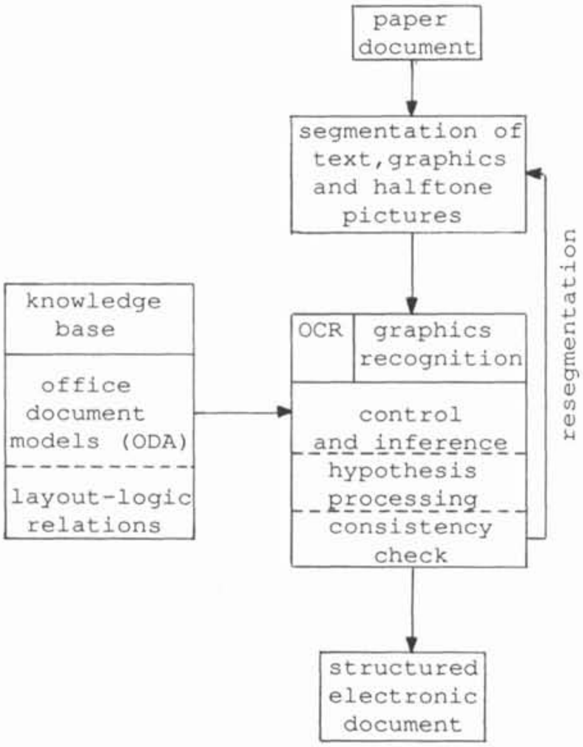

Figure 1: Overview of the System for Office Document Analysis (SODA).

tem for Office Document Analysis) is shown in Figure 1. SODA is a flexible and expandible testbed for different document analysis approaches. A top-down approach of a layout analysis using horizontal and vertical cuts through the document is reported in [5], and an augmented transition net (ATN) is used in [2].

A prototype of the system has been implemented on a VAX and a Symbolics Lisp machine. The connected component analysis, text/graphics/halftone classification, and the optical character recognition are performed on the VAX. The layout analysis and the knowledge-based analysis are implemented on the Lisp machine. Currently our document model only comprises the restricted class of (typewritten) business letters. In the sequel we concentrate on a brief description of the layout segmentation process and - in some more detail - the knowledge-based analysis.

LAYOUT SEGMENTATION

The analysis of size and arrangement of text blocks on a page is the first step for the knowledge-based analysis. Therefore an important part of the analysis of textual documents is the recognition of their layout structure, that is the grouping of characters into words, words into lines, and lines into text-blocks. Standard procedures use horizontal and vertical projections, subsampling, or dilation operations on the binary image (see e.g. [1,7]) to find these components. Our approach is based on the more abstract data structure that is the result of a connected component analysis and the text/-graphic/halftone classification [10] . Furthermore, instead of building successively larger structures, we first search for the largest and most robust (with respect to noise in the image) structures, namely text-blocks. These are also the most characteristic elements of a document layout. So, to group connected components into text-blocks, we collect all those that are "close enough" to each other. This distance criterium of cause depends on the size of the text font in the document. Therefore we choose the limiting distance according to the average character height. As long as the font sizes used in the document differ not excessively, this procedure yields results that are not very sensitive to the exact choice of the distance parameter.

The next two stages consist in finding the lines within each of the text-blocks and the words within each of the lines. Here the geometric grouping is more sensitive. The average character height therefore is determined for each text-block separately. Furthermore, the characters that consist of more than one connected component, like for example "i", have to be combined into a single entity. Otherwise the separate parts of these characters would modify the statistics for the average character height, possibly leading to erroneous results for the line and word finding. For each of the layout objects, we keep the parameters used for their segmentation. In this way we are able to perform a resegmentation of the layout objects, should it prove necessary later in the course of the analysis.

KNOWLEDGE-BASED ANALYSIS

Knowledge Base: The knowledge base contains domain specific knowledge about a representative sample of business letters and heuristic knowledge for control of the analysis. In SODA this knowledge is represented in frame-like concepts [4]. A concept is defined by a name, necessary and optional relations and generic functions. The implementation is an extension of the Symbolics Flavor System with its inheritance and instantiation mechanisms. The specific document knowledge is represented by a class hierarchy (relation "is-a") and hierarchy of document parts (relation "parts") for the layout and logical structure of business letters in accordance to the office document architecture (ODA, [8]). But ODA knowledge is not sufficient for the analysis task. With additional relations (e.g. layout-logic relations) and procedural knowledge (e.g. generic functions for geometric

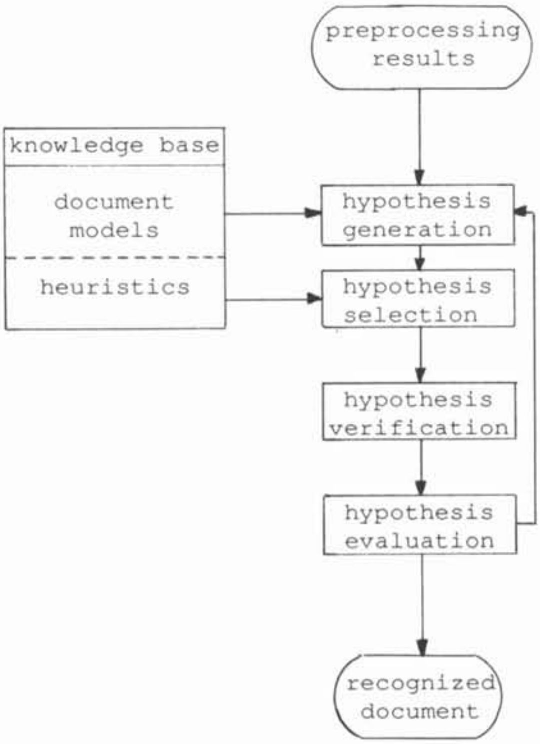

Figure 2: Hypothesis processing module

neighbourhood) our knowledge base allows a flexible and situation dependent analysis. As an aside we note that these concepts enable the representation of rules as predicate-calculus formulas like: if is-a (x, name) and part-of (x, y) then is-a (x, address).

Control and Inference: The document analysis begins with a start-hypothesis, which is predefined by a heuristics concept in the knowledge base, for example the search for a frequent document class or a typical part. The processing of this and the following hypotheses is done in a four step cycle (Figure 2). Repeated looping in this cycle builds a growing network of hypothesis nodes over the document. A hypothesis is defined as a predicate-calculus formula in an object-oriented way by instantiation of a concept, which finally has the truth value true or false. The inheritance property of concepts induces a dependency of hypotheses, which has to be supervised by the control module. For more complex documents a truth mainenance system (see e.g. [6]) is more convenient.

Hypothesis Generation and Selection: The hypothesis generation module may produce new hypotheses by specialization or generalization of the start-hypothesis in accordance with the given heuristic concept. It is controlled by the heuristics concepts providing search strategies such as best-first search or dependency directed backtracking. This avoids exhaustive search or deletion of possibly true hypotheses in the growing network (focus of attention).

Verification and Consistency Check: The verification is composed of a relational and a procedural test of the current hypothesis. The relational verification has to consider all hypotheses which are induced by the necessary relations in the actual concept. The procedural verification applies the relevant generic functions in the concepts, e.g. test of the shape and position of an address frame and calculation of confidence measures. If this step results in a successful verification and yields a confidence value above some threshold, then the result is passed as a true hypothesis to the next module (consistency check) and remembered for future references. The consistency check is necessary if more than one true hypothesis has to be evaluated. In this case the intermediate results like confidence values and supporting hypotheses in the focus have to be newly assessed, which might even involve the invocation of preprocessing steps with modified parameters and/or methods (resegmentation). The final result depends on the choice of the concept for the combination rules of different knowledge sources, e.g. probalibistic or fuzzy methods. In our example below we used a simple cut off, because only one hypothesis succeeded (Fig. 3).

RESULTS

The knowledge-based analysis is implemented and tested in Common Lisp and Flavors on a Symbolics 3640. The current knowledge base consists of about 150 concepts. New concepts of document models or heuristics may be defined by a graphical oriented knowledge acquisition module. In Figure 3 we visualize the recognition of a business letter with the start-hypothesis CCITT-letter (corresponding to the facsimile test document No. 1 of the CCITT) and a successful verification without inconsistencies. The scanned letter is segmented into a hierarchy of layout-part frames. The CCITT layout model is shown by the dashed overlay. The recognized letter components are marked by bold rectangles and corresponding names, and the whole document is classified correctly. While this example only makes use of the layout information, future developments will include the logical structure and content of the document in the analysis concept.

Acknowledgements: This work was supported in part by the German Ministry of Research and Technology, under contract No. 413-5839-ITM 8501B/7.

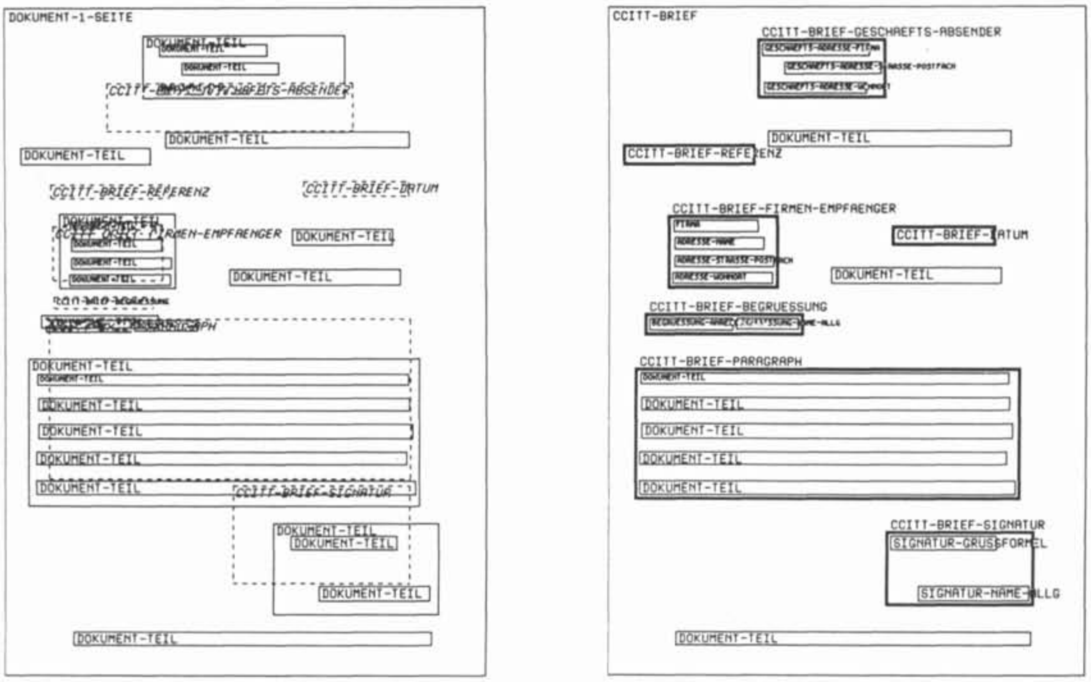

Figure 3: Scanned and segmented business letter with dashed overlay of the document model (left hand side), and the recognized document class and document parts (indicated by bold rectangles and descriptions on the right hand side).

REFERENCES

[1] Baird, H. S.: Proc. Int. Workshop on Syntactic and Structural Pattern Recognition, Nancy, France (1988).

[2] Bergengruen, O., Luhn, A., Maderlechner, G., and Ueberreiter, B.: *Dokumentanalyse mit ATN's und unscharfen Relationen*, Informatik Fachberichte 149, p.78-81 (Springer Verlag 1987).

[3] Bernhardt, L.: *Three Classical Character Recognition Problems, Three New Solutions*, Siemens Research and Development Reports 13, p.114-117 (1984).

[4] Brachman, J.R. and Schmolze, J.G.: *An Overview of the KL-ONE Knowledge Represenation System*, Cognitive Science 9, p.171-216 (1985).

[5] Dengel, A., Luhn, A., and Ueberreiter, B.: *Model Based Segmentation and Hypothesis Generation for the Recognition of Printed Documents*, SPIE vol. 860, p.89-95 (1988)

[6] deKleer, J.: *Assumption-Based TMS*, Artificial Intelligence 28, p.127-162 (1986).

[7] Nagy, G., Seth, S., and Stoddard, S.: *Document Analysis with an Expert System*, Pattern-Recognition in Practice II, p. 149-159 (1986).

[8] ISO 8613: *Office Document Architecture (ODA) and Interchange Format (ODIF)*, March 1988.

[9] Postl, W.: *Halftone Recognition by an Experimental Text and Facsimile Workstation*, Proc. of the ICPR, Munich p.489-491 (1982).

[10] Scherl, W.: *Unified Analysis of Complex Document Patterns*, Proc. 4th Scandinavian Conf. on Image Analysis, Trondheim, p.873-880 (1985).