THE CONCEPTUAL APPROACH TO CAD EDUCATION

Doris Smith Shaw
U.S. Army Construction Engineering Research Laboratory
Champaign, Illinois

ABSTRACT

Recent research at the Corps of Engineers Construction Engineering Research Laboratory (CERL) investigated embedded computer-based instruction for AutoCAD. The results of this study, which are the focus of this paper, indicated that the only factor which correlated with success in completing the final test was previous experience with another CAD system. Those who knew another CAD system had higher scores and required less than half the time to complete the lessons. Presumably their conceptual knowledge about CAD transferred to the new software environment, even though the Corps' study showed that they were initially biased against learning the new system. Such biased attitudes have been observed when users are asked to learn a second similar software of any kind.

Architects who are deeply involved in computer-aided design have stated that one must learn to program the computer to build the conceptual framework for the creative process. We at CERL agree that an understanding of underlying graphics concepts is essential to the designer. Our research shows that giving students the freedom to explore an existing software program can result in the development of conceptual knowledge. Interviews also reveal that students can invent ways to meet individual objectives when "guided discovery" learning is encouraged.

COMPUTER ASSISTED DESIGN PREREQUISITES

The aforementioned study included a questionnaire that measured attitudes toward computer-aided design. Architects and design engineers were in unanimous agreement with the statement, "Computers are helpful in the design process." However, there is wide disagreement about how that help transpires. Does the computer serve as a human extension in much the same capacity as paper and pencil, or does it add new capabilities that have been previously limited to a few creative geniuses? Either way, designing with a computer poses questions about the required level of expertise in underlying graphics concepts and how we should teach that expertise.

The ACADIA '87 Workshop focused upon "Integrating Computers into the Architectural Curriculum." There was a convincing theme throughout the talks which indicated that computers could certainly contribute to the design process. However, a wide range of roles for the computer were advanced by the various presenters. The contrast was especially striking in the session "Drawing from the Past: Looking to the Future." One paper emphasized that historically recognized "stages of design" must provide the basis for the development of CAD systems and human interaction with them (Herbert, 1987). The computer was viewed as a transparent extension of the designer in the most fundamental sense. The next presentation demonstrated that innovation in design came about as a function of programming the computer to generate forms and transformations heretofore too complex to occur to any but, perhaps, the most creative designers (Yessios, 1987). The computer here assumed the character of a consciously invoked tool to enhance the imaginative powers of the designer. In one view, the technical capabilities of the
computer were secondary to the human purpose and, in a second view, the computer's unique powers contributed to the human inspiration.

To assess the perceived utility of computers by Architectural practitioners the Corps' study included a final questionnaire. On that study, 100% agreed with the statement, "Computers are helpful in the design process." Eighty three percent agreed that, "The computer helps me consider different possibilities in my design" and 75% concurred that, "The computer allowed me to express my design ideas as well as I can manually" (Shaw & Gellish, 1988). Of interest here are the possibilities suggested by those who disagreed with the last two statements, but who agreed with the first. CAD educators are again faced with the fact that CAD users have many different impressions concerning it's scope of applicability and utility in design. The variety of perceptions relate in part to an individual's understanding of the CAD environment.

It is a common opinion that architects need to know how to program a computer in order to use computers in design. Programming is a curriculum requirement in many architecture schools. A recent textbook for programming computer graphics (Mitchell, Liggett, & Kvan, 1987) represents the program as an art form that it to drawing as a score is to music. Certainly learning to program teaches certain computer principles. Our study explored another route to understanding underlying graphics concepts. The important questions that must be answered by those who teach designers to use CAD are:

1. What computer graphic concepts are prerequisite to the design process?
2. What methods may be used to teach the necessary concepts?
3. How can we allow for the individuality that designers require?

EMBEDDED INSTRUCTION

Embedded instruction, or training which takes place within a computer program, was designed in order to avoid the problems which occur when a student learns in one environment and then attempts to transfer his knowledge to another environment. Experimentation with a CAD system is educational, and embedded instruction can allow for experimentation. The instruction uses the CAD system to display the lesson screens and encourages the student to experiment with the system's features. In the embedded lessons used in this study, there were no artificial limits imposed upon experimentation. The lessons were sequenced according to difficulty, offering customized menus with only a few menu options at first. In the final lessons the student had access to the full menu with the opportunity to explore the complete program. The student was encouraged to compare different methods for solving problems in order to develop a conceptual base rather than memorize a collection of procedural information.

Effective computerized instruction is based upon a knowledge of three areas: the subject area itself, the educational media, and the stages in the development of expertise in the learner. Since computer-based instruction attempts to reduce the need for human intervention in training, thorough analysis of the learning process is necessary. The usual accommodation of teacher to student must be predicted and simulated as closely as possible.

The "subject area" of this study involved not only procedural knowledge of a computer program but the concepts involved with designing on a computer. Design has traditionally been thought of as a product of the imagination, not subject to computerization. To test this hypothesis a case-study approach was used to study patterns in thinking as architects and engineers learned to use AutoCAD (Shaw, 1986). AutoCAD was chosen because of its open design which allowed the possibility of embedding instruction and also because most of the Corps offices were using AutoCAD for some of their work. A pilot study was conducted to aid in designing the embedded AutoCAD lessons. That study resulted in the following guidelines:
1. The course content would be represented better by a concept specification hierarchy than by a traditional task analysis of skills (Birenbaum, & Shaw, 1985). A concept hierarchy is more broad-based and can therefore address different procedural approaches.

2. The subject would be encouraged to think of the computer as an extension of his own design processes. There was little acceptance by designers of changing design practices to comply with constraints set by the computer.

3. It would be necessary to allow the designer time and space for individualized experimentation. The pilot study showed wide variations in such needs.

4. In the interest of efficiency, however, some structure would be necessary in the presentation. The first lessons were aimed at developing a general familiarity, the next lessons with applications showing the power of the software, and the remaining two with more detailed procedural knowledge.

On the basis of the pilot study observations, conceptual difficulties were predicted and six lessons were developed. The lessons were programmed in AUTOLISP and made use of AutoCAD graphics to create the screen displays. The test subjects were monitored by the lesson program and provided with feedback if they committed expected errors or failed to practice a command. Extra help was offered from some lesson screens when the student typed the letter X. This help was context-sensitive; the same keypress resulted in different help for different lesson screens. The commands CONTINUE, BACK, and INDEX were added to the AutoCAD screen menu. Typing the letter B at the Command prompt brought up a blank screen for practice. Examples from the lesson screens are shown in Figures 1, 2, 3, and 4.

**FIGURE 1. Context Sensitive Help.**

If the student does not fully understand base points and displacement points, he may type X to see context sensitive help.

77
The base point or displacement is a point that refers to the object to be moved or copied to a new location.

The second displacement point is a corresponding point on the object in its new or copied position.

Select CONTINUE to return to the lesson.

FIGURE 2. Help Example.
When help is requested, more detail is offered. Often the help includes graphics or animation.

If the copy or moved object disappeared from your screen, it is likely that you pressed the Enter or Return key rather than the pick button for the second point.

Be sure that the first point that you pick has reference to the object that you want to move or copy. Then select a corresponding point in the new position.

Select CONTINUE to return to the lesson.

FIGURE 3. Error Feedback.
If a common error is made (pressing the Return key rather than the pick button) this message offers further explanation.
This display is seen if the student fails to try the exercise when practice is provided.

The directions to practice were consistently printed in red throughout the lessons. If the student did not practice the command, he received a reminder (Figure 4), but he could go ahead without practicing. He was permitted to experiment with any command that he found on the screen menu, if his curiosity inspired him to do so. The menu was abbreviated for the first two lessons, but the entire AutoCAD menu was available from lesson three through six. The AutoCAD on-line help was introduced in the first lesson and the student was encouraged to call upon it any time he wished. Many students spent much of their time exploring and experimenting with various AutoCAD commands.

RESEARCH FINDINGS

Sample Demographics

The test was conducted with a population ranging from graduate students to practicing professionals. Eighty-four subjects participated in some part of the test. Seventy-nine completed a final drawing test and sixty-four answered the survey questionnaire. Nineteen percent of the test subjects were female. Fifteen percent of the sample were architects, forty-eight percent engineers, and thirty-six percent draftsmen and technicians. With respect to computer literacy, engineers had the most previous experience in the sample group, closely followed by draftsmen and technicians.

The architects had the least computer experience.

The average age of the test subjects was thirty-seven. There was a noticeable difference in average age for those with little computer experience and those with extensive experience on computers. The average age for subjects with more than one year of computer experience was thirty-four while the age of those with less than two months of experience was forty-one.
The average time spent on the six lessons was 730 minutes or about 12 hours. The range, however, spread from 46 minutes to 3300 with a standard deviation of 673. Forty-six subjects kept a record of their time. The graph shows that 82% required less than the average time (Figure 5).

There was no significant correlation (correlation coefficient, .33) between time spent on the lessons and the age of the subject (Figure 6). It is interesting to note though, that all the peaks representing extremely long periods of time in the lessons occurred in the over age 40 group. We suspected that those subjects who required extremely long periods of time to complete the lessons might be poor candidates for computer-based instruction, but that did not prove to be the case. Individual interviews revealed that many of them explored the program extensively and attained a high level of expertise in the system. In this professional test sample, the time on task for CAD beginners was not a predictor of success. Apparently the time variation is related to individual traits such as learning style, personality, or motivation. Future study of these factors may lead to better adaptation of CAI to learner's needs.
FIGURE 6. Plot of Subject Age and Time Spent on the Lessons.

The most notable data regarding time was its relationship to previous CAD experience. The subjects with experience using a CAD system other than AutoCAD required about half the time to complete the lessons as those with no other CAD experience. The results are shown in Figure 7 and Table 1.

FIGURE 7. Time Spent in Lessons by CAD Experience Other than AutoCAD.
TABLE 1
CAD Experience other than AutoCAD by Time Spent on the Lessons.

<table>
<thead>
<tr>
<th>CG EXP</th>
<th>AV TIME</th>
<th>STD</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>MORE 1 YR</td>
<td>491</td>
<td>470.9</td>
<td>8</td>
</tr>
<tr>
<td>(479)</td>
<td>(325)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 mo - 1 yr</td>
<td>471</td>
<td>202.6</td>
<td>12</td>
</tr>
<tr>
<td>NONE</td>
<td>950</td>
<td>807.1</td>
<td>26</td>
</tr>
</tbody>
</table>

$t = 2.45$ significant at the .01 level - two-tailed test (99% certainty)

The first two groups, representing some experience in CAD systems other than AutoCAD, were grouped for the test. The average and standard deviation are shown in brackets [ ].

Scores

The drawing test scores averaged 82.8% with a standard deviation of 12.3. It was found that neither age, sex, professional background, time spent on the lessons, nor general experience with computers significantly influenced scores on the final drawing. The group that did reflect a difference in scores and who required a significantly shorter time for their learning were those who had used a different CAD system extensively (Figure 8 and Table 2).

FIGURE 8. Drawing Scores by CAD Experience other than AutoCAD.
TABLE 2
CAD Experience other than AutoCAD by Drawing Score.

<table>
<thead>
<tr>
<th>CG EXP</th>
<th>AV SCORE</th>
<th>STD</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>More 1 yr</td>
<td>89</td>
<td>7</td>
<td>15</td>
</tr>
<tr>
<td>2 mo - 1 yr</td>
<td>82</td>
<td>(13)</td>
<td>22</td>
</tr>
<tr>
<td>None</td>
<td>81</td>
<td></td>
<td>42</td>
</tr>
</tbody>
</table>

\[ t = 2.349 \text{ significance at the .025 level (97.5\% certainty for a two-tailed test) } \]
\[ t \text{ statistic needs to be 2.37 for .01 or 99\% certainty} \]

The last two groups, representing less than expert experience in CAD systems other than AutoCAD were grouped.

PROJECTIONS

Concept Development

The issue of benefits from learning a second CAD system is extremely important. The observed bias against learning a second CAD system seems to be centered upon the hesitancy to learn a different set of procedures in a new system thereby threatening an individual with memory overload. The evidence from our study suggests that conceptual understanding facilitates faster and better learning. When the concepts are established, the procedures are no longer a problem. For example: manipulation of complex entities can be performed in a CAD system simply by following prompts. The essential “conceptual” qualities of the feature are formed, but not fully realized, with a first CAD system. However, when that feature is discovered in another CAD system, its characteristics become more meaningful and the designer imagines ways to explore its possibilities. Gaining conceptual understanding is thus a matter of hypothesis formation and testing. Embedded instruction offers an especially useful environment for this kind of testing to take place. Such concept formation is explained by Mayer (1983), “...when people try to solve problems, they sometimes begin by forming rules or hypotheses that are based on the wrong or too few dimensions ... but as more experience is acquired, correct performance increases.”

The use of Computer-aided Design tools involves a set of concepts rooted in mathematics and cybernetics. Examples of these concepts are: manipulating complex entities, perceiving the algorithmic properties of design, comprehending the mathematical database as a part of the design, and realizing the function of rule application. The final drawing test and the interviews with subjects revealed instances of partial or incorrect concept formation as well as evidence that some students had formed useful concepts. One test item which involved creating and inserting a block in the final drawing was missed by most of the test subjects who were learning their first CAD system. The item was not missed due to a lack of knowledge in the use of the BLOCK command. It was missed because the test subjects failed to correlate the significance of “insert a title block” as a direction to manipulate a complex entity. New CAD users did not appreciate the concept behind
the command; those who had previous CAD experience interpreted the instruction and used the command correctly.

In an effort to learn more about concept development, interviews were conducted with several test subjects. In one interview a young user of three CAD systems stated that one might expect errors when using the mirror command because any error in placing the mirror line would be doubled when the line was placed. He had no programming experience, but seemed to understand the algorithm involved. Another student reported that he liked to clean up his drawings using the change command whenever he noticed that two entities could be represented by one. Not only did he prefer the appearance when the drawing was redrawn, but he suspected that the information stored by the program was more concise.

Both of the interviews above were with technicians, who tended to reply that editing features were the most powerful aspects of CAD that they had noticed. Architects, on the other hand, often commented upon the ability to create and manipulate complex entities. One told me that he had thought of this marvelous scheme in which an abstraction of the "footprint" of the structure could be used in many transformations of scale and rotation in the building detail. Architects often referred to the power of CAD for repetition and commonly kept libraries of frequently used objects. The interviewed generally had developed some "special" technique which they were eager to tell about. These personal routines represented a wide variety, ranging from working out sequences for bringing up standard views to planning layering schemes to allow for temporary sketches underlying the permanent drawing. This development of personalized processes indicated that they were acquiring conceptual understanding of CAD features.

**Learning Style Factors**

Early in the testing we found that subjects with certain learning styles had more difficulty with the computer-based tutorial than others. It was noted in a previous report (Stoddard, 1985) that students with a "guided discovery" mode of learning made better use of context-sensitive embedded help. Those learners also found our tutorials more helpful. "Guided discovery" learners are characterized by their willingness to seek assistance from the on-screen lessons while learning the system. They are independent enough, however, to follow their own instincts to explore or test features that are of special interest to them. "Discovery" learners often left the lesson format to experiment on their own without guidance from the lessons. While they did learn a great deal, they may not have used their time as efficiently as those who followed the lesson format. Users with a "structured" learning style were often frustrated by the freedom to explore and the lack of a detailed course of study and user manual. These people preferred not to set their own goals, but wished for specific objectives and less opportunity to make errors.

The test scores were uniformly good even though the time spent and the learning strategies of the subjects varied widely. The test results lead us to believe that the professionals involved in this test sample understood their own educational needs well enough to exercise judicious control over many program options. In some studies of computer-aided instruction, with less sophisticated users, this has not proved to be true (Steinberg, 1977).

**RESULTS**

The test results indicated that, even though there were many new procedures to learn, the exploration that the embedded lessons allowed resulted in development of some underlying concepts of CAD which produced high scores and quick learning. Many of those responsible for teaching design students to use CAD profess that programming graphics is the best way to develop CAD design skills. Others believe that if the software interface is intuitive enough, computer
novices can produce early design drawings. Our studies indicate that when designers are led to
discover the underlying concepts common to all CAD systems, they will invent individual
approaches to using this design tool, though some systems are more conducive to that experience
than others. The use of embedded intelligent instruction in CAD software seems to foster
conceptual learning, especially when the students learn more than one CAD system.

LIST OF FIGURES
FIGURE 2. Help Example.
FIGURE 3. Error Feedback.
FIGURE 5. Time Spent in the Six AutoCAD Lessons.
FIGURE 6. Plot of Subject Age and Time Spent on the Lessons.
FIGURE 7. Time Spent in Lessons by CAD Experience Other than AutoCAD.
FIGURE 8. Drawing Scores by CAD Experience other than AutoCAD.

LIST OF TABLES
TABLE 1. CAD Experience other than AutoCAD by Time Spent on the Lessons.
TABLE 2. CAD Experience other than AutoCAD by Drawing Score.

REFERENCES
Birenbaum, M. and Shaw, D., "Task Specification Chart: A Key to a Better Understanding of

Herbert, D.M., "Study Drawings in Architectural Design: Applications for CAD Systems," In
Novitski, B. (Ed.), Integrating Computers into the Architectural Curriculum. ACADIA Workshop

Mayer, R.E. Thinking, Problem Solving, Cognition. New York: W. H. Freeman and Company,
1983, p 85.

Mitchell, W.J., Liggett, R. S., & Kvan, T., The Art of Computer Graphics Programming. (Van

Shaw, D., "Case studies in architectural CADD education," In Turner, J. (Ed.), Architectural
Education, Research, and Practice in the Next Decade. ACADIA Workshop '86 Proceedings
(University of Houston, Oct.1986).

(in publication) (U.S. Army Construction Engineering Research Laboratory, 1988).

Steinberg, E., "Review of Student Control in Computer-assisted Instruction," Journal of
Computer-Based Instruction, 3(1977), pp 84-90.


Yessios, C., "A Fractal Studio," In Novitski, B. (Ed.), Integrating Computers into the