C

COMPUTER-AIDED INSTRUCTION

INTRODUCTION

Background of CAI

Historically, computer-aided instruction, which is also called *computer-assisted instruction* (CAI), has roots in Pressey's 1925 multiple-choice machine and the punch-board device, which foreshadowed the network-supported tutorials of today. Pressey's multiple-choice machine presented instruction, tested the user, waited for an answer, provided immediate feedback, and recorded each attempt as data. In 1950, Crowder developed a process for the U.S. Air Force in which a CAI program presented some content as text, tested the user, provided some feedback, and then branched to corrective instruction or new information based on supplied responses. Branching was thought to be an advance on Pressey's multiple-choice machine (1). In 1954 at the University of Pittsburgh, Psychologist B. F. Skinner demonstrated a teaching machine for "reinforcing," not just teaching, spelling and arithmetic with a machine.

> The user may access auditory material, listen to a passage as often as necessary then transcribe it. The machine then reveals the correct text. He may listen to the passage again to discover the sources of any error (Ref. 2, p. 147).

Developers applied principles of data transmission (3,4) and reinforcement theory (2) to a variety of educational situations. Skinner used reinforcement theory to downplay the role of punishment in changing behavior. Instead, he was convinced that behavior could be altered by simply using positive and negative types of reinforcement. Positive reinforcers presented rewards (good grade or congratulatory comment) after the user achieved a desired behavior. Negative reinforcers remove aversive stimuli after the user failed to achieve a desired behavior. Crowder (5) applied these ideas to "intrinsic programming" so that a user's responses determined the material to be presented next. The main advantage of intrinsic programming was that it did not waste the time of the fast learner with unnecessary repetition. Its disadvantage was that it required a large amount of content to present even a relatively small amount of material (6).

Figure 2 shows an extended approach to alternative sequences of that consist of multiple frames. Figure 3 depicts a simple "wash-back sequence," in which users who are struggling with a concept are routed back through earlier parts of the program. Figure 4 illustrates a "wash-ahead sequence" that moves the users along faster if they grasp concepts quickly. Figure 5 shows a complex strategy in which incorrect answers were weighted for seriousness, and the student "washed back" one, two, or three steps, depending on how he or she answered.

Later, CAI researchers (7,8) observed that algorithms for teaching with CAI had to incorporate both the physical programming or authoring to run the computer program and the instructional programming required to learn from the program.

Pros and Cons of CAI

Contemporary CAI is either downloaded from an Internet site and run locally, or it is shipped on DVD with a colorful reader and links to a companion website. Some CAI programs even run interactively online. The main advantages of developing CAI include the following:

- Modular – easy to change the instruction, practice examples, and tests.
- Self-administered – "just in time".
- Feedback can be either immediate or delayed, comprehensive or partial.
- Good lesson integrity.
- Individualized, customized for users who need specific skills.
- Automatic record keeping.
- Program–user interaction can often be increased, which offers more lesson control for more advanced learning that can be motivating for some users.

The main disadvantages of CAI include the following:

- Modular – can be boring.
- Self-administered – therefore anonymous (who's taking the instruction?).
- Tends to "teach to the test", which promotes convergent thinking (e.g., answer look-up), instead of teaching for comprehension, which promotes divergent thinking (e.g., answer construction).
- Long development time, and a short shelf life.
- More costly to develop than hiring an instructor.
- Assumes reading ability and spatial acuity.

Design Guidelines for CAI

Computer-assisted instruction (CAI) means different things to different people. An active CAI developer will design a project based on a preferred method or guideline. The guideline can usually be associated with a particular definition of multimedia learning. In fact, at least eight different design guidelines for CAI represent the state of the art in CAI.

Whatever Works. Novice developers, students, and even experienced developers from time to time, will use "whatever works" as a quick design guide. Whatever works is based on the expert intuition and experience of the developer. The assumption is that whatever worked last time will work this time. So, CAI is grounded effectively in previous CAI designs.

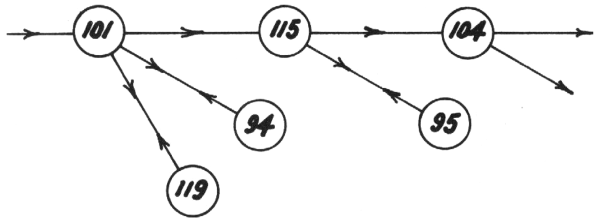

Figure 1. A simple intrinsic programming sequence in which single alternative frames exist to reinforce concepts that seem difficult to some students (Refs. 5 and 6, p 156).

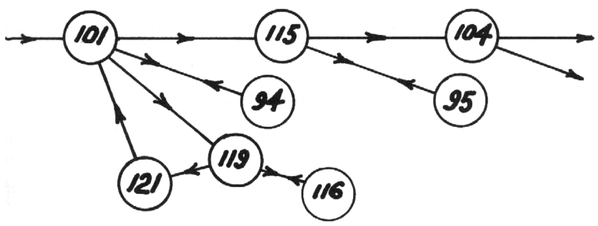

Figure 2. Alternative intrinsic programming sequences that consist of multiple frames (Refs. 5 and 6, p. 157).

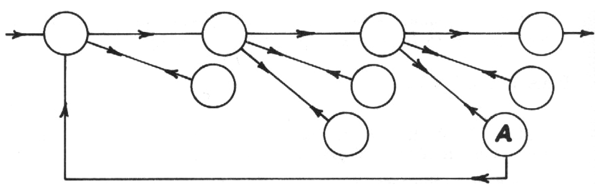

Figure 3. A simple "wash-back sequence" in which the user who struggled with a concept was routed back through earlier parts of the program (Refs. 5 and 6, p. 157).

Design-by-Type. According to "design-by-type," CAI is one of four types of instructional computer programs that define the field of CAI, which are distinct from one another: tutorial, drill and practice, simulation, and game.

Favorite Feature. The "favorite feature" guideline states simply that CAI is the most appropriate technology for teaching people something new or for giving them automated practice. In general, newer technologies can offer better instruction than older technologies.

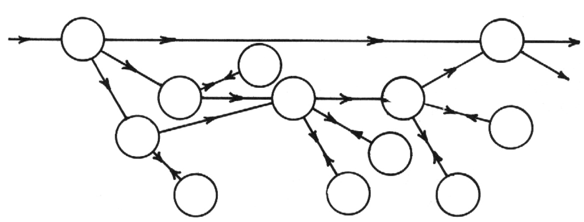

Figure 4. A "wash-ahead" sequence that moves students along faster if they grasp concepts quickly (5).

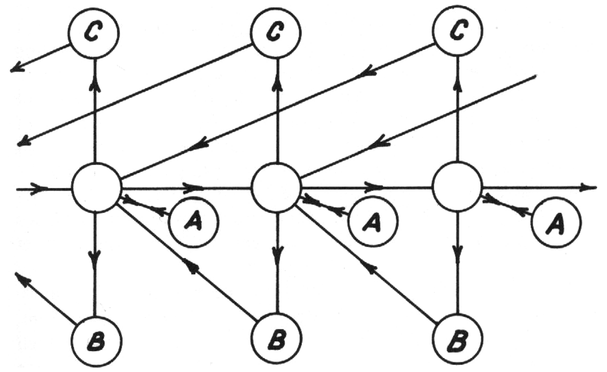

Figure 5. A complex strategy in which incorrect answers are weighted for seriousness, and the student may be "washed back" one, two, or three steps depending on how he or she answers (5).

Favorite Method. The "favorite method" guideline says that CAI is a teaching method that is especially good for teaching novices. Contemporary favorite methods proffer the discovery teaching method affords better use of a computer than direct instruction method.

Balance-the-Input. According to the "balance-the-input" guideline, CAI is a mixture of two separate kinds of representations in the user's mind, "logogens" and "imagens." Logogens are the private mental representations of words we hear or read — either spoken or printed on-screen text. Imagens are the private mental representations of graphics and white space that we observe. CAI can produce spatial and language-like representations sufficient to produce learning in the user (9,10).

Maximum Impact. The "maximum impact" guideline recognizes CAI as a more complex mixture of private acoustic images, inner eye sensations, and subvocalizations from an inner voice to our inner ear (11). The implication is that the user's senses should be regularly bombarded by ever-deepening sensations to maintain the appropriate mixture.

Cognitive Load, First. The "cognitive load, first" guideline maintains clear preference for process over outcome — look to the user's mental processing first, and the learning outcome will take care of itself. Corresponding idea elements presented in visual and auditory load the user's memory (extrinsic load) that can consume the cognitive resources. Depending on program design and user's prior knowledge, CAI does little else than increase intrinsic load and extrinsic load in the user that reduces performance, the debilitating effects of which can be changed into more productive germane load.

Structuring Sound Functions. According to the "structuring sound functions" guideline, CAI is a method of helping a user to focus their own attention during

Table 1. A summary of Design Guidelines of Computer-assisted Instruction (CAI) and Corresponding Working Definitions of Learning from Media

Guideline	Supporting Definition
Whatever works	Intuition of the developer.
Design-by-type	Taxonomy of instructional software.
Favorite feature	Technology selection.
Favorite method	Teaching method.
Balance-the-input	Logogens and imagens.
Maximum impact	Auditory memory vs. visual memory.
Cognitive load, first	Intrinsic, extrinsic and germane.
Structuring sound functions	Attentional control.

interaction with a computer. Table 1 summarizes these different approaches.

WHATEVER WORKS

As a guideline for designing CAI, whatever works is based on the intuition of the developer, that is to say the preferences, predispositions, and experiences of the designer. Under certain conditions, these factors can be illustrative to the user, which serve to make the unfamiliar familiar. In other situations, they can aid exploration, helping the user to identify questions, select constructs, or develop measures. In the early days of CAI, however, developers knew little about the effects of auditory manipulations on language learning (12). Descriptions of human interactions with technology were explained as a function of delivery system performance instead of how users' divided their attention on difficult and unfamiliar tasks. In the early days, *whatever works* generalized the results from one context to another; the guidelines were frequently too general or not generic enough to fit the need. Users were required to determine the trade-offs between conflicting guidelines. Early approaches to learning from auditory-visual technology were found to be ad hoc, unproven, and required frequent reexamination and clarification (13). Users were required to control their own processing of these sensations. It was not until the 1980s, with the advent of the microcomputer, that most educators, especially those in schools, could do much with the design and programming of CAI. In the 1980s, calls went out for better guidelines, "a systemic model for designing sound and visuals together on computers is needed - designer sound for computer systems" (Ref. 14, p. 1).

> There is a lack of guidelines for the proper use of audio in computer based instruction . . . Designers must decide if audio should replace, enhance, or mirror the textual information on a CAI screen (Ref. 15, p. 277).

In response to this call, "whatever works" guidelines began to appear in journals, magazines, and the Internet. Some writers collected other people's lists and suggestions from the Internet on whatever worked (16,17). Others assembled real-world criteria for Internet resources (18) or developed new learner-centered principles for placing the user at the center of a web-based learning environment (19), and practical guidelines for media integration (20). Maddux and Johnson (21) offered ergonomic guidelines, which include deciding on proper page length; making frequent revisions; and setting up Internet child safety rules, including text-only browsers, and soon. Mikovsky (22) offered a long list of distance-learning interactions with the inference that communication skills should be developed over the Web. Boettcher and Cartwright (23) recommended three modes of dialog and communication that should work: (*1*) the dialog between the faculty member and student, (*2*) the dialog among students, and (*3*) the dialog between the student and instructional resources. Still other authors focused on the tools for developing Web-based CAI. King (24) offered a guide for using web tools. Reynolds (25) recommended that instructional designer need to know HTML. Despite the finding that the application of a stepwise design approach to a metaphorical user interface was both problematic and difficult-to-build; Ohl and Cates (26) wrote a general guideline for CAI designers.

Although more is known today about what really works, the results are still mixed (27–29). The fundamental problem with "whatever works" advice is that it can only offers developers a stochastic method of combining sound and visual instruction in a computer. Stochastic methods employ a hit-and-miss approach to design that describes human interactions with technology as a function of the playback systems instead of how to help users to manage the incoming visual and auditory sensations. Stochastic guidelines usually reflect the following: (*1*) outdated bottleneck theories of human attention, (*2*) random cueing, (*3*) message redundancy (audio-redundant and audio-accompaniment), (*4*) oral reporting (overlapping from a visual display, continuous prompting or requesting), and (*5*) random auditory feedback (30). Subsequently the call has gone out for more purposeful advice on the design of instruction for multimedia learning based on learning theory (31). The respisne has been that some developers continued to rely on their instinct, others hoped to design by type.

DESIGN-BY-TYPE

Before 1988: Tutor, Tool, Tutee

Before 1988, three types or modes of program design were classified together in a taxonomy (32). The first type was called the tutorial or "tutor" mode of computer operation required content to be developed in a specific content area with substantial coding by expert computer programmers. Computer tutors were purported to accommodate a wide range of user differences. Psychologists and educators who adopted the tutor mode of computer operation recommended that the programs be designed to teach all manner of knowledge and skills to a wide range of audiences.

The second type was called a "tool," where the computer was recognized to help users to learn a variety of subjects, as typewriter or slide-rule would. In school, students used a calculator program for math or science assignments, or a map-making tool in geography, or a text editor to write and edit their essays.

The third type or mode of program design was the "tutee." Using the computer as tutee was to "tutor the computer," which meant the user had to learn to program the computer and to *understand* the computer. The benefits of computer as tutee was that the users would (*1*) learn something about how computers work, (*2*) gain new insights into their own thinking through learning to program, and (*3*) improve the math. Only the first of these claims was justified. Proponents of the *tutor use* criticized the *tutee* camp for delaying the acquisition of skills that would be needed to make "real" discoveries.

After 1988: Tutorial, Drill, Simulation, Game

After 1988, the three types or modes of program design were reclassified into an instructional software taxonomy, which is based on two seminal books: one by Hannafin and Peck (8) and another by Alessi and Trollip (7). These books were both practical and theory-based. Both books recommended four categories of instructional software: tutorial, drill and practice, simulation, and game. Users wanted to learn how to develop their own instructional software, many following the taxonomy. Authoring systems such as *HyperCard*, *ToolBook*, and *PC-PILOT* were being purchased by colleges and schools for use by educators and even students (33). Users in colleges and schools who worked in the MS-DOS environment learned to use *ToolBook* to develop applications for on-the-job computer-based training. Educators working on the Apple platform learned to use HyperCard and its powerful programming language HyperTalk to produce pleasing visuals and sound. The rapid prototyping capabilities of HyperCard, HyperSound and MacRecorder made them appropriate developmental tools for developing CAI (34). Teachers using any of these programs could learn to design learning materials or let their users take an active role in the design of their own materials.

The 1990s: Consensus Change

During the 1990s, the consensus of opinion changed. There were more crucial concerns than developing one's own instructional software (35–37). The primary interest in most colleges and schools turned to acquiring enough ready-made curricular and applications software for students and instructors to use. A second concern was that few educators in their job situations had the time or the need to develop CAI software. Third, and perhaps most critically, authoring software was often unavailable in most schools and school boards, so instructors did not have the authoring software required to develop their own instructional software. And finally, although CAI tutors were believed to benefit to some students, the developer was now believed to be the one who derived the most benefit from its development.

FAVORITE FEATURE

Since the 1940s, instructional designers chose technologies for their features (38–40), comparing one technology or feature of a technology with another. Over the years "favorite feature" was known as "delivery media view" (31), "media comparison" (41) and "technology selection" (42), although they all meant the same thing—technology A is better than technology B.

The "favorite feature" guideline persisted with early adopter's continued use of certain software and certain features in software, to the exclusion of other features. Guidelines were developed that subsequently helped to popularize the view that learning from CAI simply meant selecting an appropriate feature or plug-in technology to "deliver" instruction. The procedure requires setting some learning objectives and a test, and drawing a flowchart showing that showed the sequence of learning objectives. A videotape or video clip, for example, was believed to afford the user live action motion. A book or multimedia encyclopedia can afford the user-indexing capability. Selecting a sound clip would decorate, enhance, create fantasy, reward, or represent that which would have otherwise been least effectively communicated as text or numbers (43,44). This media was sound as add-on technology. Even haptic events were becoming learning environments, such as hands-on simulations, hand-sensing gloves, and even simulated reach-in-and-grab technologies that could be downloaded into content and computer applications for user interface navigation.

The favorite feature guideline is still evident in reports about production effects in television programs on children's attention (45) and media comparison research with adults (46). Even today, favorite feature guidelines are widely published in the literature. Favorite feature research usually compares computer- or web-based material presented in a lab with an instructor-based presentation in a classroom (e.g., Refs. 47–49), or distance education technologies with classroom instruction technologies (28). Favorite feature research is likely to continue as long as evolving hardware and software permit designers and users adaptive and nonlinear interactions and a greater differentiation of visual events and sound.

FAVORITE METHOD

Critics of the favorite feature guideline tended to be advocates of the "favorite method" guideline. The criticism of the favorite feature guideline is either that research about whether "technology A is better than technology B" is an unproductive line of questioning (31,50), or that differences of their effects on users was usually statistically nonsignificant because most learning effects are actually attributable to instructional design (41,50). Because new technology *per se* seems to have little or no affect on learning (41), the advice has been to use older, less-costly technology whenever possible to accomplish similar results because technologies will produce more or less equivalent results (50). The favorite method guideline therefore, is based on the most popular teaching method at the time. One of the most favorite teaching methods in CAI is called the events of instruction. A central aim of the favorite method guideline was to implement a particular teaching method into a computer program using some or all nine events of instruction (51).

1. ***Gaining attention*** to important aspects of the computer program is the first event of instruction. Attention is critical to learning; without it, no learning can occur.
2. ***Informing*** the user of the objectives in the program is the second event of instruction. A learning objective is a statement for users in an "ABCD" format; that is to say, the audience (A), behavior (B), condition (C), and degree (D). Communicating objectives takes little time and can help the user to stay on track.
3. ***Showing*** users the relationship between what they will be learning in the computer program and what they already know to this point is the purpose of the third event of instruction. This step also lends relevance to the learning process.
4. ***Presenting some material*** is the fourth of the nine events of instruction. Two basic choices can be used for presenting some material in CAI: either direct or constructivist. The aim in direct instruction in CAI is to transmit some skill and knowledge in a structured way, monitor their performance, and then provide some feedback. The aim in constructivism in CAI on the other hand, is only to guide the user as they explore, then facilitate the learning process as they collaborate "use" resources, and generate their own knowledge.
5. ***Providing guidance*** in the computer program is the fifth event of instruction. The type of guidance depends on the choice of instructional method adopted. The direct instruction method will have plenty of guidance and moral support. The constructivist method will have less direct guidance.
6. ***Eliciting evidence*** of user cognition and performance through fault-free questions or ungraded responses is the sixth of the nine events of instruction. The type of evidence collected will, again, depend on the instructional method adopted. Direct instruction would have a product developed to exact specifications and matched to objectives, such as a test following a pod cast lecture, observation data of a complex skill performed to mastery, or multiple-choice data after practicing a CAI drill. Constructivist instruction could have an electronic portfolio of original products developed by the user with little or no assistance, such as an original CAI lecture uploaded into voice-over IP technology and presented to peers for their criticism, or even the user's thoughts and feelings recorded (spoken or written) in a diary about the learning process.
7. ***Providing feedback*** is the seventh of the nine events of instruction. Not all practice in CAI requires feedback. If the expectation is that the user will rehearse some knowledge or a skill that was previously introduced, then feedback *is* required in the CAI. If, however, the expectation is that the user will experience some new knowledge or explore a new skill for the first time, then feedback *is not* required in the CAI.
8. ***Assessing cognition and performance*** through assessment is the eighth event of instruction. In direct instruction CAI, cognition and performance would be assessed by a written test or the development of a product to specification. The measure of success would be the same for all users. In constructivist CAI however, cognition and performance could be assessed by observation, peer- or self-assessment, product portfolios against a rubric, or a checklist. The measure of success could be different for each user.
9. ***Enhancing retention*** through transfer tasks is the ninth event of instruction. Presenting users with varieties of tasks that are related to what they have already learned can help them to transfer original learning to the new context. Low road transfer happens when the conditions in the new context are sufficiently similar to those in the original learning context to trigger well-developed, semi-automatic responses. High road transfer in contrast, requires mindful abstraction from the original learning context or application and a deliberate search for connections.

BALANCE THE INPUT

According to this guideline, CAI should be a balance of logogens (words, spoken or in text) and imagens (pictures or graphics), based on Paivio's dual coding theory (9), which distinguishes between a language-like system of verbal codes and a separate nonverbal system of spatial codes. The implication is that users learn better when the instructional material does not require them to split their attention between multiple sources of mutually referring information. A balanced input is achieved when CAI generates logogens and imagens in the user's mind. The words, (spoken or in text) get weeded out and off loaded into narration for processing separately (52), because users learn better when the verbal information is presented auditorily as speech rather than visually as on-screen text (10,31).

MAXIMUM IMPACT

The "maximum impact" guideline seeks to present high-resolution graphics, video, and sound in CAI to maintain sensations in the user's auditory and visual working memories from which they can create coherent episodes to change their long-term memory. Notably, sound is given an important role in the maximum impact guideline because of its durability and resistance to interference and forgetting. A strong rationale for using sound in CAI is that memory for material presented in sound is more durable and resistant to interference from other modalities than visually presented material, and it is more resilient to forgetting than visual traces (42). Neurological evidence confirms that sound stimulation can evoke responses in visual areas of the brain, even in very young children (53).

The CAI designer, therefore, aims to organize stylistically rich audio-visual sensations into a personal, cultural, or political expression that is characteristic of film style of the early film era. The user activates and combines these separate sensations in their attentional controller sensory codes. The maximum impact guideline is based on a contemporary theory of human working memory (11) of sensory regulation. This method has been applied to a cognitive theory view of multimedia learning (31) with the aim of dispensing separate sensations for auditory and visual memory systems. Like the balanced-input guideline for CAI, the quantities of sound and visuals of the maximum impact guideline must be about the same to reduce the probability of diverting attention in the novice's working memory.

COGNITIVE LOAD, FIRST

CAI can deliver large amounts of cognitive load to the user that can overload or underload his or her working memory and reduce learning effectiveness. The cognitive load, first guideline states simply "never overload (or under-load) the user," based on cognitive load theory (54). Cognitive load theory (55–57) distinguishes between three types of cognitive load in working memory: intrinsic, germane, and extraneous.

Intrinsic Load

Intrinsic load refers to the number of information elements the user needs to hold in working memory simultaneously to comprehend the information. Element interactivity is intrinsic to the material being learned and cannot be altered by instructional manipulation. Only a simpler learning task that omits some interacting elements would reduce this type of load. The omission of essential, interacting elements would compromise sophisticated understanding but may be unavoidable with very complex, high-element interactivity tasks.

Extraneous Load

Extraneous load is that which is increased in a learning environment or instructional methods that require users to engage in activities not required for new schema construction or automation. Extraneous cognitive load is primarily important when intrinsic cognitive load is high because the two forms of cognitive load are additive. If intrinsic cognitive load is low, then levels of extraneous cognitive load may be less important because total cognitive load may not exceed working memory capacity. As a consequence, instructional designs intended to reduce cognitive load are primarily effective when element interactivity is high. When element interactivity is low, designs intended to reduce the load on working memory have little or no effect. The designer who aims to keep from overloading (or underloading) the user with sensations may follow the cognitive load guideline.

Germane Load

Whereas extraneous cognitive load interferes with learning, germane cognitive load enhances learning. Instead of working memory resources being used to engage in search, for example, as occurs when dealing with extraneous cognitive load, germane cognitive load results in those resources being devoted to schema acquisition and automation. Note that increases in effort or motivation can increase the cognitive resources devoted to a task. If relevant to schema acquisition and automation, such an increase also constitutes an increase in germane cognitive load (58).

The cognitive load, first guideline aims to maximize germane load by keeping a lean CAI design, thereby reducing extraneous load. The key is to manage your CAI for the amount of cognitive load on the users' working memory (58,59).

THE SSF MODEL

The structured sound function (SSF) model recommends applying sound to help users to control their attention to the visual events in the CAI (42). Figure 6 shows the five functions and three structures in the SSF model that when combined can help users to focus their attention on important visual events in CAI.

Goal setting is an important aspect of attentional control. Goal setting can promote convergent thinking in the user (e.g., answer look-up) or divergent thinking (e.g., answer construction). A sound prompt that promotes convergent thinking would have the user's attention directed toward a solution in a single, stepwise procedure. A sound prompt that promotes divergent thinking would have the user's

	< Structuring the sound with a visual event >		
	The goal: is convergent or divergent	**The constancy:** is continuous or discontinuous	**The density:** is massed, spaced or summarized
< Giving the sound a function >			
A temporal prompt: that cues that counterpoints that dominates that undermines			
A point of view: objective, subjective, performer, political, socio-cultural			
A locale: real, imaginary			
An atmosphere, feeling, mood			
A character's: past, future, personality			

Figure 6. The SSF model to help users focus their attention in CAI.

attention directed toward visual authoring tools, libraries of resources, or advice that permits easy access and experimentation in an open environment. During convergent goal setting the user is encouraged to use a variety of sources to bear on a problem to produce "the correct result." This kind of thinking is appropriate in domains of science and mathematics. During divergent goal setting, the user is encouraged to brainstorm possible solutions, generate multiple accounts, and test analogies from interdisciplinary sources that may arrive at a short list of possible scenarios.

Constancy is another critical part of attentional control. The constancy of a sound describes its duration and is either continuous or discontinuous with the visual event. A continuous sound structure has uninterrupted sound at regular intervals throughout the sequence or entire program. Continuous sound requires a specific purpose or function for the visual event—temporal, POV, locale, atmosphere, or character.

Density is another critical part of attentional control. The density of a sound describes the periodicity for each chosen sound function in a script or sound with a visual event. When fit with a temporal, POV, locale, atmosphere, or character function, the density of a sound can be massed, spaced, or summarized with the visual event. A spaced, massed, or summarized sound describes when and how often an auditory warning, music or speech is reviewed with a visual event. Problem solving skills can be presented effectively with a spaced sound density. Personal goal areas on video clips or still graphics can be presented all at once and even out of program context using a massed sound density. Corporate applications tend to use a review sound density to reinforce role-modeling techniques. Internet news overdubs use summarized sound density to recap main stories.

For best results, the SSF model should be used with a strong domain or curricular focus, and delayed posttests to assess long-term learning effects. As a job aid or method of designing multimedia instruction, research with the SSF model should also consider the environmental, content, and learner factors that will always (to some degree), affect user learning and performance.

CONCLUSIONS

Computer-aided instruction, which is more often referred to as computer-assisted instruction, has evolved from an add-on to a learn-from technology. Eight conclusions can be drawn from this discussion, one for each design guideline.

1. Whatever works describes the preferences, predispositions, and experiences of the developer in making unfamiliar topics familiar to the user. The fundamental problem with whatever works is its stochastic method of combining sound and visual instruction in a computer.
2. Design-by-type is supported by a well-known taxonomy of instructional software. However, the types are not distinct and are not restricted to computer-assisted instruction.
3. Favorite feature is driven by the need to compare one medium with another on particular tasks. Unfortunately, recent meta-analyses and expert opinion do not support the view that media comparisons are beneficial. Nevertheless, media comparison research can still be found in the literature.
4. Favorite method usually incorporates specific instructional events into the design of the interface, and it is based on the most popular teaching method at the time.
5. Balance the input distinguishes spatial presentations (photos and graphs) from language-like presentations (text and speech), which is based on Paivio's dual coding theory. Long-term educational effectiveness is still untested, however, as studies using this guideline are all impact studies with adults (undergraduate psychology students).
6. Maximum impact seeks to present high-resolution graphics, video, and sound in CAI to maintain sensations in the user's auditory and visual working memories from which they can create coherent episodes to change their long-term memory.
7. Cognitive load, first aims to reduce intrinsic and extraneous load in the user's working memory by maximizing germane load and by keeping a lean CAI design. Again, long-term educational effectiveness is still untested, however, as studies using this guideline are all impact studies with adults (in-service teachers or undergraduate psychology students).
8. Structured sound function model can address the problem of users ignoring or forgetting to read important instructions and feedback presented in text or other visual displays. The SSF model can be applied as a job aid or as a method of designing CAI. Research with the SSF model should consider the environmental, content, and learner factors that will affect learning and performance.

All but one guideline relies for support on impact studies alone, testing users immediately after the treatment. Alternatively, the SSF model is based on studies conducted on attentional memory in adults *and* children measuring their performance over several weeks. The assumption is that young children are different, not simply little adults, and they are not capable of reasoning as an adult until they reach the age of 15 (60). Although space limitations prohibit another section here, active CAI developers and researchers would benefit greatly by exploring the history of computer-assisted instruction, especially the contributions of the great pioneers of programmed instruction and teaching machines to Internet-supported CAI.

ACKNOWLEDGMENT

Special thanks to Visiting Scholar Zhao Yi for her editorial assistance on early versions of this article.

REFERENCES

1. P. Saettler, *The Evolution of American Educational Technology*.Englewood, CO: Libraries Unlimited, Inc., 1990.
2. B. F. Skinner, Teaching machines, in A. Lumsdaine and R. Glaser (eds.), *Teaching Machines and Programmed Learning: A Sourcebook*. Washington, DC: Department of Audiovisual Education, National Education Association, 1960.
3. C. Shannon and W. Weaver, *The Mathematical Theory of Communication*. Urbana, IL: University of Illinois Press, 1949.
4. N. Weiner, *The Human Use of Human Beings: Cybernetics and Society*. New York: Aron Books, 1950.
5. N. A. Crowder, Automatic teaching by intrinsic programming. In A. A. Lumsdaine and R. Glaser (eds.), *Teaching Machines and Programmed Learning: A Source Book*. Washington, D.C.: National Education Association of the United States, 1960, pp. 286–298.
6. J. M. Heines, Technology for teaching: past masters versus present practices, In G. Kearsley (ed.), *Online Learning: Personal Reflections on the Transformation of Education*. Englewood Cliffs, NJ: Educational Technology Publications, 2005.
7. S. M. Alessi and S. R. Trollip, (1991). *Computer-Based Instruction: Methods and Development*, 2nd ed. Englewood Cliffs, NJ: Prentice Hall, 1991.
8. M. J. Hannafin, and K. L. Peck, *The Design, Development and Evaluation of Instructional Software*. New York: MacMillan Publishing Co., 1998.
9. A. Paivio, *Mental Representations: A Dual Coding Approach*. Oxford, UK: Oxford University Press, 1986.
10. R. E. Mayer, Multimedia learning: are we asking the right questions? *Educat. Psychol.* **32**(1): 1–19, 1997.
11. A. D. Baddeley, Is working memory still working? *Euro. Psychol.***7**(2): 85–97, 2002.
12. D. Porter, A report on instructional devices in foreign language teaching, in I. Lumsdaine and R. Glaser (eds.), *Teaching Machines and Programmed Learning: A Sourcebook*. Washington, D.C.: Department of Audiovisual Education, National Education Association, 1960, pp. 186–205.
13. J. Conway, Information presentation, information processing and the sign vehicle. *AV Commun. Rev.***16**(4): 403–414, 1968.
14. W. Buxton, Introduction to this special issue on nonspeech audio. *Hum.-Comp. Interac.***4**: 1–9, 1989.
15. A. Barron and M. Kysilka, The effectiveness of digital audio in computer-based training, *J. Res. Comput. Educat.* **25**(3), 277–289.
16. C. McCormack and D. Jones, *Building a Web-Based Education System*. Toronto: John Wiley and Sons, 1998.
17. L. Porter, *Creating the Virtual Classroom: Distance Learning with the Internet*. Toronto: John Wiley & Sons, 1997.
18. G. L. Wilkinson, L. Bennett, and K. Oliver, Consolidated listing of evaluation criteria and quality indicators, *Educat. Technol.* 1997.
19. Bonk and Cummings, 1998.
20. P. Riley, Designing, developing and implementing WWW-based distance learning, *J. Interact. Devel.* **10**(4): 18–23, 1998.
21. C. Maddux and L. Johnson, The World Wide Web: history, cultural context, and a manual for developers of educational information-based Web sites. *Educat. Technol.* **37**: 5–12, 1997.
22. E. Mikovsky, Techniques for distance learning instruction, *Media and Methods* **34**: 24, 1998.
23. J. Boettcher, and G. P. Cartwright, Designing and supporting courses on the web. *Change* **29**(5): 10, 62–63, 1997.
24. K. King, Course development on the World Wide Web. *New Direct. Adult Contin. Educ.***78**: 25–32, 1998.
25. J. Reynolds, What the instructional designer needs to know about HTML. *Internat. J. Instruct. Media* **25**(2): 161–169, 1998.
26. M. Ohl and M. Cates, Applying Metaphorical Interface Design Principles to the World Wide Web. *Educat. Technol.* 37, 25–38, 1997.
27. S. Adams, B. L. Mann, H. Schulz, Can seventh graders learn fractions from a Web-based pedagogical agent? Using comparison groups three times over several weeks, in B. L. Mann (ed.), *Selected Styles in Web-Based Educational Research.*, Hershey, PA: Idea Group Publishing, 2006, pp. 332–346.
28. R. M. Bernard, P. C. Abrami, Y. Lou, E. Borokhovski, A. Wade, L. Wozney, P. Wallet, M. Fiset, and B. Huang, How does distance education compare with classroom instruction? A meta-analysis of the empirical literature. *Rev. Educ. Res.***74**: 3, 379–439, 2004.
29. C. Koroghlanian, and J. Klein, The effect of audio and animation in multimedia instruction. *J. Educa. Multimed. Hypermed.* **13**(1): 23–46, 2004.
30. B. L. Mann, Evaluation of presentation modalities in a multimedia system, *Comput. Educat. Internat. J.* **28**(2): 133–143, 1997.
31. R. E. Mayer, *Multimedia Learning*. Cambridge, UK: Cambridge University Press, 2001.
32. R. Taylor (ed.), *The Computer in the School: Tutor, Tool, Tutee*. New York: Teacher's College Press, 1980.
33. L. Brader, Tools of the courseware trade: a comparison of ToolBook 1.0 and HyperCard 2.0. *TechTrends*, **35**(5), 10–17, 1990.
34. B. L. Mann, P. Newhouse, J. Pagram, A. Campbell, and H. Schulz, A comparison of temporal speech and text cueing in educational multimedia, *J. Comp.-Assis. Learn.***18**(3): 296–308, 2002.
35. P. Geisert, and M. Futrell, *Teachers, Computers and Curriculum: Microcomputers in the Classroom*, 3rd ed. Boston, MA: Allyn & Bacon, 1999.
36. J. Lockard, and P. Abrams, *Computers for Twenty-First Century Educators* 3rd ed. Boston, MA: Allyn & Bacon, 2000.
37. C. D. Maddux, D. L. Johnson, and J. Willis, *Educational Computing: Learning with Tomorrow's Technologies* 3rd ed. Boston, MA: Allyn & Bacon, 2001.
38. E. Dale, *Audio-Visual Methods in Teaching*. New York: The Dryden Press, 1946. W. Dick, and L. Carey, *The Systematic Design of Instruction* 4th ed. New York: HarperCollins, 1996.
39. R. Heinich, M. Molenda, J. D. Russell, and S. E. Smaldino, *Instructional Media and Technologies for Learning*. Upper Saddle River, NJ: Prentice-Hall, 1999.
40. R. A. Reiser, and R. M. Gagné, *Selecting Media for Instruction*. Englewood Cliffs, NJ: Educational Technology Publications, 1983.
41. R. E. Clark, *Learning from Media: Arguments, Analysis and Evidence*.Greenwich, CT: Information Age Publishing, 2001.
42. B. L. Mann, The evolution of multimedia sound, *Comp. Educa. Internat. J.* **48**(4), 2007.
43. C. Daiute, *Writing and Computers*. Reading, MA: Addison Wesley, 1985.
44. T. Malone, Toward a theory of intrinsically motivating instruction. *Cogn. Sci.***4**: 333–369, 1981.

45. W. Schramm, *Quality Instructional Television.*Honolulu, HI: University of Hawaii Press, 1972.
46. B. Seels, K. Fullerton, L. Berry, and L. J. Horn, Research on learning from television, in D. Jonassen (ed.), *Handbook of Research on Educational Communications and Technology*, 2nd ed. Mahwah, NJ: Lawrence Erlbaum Associates, Publishers, 2004, pp. 262.
47. J. Jones, C. Morales, and G. Knezek, 3-Dimensional online learning environments: examining attitudes toward information technology between learners in internet-based 3-dimensional and face-to-face classroom instruction. *Educ. Media Internat.* **42**(3): 219–236, 2005.
48. J. Steelman, Multimedia makes its mark, *Learn. Lead. Technol.* **33**(1): 16–18, 2005.
49. W. Zhang, K. Perris, and L. Yeung, Online tutorial support in open and distance learning: learners' perceptions. *Brit. J. Educat. Technol.* **36**(5): 789–804, 2005.
50. R. E. Clark, and D. F. Feldon, Five common but questionable principles of multimedia learning, in R. Mayer, (ed.), *Cambridge Handbook of Multimedia Learning*. Cambridge, UK: Cambridge University Press, 2005, pp. 1–23.
51. R. M. Gagné, W. Wager, K. Golas, and J. Keller, *Principles of Instructional Design*, 5th ed. Wadsworth Publishing, 2004.
52. R. E. Mayer, The promise of multimedia learning: using the same instructional design methods across different media, *Learn. Instruc.***13**: 125–139, 2003.
53. U. Goswami, Neuroscience and education, *Bri. J. Educ. Psychol.* **74**: 1–14, 2004.
54. P. Chandler, and J. Sweller, Cognitive load theory and the format of instruction. *Cognit. Instruc.* **8**: 293–332, 1991.
55. J. Sweller, and J. Chandler, Why some material is difficult to learn, *Cognit. Instruc.***12**: 185–233, 1994.
56. J. Sweller, Cognitive load during problem solving, *Cogni. Scie.***12**: 257–285, 1988.
57. J. Sweller, *Instructional Design in Technical Areas*. Melbourne, Australia: ACER Press, 1999.
58. F. Paas, A. Renkl, and J. Sweller, Cognitive load theory and instructional implications of the interactions between information structures and cognitive architecture, *Instruct. Sci.* **32**: 1–8, 2004.
59. J. Sweller, J. van Merrienboer, & F. Paas, Cognitive architecture and instructional design. *Educati. Psychol. Rev.***10**(3): 251–296, 1998.
60. J. Piaget, *The Psychology of the Child*. New York: Basic Books, 2000. (French: Psychologie de L'Enfant. Presses. Paris: Presses de Universitaire de France, 1966).

FURTHER READING

T. Borsook and N. Higginbotham-Wheat, Interactivity: what is it and what can it do for computer-based instruction? *Educa. Technol.* **31**(10): 11–17, 1991.

H. Gardner, (2000) Intelligence reframed: Multiple intelligences for the 21st century. New York: Basic.

M. J. Hannafin, & J. R. Hill, Epistemology and the design of learning environments, in R. A. Reiser and J. V. Dempsey (eds.), *Trends and Issues in Instructional Design and Technology*, 2nd ed. Saddle River, NJ: Merrill/Prentice Hall, 2006, pp. 53–56.

B. L. Mann, The SSF model: structuring the functions of the sound attribute, *Canad. J. Educat. Communicat.* **21**(1): 45–65, 1992.

B. L. Mann, Enhancing educational software with audio: assigning structural and functional attributes from the SSF Model, *B. J. Educat. Technol.* **26**(1): 16–29, 1995.

B. L. Mann, Focusing attention with temporal sound, *J. Res. Comput. Educat.* **27**(4): 402–424, 1995.

B. L. Mann, Shifting attention in multimedia: stochastic roles, design principles and the SSF model, *Innovat. Educa. Train. Internat.* **34**(3): 174–187, 1997b.

B. L. Mann, (2000) Adding digitized speech to web courses, in B. L. Mann (ed.), *Perspectives in Web Course Management*. Toronto, Ontario: Canadian Scholar's Press, 2000, pp. 135–147.

B. L. Mann, Making your own educational materials, Part I. *Internat. J. Instruct. Technol. Dist. Learn.* **10**(2), 2005. Available: http://itdl.org/Journal/Dec_05/article02.htm.

B. L. Mann, Making your own materials, part II: Multimedia design for learning, *Internat. J. Instruct. Technol. Dist. Learn.* **3**(10), 2006. Available: http://www.itdl.org/Journal/Oct_06/article02.htm.

I. Neath, *Human Memory: An Introduction to Research, Data, and Theory*. Pacific Grove, CA: Brooks/Cole, 1998.

G. Parchoma, User-centered instructional design and development: two examples of success, *J. Dist. Educa.***18**(2): 35–60, 2003.

R. Pettersson, Teachers, students and visuals, *J. Vis. Lite.***10**(1): 45–62, 1990.

M. Pressley and C. McCormick, *Advanced Educational Psychology for Educators, Researchers and Policymakers*. New York: Harper Collins, 1995.

R. Ragsdale, *Permissible Computing in Education: Values, Assumptions and Needs*. New York: Praeger Press, 1988.

G. Salomon, Television is "easy" and print is "tough": the differential investment of mental effort in learning as a function of perceptions and attributions, *J. Educat. Psychol.* **76**(4): 647–658, 1984.

G. Salomon, and H. Gardner, Computer as educator: Lessons from television research. *Educat. Research.* (**1**): 13–19, 1986.

P. Suppes, Computer-assisted instruction. Intellectual Autobiography, The Patrick Suppes website, originally written 1978. Stanford, CA: Stanford University, 2007. Available http://www.stanford.edu/~psuppes/autobio21.html.

R. D. Tennyson, D. L. Christensen, and S. I. Park, The Minnesota Adaptive Instructional System: An intelligent CBI system, *J. Compu. Bas. Instruc.***11**: 2–13, 1984.

J. J. G. Van Merriënboer, J. G. Schuurmann, M. B. M. de Croock, and F. Paas, Redirecting users' attention during training: effects on cognitive load, transfer test performance and training efficiency, *Learn. Instruct.* **12**: 11–37, 2002.

BRUCE L. MANN
Memorial University
Newfoundland, Canada