The Journal of Experimental Education, *2002, 70*(3), 215–239

Multimodal Versus Unimodal Instruction in a Complex Learning Context

MARK GELLEVIJ
HANS VAN DER MEIJ
TON DE JONG
JULES PIETERS
University of Twente, The Netherlands

ABSTRACT. Multimodal instruction with text and pictures was compared with unimodal, text-only instruction. More specifically, 44 students used a visual or a textual manual to learn a complex software application. During 2 103–116-min training sessions, cognitive load, and time and ability to recover from errors were measured. After training, the authors tested students' learning on trained and untrained tasks. The results for cognitive load, training time, and learning effects initially supported dual coding theory. The results show that even in this complex situation, multimodal instruction led to a better performance than unimodal instruction. That is, the multimodal manual led to a stronger mental model of the computer program, improved identification of window elements and objects, and speeded up the location of window elements and objects.

Key words: cognitive load, documentation, dual coding, screen captures, usability visualizations

INSTRUCTIONAL MATERIAL in which various media are integrated improves learning more than instruction with only one medium (Kulhavy, Stock, Peterson, Pridemore, & Klein, 1992; Mayer, 1999; Mayer, Moreno, Boir, & Vagge, 1999; Mayer & Sims, 1994; Mayer & Gallini, 1990; Robinson, Robinson, & Katayama, 1999). Mayer and his coauthors have repeatedly shown that users benefit from a multimodal approach, whose most common form is a mixture of words and pictures. In this study, we examined the value of text–picture combinations for a situation in which participants must learn to use a complex computer program through self-instruction.

Address correspondence to Mark Gellevij, Department of Educational Science and Technology, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands. E-mail: gellevij@edte.utwente.nl

The typical presentation platform for such multimodal instruction is a visual manual in which “screen captures” (i.e., displays of the computer screen) are combined with text (Van der Meij & Gellevij, 1998). The context is of research interest because screen captures in a manual also appear on the user’s monitor when they process the instructions. In such a case, previous findings from dual coding theory are challenged by those from cognitive load theory. Whereas dual coding theory predicts that users benefit from a mixture of words and pictures because of the simultaneous processing in two distinct memory systems, cognitive load theory predicts that users are likely to suffer cognitive overload because of the redundancy of the screen captures. In this study, we aimed to examine which of these two theories best fits the learning that takes place in this context. The study was conducted with a real world task: The possible advantage of multimodal over unimodal instruction was tested in a realistic, “on the job” learning context. Another goal of the present study was to validate the functions of screen captures. More specifically, we examined whether adapting the design of screen captures optimizes learning.

Processing Text and Pictures: Dual Coding

According to dual coding theory, more learning occurs when words and pictures are combined because of the way that learners process information in their working memory. Dual coding, or dual processing, presupposes that working memory consists of two distinct systems: a verbal and a nonverbal system. Using the capacity of both systems leads to the processing of more information than using only one of the systems. In addition, it also yields better results because the simultaneous processing connects the two systems. This connectivity in turn contributes to the construction of a strong mental model (Mayer, 1999; Paivio, 1990). That is, it leads to

> a model that evolves in the mind of a user as he or she learns and interacts with a computer system. . . . It represents the structure and internal relationships of a system . . . and is the source of the user’s expectations about the effects of actions; it can guide navigation or planning of actions and contribute to interpretation of feedback. (Jih & Reeves, 1992, p. 45)

This definition was derived from Gentner and Stevens (1983), Norman (1983), and Van der Veer (1989).

Processing Text and Pictures: Cognitive Load

Working memory is limited. When people face a task that is already quite difficult, additional instructions may be more of a burden than a help. The burden from additional information, in short, is the key tenet of cognitive load theory.

The theory may prove to be valuable for the present study because the experimental task is complex. One of the complicating factors is that learning to use a computer program requires the use of more information sources and the handling of more devices than just the instructional material. Users must not only process the manual but also attend to the keyboard, the mouse, and the computer screen. Cognitive load theory indicates that this situation poses two potential risks: redundancy and split attention effects (Chandler & Sweller, 1991; Sweller, 1994; Sweller & Chandler, 1994).

Redundancy occurs when the manual presents screen captures that the user can also see on the computer monitor. The *redundancy hypothesis* predicts that offering the same information twice, as in the case of a depicted computer screen, requires the user to process the information twice. Double processing is redundant and takes up unnecessary memory space that could be used to process other information. According to Sweller and Chandler, redundant information is therefore likely to increase training time and decrease learning.

In a series of experiments to test the redundancy hypothesis, Chandler and Sweller (1991) found that when a picture can be understood by itself, adding explanatory text is redundant and decreases performance. A similar result was found in another experiment by Sweller and Chandler (1994), who compared the use of a text-plus-picture (i.e., screen capture) computer training manual both with and without the use of the computer. In that study, they found that not using the computer led to more learning about the computer program because the information on the computer screen was redundant. That is, the pictures in the text-plus-picture manual already offered the necessary information.

Split attention effects occur when people must attend to multiple sources of information simultaneously. The *split attention hypothesis* predicts that learning is hampered when those sources must be integrated. According to Sweller and Chandler, a user’s learning can be obstructed by split attention effects when a text fragment is needed to understand a picture, and vice versa, because the user has to process two distinct information sources at the same time. Split attention effects occur only when the text and the pictures in the manual are mutually dependent. If both text and pictures can be understood independently, users are likely to opt for only one of the two presentation modes.

For visual manuals, the separation between the information sources—text and pictures—is small. Sweller and Chandler (1994) also examined split attention effects in a situation with a stronger physical separation when they compared a text-plus-picture manual without using a computer with a textual manual with the use of a computer. They found a negative split attention effect in the latter case because the user had to switch attention between the computer and the manual. The manual with text and pictures yielded good results, probably because text and pictures were integrated in the manual, preventing redundancy. They were thus mutually dependent; neither text nor pictures could be understood without the other.

Processing Text and Pictures: Cognitive Load or Dual Coding?

According to cognitive load theory, the use of a multimodal manual, even when text and pictures are integrated, is not preferable to unimodal instruction when the manual and the computer are used at the same time. The theory posits that screen captures are redundant because the same information is already presented on the computer screen, thus increasing the load on working memory, increasing training time, and decreasing learning. In contrast, Mayer and his colleagues (Mayer, Moreno, Boir, & Vagge, 1999; Mayer & Sims, 1994; Mayer & Gallini, 1990) showed that a multimodal approach of text–picture combinations exceeds text-only instruction in its effect on learning. Text and picture information are better integrated in the mind, thanks to their different nature and the way that similar content information is processed separately, which results in a stronger mental model and learning.

This contrast only applies when a large amount of memory is necessary to process the content to be learned. Sweller (1994) divided cognitive load into two types: *intrinsic cognitive load*, caused by the content of the instruction, and *extraneous cognitive load*, caused by the design of the instruction. Intrinsic cognitive load is caused by element interactivity, which is a critical part of cognitive load theory. When elements can be learned in isolation, there is low element interactivity and, thus, low intrinsic cognitive load, which leaves enough memory space for extraneous cognitive load. In such a case, the quality of the instructional design is not critical because there is enough memory space remaining to compensate for any possible flaws. The quality of instructional design matters only when element interactivity is high and the available space for extraneous cognitive load is limited. Reducing extraneous cognitive load is therefore beneficial only when intrinsic cognitive load is high (Sweller & Chandler, 1994).

In the present study, the complexity of the computer program to be learned was high. The content had high element interactivity, causing high intrinsic cognitive load. For example, (a) an action step was meaningless when considered in isolation—only series of (mutually dependent) action steps served as meaningful entities; (b) sections of explanatory texts became meaningful only in combination with the accompanying action steps, and vice versa; and (c) multiple series of action steps together formed meaningful procedures.

Because of the high intrinsic cognitive load of the experimental task and the redundancy of using the computer screen in combination with a visual manual, learning may be hindered. In the next section, we offer suggestions for the use of screen captures in manuals to achieve the maximum reduction in extraneous cognitive load caused by the design of the instruction.

Instructional Functions for Screen Captures

Research on the use of screen captures in software documentation has yielded varied results. In some studies, a multimodal manual has been found to yield better results than a unimodal manual (Sweller & Chandler, 1994; Van der Meij, 1996). Other researchers found no difference or better results for the textual manual compared with one type of visual manual (Gellevij, Van der Meij, de Jong, & Pieters, 1999; Nowaczyk & James, 1993). The difference can perhaps be explained by examining cognitive load theory and dual coding theory. In addition, it may be advantageous to use a dedicated theory on screen captures to supplement more general views on memory processing. In the next section, we examine the variations between studies and suggest how screen captures can best be presented in manuals.

To date, only a few experiments on screen captures have been conducted. Besides our own research (Gellevij et al., 1999; Van der Meij, 1996; Van der Meij, 2000), we found only two other studies (Nowaczyk & James, 1993; Sweller & Chandler, 1994). These studies differ on a number of critical factors. First, the designs of the screen captures vary tremendously. For example, the full range of screen possibilities—pictures of the full screen, of partial screens, and icons—were depicted in the manual. Second, for positioning text and pictures, the text was placed on the left-hand side, on the right-hand side, or to either side of the picture. Third, the type of manual varied; in one study, the instructions were given in the form of a job aid, whereas rather extensive tutorials were used in others. As a result, the magnitude of the task ranged from short and simple to long and complex. Finally, there were additional variations on the experience of the user group (i.e., novice vs. experienced users) and the possibilities of using the application during training.

To regulate research and practice, Van der Meij and Gellevij (1998) have proposed four functions of screen captures. These functions focus on maximizing the user support for selecting, organizing, and integrating information, in line with Mayer's SOI-model of constructivist learning from words and pictures (1999). In the framework of Van der Meij and Gellevij (1998), the main function for screen captures is supporting the development of a mental model. The other functions in the framework are subordinate, contributing in different ways to mental model development. These other functions are switching attention, verifying screen states, and identifying and locating window elements and objects.

Developing a mental model of the program. Each computer program (or series of computer programs) has its own look and feel that influences the development of a mental model. This mental model helps users understand how the program works and allows them to predict what happens when they carry out a certain action. Screen captures can help users develop such a mental model by acquainting them with the interface between their action and the program's corresponding reaction (e.g., showing how windows are arranged and how the interface changes during task execution).

Switching attention. When learning a computer program with a manual, users often fall into a trap known as the *nose-in-the-book syndrome*. Users should be reminded when they should look up from the manual and examine the computer screen. The manual should stimulate users to look back and forth from the manual to the screen regularly. Screen captures can prompt the user to do so. This function is not further detailed in this article because it was not tested empirically.

Verifying screen states. Novice users especially may fear that their actions will damage the computer or the application and therefore are often reluctant to work on a computer unimpeded. The manual can alleviate these fears by presenting information that allows users to verify that they are on the right track. Even for more experienced users, such verification remains important. It provides positive feedback and reinforces motivation when they are on target. In addition, it can help them detect errors early on and thus facilitate error management. Screen captures are optimally suited for this verification process. Comparing a picture of the screen with the actual screen is easier than comparing the content of written text with a screen.

Identifying and locating window elements and objects. Modern interfaces show so much information that users are faced with two problems: They must learn the meaning of specific icons and buttons, and they must acquaint themselves with the position of the icons and buttons on the screen. Screen captures can help solve both problems. They can be used to explain screen elements, and they can help users focus on the relevant part of the screen.

In the current study, we compared visual and textual manuals. The visual manual contains screen captures designed to optimally support three of the four key functions of such pictures (the function of switching attention was not tested in this study). We tested the following predictions:

1. We expected no differences for cognitive load because the visual design of the instructional material is seen as functionally redundant. Thus, although there is overlap between the screen capture and the actual screen, we did not expect this display to impose extraneous cognitive load.

2. We expected users of the visual manual to develop a better mental model than users of the textual manual for two reasons. First, text and pictures can be integrated more easily because they appear on the same medium (i.e., manual). Second, in contrast to monitor displays, the visual manual continuously displays a series of images in sequential order; therefore the images can be studied to help the user build a stronger visual image of the program.

3. We expected users of the visual manual to verify screen states faster and to recover from errors more quickly than users of the textual manual because the

displayed screens in the manual can be compared with the computer screen more easily.

4. We expected users of the visual manual to perform better at locating and identifying screen elements and to complete training faster because of the visual support provided by the screen captures.

Method

Participants

Forty-four teacher education students (29 males and 15 females) participated in the study. These students were selected because of their relatively high level of knowledge of physics (the basis for the study), their intermediate to high level of general computer knowledge, and their low level of domain-specific knowledge of the computer program to be learned. We saw the presence of a minimum level of domain-specific prior knowledge of the program as a necessary condition for users to benefit from text–picture combinations in instruction (Mayer & Gallini, 1990; Mayer & Sims, 1994). The study was presented as an introductory course that the participants could take as an extracurricular activity. The participants were randomly assigned to experimental conditions.

Materials

Computers. The sessions were held in three different computer rooms with 16 to 20 IBM compatible computers in each room. During the experiment, the participants' actions with the computer program were automatically logged.

SimQuest and motion application. The computer program SimQuest (de Jong et al., 1998) consists of two parts: learning (application) and authoring. SimQuest applications offer users a learning experience in which they can explore a specific domain and discover elements within that domain. Their discovery behavior is evoked through simulations. Learners receive assignments and explanations to support the discovery process in forms such as video, text, pictures, and audio.

Teachers and instructional designers use the SimQuest authoring environment to modify an existing learning environment or create a new one. SimQuest is based on an object-oriented approach, which means that a collection of provided elements can be used to create an application or program. In the reported study, the participants learned to work with the SimQuest authoring environment.

The participants were expected to be interested in learning how to use the SimQuest software. As future teachers, they were informed about the software as a new form of instructional material. As future designers of their own lessons, they were informed about the possibilities of modifying and creating learning environments through SimQuest's authoring environment.

We chose a particular SimQuest application to demonstrate the use, modification, and creation of a discovery learning environment that dealt with the physical issue of motion. In the program, the students could explore the relationships among initial velocity, velocity at a certain point in time, and acceleration. The participants were shown various simulations with moving motorcycles, trains, cars, scooters, and other objects. Assignments allowed the students to check the correctness of relations discovered between variables. Video and textual explanations introduced and discussed the variables in the simulations and assignments.

The manual focused on the basic tasks in handling SimQuest and discussed the modification and creation of interfaces and assignments.

Manuals. The way screen captures are designed has an important influence on its potential effects because of the way function and form interact. The following designs were used in the study.

The designs used in the study for developing a mental model were based on two major features of illustrations known to support learners in building a mental model as described by Mayer and Gallini (1990). These two features are *system topology* and *component behavior*.

System topology means that the important elements of the screen are shown and explained within their immediate context. In this case the important elements are windows, parts of windows, icons, and buttons. The immediate context is a screen capture of the complete computer screen. System topology is mainly applied in explanatory sections of the manual (see Figure 1).

Component behavior means showing “each major state that each component can be in and the relation between a state change in one component and state changes in other components” (Mayer & Gallini, 1990, p. 715). Applied to the design of screen captures in the procedural sections of the manual, this means showing how the computer screen changes as a result of carrying out various actions of the computer program (see Figure 2).

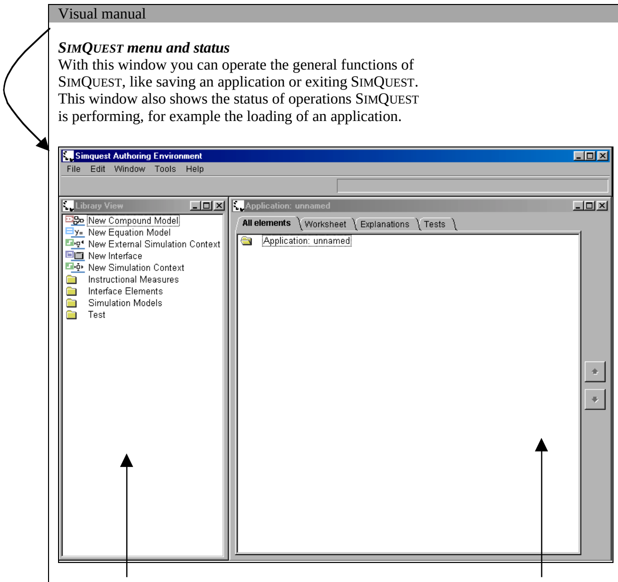
The following were the design arguments for verifying screen states: (a) providing a size that is legible to make sure that the specific content can be easily verified, (b) showing only the relevant parts of the screen that help focus attention, and (c) using, if necessary, a cueing technique to point to the essential verification element (see Figure 3). Note that the layout is basically the same as for developing a mental model. The flow of steps and accompanying screen captures stays intact to show component behavior and to keep the design consistent throughout the manual. The essential difference is that only the relevant window is shown—not the full screen—to improve legibility.

The design for identifying and locating window elements and objects was mainly based on considerations of system topology and component behavior. In addition, we used cueing to point to essential parts of the screen, and used callouts to explain those parts. For locating, providing context is important to find

FIGURE 1. Screen capture design for explanatory sections in the visual and textual manuals.

Visual manual

SimQuest menu and status
With this window you can operate the general functions of SimQuest, like saving an application or exiting SimQuest. This window also shows the status of operations SimQuest is performing, for example the loading of an application.



Library
The library window contains standard elements you can use to create a new application.

Application
The application window shows the elements that you have put into your application. The application window also shows the structure of your application.

Textual manual

SimQuest menu and status (window on top)
With this window you can operate the general functions of SimQuest, like saving an application or exiting SimQuest. This window also shows the status of operations SimQuest is performing, for example the loading of an application.

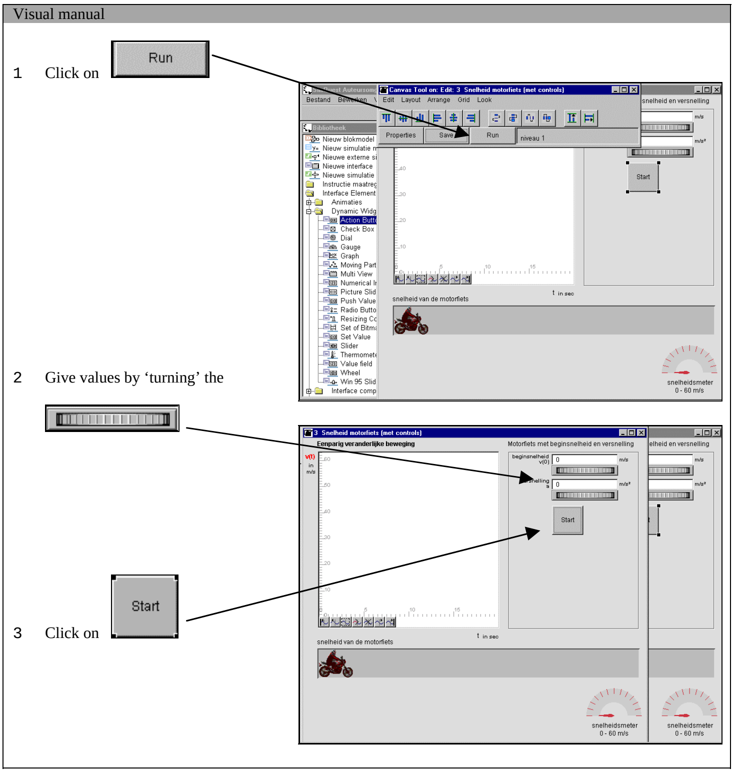
Library (window bottom left)
The library window contains standard elements you can use to create a new application.

Application (window bottom right)
The application window shows the elements that you have put into your application. The application window also shows the structure of your application.

the relevant part of the screen quickly. For identifying, context is less important. Locating and identifying window elements through screen captures can take place in both procedures and explanations. Figure 2 shows an example of the design for a procedure. Figure 1 shows the design in an explanatory section. As previously noted, these designs are the same as for mental model development. These screen captures thus serve more than a single function.

The two manuals (textual and visual) were written in Dutch and consisted of three chapters:

FIGURE 2. Screen capture design for procedural sections in the visual and textual manuals.



Visual manual

1 Click on

2 Give values by ‘turning’ the

3 Click on

Textual manual

1 Click on **Run** in the window *Canvas Tool*

2 Give values by ‘turning’ the **Wheel**

3 Click on **Start**

Chapter 1: Introduction. Basic operations like starting SimQuest, loading an application, and handling the different windows are addressed, and the basic structure and components of the system are explained.

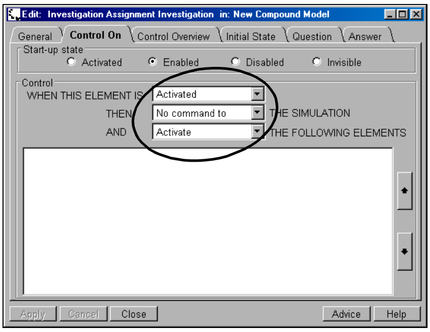
Chapter 2: Modifying and Creating Interfaces. An interface for presenting and manipulating the values of the model variables is created.

Chapter 3: Modifying and Creating Assignments. Variants for promoting and practicing with discovery behavior are explained.

FIGURE 3. Screen capture design for verifying screen state in the visual and textual manuals.

Visual manual

1 Check if the **Control** is specified as in the picture on the right. If this is not the case, you should change it.



Edit: Investigation Assignment Investigation in: New Compound Model

Start-up state
○ Activated ● Enabled ○ Disabled ○ Invisible

Control
WHEN THIS ELEMENT IS Activated
THEN No command to THE SIMULATION
AND Activate THE FOLLOWING ELEMENTS

Textual manual

1 Check if the **Control** is specified as: Activated, No command to, Activate is not the case, you should change it.

Chapters 1 and 2 of the visual manual were designed to support the development of a mental model and to assist the user in locating and identifying window elements and objects. The difference between the textual and visual manual was the absence of screen captures in the textual manual. Chapter 3 of the visual manual was designed to support screen verification. The difference between the visual and textual manuals remained the same as the previous chapters. We made the layout and typography of the visual and textual manuals as similar as possible to avoid differences in appearance and to make sure that any differences could be attributed to the screen captures. The presence of screen captures obviously led to manuals of different sizes. Table 1 contains the number of pages in each chapter of the manual. The number of screen captures used are presented in Table 2. The number of screen capture by screen capture type are presented in Table 3.

Questionnaires and tests. The participants received a paper questionnaire with general questions about their name, age, topic of study, grade level, and 11 questions about their computer experience. For this purpose, items were taken from the Computer Self-Efficacy Scale questionnaire (Murphy, Coover, & Owen, 1989) and translated into Dutch. The questionnaire uses a 5-point Likert-type scale. The following are some examples of items used: “I feel confident understanding terms/words relating to computer software,” “I feel confident learning a variety of computer programs,” and “I feel confident explaining why a program will or will not work on a given computer.”

While working with the manual, the participants encountered some pages that asked them to fill in the current time (which was displayed on the computer screen) and to rank their current cognitive load. Training time is a general measure for how easily the information in a manual can be processed (assuming that

TABLE 1
Number of Manual Pages in the Textual and Visual Manuals

Variable	Textual manual	Visual manual
All chapters	66	96
Chapter 1	13	22
Chapter 2	22	39
Chapter 3	31	35

TABLE 2
Number of Screen Captures in the Textual and Visual Manuals

Variable	Textual manual	Visual manual
Chapter 1		
Mental model, identify, and locate		
Explanation (see Figure 1)	0	4
Procedures (see Figure 2)	0	22
Verify screen state (see Figure 3)	0	0
Chapter 2		
Mental model, identify, and locate		
Explanation (see Figure 1)	2[a]	3
Procedures (see Figure 2)	0	49
Verify screen state (see Figure 3)	3[b]	3[b]
Chapter 3		
Mental model, identify, and locate		
Explanation (see Figure 1)	0	1[c]
Procedures (see Figure 2)	0	0
Verify screen state (see Figure 3)	0	30

Note. Screen capture designs in procedures for mental model, identify, and locate consist of a full-screen capture plus one or more captures of icons, buttons, or objects (see Figure 2).

[a] The design of the interface to be created by the students is very difficult to describe. It was considered necessary to show this with screen captures, even in the textual manual. [b] Creating a new interface was a complex task. To keep users on the right track, it was considered necessary to provide feedback in the form of verification information, although this not the intention in this chapter. [c] The unintended presence of this screen capture was caused by a copying error. It is considered to have had no influence on the results.

TABLE 3
Number of Screen Captures, by Type, in the Textual and Visual Manuals

Screen capture section/type	Manual Textual	 Visual
Chapter 1		
Procedure (32 steps)		
Full	0	22
Window	0	0
Icon/button/object	0	26
Explanation		
Full	0	3
Window	0	3
Icon/button/object	0	0
Chapter 2		
Procedure (32 steps)		
Full	0	49
Window	3[a]	3
Icon/button/object	0	78
Explanation		
Full	0	0
Window	2[a]	3
Icon/button/object	0	0
Chapter 3		
Procedure (32 steps)		
Full	0	3
Window	0	27
Icon/button/object	0	0
Explanation		
Full	0	0
Window	0	1
Icon/button/object	0	0

[a] The design of the interface to be created by the students is difficult to describe. It was considered necessary to show this with screen captures, even in the textual manual.

learning will not be hindered by reduced training time). This ease of processing results from coordination between the manual and the computer screen. More specifically, the support given by the manual for locating a screen object or screen element directly affects the time needed to complete the instruction.

To measure their cognitive load, we asked the participants, “How difficult would you rate the task at this moment?” Their answers could range from “extremely difficult” to “extremely easy” on a 9-point Likert-type scale. Paas, Van Merriënboer, and Adam (1994) have shown this to be a reliable and sensitive way to measure cognitive load. They compared this subjective measure with an objective measure, namely, heart rate variability. With heart rate variability, they were only able to show differences between mentally active and inactive periods. For studies such as this, measurement of cognitive load must be sensitive enough to register relatively small changes in mental effort. In two studies, Paas, Van Merriënboer, and Adam (1994) proved that the 9-point rating scale technique could detect "differences in type of practice problem and variability in the training conditions, the effects of the training condition on transfer performance, and two levels of task complexity" (Paas et al., 1994, p. 428). As an alternative, a secondary task measure could have been used. A drawback of this measure is interference with task execution. In the present study, such interference was relatively small, because the measures of time and cognitive load were taken directly after rounded-off action steps or tasks. In chapter 1, time and cognitive load were measured four times; chapter 2 had seven such measurements; and chapter 3 contained eight.

After each chapter, the participants received a paper test. This test was designed to measure their mental model and their ability to identify and locate window elements and objects. The distinction between mental model development and identifying and locating skills is functional but artificial. It helps to design and formulate instructional materials because one can present information exactly in accordance with its primary goal or function. It is artificial to measure effects of such functions. Developing a mental model of the program always involves both identifying and locating. We asked users to name or draw windows and screens, to predict successive screens, and to point out errors in screen displays to test their mental model development. In addition, we asked users to name specific elements; to describe the meaning or function of icons, buttons, and screen parts; and to determine the correct place of elements in order to test their identification and localization capabilities.

The items in these tests had different modes for responding. Some items asked the participants to write the answer in text, whereas other items asked them to draw the answer in a picture. We balanced this mode for the whole test to eliminate encoding differences (textually or visually) as a disturbing factor. Furthermore, the test consisted of items that measured trained tasks (i.e., exercises that were the same as those practiced during training) and items that measured untrained tasks (i.e., new tasks that were not practiced or discussed in the manual).

Effects of screen captures on screen verification were measured as follows: (a) the time used for verification, according to the log-registration; (b) the number of errors made; and (c) the rate of error recovery. To measure such error verification, the manual contained three intentional errors, in which the information that the user must verify in the screen capture text deliberately differs from the correct situation. An example of an intentional error is given in Figure 3. Instead of "Activated, No command to, Activate" as shown in the manual, the screen displays "Activated, No command to, Close."

Procedure

The experiment was conducted in the school for teacher education. Each participant attended two sessions that lasted approximately 2 hr each. Sessions were held in 2 successive weeks, scheduled directly before or after the regular classes.

The participants were seated in front of a computer where they found the short introduction sheet, the paper questionnaire, and the first chapter of the manual. The introduction sheet told the participants that they should learn to work individually and independently with SimQuest, using the manual as their only source of information. They were allowed to ask questions only when they encountered problems that they could not solve with the manual. The introduction sheet also stressed the importance of filling in the time and the cognitive load sheets in the manuals.

When the participants finished chapter 1, they took the test for chapter 1, which included the instruction to "try to perform as best as they possibly can, even when some questions concern issues that they have not recently practiced with the manual." The participants had no access to the manual or the computer during the test phase.

After the participants finished the first test, chapter 2 of the manual was provided, again followed by a test. After finishing this test, the participants were asked to return for the second session. The second session was set up in the same way as the first session. The participants received chapter 3 of the manual and took the corresponding test after finishing the chapter. After completing the experiment, the participants received the original SimQuest software and example applications, together with a visual SimQuest manual.

Coding and Scoring

Number of participants in the analyses. All 44 participants filled in the questionnaire on computer experience and the other general topics. Also, all participants worked on the first chapter of the manual. Four participants (2 from each condition) missed chapter 2 of the manual because of obligatory classes. Four other participants (2 from each condition) missed chapter 3 of the manual, some because of illness and others again because of curricular obligations. Consequently, in each condition for chapter 1, the data from 22 participants was used in the analyses. There were data from 20 participants in each condition for chapters 2 and 3. There were data from 18 participants in each condition when adding the results of all three chapters.

Computer experience, age, gender, topic of study, and grade level. We calculated means of the scores on the computer experience questionnaire to determine the participants' level of computer experience. Data on age, gender, topic of study and grade level were used in the analyses untransformed.

Cognitive load. For chapter 1, we used all four cognitive load measurements to test for differences between the two conditions. The first and last measurements from chapters 2 and 3 (which were, unlike in chapter 1, taken before starting the task) were removed. We removed the last cognitive load measurement because part of the task prior to the cognitive load measure required interference from the experimenter. More specifically, the last step of the task was saving the work. Because saving the work appeared to be technically problematic, the experimenter intervened, leading to an unreliable last measurement of the participant's cognitive load. Consequently, for chapters 2 and 3, the number of cognitive load measurements used in the statistical analyses was five and six, respectively.

Training time on manual chapters. Timesheets provided information on the training time that was used for each of the various chapters. For chapter 1, the completion time was subtracted from the starting time. For chapters 2 and 3, the penultimate time measurement was subtracted from the starting time.

One user of the visual manual did not record the final training time measurement from chapter 2. Consequently, we used the data from only 19 participants for the visual condition when analyzing the training time from that chapter.

Verification. The time taken by the participants to verify whether the screen showed the same information as was in the manual was extracted from data from chapter 3. The time of the first identifiable step before the verification was subtracted from the time of the first identifiable step after verification. Verifying behavior was triggered in 17 instances, which were summed to create a total time for verification.

We discovered some problems that made it difficult to be sure of the participant's behavior when determining their total verification time. Only data that gave a clear view of the users' behavior was used in the time analyses, resulting in data for 10 participants for the text version and data for 16 participants for the visual version of the manual.

As previously noted, the verification information presented in the manual intentionally did not address the same information as was on the screen in three cases. We put these errors in the manual to see whether verification information was indeed used and to see whether the participants using the visual condition were better in rectifying these errors. Rectifying the errors involved changing settings and adding elements. Changing to correct settings and choosing the right elements to add were scored as one point each.

Learning effects. Each correct answer from the test after each chapter counted as one point. Examples of correct answers include (a) describing a single step, (b) drawing a screen part, (c) pointing to a specific error, (d) explaining an object, and (e) naming a location. Answers varied in number of answer-elements. Therefore, the maximum number of points for each question differed. Table 4 contains the number of questions used in the tests and the maximum number of points.

Results

Check for Randomization

We used analysis of variance (ANOVA) to examine the random distribution of participants over the two conditions, comparing computer experience and age as the variables. Chi-square tests were used on the covariates gender, major (physics, mathematics, biology), and grade level. Results of these analyses showed random distribution on all variables.

Furthermore, we examined whether the distribution of the participants' computer experience was comparable over conditions. Results showed that both groups had variance in computer experience, which means that users with different levels of computer experience participated in the study. The results also showed that the distribution in both groups was comparable (Mann Whitney $U = 227$, $Z = -0.353$, $p = .724$) and was indeed random over the two conditions.

Cognitive Load

Cognitive load theory predicts that the visual manual would impose a higher cognitive load on the user because of the redundancy of the screen captures. Earlier, we discussed arguments that led us to predict that there would be no difference in cognitive load for the two manuals.

TABLE 4
Number of Questions and Maximum Number of Points in the Tests

	No. of questions		Maximum no. of points	
Chapter/action	Trained	Untrained	Trained	Untrained
Chapter 1				
Mental model	4	1	26	4
Locate	6	3	6	3
Identify	6	3	6	3
Chapter 2				
Mental model	5	3	22	14
Locate	7	4	8	4
Identify	10	4	14	4
Chapter 3				
Mental model	4	2	16	5
Locate	2	3	2	3
Identify	9	3	9	3

A one-way ANOVA showed no differences between conditions on the cognitive load of users while working with the manual (see Table 5). Multivariate repeated measures analyses pointed to an interaction between cognitive load and condition in chapter 1, $F(3, 32) = 6.081$, $p = .002$. Figure 4 shows that, contrary to expectations from cognitive load theory, the users of the visual manual had a lower cognitive load. Such interactions were absent for chapters 2 and 3.

Training Time

Comparing training time for the textual and visual manuals resulted in a statistically significant difference in favor of the visual manual (see Table 6). Users of the visual manual completed their training 11% faster than did users of the textual manual. The effect sizes for training time in this study ranged between 0.74 and 0.88. An effect size of 0.25 is generally seen as a small effect, 0.50 as a medium-sized effect, and 0.75 as a large effect (Cohen, 1962); therefore, the effect sizes for training time can be considered fairly high, which is in accordance with Peecks' (1993) view that that the expected effects of presenting screen captures in manuals are mediocre to high.

Verification

We used two measures to determine whether the screen captures meant to support the process of verifying screen states had their intended effects. One measure was the time used for verification. The other measure related to the detection and correction of the aforementioned intentional errors.

TABLE 5
Means and Standard Deviations for Cognitive Load

Chapter/type of manual	*M*	*SD*
All chapters		
Textual	3.30	1.02
Visual	3.65	0.57
Chapter 1		
Textual	3.33	1.22
Visual	3.09	1.10
Chapter 2		
Textual	3.50	1.04
Visual	3.79	0.70
Chapter 3		
Textual	3.37	1.04
Visual	3.73	0.92

Note. The theoretical range for cognitive load ranged from 0 (*very easy*) to 9 (*very difficult*). The actual range used was 0 to 6.

FIGURE 4. Cognitive load over time.

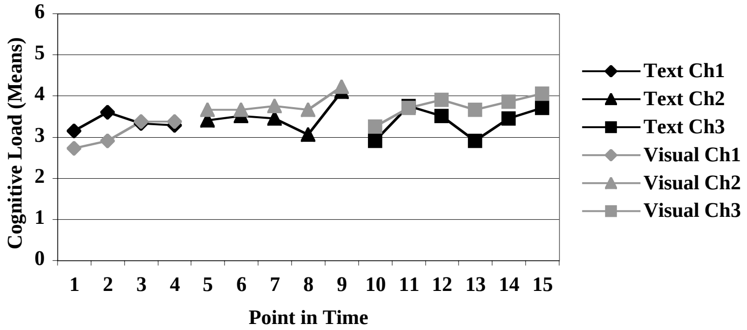


TABLE 6
Means and Standard Deviations for Training Time (Min)

Variable	*M*	*SD*
All chapters		
Textual	115.61	17.26
Visual	**102.83**	15.61[a]
Chapter 1		
Textual	23.23	3.78
Visual	**19.91**	3.80[b]
Chapter 2		
Textual	41.15	6.56
Visual	**35.58**	7.27[c]
Chapter 3		
Textual	52.45	11.69
Visual	48.10	7.09

Note. Statistically significant results are printed in boldface. $ES = (M_2 - M_1)/SD$. [a]$F(1, 34) = 5.425$, $p = .026$; $MSE = 270.85$; $ES = 0.74$; $\eta^2 = .14$. [b]$F(1, 42) = 8.426$, $p = .006$; $MSE = 14.73$; $ES = 0.88$; $\eta^2 = .17$. [c]$F(1, 37) = 6.324$, $p = .016$; $MSE = 47.82$; $ES = 0.85$; $\eta^2 = .15$.

Table 7 contains the time used for verification. Total time can be divided by time used for “normal” verification steps and by verification of screen captures that were deliberately wrong and thus required correcting the error. One-way ANOVAs showed no difference between manuals.

Table 8 contains the scores for making the corrections after comparing the verification information in the manual with the information on the screen. A one-way ANOVA revealed no differences in these scores, likely because of ceiling effects. Almost all (88 to 97%) of the scores were correct, leaving little room for statistically significant differences.

Learning Effects

In this section, we report the influence of screen captures on the outcomes of the instruction. These learning effects were measured with tests that determined the users' mental model of the program, their capability to identify, and their ability to locate window elements and objects.

A one-way ANOVA on the total test scores showed a statistically significant overall effect on trained tasks (see Table 9). Users of the visual manual scored 14% higher than users of the textual manual. A one-way ANOVA for the specific functions showed that users of the visual manual scored significantly higher on questions that measured the strength of their mental model and were significantly better at identifying window elements and objects than users of the text manual. Users of the visual manual therefore (a) gained superior knowledge of the structure of the program, (b) became more capable of explaining how the program works, (c) became better at predicting what will happen when carrying out certain actions, and (d) gained more conceptual knowledge of the program than users of the textual manual. The effect sizes for learning effects ranged between

TABLE 7
Means and Standard Deviations for Time Used on Verification (s)

Measure	*M*	*SD*
Time total		
Textual (n = 10)	1,275	323
Visual (n = 16)	1,213	226
Time		
Textual	97	66
Visual	74	23
Time errors		
Textual	1,178	291
Visual	1,140	217

TABLE 8
Means and Standard Deviations on Correcting Errors (Range = 0–3)

Measure	*M*	*SD*
Elements		
Textual	2.85	0.37
Visual	2.90	0.31
Settings		
Textual	2.65	0.59
Visual	2.85	0.49

TABLE 9
Means, Standard Deviations, and Percentage of Maximum Score for Learning Effect Test Scores for Trained Tasks

	M		*SD*		% of max. score	
Chapter	Visual	Textual	Visual	Textual	Visual	Textual
			Total test			
All chapters	**61.83**	54.06	8.93	11.22	57[a]	50
Chapter 1	23.00	20.64	4.30	5.03	61	54
Chapter 2	21.60	20.20	5.97	5.80	50	47
Chapter 3	**15.95**	13.05	3.47	3.27	59[d]	48
			Mental model			
All chapters	**36.17**	31.17	4.55	7.90	57[b]	49
Chapter 1	13.82	12.09	3.61	4.28	53	47
Chapter 2	13.05	11.75	2.31	3.48	59	53
Chapter 3	**8.75**	7.05	1.94	2.52	55[e]	44
			Locate			
All chapters	10.33	10.00	1.97	2.47	65	63
Chapter 1	5.32	5.23	0.65	0.87	89	87
Chapter 2	3.55	3.85	1.96	1.76	44	48
Chapter 3	1.15	1.05	0.37	0.51	58	53
			Identify			
All chapters	**15.33**	12.89	4.19	2.89	53[c]	44
Chapter 1	3.86	3.32	0.89	1.04	64	55
Chapter 2	5.00	4.60	2.73	1.70	36	33
Chapter 3	6.05	4.95	2.24	1.47	67	55

Note. Statistically significant results are printed in boldface. $ES = (M_2 - M_1)/SD$.
[a] $F(1,34) = 5.299$, $p = 0.028$; $MSE = 102.75$; $ES = 0.69$; $\eta^2 = 0.14$. [b] $F(1,34) = 5.414$, $p = 0.026$; $MSE = 41.56$; $ES = 0.63$; $\eta^2 = 0.14$. [c] $F(1,34) = 4.158$, $p = 0.049$; $MSE = 12.94$; $ES = 0.84$; $\eta^2 = 0.11$. [d] $F(1,38) = 7.399$, $p = 0.010$; $MSE = 11.37$; $ES = 0.89$; $\eta^2 = 0.16$. [e] $F(1,38) = 5.699$, $p = 0.022$; $MSE = 5.07$; $ES = 0.67$; $\eta^2 = 0.13$.

0.63 and 0.89. Similar to the results for training time, pursuant to Peeck (1993), these effects show a large potential for the visual design.

A one-way ANOVA showed no statistically significant difference on correctly locating window elements and objects. In contrast, as previously shown, there was a statistically significant effect of training time on locating favoring the visual manual.

Discussion

In this study, we aimed to address three main questions: (a) Which of two contradictory theories on information processing—dual coding or cognitive load—

best explains how information from multimodal instruction with text–picture combinations is processed; (b) Does the superiority of the multimodal approach also hold in the case of realistic, "on the job" self-instruction; and (c) Does a framework of functional screen captures in software manuals contribute to higher and more efficient learning? We discuss these issues in reverse order.

The framework of the visual manual's development has contributed considerably to its success. The manual was developed from an instructional, functional perspective that featured a dedicated theory on how to optimize the presence of screen captures in a software manual. Tests on the effects of screen captures yielded convincing evidence favoring the visual manual.

For the construction of a mental model and identifying window elements and objects, we found that the users of the visual manual learned more than the users of a textual manual. Therefore, we concluded that the study satisfies the four conditions for effective illustrations as proposed by Mayer and Gallini (1990, p. 716): Our study had explanative text ("The text must present a cause-and-effect system that allows for qualitative reasoning"); explanative illustrations ("Illustrations must help the learner build a runnable mental model of the system"); sensitive tests ("The performance measure must evaluate the learner's understanding and qualitative reasoning about the system"); and inexperienced learners who needed support ("The students must not spontaneously engage in active earning processes such as the construction of a runnable mental model of the system"). In this study, the explanative illustrations were screen captures. The presence of a statistically significant difference in the test scores for chapter 3, and not in chapters 1 and 2, seems surprising in this respect (see Table 9). The explanation may be that chapters 1 and 2 of the visual manual may have had a positive effect on the way users processed the information in chapter 3, which was meant to primarily to support verification. In other words, there may have been a delay in the creation of a mental model. Over time, the added effects of the visual manual became visible. Interestingly, this effect occurred with a delay in the processing of the manual, because there was a weeklong gap between processing the first two chapters of the manual and processing the third chapter. Although the specific effects of the illustrations could not always be proven directly, they did yield their intended effect, albeit delayed. This nicely accords with the observation that the positive effect of pictures in instruction may grow over time (Peeck, 1974).

The results for training time clearly show that locating window elements and objects was facilitated by the visual manual. The primary factor for reduced training time with the visual manual seems to be the process of locating, because only in chapters 1 and 2 was the difference statistically significant. In these chapters, the screen captures were specifically meant to support locating window elements and objects, which was not the case in chapter 3 (see Table 6). Screen captures for verification should also ease task execution. However, this was not found for chapter 3, in which the screen captures were designed to support exactly this purpose. Two reasons may explain why no effect on training time was found for this chapter.

First, there were fewer screen captures in chapter 3. Chapter 1, with 22 pages in the visual manual, had an average of 2.36 pictures per page (the textual manual consisted of 13 pages). Chapter 2 of the visual manual contained 39 pages with an average of 3.41 pictures per page (the textual manual consisted of 22 pages). Chapter 3, however, contained only 0.88 pictures per page on a total of 35 pages (the textual manual consisted of 31 pages). The sheer abundance of pictures in chapters 1 and 2 may have been beneficial for learning. Indeed, it more than offset the presence of more than twice as many pages in the visual manual. In chapter 3, the presence of pictures was not as dominant, which may not support a multimodal form of encoding.

Second, the substantial difference between locating and verifying may also account for the finding. Locating refers to the process of finding a relevant screen object or element as quickly as possible. Verifying, in contrast, refers to a process of carefully checking the screen for possible errors. When verifying information is clear and easy to use, users are likely to use it, taking time to verify. In such a case, they may not speed up the verification process but rather take a good deal of time verifying whether their screen is correct. When verifying information is less clear and harder to use, users are more likely to skip it and hence gain training time. This process is more likely to occur with the textual manual.

In contrast to the clear differences in the speed of locating elements on the screen, no differences were found in the ability to correctly locate objects on the posttest. These findings seem to be inconsistent, but the explanation is quite straightforward. Both the visual and the textual manual contained extensive support for locating the relevant part of the screen. In the textual manual, the information in the screen capture relevant to locating was well translated into text. This translation was successful, as shown in the finding that users of both manuals were equally successful in correctly locating screen elements. However, users of the visual manual were faster at locating the elements.

Screen captures were not found to support verification. This may have been an effect of the difficult process of measuring verification time. The recorded time intervals were not detailed enough to measure the quick nature of the user's actions. In addition, time may not be the most appropriate measure of this function because the main goal of verifying is to do a thorough job instead of a quick one. Our effort to estimate the verification process by inserting deliberate errors unfortunately failed because the errors were too easily noticed. The test thus lacked the sensitivity that is necessary for finding positive effects of effective illustrations (Mayer & Gallini, 1990). As a consequence, the benefits of presenting effective visual information to help verification could not be demonstrated.

The task used in this experiment differed from those in Mayer's experiments in the larger number of information sources (manual and computer screen) and input devices (keyboard and mouse) used at the same time, and the required interaction between these sources and devices. From the findings on learning effects and training time, one can also conclude for this realistic setting that the multimodal instruction has proven to be superior over a unimodal one. As such, the present study extends the existing findings in support of dual coding theory.

Finally, cognitive load did not seem to be adversely affected by text–picture redundancy. The scores for cognitive load were at an optimal, moderate level, neither low nor high. A low cognitive load will result in boredom, whereas a high cognitive load will frustrate learning. The results contradict the cognitive load theory predictions that users will become overloaded by the introduction of visual information along with pictures on the computer and in the manual. The expected extraneous cognitive load caused by this redundancy did not occur. The visual design of the instruction led to improved learning, despite the redundant instructional setting. We ascribe this abrogation of extraneous cognitive load to the functional use of screen captures. That is, the design of the instruction may have decreased extraneous cognitive load, possibly neutralizing extraneous load caused by redundancy. Dual coding theory gives a valid explanation for the way combinations of text and pictures are processed in this study, as shown by the decrease in training time and the increase in learning found for the visual manual.

In conclusion, we showed that in a realistic context, multimodal instruction leads to better outcomes than unimodal instruction. Training time is shortened, learning is improved, and cognitive load is not altered. In addition to Mayer's integrated design criterion for the use of visuals in instruction, the conditional use of screen captures seems responsible for these findings. Further research is needed to gain more detailed insight on the processing of information in complex multimodal learning. In future research, exploration of several interacting information sources and input devices simultaneously, with the associated consequences on processing in memory, may go beyond the current explanations given by dual coding theory and cognitive load theory. Also, further exploration of screen capture roles could provide for subsequent strengthening of multimodal instruction in realistic contexts.

REFERENCES

Chandler, P., & Sweller, J. (1991). Cognitive load theory and the format of instruction. *Cognition and Instruction, 8,* 293–332.

Cohen, J. (1962). The statistical power of abnormal–social psychological research: A review. *Journal of Abnormal and Social Psychology, 65,* 145–153.

Gellevij, M., Meij, H. van der, Jong, T. de, & Pieters, J. (1999). The effects of screen captures in manuals: A textual and two visual manuals compared. *IEEE Transactions on Professional Communication, 42,* 77–91.

Gentner, D., & Stevens, A. L. (1983). *Mental models.* Hillsdale, NJ: Erlbaum.

Jih, H. J., & Reeves, C. T. (1992). Mental models: A research focus on interactive learning systems. *Educational Technology Research and Development, 40,* 39–54.

Jong, T. de, Joolingen, W. R. van, Swaak, J., Veermans, K., Limbach, R., King, S., & Gureghian, D.

(1998). Self-directed learning in simulation-based discovery environments. *Journal of Computer Assisted Learning, 14,* 235–246.

Kulhavy, R. W., Stock, W. A., Peterson, S. E., Pridemore, D. R., & Klein, J. D. (1992). Using maps to retrieve text: A test of conjoint retention. *Contemporary Educational Psychology, 17,* 56–70.

Mayer, R. E. (1999). Designing instruction for constructivist learning. In C. M. Reigeluth (Ed.), *Instructional-design theories and models Volume II: A new paradigm of instructional theory.* Hillsdale, NJ: Erlbaum.

Mayer, R. E., & Gallini, J. K. (1990). When is an illustration worth ten thousand words? *Journal of Educational Psychology, 82,* 715–726.

Mayer, R. E., Moreno, R., Boir, M., & Vagge, S. (1999). Maximizing constructivist learning from multimedia communications by minimizing cognitive load. *Journal of Educational Psychology, 91,* 638–643.

Mayer, R. E., & Moreno, R. (1998). A split-attention effect in multimedia learning: Evidence for dual processing systems in working memory. *Journal of Educational Psychology, 90,* 312–320.

Mayer, R. E., & Sims, V. K. (1994). For whom is a picture worth a thousand words? Extensions of a dual-coding theory of multimedia learning. *Journal of Educational Psychology, 86,* 389–401.

Murphy, C. A., Coover, D., & Owen, S. V. (1989). Development and validation of the Computer Self-Efficacy scale. *Educational and Psychological Measurement, 49,* 893–899.

Norman, D. A. (1983). Some observations on mental models. In D. Gentner & A. L. Stevens (Eds.), *Mental models.* Hillsdale, NJ: Erlbaum.

Nowaczyk, R. H., & James, E. C. (1993). Applying minimal manual principles for documentation of graphical user interfaces. *Journal of Technical Writing and Communication, 23,* 379–388.

Paivio, A. (1990). *Mental representations: a dual coding approach.* New York: Oxford University Press.

Peeck, J. (1993). Increasing picture effects in learning from illustrated texts. *Learning and Instruction, 3,* 227–238.

Peeck, J. (1974). Retention of pictorial and verbal content of a text with illustrations. *Journal of Educational Psychology, 66,* 880–888.

Robinson, D. H., Robinson, S. L., & Katayama, A. D. (1999). When words are represented in memory like pictures: Evidence for spatial encoding of study materials. *Contemporary Educational Psychology, 24,* 38–54.

Sweller, J. (1994). Cognitive load theory, learning difficulty, and instructional design. *Learning and Instruction, 4,* 295–312.

Sweller, J., & Chandler, P. (1994). Why some material is difficult to learn. *Cognition and Instruction, 12,* 185–233.

Van der Meij, H. (1996). A closer look at visual manuals. *Journal of Technical Writing and Communication, 26,* 371–383.

Van der Meij, H. (2000). The role and design of screen images in software documentation. *Journal of Computer Assisted learning, 16,* 294–306.

Van der Meij, H., & Gellevij, M. (1998). Screen captures in software documentation. *Technical Communication, 45,* 529–543.

Veer, G. V. van der. (1989). Individual differences and the user interface. *Ergonomics, 32,* 1431–1449.