

Demonstrating the Electronic Cocktail Napkin

Gross, M. D. and E. Do

*Conference Companion, ACM Conference on Human Factors
in Computing (CHI '96) ,*

pp 5-6.

Vancouver, 1996

design machine group
University of Washington
Seattle WA USA 98195-5720
<http://depts.washington.edu/dmachine>

Demonstrating the Electronic Cocktail Napkin: a paper-like interface for early design

Mark D. Gross

Environmental Design
University of Colorado
Boulder CO 80309-0314
+1 303 492 6916
mdg@cs.colorado.edu

Ellen Yi-Luen Do

College of Architecture
Georgia Institute of Technology
Atlanta GA 30332-0155
+1 303 492 2807
ellendo@cc.gatech.edu

ABSTRACT

We demonstrate the Electronic Cocktail Napkin, a pen based interface for conceptual design. The project goal is to support design by recognizing, interpreting, and managing drawings, and to serve as an interface for knowledge-based critiquing, simulation, and information retrieval. We demonstrate the Napkin's facilities for end-user programmable recognition and interpretation, drawing management, and multi-user collaboration. We show applications of the Napkin: (1) indexing visual databases and (2) a front end to a local area network design program.

Keywords Pen based systems, design environments, constraint-based graphics.

SUMMARY

The Electronic Cocktail Napkin* is a pen based, multi-user environment for conceptual design. Its goal is to support freehand drawing and thinking of the sort that designers often do on the back of an envelope or a cocktail napkin. Free hand drawing offers a more direct means to explore and record decisions than structured mouse-menu interaction and it enables designers to make ambiguous and imprecise representations. Beyond the act of drawing, the Cocktail Napkin aims to support intelligent design critiquing, retrieval of relevant items from a case library, and simulation. Therefore, the Napkin attempts to recognize and interpret the elements and configurations of drawings. Figure 1 shows the Cocktail Napkin drawing board and its associated sketchbook. The Napkin supports:

- trainable recognition of multi-stroke glyphs
- end-user programming of graphical rewrite rules
- use of context to resolve ambiguity
- constraint inference and maintenance
- graphical search with user-tunable similarity measures
- semi-transparent trace overlays
- collaborative drawing



Figure 1a,b. Napkin Drawing Board and Sketchbook.

RECOGNITION

The Electronic Cocktail Napkin uses a simple, trainable-on-the-fly recognition scheme to identify multistroke glyphs drawn using a Wacom digitizing tablet, mouse, or Newton PDA. Similar to early recognition efforts like Teitelman's [8] our scheme looks at pen path through a 3x3 grid, number of strokes and corners, aspect ratio and size. Unlike others in current use (e.g. [2], [9]) our recognition scheme handles multiple stroke glyphs. We delay recognition of ambiguous glyphs and use context and role in larger configurations to later resolve ambiguity.

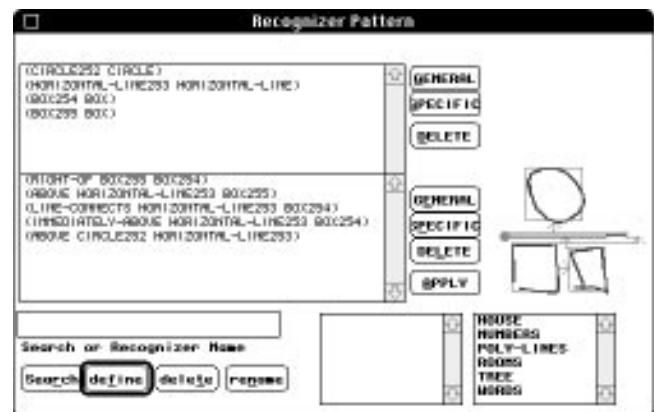


Figure 2. End users define recognizers by example.

* <http://wallstreet.colorado.edu/Napkin>

End users program recognizers for higher-level configurations by drawing examples. The Napkin identifies elements and spatial relations, and constructs a function to recognize instances of the configuration. The user may adjust the recognizer pattern by adding, deleting, and refining the constraints that define the recognizer (Figure 2). The user may provide several examples to construct the pattern either conjointly (any of the given patterns match)

or disjointly (only constraints common to all the examples define the recognizer). Users may also provide a picture to replace a recognized configuration of elements, in effect programming a graphical rewrite rule.

DRAWING ENVIRONMENT

The Cocktail Napkin provides semi-transparent trace overlays to simulate tracing paper designers use to copy and combine drawings. Users can set layers aside for later use (top, figure 1a). A sketchbook (figure 1b) is provided for keeping interesting drawings. The Napkin can search the sketchbook for drawings similar to a given query.

The Cocktail Napkin infers constraints from relations it finds in the drawings, and maintains them as the user edits the drawing. Unlike others e.g., [3] we use a domain-oriented approach to inferring constraints. When the Napkin recognizes a configuration, it asserts as constraints only those relations that define the configuration. For example, if it recognizes a graph, it asserts only connectivity constraints among the nodes and arcs, not spatial layout relations. The constraints are displayed visually in the drawing and the user can edit them.

RELATED WORK

Landay and Myers' work on SILK [5] shares our goals; however they don't emphasize end-user programming of visual parsing, and don't support the larger range of drawing management functions for a design environment. Ishii and Kobayashi's Clearboard project [1] shares our interest in collaborative drawing, but does not attempt recognition. Moran et al's work on shared whiteboards [7] likewise shares our aims, but capitalizes on structures specific to their domain. We have learned from discussions of visual parsing (e.g., Lakin's [4]) in formal and informal drawing.

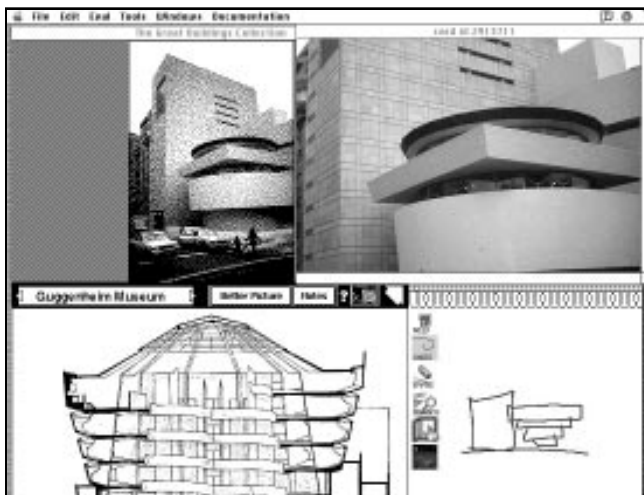


Figure 3. We used the Napkin to build diagram based indexing schemes for several databases.

APPLICATIONS

We are interested in using the Cocktail Napkin as an interface to other applications. We have used it to build diagram indexing schemes for several databases, including

The Great Buildings Collection CD-ROM [6], the Archie II case base of design evaluations [10], and Netscape. A user browses databases making diagrams to serve as bookmarks, then makes a similar diagram to retrieve an indexed item. We are using the Napkin to build a sketching interface to ProNet, a local area network simulation.

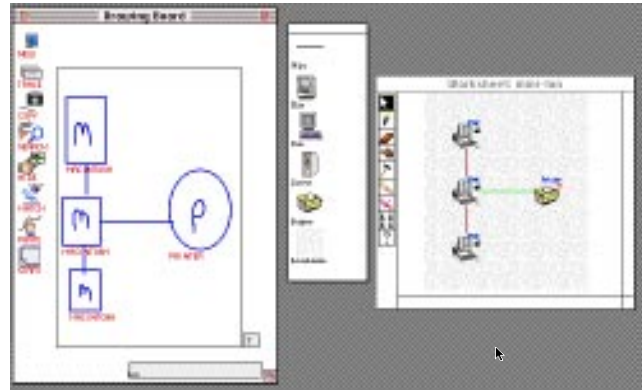


Figure 4. Cocktail Napkin as interface to a LAN tool.

ACKNOWLEDGMENTS

We appreciate support from NSF grant DMII-93-13186, and permission to reprint the screen in Figure 3, from K. Matthews of the Integration Lab at University of Oregon. Kyle Kuczun built the connection to the ProNet LAN design tool, Jim Sullivan's PhD project in Fischer's Human-Computer Communication group at Colorado.

REFERENCES

1. Ishii, H., Kobayashi, M. Clearboard: A seamless medium for shared drawing and conversation with eye contact. CHI '91. (Monterrey, CA, 1991), pp. 525-532.
2. Kimura, T.D., Sengupta, S., Apte, A. A Graphic Diagram Editor for Pen Computers. Software Concepts and Tools. (1994), pp. 82-95.
3. Kurlander, D. Graphical Editing by Example. InterCHI '93. (Amsterdam, 1993), pp. 529.
4. Lakin, F., Wambaugh, J., Leifer, L., Cannon, D., Steward, C. The electronic notebook. Visual Computer, 5,(1989) pp. 214-226.
5. Landay, J.A., Myers, B.A. Interactive Sketching for the Early Stages of Interface Design. CHI '95 (Denver, Colorado, 1995), ACM Press, pp. 43-50.
6. Matthews, K. *Great Buildings Collection*. Van Nostrand Reinhold, New York, 1994.
7. Moran, T.P., Chiu, P., Melle, W.v., Kurtenbach, G. Implicit Structures for Pen-Based Systems within a Freeform Interaction Paradigm. CHI '95 (Denver, Colorado, 1995), ACM Press, pp. 487-494.
8. Newman, W.M., Sproull, R.F. *Principles of Interactive Computer Graphics*. McGraw Hill, New York, 1973.
9. Rubine, D. Specifying Gestures by Example. Computer Graphics, 25,4, (1991) pp. 329-337.
10. Zimring, C., Do, E.Y.-L., Domeshek, E., Kolodner, J. Creating a case-based design aid for architecture. Proc. *Design Decision Support Systems II*. (Vaals, Netherlands, 1994).