The Right Tool at the Right Time -- drawing as an interface to knowledge based design aids

Ellen Yi-Luen Do

Proceedings, 1996 National Conference, Association for Computer Aided Design in Architecture 1996 ,(ACADIA 96), Filiz Ozel and Patricia McIntosh (eds.), University of Arizona, Tucson. p. 191-199.

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

The Right Tool at the Right Time
-- drawing as an interface to knowledge based design aids

Ellen Yi-Luen Do
College of Architecture
Georgia Institute of Technology
and
Sundance Laboratory for Computing in Design and Planning
University of Colorado at Boulder

ABSTRACT

Designers use different symbols and diagrams in their drawings to explore alternatives and to communicate with each other. Therefore, a useful design environment should attempt to infer the designer's intentions from the drawing and, based on this inference, suggest appropriate computational tools for the task at hand. For example, a layout bubble diagram might activate design cases with similar configurations. Scribbles of view lines on a floor plan might bring up a spatial analysis tool. This research aims to develop an integrated digital sketching environment to support early design activities. The paper proposes RT^2, an intelligent sketch environment that provides the designers with the right tools at the right time.

1. INTRODUCTION -- WHY THE RIGHT TOOLS AT THE RIGHT TIME?

Imagine a workman doing household jobs. He has a tool box filled with various tools, hammer, screw driver, pliers and a drill. All these tools are useful, but only if they are applied to the right job at the right time. For example, no matter how useful a sanding kit is, it will not drive screws or fasten nuts and bolts. A hammer will be useful for striking but not for drilling holes. An experienced assistant watching the job can hand the appropriate tools to the workman at the right time.

This paper proposes RT^2, a computer based freehand sketching environment for design that delivers the right tools at the right time. Rather than asking the designer to find and select tools for specific design tasks, we explore the idea of automatically invoking various computational tools based on the designer’s drawing. We are developing RT^2 for two reasons. First, to be really useful, knowledge based design tools must be available at the right time. Design involves many different activities: exploring ideas, finding references, retrieving information, manipulating form, evaluating performance, and making functional analyses. Many knowledge based design tools have been built to support design activities; however designers do not use them because these systems require designers to stop designing and translate their intentions into obscure system commands.

Second, examining design drawings can provide a way to decide the right time to activate design tools. Freehand sketching is traditionally the main medium in design; in addition to the specific design configuration it can convey information about design intentions and context of the task at hand. Current Computer Aided Design construction tools, though beneficial for design drafting, are cumbersome to use in early design. Their structured command and text oriented interfaces tend to obstruct the flow of design. Most designers would prefer to sketch their design ideas on paper with a pen or pencil (Do 1993b). Therefore, freehand drawing is an obvious interface to access design tools.

The rest of the paper is organized as follows: In Section 2 we describe a scenario suggesting how different activities and design tools might be used in a design process. Section 3 presents our observation of design activities. Section 4 argues for the importance of drawing in design. Section 5 discusses the implementation of RT^2, an intelligent sketching environment that activates different design tools at the right time based on design drawing.

2. A SCENARIO -- DESIGNING WITH DIFFERENT TASKS AND TOOLS

Imagine a novice designer Lynn designing a public building. Her assistant Arty (RT) sits quietly watching and ready to help. Lynn starts the design from scratch by making a bubble diagram to think about spatial arrangements. From Lynn's diagram (figure 1a), Arty recognizes that she is working on the spatial relations of lobby and other spaces, and calls up a story from a case base, Archie, (Kolodner 1991; Zimring and others 1994) that discusses lobby arrangements (figure 1b).

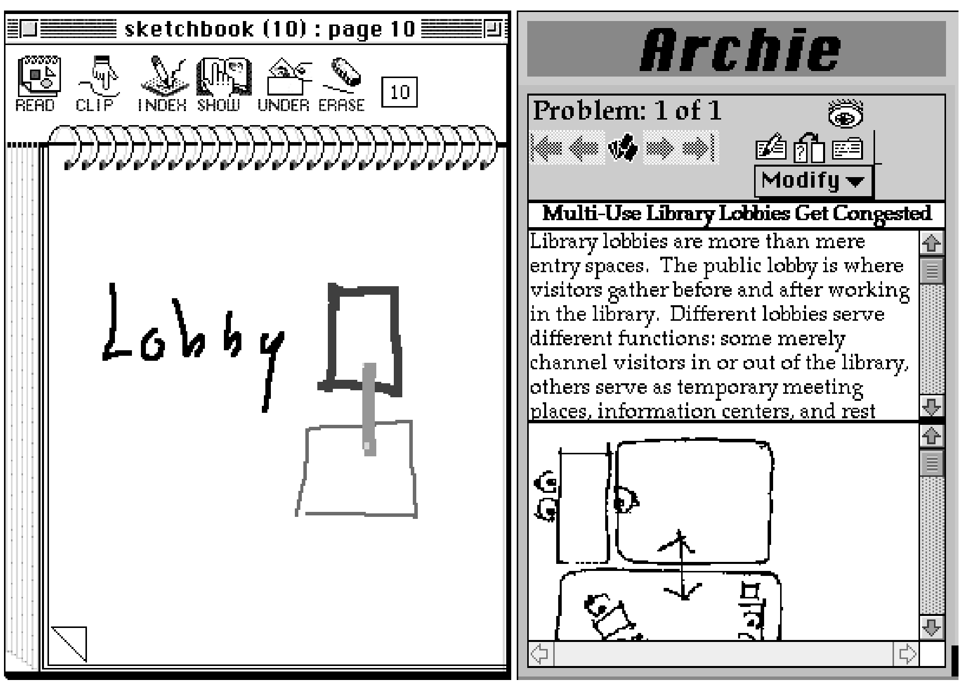

Figure 1. Designer's diagram retrieves a relevant story from a case library, Archie. (a) Bubble diagram of spatial arrangements in a sketchbook; (b) Archie case story of a library lobby with similar configuration.

Lynn finds the case provided by Arty useful and incorporates the arrangement into her design (reminded by the Archie case story that lobby can serve as information center, Lynn adds another space bubble into the diagram). She continues to draw. When she stops, Arty hands Lynn a hard line version of her sketches (figure 2). Arty then uses the cleaned up drawing to check the total square footage against the design brief requirements. Meanwhile, Lynn starts to draw diagrams to analyze the sense of visual enclosure in her design from different view points. Arty calls up a viewshed simulation program, Isovist (Do 1993a; 1995a) to help her with analysis (figure 3).

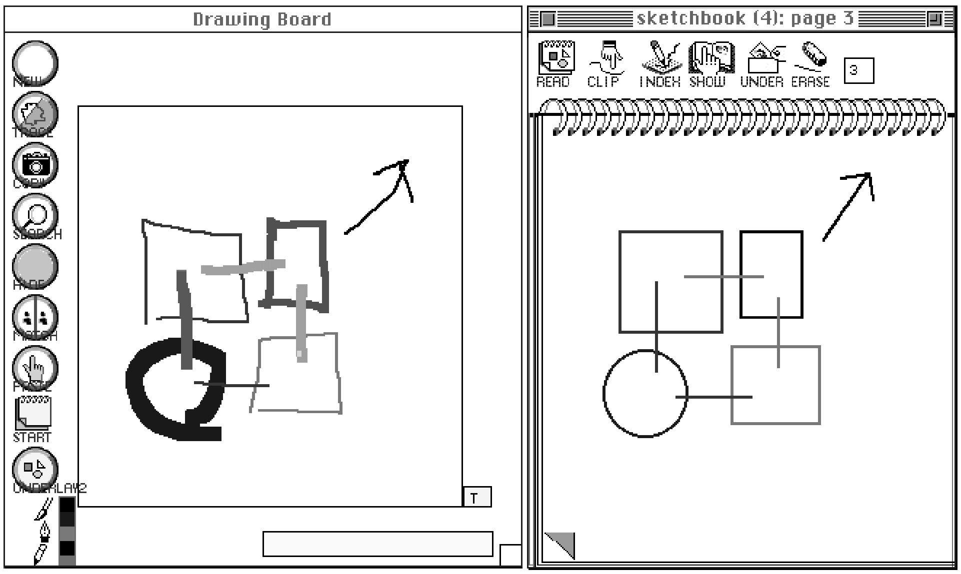

Figure 2. Designer's drawing can be cleaned up and rectified if desired. (a) Sketchy drawing of spatial arrangements; (b) Rectified geometric shapes of the sketchy drawing.

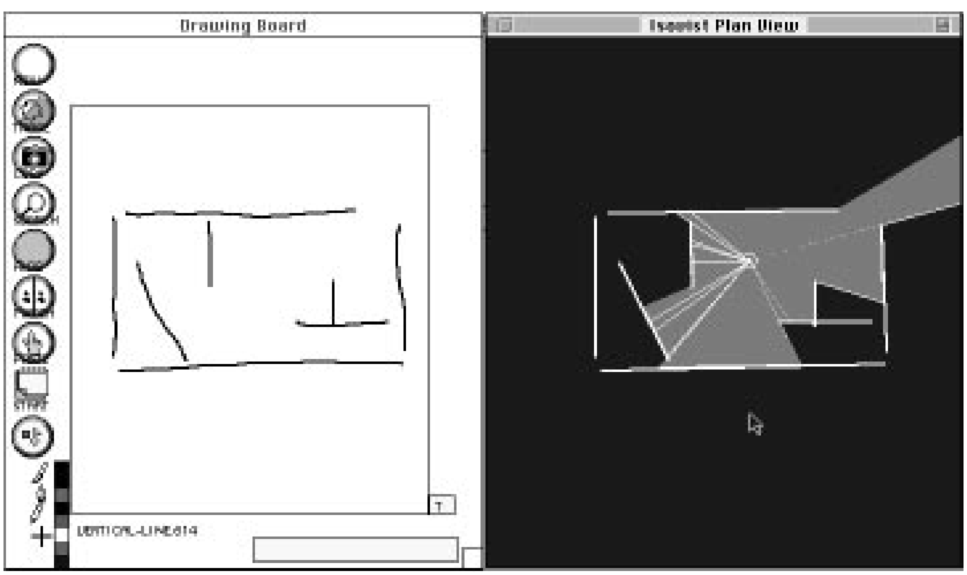

Figure 3. Designer's drawing of visual enclosure brings up a viewshed simulation program, Isovist. (a) Diagram to explore visual access; (b) Isovist viewshed analysis tool.

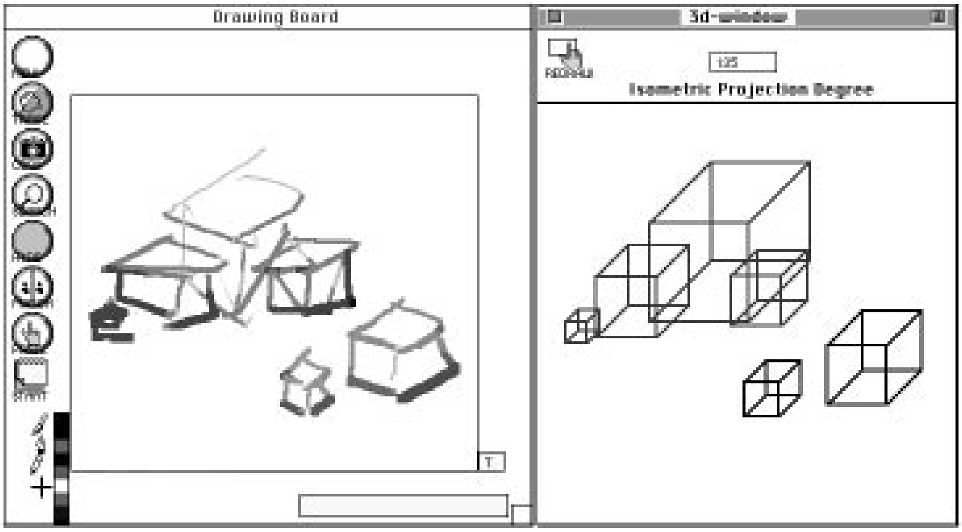

Figure 4. Designer's sketches can be transformed into a 3D CAD model. (a) Sketches of massing study; (b) Translated 3D model of the sketches that allows changes of isometric projection angles.

Satisfied with her basic spatial arrangement, Lynn decides to work on form. She sketches several cubes to examine the massing of her design. Based on Lynn's sketch, Arty quickly builds up three dimensional computer models (figure 4) for Lynn to explore viewing perspectives and perhaps create a walk through animation. To make her building look more monumental (!), Lynn draws a facade composed of a triangle pediment and colonnade. Arty recognizes the drawing as a temple and finds for Lynn slides and a QuickTime animation of the Parthenon from The Great Buildings Collection CD ROM (Matthews 1994) (figure 5).

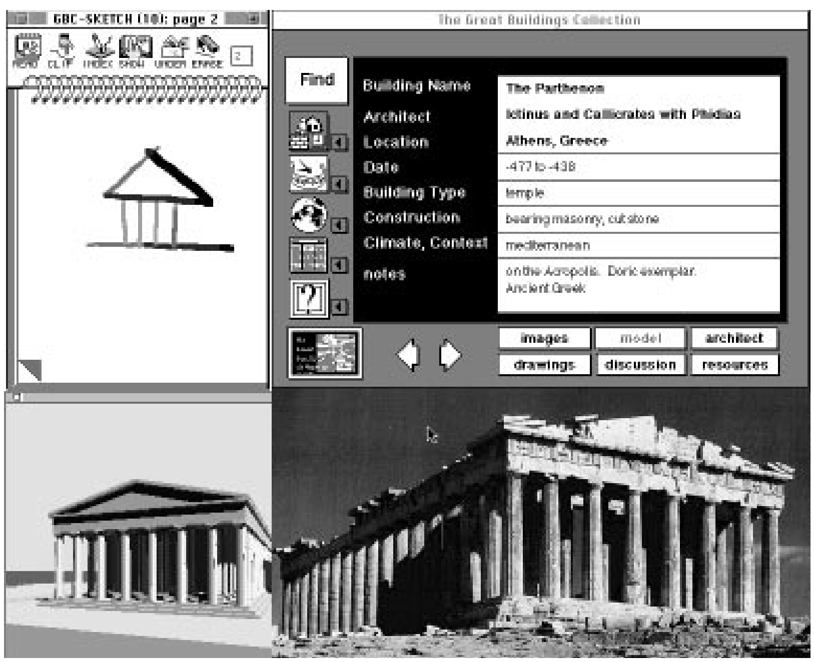

Figure 5. Designer's diagram retrieves information from a multi-media database. (a) Diagram of a triangle pediment and colonnade; (b) Textual information, picture and QuickTime animation of the Parthenon from the Great Buildings Collection (Matthews 1994).

The story can go on. Though different designers may approach design differently, all design processes have two things in common. First, our designer and her assistant engage in various activities during the design process: finding references, functional analysis, modeling, and visualization. Tools such as design case libraries, drafting and modeling, viewshed analysis support these activities. Second, as she works, the designer makes various kinds of drawings. Her computer based assistant Arty recognizes what she is doing by looking at these drawings and then suggests an appropriate tool. In the following sections we discuss design activities and drawing.

3. ACTIVITIES IN A DESIGN PROCESS

Design involves many different activities. However, most current Computer Aided Design tools assume a single task – tool approach. We propose that a useful design environment should provide a variety of tools to support various design activities. Design studies researchers identify different phases of design during which different kinds of activities take place. For example, Sanoff describes design as transforming information through stages of analysis, synthesis and evaluation (Sanoff 1977). Markus describes design as decision making through problem identification, finding relationships between issues, model buildings and optimization (Markus 1969). This notion of design involving many different activities is shared among researchers and designers.

In interviews with four professional architects (Do and Zacherl 1993), we found that though personal styles vary, designers use diagrams and drawings to develop ideas, employ different tools, conduct case studies, and perform various analyses.

Based on design studies (Akin 1984) and interviews (Do 1993b), we observe that the early design process involves three major activities, which we call Organization, Ideation and Fabrication. First, Organization deals with abstract manipulation of spatial requirements as derived from the program brief or problem analysis. During Organization, designers use cut-out paper cards or bubble diagrams to explore functional arrangements. They trace over bubbles or move them around to find design alternatives. Designers explore adjacency, circulation, and approximate construction costs. The drawings of this activity deal mostly with “function.”

In Ideation designers sketch to express and explore formal or analogical thoughts. For example, in the interviews designers mentioned that at certain times they would find ideas and visual references to incorporate into their design. They are open to ideas that lead to physical forms, and concerned with form configurations and shape manipulations. In Ideation, they use sketches to explore and understand formal aspects. Drawings in this activity deal mostly with form.

The third activity is Fabrication. When designers refine design solutions they make more structured drawings. For example, grids are used to guide column or wall locations in the floor plan and isometric sketches are used to depict three dimensional massing. In this activity, design drawings are not as diagrammatic or sketchy as in the first two phases, and can easily be hard lined. In Fabrication activities, designers make harder line drawings in this stage to help prepare for constructing the future working drawing.

4. IMPORTANCE OF DRAWING IN DESIGN

In early design, designers draw diagrams and sketches to explore ideas and solutions. Drawing on paper involves recording ideas, recognizing functions and meaning from the drawings, and finding and adapting new forms into design. Several books have described drawing as a means for design development. For example, *Graphic Thinking* (Laseau 1980) guides the designer to make drawings for working out problems, and communicating with others. *Envisioning Architecture* (Fraser and Henmi 1994) looks at how techniques used to make different drawing types influence the making of architecture. *Why Architects Draw* (Robbins 1994) examines the work of well-known professional architects, focusing on the role of drawings in architectural practice. *Architectural Study Drawings* (Herbert 1993) examines the graphical media and design processes of six practicing architects. It argues that drawings are more than just a convenient strategy for solving design problems and that in fact they are “the designer’s principal means of thinking” (p. 1).

4.1. Drawing conventions -- results from empirical studies

We have conducted several empirical studies to explore: 1) drawing to illustrate concepts and impressions of buildings from memory, 2) drawing to design and solve design problems, and 3) drawing to illustrate textual concepts and writing for design drawings. From the studies we found a number of drawing conventions that designers shared.

The first study asked design students to draw their impression of famous buildings, illustrate design concepts, and propose solutions. The second study asked designers to design with given concepts, and to illustrate a given artifact. The third study focused on the feasibility of using diagrams to access a case based design aid (Archie). We designed the experiment to see whether designers would employ conventional drawing techniques in diagramming architectural design problems. The tasks were making diagrams from stories, writing stories from given diagrams, pairing drawing and texts and commenting on existing Archie diagram-story pairs.

We found from the above three empirical studies that designers share drawing conventions. First, designers only use a small set of symbols in their drawing in conventional and consistent ways (figure 2 shows the lexicon of symbols they used), hence we can train the computer to recognize these symbols. Second, designers use different view preferences for different concepts (e.g., plans or sections) to illustrate different sorts of problems (e.g., spatial arrangement versus getting light into a building), so we can use view preference to identify context and design tasks. For example, a stick person symbol indicates a section or an elevation drawing while a row of circles represent columns in a plan view. Most importantly, the studies show that designers can read and understand each others' drawings (Do 1995b). Therefore, drawing can be used to support the task of finding the right tools at the right time.

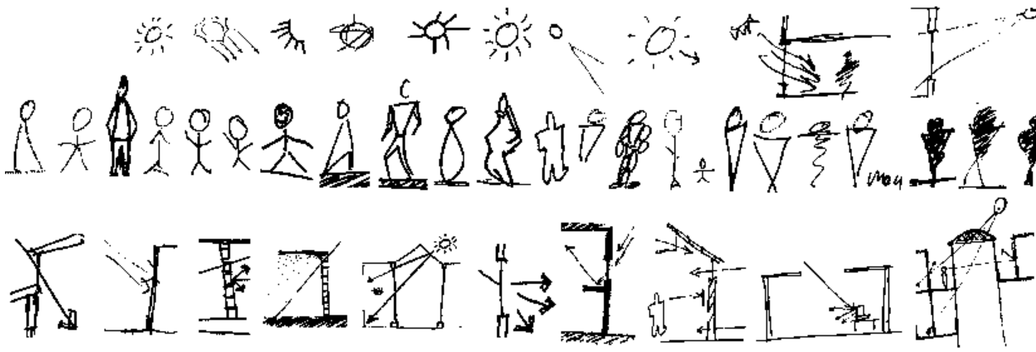

Figure 6. Designers used conventional symbols for architectural concepts in diagrams.

5. RT^2 -- THE RIGHT TOOLS AT THE RIGHT TIME

5.1. Computational support for design

Since the early sixties, researchers have been investigating using computers to help design. Evaluation tools and expert systems (e.g., (Maher 1985; Rittel and Kunz 1970)) have been developed to help design decision making by providing design rationale and guidelines. The approach of case libraries (e.g., (Kolodner 1991; Oxman 1993; Zimring and others 1994)) proposes using past experience (design precedents) to help designers avoid previous mistakes, and to adapt old design cases into new situations. Shape grammar systems (e.g., (Flemming 1987; Koning and Eizenberg 1981; Stiny and Mitchell 1978)) compute geometric information to guide generation and transformation of design artifacts such as floor plans and facades. Geometric modeling and drafting programs (e.g., Form-Z, AutoCAD) help speed the production of working drawings.

Computer Aided Design (CAD) programs only partially support the design process. They fail in one important respect: they do not provide an environment suited to design tasks. First, designers draw to design. A freehand sketching environment is more appropriate for design than menus, buttons and dialogs. Second, no matter how full of knowledge these systems are, designers will not use these tools if they are not readily available when they are needed. Owning a box full of useful tools does not guarantee a job well done. To use current knowledge based design assistants, designers need to spend time choosing tools and train themselves to be familiar with the systems' commands. Stopping designing to choose and specify commands can obstruct the design flow so designers are understandably reluctant to use these computer systems (Do and others 1994). Therefore, we propose a drawing environment to deliver appropriate assistance at the right time to help designers make better use of available knowledge based design tools.

5.2. Activation of design tools from drawing

We are developing RT^2 to serve as an interface to a variety of knowledge based design tools. Rather than asking the designer to select and apply tools to perform specific tasks (e.g., 'now let's look at design precedents') I explore the idea of automatically invoking various tools based on the designer's drawing. The front end of RT^2 is a sketching environment that accesses information and design tools. RT^2 will try to guess from the drawing what the designer is working on, and then deliver knowledge based support appropriate to the task at hand.

The drawings designers make reflect the task they are working on and can serve as an indicator of intention. We can use drawing to identify design context and call up the right tools at the right time. Therefore, we are building RT^2 to recognize drawing context. For example, in Organization activities, architects often draw a bubble diagram to explore functional arrangement of spaces; they draw sight lines and viewsheds when working on visual analyses. Therefore, when the designer draws a bubble diagram, RT^2 will retrieve a design case with a similar spatial configuration. When the designer draws view lines in a floor plan, RT^2 will bring up a spatial analysis program. When the designer is engaged in Ideation activities, RT^2 will find visual references from building slides or natural artifacts based on similar shapes or concepts. When the designer is engaging in Fabrication activities, RT^2 will activate construction modules for automatic drafting and model building. Each of these programs will operate on the sketch diagrams, and transform the sketch into different representations. For example, a bubble diagram can be rendered as 'clean' geometric shapes; a sketchy cube can be transformed into a three dimensional CAD model. These drafting tools might not be appropriate for early design stages (during which freehand sketching is preferable) but they would be useful for the later Fabrication stage.

5.3. Integrating design tools into sketching environment

Previous works (Do and Gross 1995; 1996; Gross and Do 1995; Gross and others 1994) have used the Electronic Cocktail Napkin (Gross 1994; 1996) to prototype an integrated drawing environment. In this work, a Sketchbook module serves as an interface to various knowledge based design tools: Archie, The Great Buildings Collection, HyperCard, FileMaker Pro documents and the World Wide Web pages. Some screen snapshots of these programs appear in figures above (1, 2, 5). The Sketchbook module enables designers to keep their personal sketches, making visual bookmarks and using diagrams to query several reference databases. In this drawing environment, the connections to various design tools are direct, one-to-one, and require the designer to take explicit action to invoke them. We are developing RT^2 to extend the system to automatically activate appropriate tools when it can identify a design context from drawing.

We are implementing the RT^2 architecture in Macintosh Common Lisp as a supervisory module in our drawing environment. Our current system architecture uses a checklist and production rule to select design tools based on context. We are building a data structure that contains tables of 'contexts' and 'intentions'. A 'context' data structure is a list of recognized graphic symbols and a list of 'intentions'. The value of the slots in the 'intention' are names of design tools. RT^2 infers a design context by comparing the symbols in a drawing with the symbols in the 'context' table entries. For example, a 'sectional drawing' context contains stick person, vertical wall lines, horizontal ground lines and slanted roof lines. The value of 'intention' slots in a 'sectional drawing' includes 'structure,' 'scale' or 'lighting analysis'. If a drawing contains instances of all the four symbols, 'sectional drawing' becomes a strong candidate for the 'context'. A 'bubble diagram' context contains lists of closed shapes symbols such as rectangles, ovals and lines. The values of the ‘intention’ slots in ‘bubble diagram’ are ‘functional configuration,’ ‘viewshed analysis’ or ‘electrical system design’. In other words, when the designer is drawing bubble diagrams, RT^2 can infer that she is working on ‘functional configuration,’ ‘viewshed analysis’ or ‘electrical system design’.

These contexts and intentions are used to call up various design tools. For example, when RT^2 finds an arrow penetrating a wall line in a ‘sectional drawing,’ it checks the table and finds that the closest match is an ‘intention’ of ‘day lighting’. A drawing that has an arrow between bubbles in a plan view returns a ‘circulation’ intention. Likewise, in a structure analysis diagram, an arrow indicates the ‘direction of force,’ while in an electrical wiring diagram, the arrow represents an ‘electricity plug’. For example, if a drawing contains sight lines, wall lines and a plan person symbol, the ‘context’ is plan view, the ‘intention’ is visual perception, and so RT^2 will call up the Isovist visual analysis program. If no particular context can be identified, an arrow simply indicates annotation, and it brings up a PostIt window for making notes. Currently, the activation sequence of the candidate design tools depends on its position in the list. We plan to improve the interface to allow designers to customize their own priority preferences for tool activation.

This paper presented the idea of RT^2, an integrated freehand sketching environment that aims to deliver the right tools at the right time. We have already connected various design tools with the drawing environment. We are incorporating additional design tools into the system, automating tool activation, and working on improving the context detection mechanism. We are conducting further empirical studies to understand about design drawing conventions, their associated tasks and relevant design tools. Detecting context from drawing is proving to be an interesting and complex problem.

6. ACKNOWLEDGMENTS

A special thanks to Mark Gross for his remarks, support and encouragement. Gratitude also goes to the Archie Group at Georgia Tech (Craig Zimring, Janet Kolodner and Eric Domeshek) for the use of Archie program, Kevin Matthews and the Design Integration Laboratory at the University of Oregon for permission to reprint the screen snapshot of The Great Buildings Collection, and National Science Foundation grant DMII 93-13186.

7. REFERENCES

Akin, Omer. 1984. An exploration of the design process. In *Developments in Design Methodology*, edited by Nigel Cross. Chichester: John Wiley & Sons.

Do, Ellen Yi-Luen. 1993a. *Imaging the Concepts of Isovist in Design Process*. Georgia Institute of Technology. Working Paper.

Do, Ellen Yi-Luen. 1993b. *Stages of Conceptual Diagramming in the Design Process*. Georgia Institute of Technology, College of Architecture. Working Paper.

Do, Ellen Yi-Luen. 1995a. *Visual Analysis through Isovist*. Georgia Institute of Technology. Working Paper.

Do, Ellen Yi-Luen. 1995b. What's in a diagram that a computer should understand. In *The Global Design Studio, Proceedings of the Sixth International Conference on Computer Aided Architectural Design Futures*, edited by Milton Tan and Robert Teh. Singapore: National University of Singapore.

Do, Ellen Yi-Luen, and Mark D. Gross. 1995. Drawing Analogies: Finding Visual References by Sketching. In ACADIA 95, Computing in Design, enabling, capturing and sharing ideas, at Seattle. Association of Computer Aided Design In Architecture, 35-52.

Do, Ellen Yi-Luen, and Mark D. Gross. 1996. Reasoning about Cases with Diagrams. In Third Congress on Design Computing, at Anaheim. edited by Jorge Vanegas, American Society of Civil Engineers, to appear.

Do, Ellen Yi-Luen, Siu-Wing Daniel Or, David M. Carson, Chang-Shin Chang, and Wesley C. Hacker. 1994. *Usability Study of A Case-based Design Aid Archie*. Georgia Institute of Technology. Technical Report GIT-CS-6752-94.

Do, Ellen Yi-Luen, and Anna Zacherl. 1993. *Requirements for a Bubble Diagram Editor for Use by Architects during Conceptual Design*. Georgia Institute of Technology, College of Computing. Archie Project Technical Report.

Flemming, U. 1987. More than the sum of parts: the grammar of Queen Anne houses. *Environment and Planning B Planning and Design* 14:323-350.

Fraser, Iain, and Rod Henmi. 1994. Envisioning *Architecture - an analysis of drawing*. New York: Van Nostrand Reinhold.

Gross, Mark D. 1994. The Fat Pencil, the Cocktail Napkin, and the Slide Library. In ACADIA '94, at St. Louis, MO. edited by M. Fraser and A. Harfmann.

Gross, Mark D. 1996. The Electronic Cocktail Napkin - working with diagrams. *Design Studies* 17 (1):53-69.

Gross, Mark D., and Ellen Yi-Luen Do. 1995. Diagram Query and Image Retrieval in Design. In 2nd International Conference on Image Processing, at Washington, D. C. IEEE Computer Society Press, 308-311.

Gross, Mark D., Craig Zimring, and Ellen Yi-Luen Do. 1994. Using Diagrams to Access a Case Base of Architectural Designs. In *Artificial Intelligence in Design '94*, edited by John Gero. Lausanne: Kluwer Academic.

Herbert, Daniel M. 1993. Architectural *Study Drawings*. New York: Van Nostrand Reinhold.

Kolodner, Janet L. 1991. Improving human decision-making through case-based decision aiding. *AI Magazine* 12 (2):52-68.

Koning, H, and J Eizenberg. 1981. The language of the prairie: Frank Lloyd Wright's pairie houses. *Environment and Planning B* 8:295-323.

Laseau, Paul. 1980. *Graphic Thinking for Architects and Designers*. New York: Van Nostrand Reinhold.

Maher, May Lou. 1985. Hi-Rise and beyond: directions for expert systems in design. In *Computer Aided Design*: Butterworth & Co.

Markus, Thomas A. 1969. The role of building performance measurement and appraisal in design method. In *Design Methods in Architecture*, edited by Geoffrey Broadbent and Anthony Ward. New York: George Wittenborn.

Matthews, Kevin. 1994. *The Great Buildings Collection*. New York: Van Nostrand Reinhold.

Oxman, R. 1993. PRECEDENTS: Memory structure in design case libraries. In *CAAD Futures '93*: Elsevier Science Publishers.

Rittel, W., and W. Kunz. 1970. *Issues as elements of information systems*. Center for Planning & Development Research, University of California, Berkeley. Working Paper 131.

Sanoff, Henry. 1977. Methods *of Architectural Programming*. Stroudsburg, Pennsylvania: Dowden Hutchington & Ross, Inc.

Stiny, George, and William J Mitchell. 1978. The Palladian Grammar. *Environment and Planning B* 5:189-198.

Zimring, Craig, Ellen Yi-Luen Do, Eric Domeshek, and Janet Kolodner. 1994. Using post-occupancy evaluation to aid reflection in conceptual design: Creating a case-based design aid for architecture. In*Design Decision Support System*, edited by H. Dimitripoulos. Vaals, Netherlands.