December 2003

The Effect of Domain Knowledge on Icon Visualization

Keng Siau
University of Nebraska-Lincoln

Fiona Nah
University of Nebraska-Lincoln

THE EFFECT OF DOMAIN KNOWLEDGE ON ICON VISUALIZATION

Keng Siau
University of Nebraska–Lincoln
ksiau@unl.edu

Fiona Fui-Hoon Nah
University of Nebraska–Lincoln
fnah@unl.edu

Abstract

Iconic interfaces are now the de facto *interface for most computer systems. Despite the popularity of iconic interfaces and the widespread belief that iconic interfaces are easier to comprehend than non-iconic interfaces, we have not come across any published study that has examined the effect of domain knowledge on end users' interpretation of icons. An understanding of the relationship between domain knowledge and its effect on interpretation will enable us to design better icons to not only facilitate end users' interpretation but also reduce misinterpretations of icons. This paper reports on an experimental study that investigates the effect of domain knowledge on novice users' interpretation of icons. The hypotheses for this study are derived from theories in cognitive psychology and are empirically tested in an experiment. The results indicate that domain knowledge has no significant effect on the accuracy of interpretation. Domain knowledge, however, significantly affects the time taken to interpret icons and the confidence level of subjects. Theoretical perspectives on the interpretation of icons are also discussed.*

Keywords: Iconic interface, domain knowledge, problem space, experimental study, human information-processing systems

Introduction

The importance of human-computer interaction has been widely recognized and studied (e.g., Galitz 2002, Hecht 2001, Landay & Myers 2001, Raskin 2000, Ware 2000, Lai 2000, Faulkner 1998, Hackos & Redish 1998, Shneiderman 1997, Mandel 1997, Card *et al.* 1983). Icon visualization is, at the present time, the *de facto* standard for human-computer interaction. As Mullet and Sano (1995, p.173) wrote, "the imagery used in existing GUI's is predominantly (though by no means exclusively) iconic." The popularity of icons is not surprising as Larkin and Simon (1987, p.65) has pointed out that "a diagram is (sometimes) worth ten thousands words."

An icon is defined as a graphical representation of concepts that symbolize computer actions (Ware 2000). Proponents (e.g., Shanks & Darke 1999, Sia *et al.* 1997) of icons argue that iconic interfaces have many advantages. First, icons offer the advantage of easy recognition (Shneiderman 1997). Second, graphic images help users memorize and recognize functions available within an application (Whitten *et al*. 2001). Third, iconic interfaces are especially important for novice users who use computer systems on an infrequent and ad-hoc basis.

The arrival of the Internet era propels the application of iconic interface to a new height. With the exponential growth of Internet users and the increasing popularity of electronic commerce, millions of novice computer users spend a significant amount of time exploring and surfing the Internet. For e-commerce to become a reality, a critical mass of Internet users is a necessary condition. The use of simple and intuitive icons as the interface is one step in making Internet and e-commerce novice-friendly. Nevertheless, while the advantages and importance of iconic interfaces have been widely accepted, experimental research that studies the relationship between domain knowledge and its effect on icons interpretation is lacking. Understanding this relationship is crucial in the development and design of icons.

This paper describes an experimental that studies how problem domain knowledge influences novice users' interpretation of icons. In this study, subjects (novice computer users) were randomly divided into two groups. One group of subjects was given a set of icons with a short textual description attached to each icon. The textual description provides the problem domain in which the icon is commonly found (e.g., word processing software, programming software, system utility software) – the domain knowledge. The other group of subjects received the same set of icons without any description of problem domains. We measured the subjects' interpretation of each icon, confidence in that interpretation, and overall time taken to complete the experiment.

The rest of the paper is organized as follows: Section 2 discusses the different types of icons, characteristics of icons, and interpretation of icons. Section 3 presents the theoretical foundation and hypotheses for this research. Section 4 depicts the research model and describes the research procedure. The experimental results are shown in Section 5. Section 6 explains the results based on the theories and discusses the implication of the results. The last section, Section 7, concludes the paper and discusses future research directions.

Icons and Interpretation of Icons

Icons, which are familiar little pictures used as buttons and other graphical computer objects, are a shibboleth of modern interface design (Raskin 2000). An icon typically denotes its object by virtue of its own likeness to, or resemblance with, that object – on the basis of some quality or characteristics inherent in the icon itself (Mullet and Sano, 1995). Despite the ubiquity of icons, the term 'icon' is not very specific and can actually represent different things (Galitz 2002). Marcus (1984) suggested that icons fall into three categories:

(1) Icon – Something that looks like what it means
(2) Index – A sign that was caused by the thing to which it refers
(3) Symbol – A sign that may be completely arbitrary in appearance

From this perspective, icons are probably a mixture of true icons, signs, and indexes (Galitz 2002). We can also categorize icons into different kinds (Rogers 1989, Lodding 1983, Nielsen 1993, Dix *et al.* 1998):

(1) Resemblance icons: An image that looks like what it means. For example, a printer image to represent a printer.
(2) Symbolic icons: An abstract image that represents something. For example, a cracked glass can represent something fragile.
(3) Exemplar icons: An image that illustrates an example or a characteristic of something. For example, a sign at a freeway exit that depicts a bed indicates a hotel/motel.
(4) Arbitrary icons: An arbitrary shape that has meaning only by domain convention. These icons can be difficult for untrained users to interpret. For example, the use of a red triangle to represent warning.
(5) Analogy icons: An image that is physically or semantically associated with something. For example, the use of a clamp (because it squeezes) to represent a file-compression utility.

Marcus (1984) suggested that icons possess technical qualities of syntactics, semantics, and pragmatics. Syntactics refers to an icon's physical structure. Semantics indicates the meaning of an icon, and pragmatics refers to how the icons are physically produced and depicted. These different aspects of icons have tremendous impact on the interpretability and understandability of icons.

Raskin (2000, p.9) stated that one should "use a machine or tool in accord with its strengths and limitations, and it will do a good job for you. Design a human-machine interface in accord with the abilities and foibles of humankind, and you will help the user to not only get the job done but also be a happier, more productive person." To fully appreciate the human aspects of human-computer interface, we will reference the cognition literature. One of the key cognitive architectures is the Adaptive Control of Thoughts (ACT) (Anderson 1996, Anderson & Lebiere 1998).

An ACT production system consists of three memories: working, declarative, and production. Working memory contains the information that the system can currently access, which comprises information retrieved from long-term declarative memory as well as temporary structures deposited by encoding processes and the action of productions (Anderson 1996, Anderson & Lebiere 1998). Declarative memory and production memory are both long-term memory. The former deals with facts while the other handles processes or procedures that operate on facts to solve problems.

Declarative knowledge refers to knowing *that* something is the case whereas procedural knowledge concerns knowing *how* to do something. *Encoding* deposits information about the outside world into working memory whereas *performance* converts commands in working memory into behavior. The *storage* process creates permanent records in the declarative memory of the contents of working memory and can increase the strength of existing records in declarative memory. The *retrieval* process retrieves information from declarative memory into working memory. During the *match* process, data in working memory is put into correspondence with the conditions of productions. The *execution* process deposits the actions of matched productions into working memory. The whole process of production matching followed by execution is known as *production application*.

One important point to note is the limited capacity of the working memory. Miller (1956) claimed that working memory holds 7 $\pm$ 2 units of information while Simon (1974) claimed that it holds only about 5 units. Whatever the actual number is, the important point is that it is small and very limited. Because of its limited capacity, working memory is often referred to as the "bottleneck" of the human information-processing system. Figure 1 shows the ACT model.

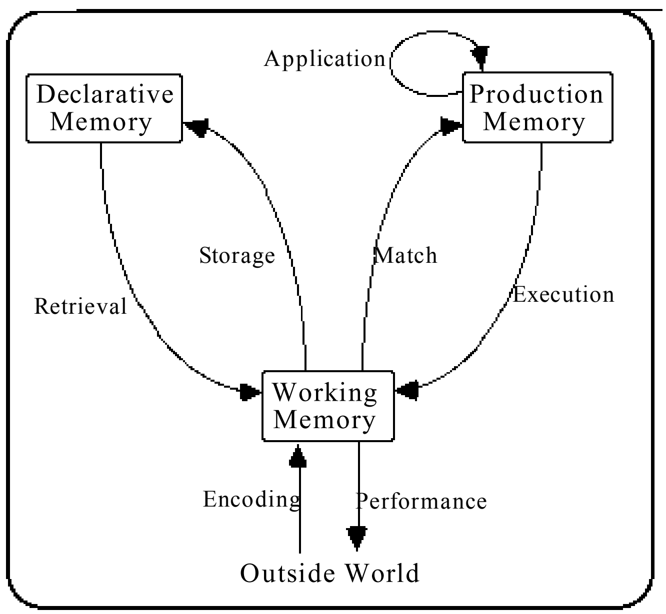

Figure 1. The Processes Involved in Interpreting an Icon

Based on the ACT model, the cognitive task of interpreting an icon involves three types of processes:

(1) the encoding and search process,
(2) the match process, and
(3) the execution process.

Encoding and Search Process

The encoding and search process operates on the data elements and data structures of the icons, seeking to locate sets of elements that satisfy the conditions of one or more productions in the production memory.

Match Process

The match process matches the condition parts of productions in the production memory to the data elements located during the encoding and search process (in the working memory).

Execution Process

Once a match is identified, the production is executed (or fired) to produce new (inferred) knowledge from the data element.

A well-designed icon should facilitate the three processes during interpretation. This icon interpretation model is used in the discussion section of the paper to help us interpret and understand the research findings.

Theoretical Foundation

The effect of domain knowledge on user interpretation of icons is discussed in terms of domain-specific knowledge, inferences, and problem space.

Types of Knowledge

Lindsay and Norman (1972) distinguished three types of knowledge that are used in problem solving:

(1) Facts – basic propositions that are immediately available to the subject
(2) Algorithms – sets of rules that automatically generate answers
(3) Heuristics – rules of thumb or general plans of actions or strategies

Holsapple and Whinston (1996) proposed two main classifications of knowledge: primary and secondary knowledge. The three primary types of knowledge are descriptive, procedural, and reasoning knowledge. The three secondary types of knowledge are: linguistic, assimilative, and presentation knowledge.

Anderson (1996) classified knowledge into declarative knowledge (facts stated in propositions) and procedural knowledge (algorithms and heuristics stated as procedures). Procedural knowledge can in turn be categorized into domain-general and domain-specific knowledge. Domain-general knowledge can be applied in more than one domain. In contrast, domain-specific knowledge applies in only one domain. The term domain refers to any defined area of content and can vary in its breadth. For example, our general ability to solve problems is domain-general knowledge whereas our ability to interpret an icon is domain-specific knowledge.

Domain-Specific Knowledge and Inferences

Problem-solving processes (such as interpretation of icons) include representing the problem, searching through the problem space, and evaluating the possible solutions. The quality of these problem-solving processes depends on the problem solvers' domain-specific knowledge. For example, several studies have illustrated the importance of domain-specific knowledge for drawing inferences (Walker 1987, Yekovich *et al.* 1990). Gagne *et al.* (1993) also stressed that the lack of domain-specific knowledge is a crucial constraint on the reasoning process.

The existing theories and prior studies, thus, suggest that subjects who are given icons with descriptions of problem domains will be able to cue the appropriate domain-specific knowledge more correctly, thus facilitating the problem solving processes and enhancing performance. On the other hand, subjects, who are given icons without any accompanying descriptions, do not have this advantage and will, therefore, exhibit relatively lower performance. This suggestion is consistent with the argument by Gagne *et al.* (1993) who stressed that problem representation is the key element to the success of problem solving, because the representation determines what knowledge will be activated in long-term memory. We, therefore, hypothesize that:

> ***Hypothesis 1:*** *Providing domain knowledge information on icons will result in a higher accuracy of interpretation.*

Problem Space

The task of interpreting icons can be viewed as finding the correct path or route through a problem space. Problem space, as pointed out by Ernst and Newell (1969) and Simon (1978), is the set of all states (or all possible sequences of operators) that the problem solver is aware of. The standard information-processing framework for defining a problem space consists of: (i) initial state, (ii) goal state, (iii) intermediate problem states, and (iv) operators (Newell and Simon 1972, Mayer 1992).

Initial state refers to the state in which the given or starting conditions are represented. In a goal state, the final or goal situation is represented. Intermediate problem states consist of states that are generated by applying an operator to a state; and Operators refer to the moves that are made from one state to the next.

Making the problem space easier to work with is the key to efficient and effective problem solving. Subjects who are given icons with problem domains specified are able to "prune the search tree", thus resulting in better performance. On the other hand, subjects who are given icons without problem domain description have a larger problem space to search. This will in turn lead to longer search time and an increased number of errors. Thus,

> ***Hypothesis 2:*** *Providing domain knowledge information on icons will result in a shorter interpretation time.*

Because users who are given icons with domain knowledge information will be able to reduce the number of possibilities and eliminate many choices, they will achieve a higher level of confidence in their interpretations.

> ***Hypothesis 3:*** *Providing domain knowledge information on icons will result in a higher confidence level.*

Although the theories suggest that knowing the specific problem domain is beneficial to the users in interpreting the icons, it is important, however, to verify the hypotheses through a controlled experiment.

Research Model and Procedure

The research framework for this experiment is shown in Figure 2. The independent variable is Domain Knowledge and the dependent variable is Performance.

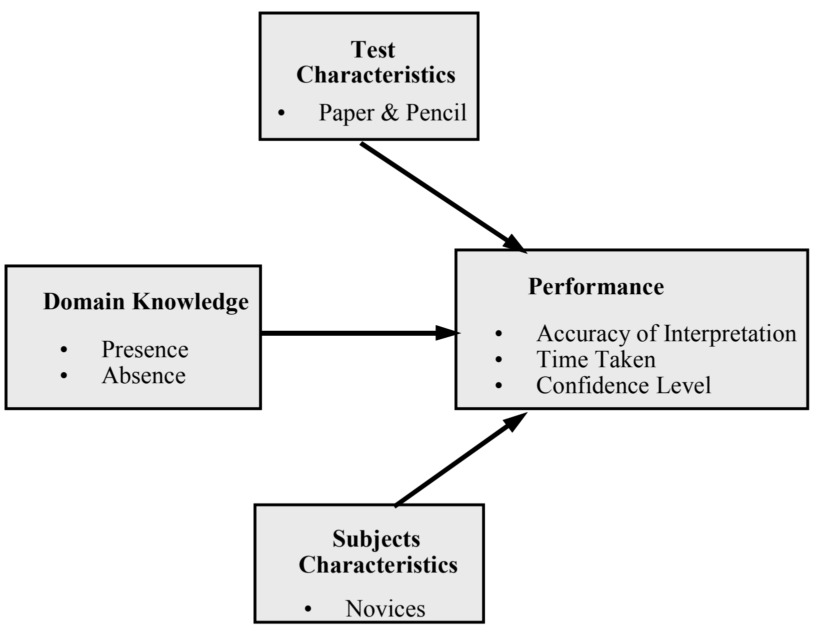

Figure 2. Research Model

Domain Knowledge

Twenty-five subjects participated in the experiment. The subjects were randomly assigned to two groups (13 in one group and 12 in the other). Depending on the group a subject was assigned to, s/he was given either the set of icons without problem domain description or the set of icons with problem domain description. Examples of problem domain descriptions would be "This icon is used for word processing" or "This icon is used for picture drawing." A set of 20 icons was used in the experiment. Some of the icons are shown in Figure 3. These icons were taken from the Icon Factory software. The icons are representative icons used in word-processing, spreadsheet, and graphics software. Each subject was given the full set of icons.

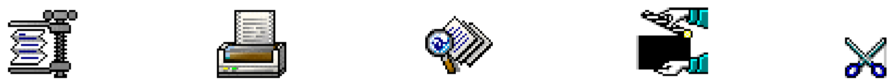

Figure 3. Examples of Icons

Performance

The dependent variable is performance, which consists of three components: the number of icons correctly interpreted, the total time to interpret the whole set of questions, and the confidence level specified by the subjects for each question. The subjects' interpretation of each icon was graded as either correct or incorrect. The time taken to complete the entire set of questions (20 questions) was timed (to the nearest minute). For the confidence level, we captured the subjects' confidence for each question (on a scale of 1-7).

Subjects Characteristics

The subjects were first year students in a university. As part of their first year program, they were required to help out in research studies. These 25 subjects decided to participate in this study. One of the criteria for participating in this study is that the subjects had only limited experience in using computers. Novice subjects were recruited for this study because the objective is to investigate the effect of domain knowledge on typical end users and not computer experts. Demographic information of the subjects was captured at the beginning of the study. The average age of the subjects is around 20.

Test Characteristics

A paper and pencil test was selected for this study. Each subject received a package containing the icons. The order of the icons was randomized to prevent any order bias. In other words, each subject received a different sequence of icons for interpretation.

Experimental Results

The T-test and analysis of variance (ANOVA) were used to test the hypotheses.

Accuracy of Interpretation

The total number of correct interpretations by each subject was recorded.

The t-test did not show a significant difference between the two groups ($p<0.216$) although the group that was given problem domain descriptions had a higher mean number of correct interpretations (10.92) than the group that received no problem domain description (9.31). This means that the presence of domain knowledge did not result in higher accuracy of interpretation.

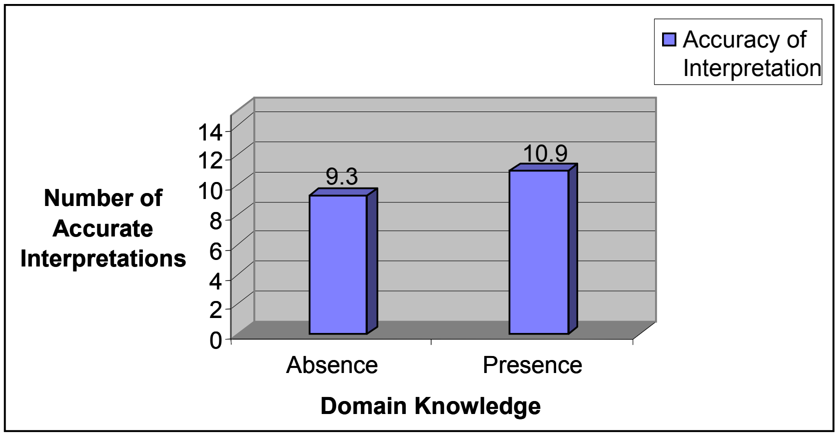

Figure 4. Accuracy of Interpretation

Time Taken

The time taken to complete the entire questionnaire was recorded. The t-test showed a significant difference between the two groups ($p<0.014$). The group that was given icons with descriptions took significantly less time (17.42 min) than the group that was not given any description (23.46 min). This is consistent with the theories, which predict that the availability of problem domain descriptions helps in "pruning the search tree", thus reducing the search time.

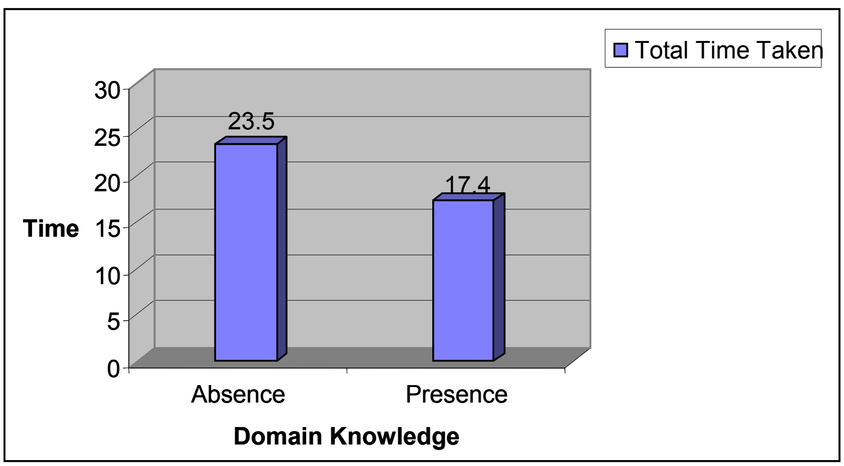

Figure 5. Time Taken

Confidence Level

The confidence level for each interpretation given by the subjects was captured (on a scale of 1-7). The analysis of variance (ANOVA) showed that there was a significant difference between the two groups ($p<0.0003$) in terms of confidence level. The group that was given icons with problem domain descriptions was more confident (4.39) in their interpretations than the group that was not given any problem domain description (3.82). Again, this is consistent with the prediction derived from the theories. With the problem domain description, the subjects could access the appropriate domain-specific knowledge and narrow the problem space. This gave the subjects more confidence in their interpretation.

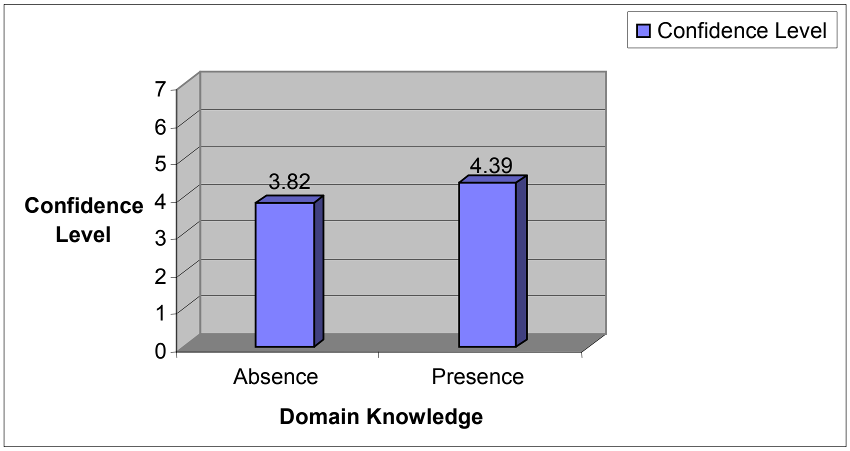

Figure 6. Confidence Level

Discussion

In this study, we looked at the effect of domain knowledge on novice users' interpretation of icons. The result shows no significant difference between the two groups of subjects with respect to accuracy of interpretation. In other words, the presence of domain knowledge did not improve the subjects' accuracy of interpretation. Hypothesis 1 is not supported.

From hindsight, the result is not surprising. Many icons are used fairly consistently across different application domains, which renders the presence of domain knowledge less vital. For example, the use of the object "printer" to represent print is almost a standard across all computer applications. The "printer" icons may look slightly different (i.e., the syntactic part of the icon) in different applications, but the semantics is the same (i.e., printing). Although these are novice subjects with little computer experience, they are likely to have prior exposure to iconic representation (though it may be somewhat limited). The use of "standard" icons in different software applications to represent the same concept facilitates knowledge transfer from one interface to the other. More specifically, it facilitates lateral transfer of knowledge. Lateral transfer refers to the transfer of known knowledge into a new setting but at the same level of complexity as the old settings (Gagne *et al.* 1993). This explanation is also consistent with Thorndike's theory of transfer through identical elements. Thorndike (1906) argued that, "positive transfer occurs because some of the elements in the to-be-learned task (B) are identical to elements that the learner has already learned from a previous task (A)."

With the presence of domain descriptions, the subjects who received icons with domain knowledge were able to apply an additional condition (i.e., the domain description) to the generation of the problem space. This enabled the subjects to significantly reduce the problem space by generating a problem space that contains less non-targets or distractors. With respect to the ACT theory, large problem space "overwhelms" the limited short-term memory. Because of the smaller problem space, the time needed to interpret the icons is significantly reduced. The presence of domain knowledge also leads to an increase in subjects' confidence level because of the smaller number of possible solutions (i.e., smaller problem space).

Informal post-experiment discussions with some of the subjects who received icons with domain knowledge indicated that they applied the "working backward" technique. Wickelgren (1974) and Polya (1965) suggested "working backward" as a technique for "pruning the tree." Instead of generating the problem space based on the icon, they generated the problem space based on the domain knowledge. In other words, they worked backward to generate a problem space that contains possible computer operations in the domain. For example, when given the icon of "a man running" and a domain description of "This icon is used for programming", a subject generated a problem space consisting of possible computer actions/commands in programming (e.g., edit, compile, run, save, etc.). This technique is helpful in interpreting some less commonly used icons – in these cases, the problem space generated from the domain knowledge can be smaller than the problem space generated from the icon itself. We also found that some subjects generated their problem space based on both icons and domain knowledge. They then tried to "prune the trees" by intersecting the two sets of problem space. Thus, it appears that domain knowledge is helpful in generating a smaller problem space and in "pruning the tree."

The results have some important implications. The results show that the presence of domain knowledge is somewhat helpful. Although the domain knowledge has no significant impact on accuracy of interpretation, it helps in both reducing the time to interpret the icons and increasing the confidence level of the subjects. Thus, developers of user interface should provide domain knowledge as far as possible. For example, icons can be grouped into categories with the category names provided (see figure 7). In Figure 7, the icons are grouped into five types with domain knowledge (i.e., WordArt, Picture, Tables and Borders, Web Tools, and Control Toolbox) listed. According to the results of our study, domain knowledge facilitates the interpretation of icons by users.

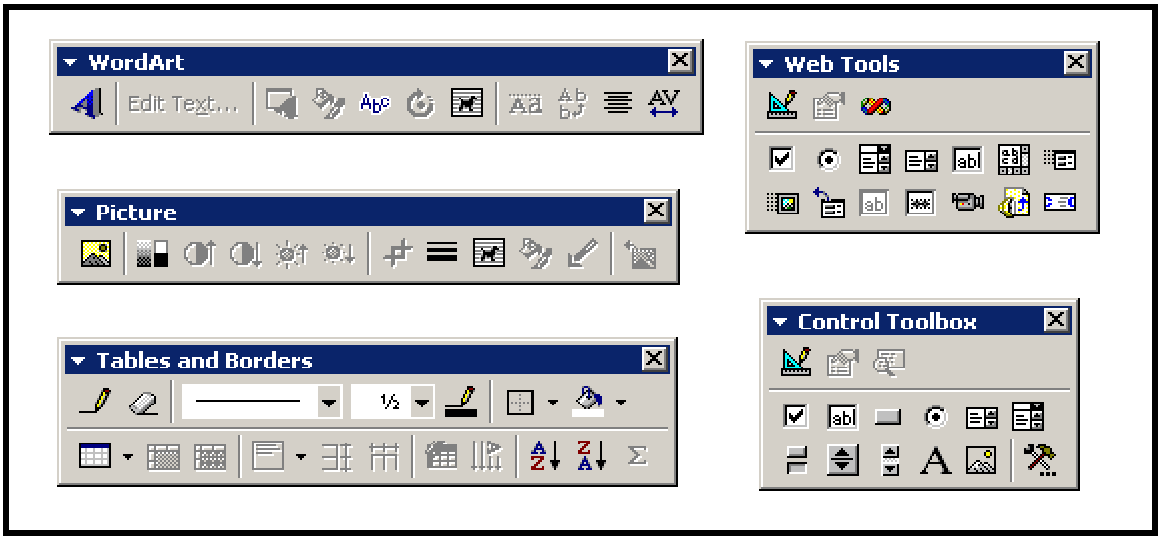

Figure 7. Examples of Icon Categories

Like all other research, this study has its share of limitations.

First, we did not differentiate between the various types of icons in this study. As discussed in the literature review, icons can be classified into a few kinds – e.g., resemblance, symbolic, exemplar, arbitrary, and analogy. It is possible that different kinds of icons will produce different results in this study. Future research will need to look into this issue.

Second, we used student subjects for this study. Some may argue that this is not the best group of subjects for this study. The generalizability of the results might be limited by the fact that this group of subjects is highly educated (i.e., freshmen in a university). The reliability and internal validity of the results, however, are not affected by the subjects' characteristics.

Third, despite our effort to ensure that subjects in this study had only limited experience with computers, there is a possibility that a few subjects might be familiar with some of the icons. However, this possibility is quite small. No outlier was found in either group. Also, the mean score (for correct interpretation) for each group is around 10 out of 20 (about 50%), which suggests that most of the subjects were not familiar with the icons.

Conclusions and Future Research

Icons have been and will continue to be an important component of user interfaces. Although iconic visualization has been a subject of much research, prior studies have been largely directed towards the design and development of icons. Studies on the use and interpretation of icons from a cognitive perspective are lacking. However, in order to design meaningful icons, we need to understand which properties of icons are important in interpretation.

In this paper, we began with a review of theories related to icon interpretation. The research model and hypotheses were then formulated based on the research question and theories respectively. An experimental study was then conducted to test the hypotheses. The results of the study showed that the presence of domain knowledge facilitates the interpretation of icons. The subjects provided with domain knowledge had slightly higher accuracy in interpreting icons, though not statistically significant. These subjects had a significantly higher confidence level and took significantly less time.

This research can be extended in several directions. First, future studies can investigate the effect of different kinds of icons (e.g., symbolic, analogy, etc.) on user interpretation. Second, this research can be replicated using different groups of subjects to validate the results obtained in this study. Third, process-tracing techniques can be used to further understand the cognitive processes involved in interpreting icons.

References

Anderson, J. R. (1976). *Language, Memory, and Thought*. Hillsdale, NJ: Erlbaum

Anderson, J.R. (1996). *The Architecture of Cognition*. Cambridge. MA: Harvard University Press.

Anderson, J. R. and Lebiere, C (1998). *The Atomic Components of Thought*. Mahwah, NJ: Erlbaum.

Card, S.K., Moran, T.P., and Newell, A. (1983). *The Psychology of Human-Computer Interaction*, Hillsdale, NJ: Erlbaum.

Dix, A., Finlay, J., Abowd, G., and Beale, R. (1998). *Human-Computer Interaction*. Second Edition. Prentice Hall Europe.

Ernst, G.W., and Newell, A. (1969). *GPS: A Case Study in Generality and Problem Solving*. New York: Academic Press.

Faulkner, C. (1998). *The Essence of Human-Computer Interaction*. Prentice Hall.

Gagne, E.D., Yekovich, C.W., and Yekovich, F.R. (1993). *The Cognitive Psychology of School Learning*. Harper Collins.

Galitz, W.O. (2002). *The Essential Guide to User Interface Design*. Wiley.

Hackos, J., and Redish, J. (1998). *User and Task Analysis for Interface Design*. John Wiley & Sons.

Hect, D. (2001). Printed Embedded Data Graphical User Interfaces. *IEEE Computer*, Vol. 34, No. 3, pp. 47-55.

Holsapple, C.W., and Whinston, A.B. (1996). *Decision Support Systems: A Knowledge-Based Approach*. West Publishing Company.

Landay, J., and Myers, B. (2001). Sketching Interfaces: Toward More Human Interface Design. *IEEE Computer*, Vol. 34, No. 3, pp. 56-64.

Lai, J. (2000). Conversational Interfaces. *Communications of the ACM*, Vol. 43, No. 9, pp. 24-27.

Larkin, J.H., and Simon, H.A. (1987). Why a Diagram is (Sometimes) Worth Ten Thousand Words. *Cognitive Science*, Vol. 11, 65-99.

Lindsay, P.H., and Norman, D.A. (1972). *Human Information Processing: An Introduction to Psychology*. New York: Academic Press.

Lodding, K.N. (1983). Iconic Interfacing. *IEEE Computer Graphics and Applications*, Vol. 3, No. 2, 11-20.

Mandel, T. (1997). *The Elements of User Interface Design*. Wiley.

Marcus, A. (1984). Icon Design Requires Clarity, Consistency. *Computer Graphics Today*.

Mayer, R. (1992). *Thinking, Problem Solving, Cognition*. Freeman.

Miller, G., (1965). The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information, *Psychological Review*, 63, pp. 81-97.

Mullet, K. and Sano, D. (1995). *Designing Visual Interfaces*. Prentice Hall.

Nielsen, J. (1993). *Usability Engineering*. Morgan Kaufmann.

Newell, A. and Simon, H.A. (1972). *Human Problem Solving*. Englewood Cliffs, NJ: Prentice Hall.

Polya, G. (1965). *Mathematical Discovery. Vol. II: On Understanding, Learning and teaching Problem Solving*. New York: Wiley.

Raskin, G. (2000). *Humane Interface: The New Directions for Designing Interactive Systems*. Addison-Wesley.

Rogers, Y. (1989). Icons at the Interface: Their Usefulness, *Interacting with Computers*, 1 (1), pp. 105-117.

Shank, G., and Darke, P. (1999). Understanding Corporate Data Models. *Information and Management*, Vol. 35, No. 1, pp. 19-30.

Shneiderman, B. (1997). *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Addison-Wesley.

Sia, C., Tan, B., and Wei, K. (1997). Effects of GSS Interface and Task Type on Group Interaction: An Empirical Study. *Decision Support Systems*, Vol. 19, No. 4, pp. 289-299.

Simon, H.A. (1978). Information Processing Theory of Human Problem Solving. In W.K. Estes (ed.), *Handbook of Learning and Cognitive Processes*. Hillsdale, NJ: Erlbaum.

Simon, H.A. (1974). How Big is a Chunk ?, *Science*, 183, pp. 482-488.

Thorndike, E.L. (1906). *Principles of Teaching*. New York: Lemke & Buechner.

Walker, C.H. (1987). Relative Importance of Domain Knowledge and Overall Aptitude on Acquisition of Domain-Related Information. *Cognition and Instruction*. Vol. 4, pp. 25-42.

Ware, C. (2000). *Information Visualization*. Morgan Kaufmann.

Whitten, J., Bentley, L., and Dittman, K. (2001). Systems Analysis and Design Methods. McGraw Hill.

Wickelgren, W. (1974). *How to Solve Problems: Elements of a theory of Problems and Problem Solving*. San Francisco: Freeman.

Yekovich, F.R., Walker, C.H., Ogle, L.T., and Thompson, M.A. (1990). The Influence of Domain Knowledge on Inferencing in Low-aptitude Individuals. In A.C. Graesser & G.H. Bower (eds.). *The Psychology of Learning and Motivation: Advances in Research and Theory. Vol. 25: Inferences and Text Comprehension.* San Diego, CA: Academic Press.