A Framework for Multi-Domain Sketch Recognition

Christine Alvarado, Tevfik Metin Sezgin, Dana Scott, Tracy Hammond, Zardosht Kasheff, Michael Oltmans & Randall Davis

Artificial Intelligence Laboratory
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

http://www.ai.mit.edu

The Problem: People use sketching to interact with one another and to record their ideas across many domains, including mechanical engineering design, software design, human resource management, and many other areas. Currently, almost all sketching is done on paper, with the result that they are frequently left behind during the remainder of the design process, or have to be laboriously transferred to the computer through menu-based systems.

While current hardware allows people to sketch directly on their computers, the ability is not useful unless the computer can understand and represent the meaning of the sketches, rather than just capture the bitmap image. Current recognition systems that attempt this task are brittle and difficult to construct.

This abstract outlines a framework we are developing that would make recognition systems across various domains more robust and easier to construct.

Motivation: We have written sketch recognition systems in various domains, including mechanical engineering designs, UML diagrams, architectural floor plans and finite state machines, and in each case, found that (as in speech recognition) knowledge of the domain provides a substantial improvement in accuracy and robustness of recognition. If we know that the user is drawing finite state machines, for example, it is likely that any closed shape the user draws is meant to be a circle.

But context must currently be built into each recognizer–the routine for recognizing circles in one domain is different from the routine to recognize circles in another. We would like to avoid building context into each individual recognizer, and allow the designer of the recognition system to specify formally both the context rules and the shapes found in a given domain.

Previous Work: Formal grammars for describing shapes and patterns have been explored in architectural design for the last thirty years, starting with work by Stiny [8, 7]. Work in this area is mainly concerned with using grammars for the generation of shapes. In contrast, our work is concerned with using these grammars to guide recognition.

Recognition has been studied in various contexts. The most closely related to our work is other work in sketch recognition. Work by Gross and Do [2], Landay and Meyers [4], and Stahovich [6] all explore various aspects of sketch interpretation within a limited domain. Forbus [1] takes a multi-modal approach to building a sketch interpretation system for military diagrams.

Pattern recognition systems that use context and shape to constrain recognition have also been proposed in the field of computer vision, where the task is to recognize a pattern in a photographic image [5, 9, 3]. This body of work presents a general framework of the type we are aiming for, but we start with hand-drawn stroke data rather than pixel data.

Approach: There are two components to our proposed solution. We first provide a grammar used to specify the shapes expected to be sketched in a particular domain. The system will then compile each shape description into a recognizer capable of sketch processing both from the bottom up—driven by matching against recognizers for simple shapes such as lines and circles—and from the top down—driven by the high-level patterns specified by the user—to recognize the subsequent strokes in the sketch.

A programmer wishing to build a recognition system in a new domain specifies the shapes in the domain in terms primitive shapes and geometric relationships. Figure 1 gives an example of the specification for an and gate.

Each shape will be represented with a fragment of a Bayesian network; low level interpretations (for example, a line and an arc in the and-gate) will provide support for higher level interpretation (the and-gate) which in turn provide support for the other low level interpretations in the pattern (the other two lines).

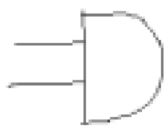
Sketch	Description	
	1. define and-gate 2. NumberOfInputs 3. line l1 l2 l3; arc a1 4. parallel l1 l2 5. same-length l1 l2 6. same-horiz-position l1 l2 7. vertical l3	8. meets l1.p2 l3 9. meets l2.p2 l3 10. semi-circle a1 11. orientation(a1, 180) 12. connected a1 l3 a1.p1 l3.p1 13. connected a1 l3 a1.p2 l3.p2

Figure 1: The formal description of an and-gate. Line 1 starts the definition and gives the name of the object. Line 2 lists attributes that are relevant to the recognized object. Line 3 lists the subcomponents that make up the and-gate. Lines 4–12 describe the relations that must be found among the subcomponents for the object to be recognized as an and-gate.

Impact: Because sketching is useful across many domains, a system that can support multiple domains with less effort from the programmer is more useful than a system tailored to one domain. Incorporating context into recognition systems will also make these systems more robust and more useful in practice.

Future Work: We have outlined a description-driven sketch interpretation system, but have not yet completed implementation, and face several challenges in doing so. For example, we have described a method for describing shapes within a domain, as well as a skeleton of a recognition system in terms of fragments of a Bayesian network. When we actually perform recognition of a user's strokes we will have to determine how to map those strokes to the recognition templates defined in our system.

Our current approach also calls for the programmer to write out a detailed description of each shape, which is a tedious process in and of itself. We would like the system to learn the relationships and subcomponents from drawings, rather than forcing the designer to specify them explicitly.

Research Support: This work is supported in part by MIT's Project Oxygen, by the National Science Foundation, through graduate research fellowships, and by the Ford / MIT Collaboration, under the Virtual Engineering Project.

References:

[1] Kenneth Forbus, R. Ferguson, and J. Usher. Towards a computational model of sketching. In *IUI '01*, 2001.

[2] Mark Gross and Ellen Yi-Luen Do. Ambiguous intentions: a paper-like interface for creative design. In *Proceedings of UIST 96*, pages 183–192, 1996.

[3] V. P. Kumar and U. B. Desai. Image interpretation using bayesian networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(1):74–77, 1996.

[4] James A. Landay and Brad A. Myers. Sketching interfaces: Toward more human interface design. *IEEE Computer*, 34(3):56–64, March 2001.

[5] Jianming Liang, Finn V. Jensen, and Henrik I. Christensen. A framework for generic object recognition with bayesian networks. In *Proceedings of the First Interational Symposium on Soft Computing for Pattern Recognition*, 1996.

[6] T. Stahovich, R. Davis, and H.Shrobe. Generalting multiple new designs from a sketch. *Artificial Intelligence*, 104(1-2):211–264, 1998.

[7] George Stiny. Introduction to shape and shape grammars. *Environment and Planning B: Planning and Design*, 7:343–351, 1980.

[8] George Stiny and James Gips. Shape grammars and the generative specification of painting and sculpture. In C V Freiman, editor, *Information Processing 71*, pages 1460–1465. North-Holland, 1972.

[9] M. E. Westling and L. S. Davis. Interpretation of complex scenes using Bayesian networks. *Lecture Notes in Computer Science*, 1352, 1997.