Form Analysis with the Nondeterministic Agent System (NDAS)

Thomas C. Henderson and Lavanya Swaminathan
School of Computing
University of Utah
Salt lake City, Utah 84112, USA
tch@cs.utah.edu

February 27, 2003

Abstract

We have been developing a framework within which multiple agents cooperate to analyze scanned images of engineering drawings. While our main focus has been on extracting dimension annotations from CAD drawings, we describe here how our method can be applied to form processing. The basic ideas behind our approach are (1) allow the nondeterministic exploration of a large search space, and (2) exploit structural constraints to rapidly prune the search space.

1 Introduction

The complete analysis of engineering drawings is still an unsolved problem. Most extant systems exhibit brittle performance when applied to real world image sets. (See Tombre [1] for an overview of the field and Kanungo et al. [2] for insightful commentary.) An approach very similar to ours is that of Bapst et al. [3]. They propose a system for Cooperative and Interactive Document Reverse Engineering (CIDRE). Its major features are that it:

- is human-assisted,
- provides automatic inference of document models,
- has a cooperative, multi-agent architecture,
- explores the search space concurrrently,
- commits to an interpretation only when confidence is high, and
- provides for local revisions of belief.

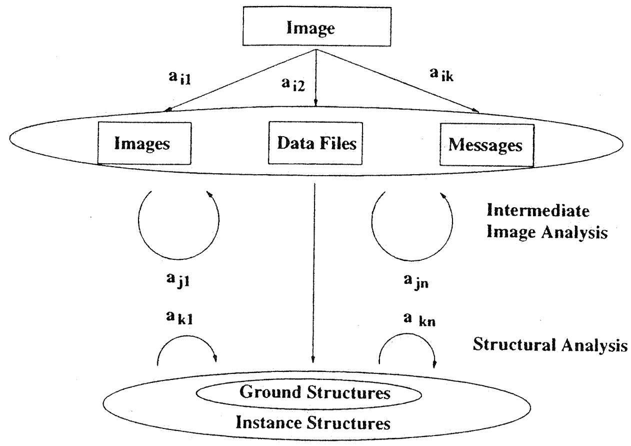


Figure 1: NDAS Agent Organization

We have developed a similar approach called the NonDeterministic Agent System (NDAS). NDAS shares some of CIDRE's philosophy; however, as the authors of CIDRE note: "one of the main drawbacks in our proposition is the present lack of a high-level reasoning language that can express intelligent behavior." Our view is that this can be addressed by putting into place a mechanism to control the combinatorics of the search space growth. We achieve this through the exploitation of structural constraints in the underlying document, and through feedback mechanisms to reinforce paths that lead to successful interpretations. See Swaminathan [4] for details on the NDAS approach as applied to technical drawing analysis.

2 Nondeterministic Agents

NDAS is basically a blackboard system with a set of independent agents, each with expertise on some particular aspect of the analysis problem. In addition, there are agents that monitor the form analysis as it proceeds and give feedback to reinforce successful results. (See Figure 1.)

An agent is a software process that:

- is autonomous: runs independently of other agents;
- has state: assertions, relations memory;
- is persistent: never terminates;
- can communicate: send and receive messages related to the analysis;
- can sense: is aware of or can detect relevant input or conditions;
- and can act: analyze or create information.

For more complete accounts, see [5,6].

An agent architecture is a software architecture for decision making with intelligent (flexible) processes embedded within it. The agents may be proactive or reactive, and should cooperate (including communicate) to achieve a goal.

We explore the use of nondeterministic agent systems (NDAS) to achieve a more flexible system for technical drawing analysis. They are called nondeterministic because the agents explore alternative parts of the solution space simultaneously, and every agent works to produce some result which may or may not contribute to the final result. The final result derives from only a subset of the work put in by all the agents. We explore nondeterminism in this problem domain since deterministic systems usually make irrevocable decisions (e.g., threshold selection) that eliminate possible solutions. The technical drawing problem domain contains many factors that vary with the drawing: thresholds, text fonts and size, noise levels, etc., and this variation makes it interesting to explore the possible solution space dynamically and in a breadth-first way.

We demonstrate this through the design and analysis of the NDAS system and provide experimental results to support the claims. (See [4] for more detail.)

3 Form Structure and Constraints

Forms may be described in terms of the layout and relations between the logical parts of the form. Several methods have been previously proposed to perform document structure analysis and to identify form classes; e.g., [7] describes a document analysis system to analyze forms from the French social services department (les Allocations Familiales). Others have reported on a variety of issues: Wang and Shirai [8] discuss a bottom-up approach that segments boxes, line segments and text into components, determines spatial relationships between boxes and text, and separates text from lines that touch or overlap. They give several examples of removing lines. Ha and Bunke [9] demonstrate good results of model-based analysis and understanding of checks; Arai and Odaka [10] perform background region analysis and extract box regions; they tested 50 types of forms; Tang and Lin [11] show how semantic information can be acquired and forms efficiently stored.

For our work here, we assume that a form model has been acquired and is expressed as a structural model including spatial relation constraints. For example, the box structure of the form in Figure 2 could be described as:

```
Rule 1: form := A + B + C
        [side(A,1)=side(B,3);
         len(A,2)=len(B,2)]

Rule 2: A := box

Rule 3: B := box1 + box2
```

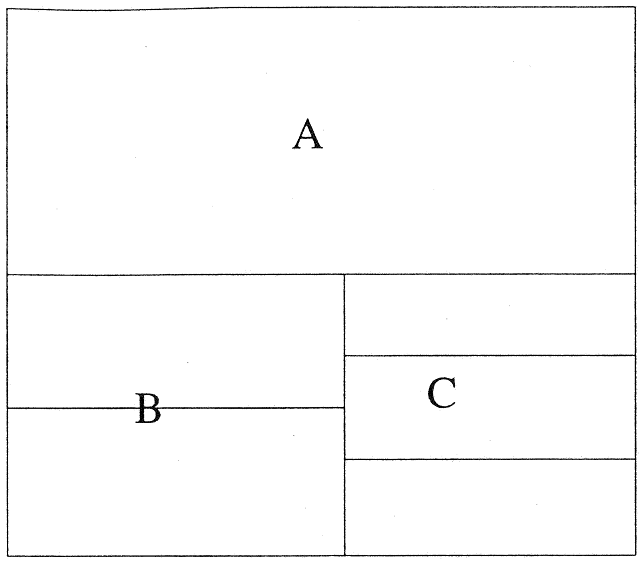


Figure 2: Boxes Example

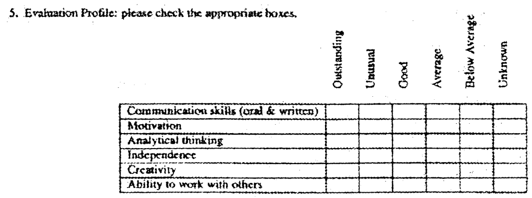
```
        [side(1,3)=side(2,1);
        len(1,2)=len(2,2)]

Rule 4: C := box1 + box2 + box3
        [side(1,3)=side(2,1);
        side(2,3)=side(3,1);
        len(1,2)=len(2,2);
        len(1,2)=len(3,2)]
```

Simple rectangles form the lowest level terminal symbols; A is a nonterminal representing the top box; B represents the lower left 2 boxes; C represents the lower right 3 boxes; $form$ represents A, B, and C in their proper configuration. The sides of boxes are numbered lower, right, upper, and left as 1, 2, 3 and 4, respectively. The square brackets in the rules contain the constraints as predicates about the geometry of the boxes.

We have done a number of detailed experiments on the analysis of technical drawings [4]. As for the analysis of document forms, we have only performed an initial study at the request of the SDIUT organizers, and thus, can only report on our first cut at using NDAS on this problem. The initial results are encouraging.

NDAS already has image analysis agents for the first steps in form processing: thresholding, edge and box extraction; we have added agents to perform the structural analysis described in the rewrite rules given above. If there are only terminal symbols (simple boxes) on the right hand side of the rewrite rule, then the agent looks for simple boxes produced directly from the image analysis; if they satisfy the required relations (if any), then an instance of the left hand



5. Evaluation Profile: please check the appropriate boxes.

	Outstanding	Unusual	Good	Average	Below Average	Unknown
Communication skills (oral & written)						
Motivation						
Analytical thinking						
Independence						
Creativity						
Ability to work with others						

Figure 3: A Graduate Reference Form

side of the rule is produced. Otherwise, the agent watches for instances of the nonterminals to be created and then goes to work once they appear.

Figure 3 shows part of the form that we use in the School of Computing in our graduate admissions application process; we would like to automatically extract the information from these legacy forms. We hope to report at the meeting more detailed results of the application of NDAS to the understanding of such forms.

4 Conclusions and Future Work

We have demonstrated elsewhere that the NDAS approach combined with structural pruning methods can be very advantageous [4] for the analysis of technical drawings. Here we have presented some preliminary data on the application of the NDAS approach to document form analysis. A coherent analysis is performed which allows for a wide range of form structures as well as the possibility to explore the large search space necessary for a robust result.

The next steps include:

- develop a more comprehensive document form model,
- perform more in-depth experiments on a wider class of documents,
- investigate feedback mechanisms from higher-level analysis agents to the image analysis and structural parsing agents.

Acknowledgements

This work was supported in part by ARO grant number DAAD19-01-1-0013. Thanks to Jim de St. Germain for help with form document scanning.

References

[1] K. Tombre. Analysis of engineering drawings: State of art and challenges. In *Graphics Recognition - Algorithms and Systems*, volume 1389 of *Lecture Notes in Computer Science*, pages 257–264, Springer Verlag, April 1998.

[2] T. Kanungo, R. M. Haralick, and D. Dori. Understanding engineering drawings: A survey. In *Proceedings of First IARP Workshop on Graphics Recognition*, pages 217–228, University Park, P.A, 1995.

[3] F. Bapst, R. Brugger, A. Zramdini, and R. Ingold. Integrated multi-agent architecture for assisted document recognition. In *Document Image Analysis*, pages 301–317, World Science, Singapore, 1994.

[4] L. Swaminathan. Agent-based engineering drawing analysis. Master's thesis, University of Utah, Salt Lake City, Utah, December 2002.

[5] V. Subrahmanian. *Heterogeneous Agent Systems.* MIT Press, Cambridge, MA, 2000.

[6] G. Weiss. *Multi-Agent Systems.* MIT Press, Cambridge, MA, 1999.

[7] S. Diana, E. Trupin, Y. Lecourtier, and J. Labiche. Document modeling for form class identification. In *Advances in Document Image Analysis*, pages 176–187, Springer-Verlag, Berlin, 1997.

[8] D. Wang and S. Shirai. Analysis of form images. In *Document Image Analysis*, pages 35–56, World Science, Singapore, 1994.

[9] T. Ha and H. Bunke. Model-based analysis and understanding of check forms. In *Document Image Analysis*, pages 57–84, World Science, Singapore, 1994.

[10] H. Arai and K. Okada. Form processing based on background region analysis. In *Proc ICDAR 97*, pages 164–169, Ulm, Germany, 1997.

[11] Y. Tang and J. Lin. Information acquisition and storage of forms in document processing. In *Proc ICDAR 97*, pages 170–174, Ulm, Germany, 1997.