INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.
Animated demonstrations versus text: A comparison of training methods

Cornett, Larry Lee, M.A.
Rice University, 1993
RICE UNIVERSITY

ANIMATED DEMONSTRATIONS VERSUS TEXT:
A COMPARISON OF TRAINING METHODS

by

LARRY L. CORNETT

A THESIS SUBMITTED
IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE

MASTER OF ARTS

APPROVED, THESIS COMMITTEE:

David M. Lane, Associate Professor, Director
Psychology

Kenneth R. Laughery, Professor
Psychology

Robert L. Dipboye, Professor
Psychology

Houston, Texas
April, 1993
ABSTRACT

Animated Demonstrations Versus Text: A Comparison Of Training Methods

by

Larry L. Cornett

The use of animated demonstrations to teach computer software procedures is becoming more common, but previous research has been ambivalent about their value. This study compared animated demonstrations and text instructions to determine if animation is indeed a useful form of instruction. Thirty-three users were trained and tested on 7 HyperCard® tasks during three sessions to assess acquisition, retention, and transfer of procedural skills. During training, users either watched and listened to an animated demonstration, read written text, or had a choice of either. They were then tested on identical, similar, and different HyperCard tasks. Whereas the animation group was slower than the text group during training, their performance steadily improved with the subsequent tests such that they were faster and more accurate one week later. These results suggest that animation may be an effective method of training, if long-term retention of skills is the ultimate goal.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>METHOD</td>
<td>11</td>
</tr>
<tr>
<td>Subjects</td>
<td>11</td>
</tr>
<tr>
<td>Design</td>
<td>11</td>
</tr>
<tr>
<td>Materials</td>
<td>12</td>
</tr>
<tr>
<td>Procedure</td>
<td>16</td>
</tr>
<tr>
<td>RESULTS</td>
<td>17</td>
</tr>
<tr>
<td>Performance for All Tasks and Sessions</td>
<td>18</td>
</tr>
<tr>
<td>Performance for Identical Tasks</td>
<td>21</td>
</tr>
<tr>
<td>Performance for Similar Tasks</td>
<td>23</td>
</tr>
<tr>
<td>Performance for Different Tasks</td>
<td>24</td>
</tr>
<tr>
<td>Analysis of Help Use</td>
<td>25</td>
</tr>
<tr>
<td>Performance on the Transfer Task</td>
<td>25</td>
</tr>
<tr>
<td>Questionnaire Results</td>
<td>27</td>
</tr>
<tr>
<td>Usage of Instructions</td>
<td>27</td>
</tr>
<tr>
<td>DISCUSSION</td>
<td>28</td>
</tr>
<tr>
<td>REFERENCES</td>
<td>35</td>
</tr>
<tr>
<td>APPENDIX A</td>
<td>39</td>
</tr>
<tr>
<td>APPENDIX B</td>
<td>43</td>
</tr>
<tr>
<td>APPENDIX C</td>
<td>48</td>
</tr>
<tr>
<td>APPENDIX D</td>
<td>54</td>
</tr>
<tr>
<td>APPENDIX E</td>
<td>58</td>
</tr>
<tr>
<td>APPENDIX F</td>
<td>61</td>
</tr>
</tbody>
</table>
LIST OF TABLES

TABLE 1. Identical, Similar, and Different Tasks in the Three Sessions ..........13
TABLE 2. Mean help times for test tasks 1, 2, and 3 (sec)..............................26
TABLE 3. Questionnaire results for 7 questions (average responses 5 point scale) .........................................................27
TABLE 4. Average performance of animation group across sessions (sec) ......62
TABLE 5. Average performance of text group across sessions (sec) .............62
TABLE 6. Average performance for all tasks across sessions (sec) ...............62
TABLE 7. Average performance for identical tasks across sessions (sec) .......63
TABLE 8. Average performance for similar tasks across sessions (sec) ........63
TABLE 9. Average performance for different tasks across sessions (sec) ......63
TABLE 10. Average performance for transfer task (110 points possible) ........63
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIGURE 1</td>
<td>Example screen from training program</td>
<td>13</td>
</tr>
<tr>
<td>FIGURE 2</td>
<td>Example screen from transfer task</td>
<td>15</td>
</tr>
<tr>
<td>FIGURE 3</td>
<td>Training group by session interaction (all tasks 1-7)</td>
<td>18</td>
</tr>
<tr>
<td>FIGURE 4</td>
<td>Training group by session interaction (percentage of correct tasks)</td>
<td>19</td>
</tr>
<tr>
<td>FIGURE 5</td>
<td>Boxplot of group performance (all tasks)</td>
<td>20</td>
</tr>
<tr>
<td>FIGURE 6</td>
<td>Training group by session interaction (identical tasks 1-3)</td>
<td>21</td>
</tr>
<tr>
<td>FIGURE 7</td>
<td>Boxplot of group performance (identical tasks)</td>
<td>23</td>
</tr>
<tr>
<td>FIGURE 8</td>
<td>Training group by session interaction (similar tasks 4 &amp; 5)</td>
<td>24</td>
</tr>
<tr>
<td>FIGURE 9</td>
<td>Training group by session interaction (different tasks 6 &amp; 7)</td>
<td>25</td>
</tr>
<tr>
<td>FIGURE 10</td>
<td>Average group scores on transfer task (110 points possible)</td>
<td>26</td>
</tr>
</tbody>
</table>
INTRODUCTION

One area of research that is vital to the understanding of human-computer interaction (HCI) is the study of training. Training in HCI is basically concerned with the process of acquiring the knowledge and skills necessary to operate a new system (i.e., hardware and/or software). Learning to use a new computer application is often a time-consuming and frustrating process. As such, people often try to avoid it because they feel temporarily “incompetent” while they attempt to master the application (Carroll, 1987). The traditional learning process usually entails sitting down with poorly written training and reference manuals, patiently reading through them, and practicing the skills. But, since people prefer to be active learners, they often discard the manuals and jump right into trying the application or ask a colleague for a quick demonstration (Carroll & Rosson, 1987; Mack, Lewis, & Carroll, 1990). Indeed, it has been discovered that users who adopt such an exploratory strategy learn faster and perform better than those who learn by more traditional methods (Kamouri, Kamouri, & Smith, 1986; Robert, 1987; Wendel & Frese, 1987). How should computer-based training be designed in order to optimize the acquisition and retention of procedural knowledge, as well as providing the learning environment that users prefer? Intuitively, it would seem that a training method which more closely approximates the learning process that people prefer might actually encourage users to utilize that method, instead of setting it aside as traditional manuals often are. One training method that seems to hold promise is the technique of using animated demonstrations. The use of animated instructions as a method of training might be one answer to the problems associated with other methods, since they simulate a skilled user demonstrating proper procedures.
Animated instructions differ greatly from traditional training methods that use written text or static images on a computer screen. They are an actual full motion recording of the computer screen as the task is being performed correctly (i.e., as if by an expert) and sound can be incorporated as well. For example, the user can watch the cursor moving on the screen demonstrating the correct sequence of object and menu selections for a particular task and listen to a human voice describing what is being done and why. This is similar to the behavioral modeling training method used by Gist et al. (1988, 1989) to teach computer software. They compared an interactive tutorial method with a modeling method which displayed a human instructor on videotape performing the tasks. They found the modeling to be superior to the tutorial for training people to use the software. The subjects in the modeling condition performed better and scored higher in self-efficacy (i.e., the belief in one's ability to perform a certain task). The authors suggested that modeling enhances learning through the vicarious experience of observing the model because subjects can observe each task being performed correctly and be shown the feedback and end states. It would seem that an animated training method would incorporate many of these advantages, since it displays the task being performed in “real time” by an expert user while a human voice explains the procedures more fully, even though the actual human expert is not visible.

The advent of graphic user interfaces (GUI) (e.g., MS -Windows), although generally easier to learn and use, places new demands on the traditional process of using text to train people to use software applications. Command-based systems can be taught with simple text instructions since the user interacts with the system through text entered in a command line. There is a certain defined sequence of commands for each operation performed with such a system. You do not have to
“show” someone how to enter text and therefore written instructions explaining the commands and sequences are probably sufficient. However, GUI require various operations at different locations on the screen and choices from specific menus. Although GUI are typically easier for new users to learn than command-based systems, it is helpful to demonstrate what the different objects look like, where they are located, etc., since the users are interacting with graphical objects. Written instructions are believed to require an additional referential step of determining the relationship between the verbal concepts and the objects and actions in the interface while animation directly shows these relationships visually. Thus, an animated training method seems more appropriate for learning to use these systems, since users would actually be able to see how to perform a procedure and could observe the immediate results. This process seems to mimic the method by which people usually learn procedures in areas other than human-computer interaction: Visual observation of another person performing the procedure, listening to the accompanying verbal explanations, and attempting to imitate the demonstrated procedure. Animated demonstrations, unlike written instructions, can also incorporate spoken material, which can be used to describe and explain the visual images in more detail (Baggett, 1984). Thus, it would seem that there are several advantages to using animation for training. It is less ambiguous than text instructions. There is no guessing involved: The demonstration shows the user exactly what procedures to use with the appropriate objects and menus, what those objects and menus look like and where they are, as well as displaying the end result of the procedure (i.e., feedback). However, a disadvantage of animation stems from these advantages; the users may be less active learners than they would be with a less dynamic training system and passive learning is not as effective as active learning for long-term
retention of skills (Avner, Moore, & Smith, 1980).

Many software packages today are incorporating animated demonstrations into their tutorial and help systems. These animation programs and files can be quite large in terms of the computer storage space required, but the development of CD ROM technology has made it much more feasible and practical. This trend would seem to indicate that people believe that showing someone how to do something is an effective training method. Macintosh® and HyperCard® (Claris, 1990) both utilize animated tutorials that show users how to get started with the basics of using the mouse, etc. Software programs, such as Microsoft Excel (Microsoft, 1991) and Lotus 1-2-3 (Lotus Development Corp., 1991), are also using animation in help systems to demonstrate how to perform certain procedures. However, although animated instructions are increasing in popularity, there is little evidence that they are a superior training method. Indeed, most research to date has not even shown a superiority of online help and documentation when compared to traditional manuals. Users still tend to prefer and perform better with hardcopy documentation (Krauss, Middendorf, & Willits, 1991).

A study conducted by Mayer and Anderson (1991) found evidence that animation is an effective instructional medium. Mechanically-naive students were taught how a bicycle pump works using various forms of instruction. The group taught with animation and a simultaneous verbal description performed better on creative problem solving tests than groups taught with just animation, just verbal descriptions, or a sequence of the verbal explanation followed by animation. They believe that the animation must be coordinated with verbal information if it is to be an effective instructional tool. However, they did not use a delayed session to assess performance or retention of the information, so it is difficult to determine
from this study if the early benefit of animated and verbal instructions would remain stable over time.

However, Palmiter and Elkerton (1991a) conducted what seems to be the most thorough research in the area of animated instructions to date and their results indicate that animation may not be effective for producing long-term retention of skills. They compared animated demonstrations and written instructions for learning procedural HyperCard authoring tasks. The subjects either watched a demonstration or read written procedures for each task and then performed the task. They were given an immediate test after the training and a delayed test about one week later. Animated demonstrations resulted in faster learning and better performance during training. However, during the later test sessions, the subjects who were trained with written instructions performed faster. The authors suggested that the subjects in the animation condition might simply have been mimicking the demonstrations which resulted in fast initial learning and performance, but this "rote learning" did not result in a deeper encoding of the procedures. These subjects were unable to generalize what they had learned to new tasks and did not retain the procedural information as well as the subjects in the written condition. Perhaps the subjects in the written text condition encoded the procedures in a "richer" manner since they were exposed to verbal (i.e., from reading the text), visual, and motoric codes (i.e., from practice) while the subjects in the animation condition were only exposed to the visual and motoric codes. It is possible that the animation group did not link the visual demonstration to the underlying procedures and thus could not generalize the procedures to new tasks.

A second study by Palmiter and Elkerton (1991b, 1991c) attempted to answer the questions that arose from their first study. Most importantly, would the
"combination of spoken text and animated instructions provide the early benefits seen with the animated demonstrations and the later retention and transfer capabilities found for text" (p. 2)? They compared three instructional methods: Animation only, text only, and an animation-spoken text combination. The spoken text was identical to the written text used in the text only condition. The procedure was similar to that of their first study with the exception of additional tasks, counterbalancing of tasks, and equal instruction presentation times. The results showed that, interestingly, the animation group and the animation-spoken text combination group performed similarly. Thus, simply adding spoken text material was not enough to solve the problems encountered with animated instruction. Once again, the animation groups performed faster and more accurately than the text group during the training session; however, the animation users were slower than the text group in the delayed testing and seemed to still have trouble generalizing procedures to similar tasks. The text group did not improve much between the first training sessions and the later sessions, but they did perform better than the animation groups by the last session. It seems that the slower, and probably deeper, processing of the text users may have resulted in better learning of the procedures and allowed for generalization of learned procedures to new situations. The animation users once again felt as if they were mimicking the task procedures and may have disregarded the spoken text, as it was redundant with the procedures being viewed. Thus, adding the spoken text did not result in better retention and transfer of procedures.

The general problem identified with animated instruction seems to be that users are simply engaging in rote memorization of the procedures. They are mimicking the demonstration and are not encoding the information as thoroughly as
users who were presented with only written instructions. The early benefit of animation of identification and orientation of objects and actions is not sufficient for the retention and generalization of the procedures. Watching an animated demonstration may lure users into believing that they understand the procedure since they can easily mimic it and perform the task correctly the first time. However, the users have not truly learned the procedures. One possible method of reducing the likelihood of mimicry and “shallow” learning would be to design animations that demonstrate the correct procedures, but are not identical to the exact tasks that the users will be performing (e.g., objects named differently and in different locations). Thus, mimicking the animations exactly will result in incorrect performance and users will be forced to apply what they have learned to a slightly different situation.

It is also possible that the animation users in Palmiter and Elkerton’s study were not given enough time to fully understand what they viewed in the demonstrations. The presentation of the demonstrations was timed so that they could only be watched for a set time limit and then they were made unavailable until the users had tried the task, failed, and requested to view the demonstration again. Many users voiced complaints in their questionnaires that the demonstrations were too fast, couldn’t be controlled, and couldn’t be easily viewed again. Thus, it would seem that users should be given more control over the presentation of the animated demonstrations, since it has been found that users need a little more time to become accustomed to and utilize them, because they are a novel instructional medium.

Another possible method of improving animated demonstrations would be to use the auditory channel to present more general, conceptual material which might help the users to process the procedures more deeply, rather than just repeating the written procedural steps as Palmiter and Elkerton did. There are many stud-
ies that have found that providing users with conceptual information helps them build a mental framework (i.e., mental model) which may help them remember procedural information. Bayman and Mayer (1988) taught subjects the BASIC computer programming language using a manual that either emphasized a conceptual model or emphasized syntactics. The subjects who learned the conceptual model had a better understanding of the programming language and performed better on tests that measured problem solving (e.g., debugging programs) than the subjects trained with the more conventional manual. A study by Halasz and Moran (1983) trained subjects in the use of an on-screen stack calculator. All of the subjects were taught the procedures, but one group was also taught a conceptual model that explained the calculator's stack mechanism. The model seemed to provide no advantage on routine problem solving. However, for novel problems, the model group performed much better than the no-model group. A final example of the benefit of conceptual information is provided by Kieras and Bovair's (1984) study. Their subjects were required to learn how to operate a complex device. One group simply learned the operating procedures, while another group was taught the conceptual model of the device before learning the procedures. In general, they found that having a device model improves learning performance and recall of operating procedures, as well as helping users infer forgotten procedures. Thus, adding conceptual verbal information to animated demonstrations, rather than just verbally repeating the procedural steps, may help users acquire and retain the procedures used in the HyperCard tasks.

This study was designed to incorporate the proposed improvements to produce both animated and written text instructions that were as close to optimal as possible. Three instructional conditions for training HyperCard authoring tasks
were compared: Written text, animation plus spoken general text, and a third condition in which the subjects were allowed to choose between the first two types of instructions. The written text was equivalent to what was presented by the animated demonstration. The animation plus spoken text included a voice explaining general HyperCard concepts. For example, the task of changing the border of a field would include the following spoken text: “To view information and settings for any object in HyperCard, select that object and choose Get info for that object”. This concept informs the learners that the procedure for getting information about and changing the settings for fields also applies to other objects (e.g., buttons). The choice condition assessed user preference for the two types of instruction at different points during training.

The animated demonstrations were optimal in that they took advantage of the “additional” auditory channel to present conceptual information and the learners were given greater control over the presentation of the demonstration. They could play the entire demonstration or shorter segments of it as often as necessary. The text instructions were optimal in that they were always available to the users for review throughout the entire training session, unlike P:\lmite and Elkerton’s written instructions which had a timed presentation. Their subjects noted in the questionnaires that they disliked not being able to read the instructions at their own pace and wished that they could have referred back to them. Thus, it was decided that the written instructions for this study should be made more available to the subjects. This is more realistic, since users in the “real world” do not put away the manuals they are using each time they attempt a task. They leave them open to the appropriate page for quick reference. Thus, there was more than one difference between the two training conditions. The method of presenting the instructions was
different (i.e., animation vs. text) and the animated demonstrations had spoken conceptual material added. Animation allows this addition, while text does not, since it would be extremely difficult to read text while trying to listen to a voice explaining additional concepts. HyperCard was chosen as the software for this study due to its good direct manipulation interface and to the wide variety of general concepts involved as well as specific procedural steps for tasks. It is also the software that was used by Palmiter and Elkerton and many of the tasks used in this study are very similar to the tasks they used. This permitted a better comparison of the results of the studies.

The subjects were given a training session during which they paced their own learning; spending more time on some tasks than others if they found it necessary. The instructional material was always available for review. Giving users more control over their training often results in greater user satisfaction and better learning (Fry, 1972; Gay, 1986; Mackay, 1992). Some tasks take longer to learn than others and the users should be able to determine, to some degree, how much time they want to spend learning a certain task. This type of controlled “self-pacing” is also more realistic in the realm of computer training since users do not typically get only one chance to see how a task is performed. However, there was a time constraint on the overall training time so they did have to pace themselves; speeding up if necessary or trying some tasks again. They were given an immediate test following the training session during which the instructional materials were available as help for some tasks. Hopefully, this also helped to discourage simple rote memorization of procedural steps during training since the subjects were not worried about the instructional material being taken away. Exactly one week later the subjects were retested on the same material as immediate test, but this time without the aid
of instructions in order to test retention. After completion of this test, they were required to attempt a more complex task of designing an entire HyperCard stack, without instructions, to measure the generalization of their knowledge to a different, more global task.

METHOD

Subjects

Thirty-three undergraduate students from Rice University (13 males and 20 females) served as subjects in this study. They received research credit for the psychology courses in which they were currently enrolled. The age of the subjects ranged from 17 to 22 years (M = 18.8 years). All were computer literate, but had no previous experience using the HyperCard programming environment. A questionnaire revealed that the subjects had a wide range of previous computer experience ranging from those who simply used computers for word processing to a few who had several years of programming experience in other languages. Subjects were randomly assigned to one of the three training groups.

Design

The study was a three variable (3 x 4 x 7), mixed-factor design. The between-subjects variable was the type of training instructions (animated demonstrations with spoken text, written text, or a choice of either type). The within-subjects variables were the 7 HyperCard tasks and 4 performance assessments. The performance of the subjects was assessed using four measures: (1) the training session while they were learning the tasks; (2) an immediate test after the training to assess skill acquisition; (3) a delayed test one week later to assess retention of skill knowledge; and (4) a transfer task following the delayed test to assess transfer of
training to a more complex task.

Materials

HyperCard Tutorial. A computer tutorial was developed that explained the basic structure and components of the HyperCard programming environment. The tutorial lasted approximately 10 minutes and explained the basic components (e.g., stack, background, card, buttons, fields) and their properties, as well as how to navigate through HyperCard stacks and use tools. A brief test was administered at the end of the tutorial that assessed basic knowledge of the concepts and required a hands-on use of some skills (e.g., selecting tools). A short practice task was also included that gave the subjects a warm-up trial using the menus and painting tools. The skills used in this task were not related to the skills that were taught during the training session.

Training Program. A training program was designed to teach the subjects the 7 original HyperCard tasks, as well as to test them on their skills after the training was completed. Figure 1 is an example of one of the training screens. The task instructions were presented below the training program window. The program allowed the subjects to progress through the tasks at their own pace, but they had only 30 minutes to complete the training, at the end of which they were taken to the immediate test. The program recorded user actions; including accuracy of performance, performance time, and usage of instructions.

HyperCard Tasks. The subjects were taught 7 basic HyperCard tasks during the training session and were tested on 7 tasks during the immediate and delayed tests. Three tasks were identical to the training tasks, two were similar (e.g., create field and create button), and two were completely different (see Table 1). There were also two practice tasks that were completely unrelated to HyperCard that gave
the subjects practice accessing the instructions: Microsoft Word (Microsoft, 1988) and Microsoft Excel (Microsoft, 1991). Subjects were presented with a goal state-

<table>
<thead>
<tr>
<th>Training Tasks</th>
<th>First Test</th>
<th>Second Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resize Field</td>
<td>Resize Field (^1)</td>
<td>Resize Field</td>
</tr>
<tr>
<td>Link Button</td>
<td>Link Button (^1)</td>
<td>Link Button</td>
</tr>
<tr>
<td>Change Field’s Border</td>
<td>Change Field’s Border (^1)</td>
<td>Change Field’s Border</td>
</tr>
<tr>
<td>Create Button</td>
<td>Create Field (^s)</td>
<td>Create Field</td>
</tr>
<tr>
<td>Copy Button</td>
<td>Change Button Style (^s)</td>
<td>Change Button Style</td>
</tr>
<tr>
<td>Create Card</td>
<td>Delete Button (^d)</td>
<td>Delete Button</td>
</tr>
<tr>
<td>Create Stack</td>
<td>Bring Field Closer (^d)</td>
<td>Bring Field Closer</td>
</tr>
</tbody>
</table>

*Note. i = Identical tasks, s = Similar tasks, d = Different tasks*

[Image of training program]

**Goal:** Change the *See Clock* button so that it takes you to the *Clock* card.

Use the instructions, then try the task by pressing the **Start** button. When you are done, press **Done** to get feedback. When you feel ready to move on to the next task, press **Next Task**.

1. Select the **Button** tool under the **Tools** menu.
2. Click on the button that you want to change.
3. Select **Button Info...** under the **Objects** menu.
4. Click on the button named "Link To..." in the Button info dialog box.
5. Click on the **arrow** button named "Go to clock card".
6. Click on the button named "This Card" in the Linking dialog box.
7. Select the **Browse** tool under the **Tools** menu.

**FIGURE 1.** Example screen from training program
ment for each task that explained what was required

**Animated Demonstrations and Written Text.** The animated demonstrations for the two practice tasks and 7 training tasks were created using Camera Man™ (Vision Software, 1992) to record the tasks being performed correctly. Spoken text was coordinated with the animations using Adobe Premier™ (Adobe Systems, 1992). The spoken text was recorded with a female speaker trained in public speaking at a rate of about 120 words/min. and synchronized with the related tasks steps (see Appendix A for the spoken text used). The animations plus spoken text were evaluated by skilled HyperCard users. This involved an iterative process of improving the clarity and timing of the demonstrations and text, as well as ensuring that the text reflected the basic HyperCard concepts associated with each task. The animated demonstrations were always available to the subjects in a window at the bottom of the screen. They were operated by a control stack that allowed users to view the entire demonstration or select shorter segments which were labeled by the procedural steps they demonstrated. For example, the demonstration for the task of resizing a field had the following segment options: Whole movie, Select tool, Select field, and Resize field.

The written text was created in the form of action-object pairs (e.g., select the browse tool) and basically followed the sequence of procedural steps necessary to complete each task (see Appendix B). The written instructions were always available during training at the bottom of the screen. Five experienced Macintosh users (three naive and two skilled HyperCard users) compared the written text instructions and animated demonstrations. Each read the text on a sheet of paper while watching the demonstration and recorded any differences or missing information. Based on these evaluations, several changes were made to make the infor-
mation contained in the two types of instructions more equivalent.

**Transfer Task.** A transfer task was designed that required the practical application of procedural skills and the transfer of that knowledge to a few novel tasks. The task involved replicating a simple HyperCard stack that resembled an address book (see Figure 2). The total points possible for perfect performance was 110 and the subjects were allowed 20 minutes to complete the task.

**Questionnaire.** A four page questionnaire was developed in which subjects provided background information (e.g., age and major), computer experience, and answered several questions assessing their opinions about the training method (see Appendix C).

**Computers.** An accelerated Macintosh II computer was used to produce the animated demonstrations. A Macintosh IIXi computer with a 13 inch color monitor was used to present the training and tests.

![Image of HyperCard stack](image)

**FIGURE 2.** Example screen from transfer task
Procedure

The experiment was conducted on two separate days, each lasting approximately 60 minutes, spaced one week apart. During the first day, the subjects were seated at the computer and given brief verbal instructions by the experimenter about the tutorial (see Appendix D for experimental instructions). They informed the experimenter when they had completed the 10 minute tutorial and were then given a brief test over the concepts and some basic procedures (e.g., select the field tool). If they had difficulty performing the tests, they were retrained and retested. The subjects then completed a short practice task using the paint tools to familiarize them with the HyperCard program and as a warm-up trial.

They were then given verbal instructions again, which emphasized similar written instructions, regarding the training program. The experimenter walked them through the first two practice tasks which simply gave the subjects practice using their type of instructions. The animation group practiced operating the demonstrations, including playing the segments. The choice group viewed the written text and then practiced operating the corresponding demonstration. The subjects then had 30 minutes to learn and practice the 7 tasks. They were informed that if they finished early they should continue practicing.

Once the time for the practice session ended, the subjects were taken to the first test and given verbal instructions that emphasized the written instructions for the testing session. They had 20 minutes to complete all 7 test tasks, with that time being divided among the tasks. They were only allowed one attempt to correctly perform each task. They were able to access help for the three tasks that were identical to the practice tasks, but had to hide the help before beginning the tests. There was no help available for the other four tasks that were similar to and different than
the practice tasks. Upon completion of the test, the subjects were excused and asked not to use HyperCard during the following week.

On the second day, one week later, the subjects received a warm-up trial on the same practice task used the first day. They then began the second test, which was identical to the first test and also lasted 20 minutes. They were only allowed one attempt to correctly perform each task. When they completed the test, they were given additional written and verbal instructions explaining the transfer task. They were allowed 20 minutes to complete this task. When the time ended, the subjects filled out a questionnaire and were debriefed regarding the purpose of the experiment (see Appendix E for the debriefing form).

RESULTS

Accuracy and time to complete the tasks were automatically recorded by the HyperCard training program, as well as usage of the instructions. The subjects were given only one chance to perform each task at Test 1 and 2. This made a situation possible where one person could perform a task incorrectly more quickly than another person who performed the task correctly. Therefore, performance times for all tasks and sessions were recoded to reflect accuracy of performance. In this manner, incorrect task performance would always be scored as slower than correct performance. Since subjects could only spend a limited amount of time on each task during the tests (e.g., the maximum time that could be spent on Test 1 was 2 minutes), the time for an incorrect performance was set to the maximum time for that task plus 10%. Times for correct performance were not recoded and thus reflect the actual time taken to complete the tasks correctly. The time taken to complete the last attempt on each task during training was used as the performance measure for
the training session. (See Appendix F for tables of mean performance times for all measures). All of the following analyses focus mainly on the changes in performance within each group over sessions, since the small number of subjects did not really allow a meaningful between-groups comparison of performance.

**Performance for All Tasks and Sessions**

Only the data for the animation and text group were compared in these analyses, since the data from the choice group was intended to be used primarily to assess users' instructional preferences. The average performance time for all tasks across all three assessment sessions was compared for the two groups. Figure 3 illustrates the effect of session for the animation and text groups. It can be seen that the average performance time for the animation group became steadily faster with each session, while that of the text group became slightly slower and leveled off. The Session by Type of training interaction was significant, $F (2, 40) = 6.309, p = .0042$. There was a significant effect of session for the animation group, $F (2, 20) =

![Graph](attachment:image.png)

**FIGURE 3.** Training group by session interaction (all tasks 1-7)
5.808, $p = .0103$, but not for the text group, $F (2, 20) = 1.608, p = .225$. The steady improvement of the animation group over sessions, while the performance of the text group remained relatively stable, indicated that there was a negative linear effect of session for the animation group. Indeed, there was a significant Group by Session (linear) interaction, $F (1, 20) = 9.803, p = .0053$. Thus, the effect of session for the animation group exhibited a significant negative linear trend, $F (1, 10) = 11.486, p = .0029$, while the linear effect of session for the text group was not significant, $F (1, 10) = 2.030, p = .1696$. This effect of session was not task dependent. With the exception of task four, the animation group’s performance improved across sessions for all tasks (see Appendix F).

The percentage of tasks performed correctly across sessions followed a similar pattern (see Figure 4). The percentage of correct tasks for the animation group steadily increased across sessions from 74.2% to 92.2%, while the text group’s per-

![Figure 4](image_url)

**FIGURE 4.** Training group by session interaction (percentage of correct tasks)
formance increased from 79.3%, but leveled off for the first and second tests at 87%. However, the interaction of group and session was not significant for this measure, $F (2, 40) = 1.132, p = .3323$, and the differences between the groups was not significant at any of the three sessions.

It is interesting to note the variance in average performance for all tasks across the first and second tests (see Figure 5). There was greater within-group variance for the animation group during training which reduced substantially over time. Also, note that both the level of average and median performance improved.

**FIGURE 5.** Boxplot of group performance (all tasks)
over sessions (i.e., the subjects became faster at performing the tasks). The variance in performance times for the text group also reduced somewhat over the sessions, however, both the average and median performance levels demonstrate that the subjects did not improve steadily over time. The difference between the groups in terms of variance reduction over sessions was not significant, using Levene’s (1960) test, $F(2, 40) = 1.157, p = .3246$. This type of within-group variance occurs quite often with computer tasks, especially programming tasks. The effect of the individual differences evident in this study probably overshadowed the effect of training method. Indeed, differences among individuals usually account for more variability than differences in training methods (Egan, 1988).

**Performance for Identical Tasks**

Tasks 1, 2, and 3 (i.e, Resize Field, Link Button, and Change Field’s Border) were identical across all 3 sessions. Figure 6 demonstrates the interaction of session and groups for these tasks. The animation group showed steady improvement

![Graph showing performance improvement over sessions](image)

**FIGURE 6.** Training group by session interaction (identical tasks 1-3)
in their performance times across sessions, while the text group’s performance improved between the practice session and the first test, but returned to the practice level of performance by the second test. An examination of the performance on these tasks revealed that once again there was a significant interaction of training group and session, $F(2, 40) = 4.615, p = .0157$. There was a significant effect of session for the animation group, $F(2, 20) = 8.5, p = .0021$, but not for the text group, $F(2, 20) = 0.749, p = .4856$. Once again, the animation group improved over sessions, while the performance of the text group remained relatively stable, indicating that there was a negative linear effect of session for the animation group for the identical tasks. There was a significant Group by Sessions (linear) interaction, $F(1, 20) = 7.423, p = .0131$. Specifically, there was a significant negative linear effect of session for the animation group, $F(1, 10) = 16.843, p = .0006$, while there was no linear effect of session for the text group, $F(1, 10) = .001, p = .9812$.

Once again, it is interesting to note the variance in average performance for the identical tasks across the first and second tests (see Figure 7). There was greater within-group variance for the animation group during training which reduced substantially over time as both the average and median levels of performance improved. However, the variance in the text group’s performance times actually increased over time and both the average and median levels of performance illustrate that they did not improve steadily over sessions. Using Levene’s test, there was a significant Group by Sessions interaction in terms of the decrease in variance over sessions, $F(2, 40) = 5.114, p = .0105$. The variance within the animation group decreased significantly over time, $F(2, 20) = 5.552, p = .0121$, whereas the variance within the text group did not, $F(2, 20) = 1.294, p = .2962$. 
Performance for Similar Tasks

Test tasks 4 and 5 (i.e., Create Field and Change Button Style) were similar to the training tasks Create Button and Change Field's Border. Figure 8 illustrates that the animation group seemed to transfer their skill knowledge from the original training tasks to the similar test tasks, as evidenced by a steady improvement in performance time, while the text group exhibited a decrement in performance as they transferred to the similar test tasks. These differences were not large enough, however, to result in a significant effect of training group, $F(1, 20) = 1.457$, $p =$

FIGURE 7. Boxplot of group performance (identical tasks)
.2414, or an interaction of Group and Session, $F(2, 40) = .682, p = .5114$. Also, for the similar tasks, there was not a significant Groups by Sessions (linear) interaction, $F(1, 20) = .546, p = .4687$.

*Performance for Different Tasks*

Test tasks 6 and 7 (i.e., Delete Button and Move Field Closer) were completely different than any of the training tasks. Since there were no related practice tasks for these two tasks, only the data from the first and second test sessions was used. These tasks assessed the subjects' ability to apply the procedural skills they had learned to a novel context. Figure 9 shows that once again the performance of the animation group improved over the two sessions, while that of the text group remained fairly stable. The effect of training group, however, was not significant, $F(1, 20) = .040, p = .843$, but the interaction of Group and Session approached significance, $F(1, 20) = 3.496, p = .0762$.

![Graph showing the mean time to perform tasks](image)

**FIGURE 8.** Training group by session interaction (similar tasks 4 & 5)
Analysis of Help Use

The time spent using help during the first test for tasks 1, 2, and 3 was examined. Table 2 shows the average times spent using help for the two training groups. The animation and text groups did not differ significantly in terms of their overall time spent using help, $F(1, 20) = 1.578, p = .2236$. However, there was an interaction of Group and Task, $F(2, 40) = 3.576, p = .0373$. The two groups spent similar amounts of time using the help for tasks 1 and 3, but the animation group spent more time using the help for task 2 (i.e., Link button). Thus, there is a significant effect of Task for the animation users, $F(2, 20) = 5.571, p = .0119$, but not for the text users, $F(2, 20) = 1.618, p = .2232$.

Performance on the Transfer Task

Figure 10 shows the average performance of the two training groups on the transfer task during the last session. None of the subjects in the two groups were able to complete the transfer task (i.e., replicating the address stack) in the allotted

![Graph showing the mean time to perform the task for different sessions.](image)

**FIGURE 9.** Training group by session interaction (different tasks 6 & 7)
time of 20 minutes. Therefore, the only measure of performance used was the total points scored out of the 110 possible. The subjects had to create three cards with
different types of buttons and fields on each one. Every characteristic of their stack that matched the example stack was worth points. For example, one of the buttons on the main card was worth a total of seven points: One point each for the location of the button, the size, the name, the style, autohiliting, and two points for linking the button correctly to another card. As the figure shows, the two training groups did not differ greatly in their performance. The animation group scored slightly

<table>
<thead>
<tr>
<th>Tasks</th>
<th>Animation</th>
<th>Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task 1</td>
<td>0 (0)</td>
<td>2.364 (6.622)</td>
</tr>
<tr>
<td>Task 2</td>
<td>34.182 (41.792)</td>
<td>8.727 (15.193)</td>
</tr>
<tr>
<td>Task 3</td>
<td>10.818 (24.244)</td>
<td>9.636 (14.193)</td>
</tr>
</tbody>
</table>

*Note. Standard deviations are in parentheses*

FIGURE 10. Average group scores on transfer task (110 points possible)
higher on average \((M = 32.182)\) than the text group \((M = 28.182)\), but the difference was not significant, \(t \,(20) = .6068, p = .5508\).

**Questionnaire Results**

There were four pages of questions on the questionnaire that asked for a variety of background information, computer experience, as well as the subjects’ preferences regarding learning methods for computer tasks. However, only seven critical questions were analyzed to examine the opinions of subjects in the animation and text groups regarding their instruction type and the training in general. See Appendix D for the questionnaire used in this study. The animation group rated their instructions as slightly more helpful and clear. They also stated that they would rather use animation again and were slightly less interested in using the other form of instruction (i.e., text) than the text group. However, there were no significant differences between the two groups for any of the questions. Table 3 displays the means for the seven questions of interest.

**Table 3. Questionnaire results for 7 questions (average responses 5 point scale)**

<table>
<thead>
<tr>
<th>Question</th>
<th>Animation</th>
<th>Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>Did you find the instructions easy to use?</td>
<td>3.818</td>
<td>3.909</td>
</tr>
<tr>
<td>Were they clear and easy to follow?</td>
<td>4.091</td>
<td>3.818</td>
</tr>
<tr>
<td>Did you like having these type of instructions?</td>
<td>3.909</td>
<td>3.909</td>
</tr>
<tr>
<td>How helpful were the instructions?</td>
<td>4.545</td>
<td>4.364</td>
</tr>
<tr>
<td>Would you have liked to use the other instruction type?</td>
<td>2.364</td>
<td>2.818</td>
</tr>
<tr>
<td>Would you want to use this type of instructions again?</td>
<td>3.727</td>
<td>3.455</td>
</tr>
<tr>
<td>Do you feel that you were mimicking the instructions?</td>
<td>3.000</td>
<td>2.909</td>
</tr>
</tbody>
</table>

*Note. 1 = Not at all, 3 = Average or Maybe, 5 = Very*

**Usage of Instructions**

The data from the choice group was analyzed to determine which type of
instruction (i.e., text or animation) was used most often. During training, although there were individual differences, on average the animated instructions were chosen 18.8% of the time, while the text instructions were chosen 81.8% of the time the instructions were used. This difference in choice of instructions was significant, \( t (10) = 4.151, p = .002 \). Frequency data from the choice and animation groups combined was analyzed to determine how the animated demonstrations were used. In general, the subjects preferred to play the demonstrations in their entirety, 92.7% of the time, rather than just playing the segments, 7.3% of the time. This difference was significant, \( t (15) = 9.798, p = .001 \). The pattern of instruction usage and task practice for the animation and text groups was compared. During the training session, the animation group appeared to be spending more time using their instructions than practicing the training tasks, while the text group spent more time actually practicing the tasks. The text group made more task attempts \( (M = 21.18) \) than the animation group \( (M = 10.91) \) and spent more time practicing \( (M = 1120.18 \text{ sec}) \) than the animation group \( (M = 776 \text{ sec}) \). The differences in number of attempts, \( t (20) = 3.886, p = .001 \), and time spent practicing, \( t (20) = 3.771, p = .001 \), were both significant.

**DISCUSSION**

The results of this study demonstrate an interesting interaction of type of training method and time (i.e., session). The group that was trained with the animated demonstrations plus spoken concepts performed the HyperCard tasks more slowly than the group that was trained with written text during the initial training session. Also, the text group spent more time and attempts actually practicing the tasks than did the animation group, who obviously spent more of their time using
the animated instructions. Thus, the animated demonstrations did not result in faster skill acquisition, as Palmiter and Elkerton (1991c) found. However, while the animation group's performance times and accuracy improved steadily from the training session and first test to the second test one week later, the text group's performance remained fairly stable and returned to a level similar to their training performance by the last test session. The animation group was actually performing slightly faster and more accurately than the text group by the last session. Therefore, it would seem that learning with animated demonstrations results in better long-term retention of skill knowledge than learning from reading written text. However, the animation group did not perform much better than the text group on the transfer task and, thus, did not exhibit a greater procedural transfer than the text group.

Why did the text group show faster skill acquisition during training? There are several possible reasons. The written text may have been easier for users to follow and apply immediately to task performance, since it could be reviewed more easily than the animated demonstrations. Indeed, the text group spent more time actually practicing the tasks and this may have allowed them to reach a faster and more accurate level of initial performance than the animation group. The animation users did note in their questionnaires that the demonstrations seemed too fast the first time they watched them and too slow the second time. It seems that the animation group spent more time using their instructions and less time practicing and therefore did not reach the same level of initial skill acquisition as the text group. It is possible that if the animation group had practiced the tasks as much as the text group, their levels of performance at the end of training would have been more equivalent. Also, the animated demonstrations were specifically designed to reduce
the level of mimicry that Palmiter and Elkerton reported. The subjects could not simply replicate what they had watched and this may have resulted in slower initial performance since they had to apply the procedures they had learned to a slightly different task situation.

However, acquisition performance has been found to be a poor indicator of actual skill learning (Schmidt & Bjork, 1992). A better measure of learning is long term retention of skills and, therefore, it is more informative to examine how the performance of the two groups changed between the first sessions and the last session seven days later. Rose (1989) notes that there are four variables that affect skill retention: the retention interval; the degree of overlearning (i.e., additional practice); the type of tasks (e.g., procedural skills tend to be rapidly forgotten); and individual differences. It would seem that the text group should have exhibited the greatest retention, based on these variables, since they practiced the tasks almost twice as much as the animation group. But, in fact, the animation group performed better on the second test. Their performance improved steadily between the initial training session and the last test, while the text group's performance remained fairly stable. Thus, it appears that training with animated demonstrations results in a better retention of skill knowledge over time than training with written instructions, even when the text users practice the skills more often.

Why did the animation group show better retention and performance on the last test if they initially practiced less during training? It would seem that the animated demonstrations plus spoken concepts are a better form of instruction than written text. However, since they were designed to make maximum use of the capabilities of that form of media, it is difficult to determine if the effectiveness of this form of training is due to the animation, the spoken concepts, or the combination of
both. The animation users may have found the procedures more meaningful since they were provided a conceptual framework and meaningful information is obviously more easily remembered (Kieras & Bovair, 1984; Rose, 1989). Also, as Schmidt and Bjork (1992) state, conditions that provide added difficulty for the learner during skill acquisition and result in poorer initial performance produce better retention performance. For example, intentional variation of a task to be learned during acquisition (e.g., having learners perform novel variations of the task) typically results in poorer initial performance than consistent practice of the same task. However, variable practice results in better retention and generalizability of knowledge to novel situations. The animated demonstrations in this study may have provided difficulty for the learners since they had to infer what the procedural steps were, decide how to apply them to new objects, and construct a conceptual framework based on the spoken text. Although this caused poorer initial performance than the text group, their performance improved steadily over time and they displayed good retention of their skills.

If animated demonstrations are a better training method than written text, why didn’t the animation group perform better on the transfer task? The transfer task required a synthesis of the separate procedural skills that had been learned and practiced, as well as the generalization of that skill knowledge to some novel tasks (e.g., create icon buttons). This task was a difficult one for relatively naive users. They had become accustomed to the training stack which was designed to take them through the tasks by clicking various buttons. However, with the transfer task, the subjects were faced with a rather intimidating blank stack. There were no buttons to take them from card to card and the greatest difficulty that the subjects seemed to have was navigating between the cards, rather than the task of duplicat-
ing the address stack. Thus, many subjects never got past creating objects on the first card. This may explain why the average performance of the groups was lower than would be expected and also why there was not a significant difference between the groups. If the subjects could have gotten past the problem of navigating between cards, there may have been greater differences in performance between the training groups.

Overall, these findings indicate that animation may be an effective method of training procedural skills in a graphic environment like HyperCard. This seems to contradict the findings of Palminter and Elkerton (1991). Their animation group showed better initial performance and poorer retention than their text group, which is the opposite of the findings of this study. However, the design of this study differed greatly from their design. They believed that the poor performance of the animation group seemed to be due mainly to mimicry of the instructions. A great deal of effort was made in this study to design instructions for both the animation and text conditions that were as close to optimal as possible. It may be that the redesigned animation plus spoken concepts actually reduced mimicry and provided a more meaningful conceptual framework for the learners and, therefore, resulted in better retention than the animation used by Palminter and Elkerton. Therefore, the goal of designing animated instructions that would result in long-term retention of skills was accomplished in this study. It would also seem that animation may be best for people who have difficulty learning computer tasks. People who are capable of learning new tasks quickly and easily will typically do well regardless of the type of instruction they are given; animation or written text. But, people who do not learn as easily will benefit the most from being trained with animated demonstrations, rather than conventional written text.
However, the preference data from the choice group raises the following question: Do people even want to use animation to learn computer tasks? The choice group, on average, chose the text instructions more often than the animated demonstrations during training. However, there were subjects who did choose the animations frequently or combinations of text and animation. The general preference for text did not seem to be based on any real differences between the two types of instruction in terms of factors such as ease of use or value of the information provided. Many subjects noted in their questionnaires that they chose text simply because it was what they were accustomed to and they did not feel like “braving” the unfamiliar animation instructions. This is similar to the production paradox described by Carroll and Rosson (1987). Users often do not want to make the extra effort to learn new and more efficient methods of performing tasks that they can still accomplish with familiar methods. This may have been the case in this study as well. The subjects did not want to make the effort to become familiar with using the animations when conventional written text was available. But, the subjects who did use the animated demonstrations noted in their questionnaires that they liked them and found them helpful. Thus, it is difficult to determine from this study if people would indeed use animated instructions if they were available.

Based on the findings of this study and previous research, some guidelines for design and implementation of animated instructions can be suggested. The goal of training in computer applications should be to support post-training performance (Schmidt & Bjork, 1992). How well users can perform their skills in the long term is much more important than how well they perform immediately following training. There are several factors that seem to support long-term retention of skill. Conditions that provide added difficulty for learners usually result in poor initial
performance, but better delayed performance (Catrambone, 1989; Charney &
Reder, 1986; Kamouri, Kamouri, & Smith, 1986; Schmidt & Bjork, 1992). These
added conditions require a deeper level of cognitive processing which results in
better learning. Also, making the procedural information more meaningful by pro-
viding a conceptual framework may delay the rate of forgetting of skills and help
learners transfer their skills to novel situations (Bayman & Mayer, 1988; Kieras &
Bovair, 1984; Rose, 1989). Baggett (1987) also found that the best retention of skill
resulted when the learners first practiced a task “hands-on” before watching a nar-
rated film of the procedures.

The most effective form of training, therefore, may be to have new learners
initially try to perform the tasks, without instructions, much like learning by explo-
ration. Then, present them an animated demonstration of the correct procedures,
which includes spoken conceptual information. However, the animation should
demonstrate the general procedures for the tasks, without direct reference to the
actual objects that the learners will be using to practice the tasks. This will discour-
age simple mimicry of the instructions and possibly promote deeper cognitive pro-
cessing of the information. For review of the procedures, the learners should be
presented with the procedural steps in written text for quick reference, since users
noted that animation seemed too slow for reviewing procedures. However, each
written procedural step should be linked to a short animated segment that will pro-
vide the users with an opportunity to see how the task is performed. This type of
instructional design would probably be a very effective method of training for long-
term retention of computer skills.
REFERENCES


Bullinger & B. Shackel (Eds.), *Human-Computer Interaction - Interact '87* (pp. 651-656). New York: Elsevier Science.


Spoken Text for Animated Demonstrations

Practice Task 1:
1. There are several functions that can be selected using Paste Function.
2. The function operates on the values you put between the parentheses.
3. The checkmark executes the function that is currently in the formula line.

Practice Task 2:
1. You can create a new table or put existing text into a table.
2. The text should be separated by tabs.

Task 1:
1. There is a specific tool for selecting and working on each object type. For this task the object is a field and we need the Field tool.
2. Once you have the right tool, click on the object to select it. Notice the appearance of the selected object.
3. To change the size and shape of a selected object, click on a corner of the object and, keeping the mouse button down, drag the corner to a new position.

Task 2:
1. For this task, the object is a button so we need the Button tool.
2. Click on the object to select it. Notice the appearance of the border of the selected button.
3. Select Info under the Objects menu to change things about the object. Since the object is a button, select Button Info.
4. Here, we are linking the button to another card.
5. To complete a button link, go to the card that you want to link to and click the button named This Card.
6. The Link To option modifies a button so that clicking the button shows another
card.

Task 3:

1. Here again, we need the Field tool.
2. Since the object is a field, select Field Info. Notice the style of this field.
3. Select the field that you want to change.
4. Notice the various styles and properties that you can select.

Task 4:

1. Create a new button by selecting New Button under the Objects menu. Other objects also use the New command.
2. To move any selected object, click on a part of the object other than a corner, hold down the mouse button, and drag it to a new location.
3. Object names are one of many things that can be changed by selecting Info about an object.

Task 5:

1. Here, we need the Button tool.
2. The Copy item under the Edit menu stores a copy of the selected object on the Clipboard, which is a temporary storage area.
3. Paste Button under the Edit menu puts a new copy of the button onto the exact location of the original button. Pasting works the same way with other objects.

Task 6:

1. The new card is placed directly after the card that you are viewing when the command is selected.
2. Notice that New Card is under the Edit menu.

Task 7:

1. Notice that New Stack is under the File menu.
2. Opening the stack in a New Window leaves your current stack open too.

3. Selecting Close Stack under the File menu or Back under the Go menu returns you to the previously displayed card in the previous stack.
APPENDIX B
Written Procedural Text

Practice Task 1:
Goal: Find the sum of the three group scores.
1. Click on the empty cell where you want the Total.
2. Select Paste Function... under the Formula menu.
3. Select Sum() and click on the OK button.
4. Select the space between the parentheses of the formula.
5. Click and drag to select the three numbers.
6. Click on the checkmark button to perform the Sum function.

Practice Task 2:
Goal: Insert the existing text into a table.
1. Click and drag to select all of the text.
2. Select Insert Table... under the Document menu.
3. Check that the settings for rows and columns is correct.
4. Click on the OK button.

Task 1:
Goal: Change the shape of the field next to the letter B to the shape of the field next to the letter A.
1. Select the Field tool under the Tools menu.
2. Click on a corner of the field that you want to change (holding down the mouse button).
3. Drag the corner of the field to a new location to change the shape of the field.
4. Release the mouse button when the field is the same shape as field A.
5. Select the Browse tool under the Tools menu.
Task 2:

Goal: Change the See Clock button so that it takes you to the Clock card.
1. Select the Button tool under the Tools menu.
2. Click on the button that you want to change.
3. Select Button Info... under the Objects menu.
4. Click on the button named “Link To...” in the Button info dialog box.
5. Click on the arrow button named “Go to clock card”.
6. Click on the button named “This Card” in the Linking dialog box.
7. Select the Browse tool under the Tools menu.

Task 3:

Goal: Make the border of the field next to the letter g the same as that of the field next to the letter d.
1. Select the Field tool under the Tools menu.
2. Click on the field that has the style of border that you want (Field d).
3. Select Field Info... under the Objects menu.
4. Note the style of this field.
5. Click on the OK button.
6. Click on the field that you want to change (Field c).
7. Select Field Info... under the Objects menu.
8. Click on the appropriate button under Style in the Field Info dialog box to choose the style that you want.
9. Click on the OK button.
10. Select the Browse tool under the Tools menu.

Task 4:

Goal: Create a button, name it Open Book, and place it centered directly above the
books.

1. Select New **Button** under the **Objects** menu.
2. Click on the **middle** of the new button and hold down the mouse button.
3. Drag the new button to the appropriate location and release the mouse button.
4. Select **Button Info...** under the **Objects** menu.
5. Type the new button’s name in the button **name field** of the **Button Info** dialog box.
6. Click on the **OK** button.
7. Select the **Browse** tool under the **Tools** menu.

**Task 5:**

Goal: Duplicate the **See Apple** button and put it **under** the field named **All about dragons**. Name the new button **See Dragon**.

1. Select the **Button** tool under the **Tools** menu.
2. Click on the button that you want to **duplicate** (**See Apple**).
3. Select **Copy Button** under the **Edit** menu.
4. Select **Paste Button** under the **Edit** menu.
5. Click on the **middle** of the new button and hold down the mouse button.
6. Drag the new button to the appropriate location and release the mouse button.
7. Select **Button Info...** under the **Objects** menu.
8. Type the new button’s name in the button **name field** of the **Button Info** dialog box.
9. Click on the **OK** button.
10. Select the **Browse** tool under the **Tools** menu.

**Task 6:**

Goal: Create a new card, placing it **after** the card named **Clock**. Name the new card
Dog.

1. Click on the arrow button named “Go to clock card”.
2. Select New Card under the Edit menu.
3. Select Card Info... under the Objects menu.
4. Type the new card’s name in the card name field of the Card Info dialog box.
5. Click on the OK button.
6. Click on the return arrow button named “Return”.
7. Make sure the Browse tool under the Tools menu is selected.

Task 7:

Goal: Create a new stack and name it Mine. You must create the stack in a new window. Then return to this stack, which is named Trainer.

1. Select New Stack under the File menu.
2. Type the new stack’s name in the stack name field of the New Stack dialog box.
3. Click on the checkbox button named “Open Stack in new window”.
4. Click on the New button.
5. Select Stack Info... under the Objects menu.
6. Check that the new stack has the name that you gave it.
7. Click on the OK button.
8. Select Close Stack under the File menu to return to the Training stack.
9. Make sure the Browse tool under the Tools menu is selected.
APPENDIX C
Questionnaire for Animation Group

Subject #______  Experiment #23  Animation Condition

Please answer the following questions:

1. Age_______  2. Sex:  Male  Female

3. Year in school:  1  2  3  4  5  Major ____________________________

5. Which type of computer(s) do you use most frequently? (You may check more
   than one)
   ___ Macintosh  ___ Apple  ___ IBM or IBM compatible
   ___ Mainframe (terminal)  ___ Sun workstation
   ___ Other (please specify) ________________________________

6. For what purpose(s) do you use a computer? (Check all that apply and estimate
   your experience using those applications.)

   Programs                      Experience

   ___ Word processing           _____ months _____ years
   ___ Spreadsheets              _____ months _____ years
   ___ Statistical packages      _____ months _____ years
   ___ Graphics                  _____ months _____ years
   ___ Programming              _____ months _____ years
   ___ Computer / Video games    _____ months _____ years
   ___ Data bases                _____ months _____ years
   ___ Communications (e.g., electronic mail) _____ months _____ years
   ___ Other ______________________   _____ months _____ years

7. How much experience do you have using HyperCard?  _____ months  _____ years

8. How much experience do you have using Macintosh computers?
   _____ months  _____ years

9. Have you used the statistical program called ASTATS (used in Psych 339)?
   ______ Yes  ______ No
10. Estimate how many computer programs you have written (e.g., in BASIC, Pascal, Fortran, C, HyperCard, etc.)

The following questions refer to your use of the training program in this experiment:

11. Did you find the animated demonstrations easy to use?

Very difficult  Average  Very easy
1            2       3        4       5

12. Were they clear and easy to follow?

Very difficult  Average  Very easy
1            2       3        4       5

13. Why were they easy or difficult to follow?

14. Did you like having animated demonstrations for the instructions?

Not at All  Some  Very much
1            2       3        4       5

15. How helpful were the demonstrations?

Not at All  Some  Very helpful
1            2       3        4       5

16. How was the pace of the demonstrations?

Too Slow  Just Right  Too Fast
1            2       3        4       5

17. Do you believe that the spoken text was redundant?

Not at All  Some  Very Redundant
1            2       3        4       5
18. How helpful was the spoken text?

<table>
<thead>
<tr>
<th>Not at All</th>
<th>1</th>
<th>Some</th>
<th>3</th>
<th>Very helpful</th>
<th>5</th>
</tr>
</thead>
</table>

19. Would you have preferred to have the animated demonstrations **without** the spoken text?

<table>
<thead>
<tr>
<th>Not at All</th>
<th>1</th>
<th>Maybe</th>
<th>3</th>
<th>Very much</th>
<th>5</th>
</tr>
</thead>
</table>

20. Do you think that you would have liked to have been presented with **Text** instructions that simply told you how to do the tasks instead of the demonstrations?

<table>
<thead>
<tr>
<th>Not at All</th>
<th>1</th>
<th>Maybe</th>
<th>3</th>
<th>Very much</th>
<th>5</th>
</tr>
</thead>
</table>

21. Would you want to use the same type of instructions again if you were learning another task?

<table>
<thead>
<tr>
<th>Not at All</th>
<th>1</th>
<th>Maybe</th>
<th>3</th>
<th>Very much</th>
<th>5</th>
</tr>
</thead>
</table>

22. Were the HyperCard tasks easy to learn?

<table>
<thead>
<tr>
<th>Very Difficult</th>
<th>1</th>
<th>Average</th>
<th>3</th>
<th>Very Easy</th>
<th>5</th>
</tr>
</thead>
</table>

23. Do you feel that you were simply mimicking the demonstration to complete the tasks?

<table>
<thead>
<tr>
<th>Never</th>
<th>1</th>
<th>Some</th>
<th>3</th>
<th>Very often</th>
<th>5</th>
</tr>
</thead>
</table>

24. Did you feel that you had enough time to learn the 7 HyperCard tasks during the 1st session?

<table>
<thead>
<tr>
<th>Not at All</th>
<th>1</th>
<th>Just Enough</th>
<th>3</th>
<th>Plenty of Time</th>
<th>5</th>
</tr>
</thead>
</table>
25. Did you refer to the Help when you took the test during the 1st session?

<table>
<thead>
<tr>
<th>Never</th>
<th>Some</th>
<th>Very often</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

26. Did you find the test during the 2nd session to be difficult or easy?

<table>
<thead>
<tr>
<th>Very Difficult</th>
<th>Average</th>
<th>Very Easy</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

27. During the 2nd session, did you find the last test of creating a stack that matched the example stack (the Address stack) to be difficult or easy?

<table>
<thead>
<tr>
<th>Very Difficult</th>
<th>Average</th>
<th>Very Easy</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

28. What did you like about the animated demonstration instructions?

29. What did you dislike about them?

30. In general, how easy is it for you to learn to use new computer software?

<table>
<thead>
<tr>
<th>Very Difficult</th>
<th>Average</th>
<th>Very Easy</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

31. Rate the degree to which the following statements describe what you usually do when you are learning to use computer software: (circle one answer for each)

a. I refer to reference manuals and/or help screens for help in using software.

<table>
<thead>
<tr>
<th>Never</th>
<th>Sometimes</th>
<th>Always</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
b. I like to be shown how to do a procedure, rather than reading how to do it.

| Never | 1 | 2 | 3 | Sometimes | 4 | 5 | 6 | Always | 7 |


c. I like to know why programs or procedures “work” a certain way, rather than just how the procedures are performed.

| Never | 1 | 2 | 3 | Sometimes | 4 | 5 | 6 | Always | 7 |


d. I like to explore a new program to see how it works and try things out on my own.

| Never | 1 | 2 | 3 | Sometimes | 4 | 5 | 6 | Always | 7 |

**Questionnaires for Text and Choice Groups**

The questionnaires for the text and choice groups were identical to the animation questionnaire with a few exceptions. The text questionnaire had questions pertaining to the written text, rather than the animated demonstrations. The choice group had questions pertaining to both the written text and animated demonstrations.
**Training Instructions**

First, I want you to try this practice task to familiarize you with doing HyperCard tasks. When I say Go, turn over the sheet and follow the instructions. Just read each step, do what it says, and so on. Ok? Go.

Ok, now we’re going to start the training. Go ahead and read the instructions now and let me know when you are finished. Let me emphasize that the instructions for each task will simply tell you how to do the procedures for that task. It is the goal statement that actually gives you the specifics of the task, so read it carefully.

If you happen to make a mistake during the session and lose your training stack or end up on the Introduction screen again, just come and get me and I will set it up again.

Now, press the Begin button and I’ll walk you through the next two practice tasks. This is an example in Microsoft Excel. You don’t need to do anything on this card but practice using the instructions. Each screen will show the goal statement here and the instructions for each task will appear here. This area here is where you’ll be practicing the tasks. Go ahead and practice using the instructions now.

Notice that you have to click on a window in HyperCard to activate it. Be patient, the system is a little slow. Press the Next Task button to go to the next task. Here is an example in Microsoft Word. Practice using the instructions again.

After using the instructions, you’ll press the Start button which will show the menu bar so you can try the task. When you’re finished, press the Done button to get feedback. You can then try the tasks again or go to the next task. Remember, you have 30 minutes to do all 7 tasks, so pace yourself. Don’t waste too much time on any one task. Make sure you get through all 7 tasks at least once. If you finish early, just keep going through and practicing until the time is up and the program
tells you to report to the experimenter. Now press the Next Task button to start the actual training.

First Test Instructions

Ok, read these instructions and let me know when you are finished. Let me emphasize that there will only be help available for the first 3 tests. You can only see it once and then you have to hide it with the Hide Help button before you start working on the test task. You have only one chance to do each test, so make sure you are finished before you press the Done button. Doing the test correctly is more important than doing it quickly, but try to perform each test as quickly as you can. Each test is timed, so if you spend too much time on one test it will take you to the next one. So, if there is one test that you can’t quite figure out, it’s best to move to the next one. Ok?

Go ahead and press the Start Test button and it will tell you to report to the experimenter when you are finished.

Second Test Instructions

First, I want you to try this practice task again to familiarize you with what you were doing last week. When I say Go, turn over the sheet and follow the instructions. Just read each step, do what it says, and so on. Ok? Go.

Ok, read these instructions and let me know when you are finished. There will be no help on this test. You have only one chance to do each test, so make sure you are finished before you press the Done button. Both speed and accuracy are important here. Doing the test correctly is more important than doing it quickly, but try to perform each test as quickly as you can. Each test is timed, so if you spend too much time on one test it will take you to the next one. So, if there is one test that you can’t quite figure out, it’s best to move to the next one. Ok?
Go ahead and press the Start Test button and it will tell you to report to the experimenter when you are finished.

Transfer Task Instructions

Now, I want you to apply the skills that you have learned to creating a simple address stack from the ground up. Here is an example of the stack that I want you to make. I want your stack to look and work exactly the same as this stack in every single detail. Ok, go ahead and create your new stack and name it Address Book. Make sure you leave the example stack open so you can use it as your reference. I’ll set this up for you so that the stacks overlap each other and you can easily go between the two stacks by clicking on each one. I want you to create your own stack so do not just try to copy buttons or fields from the example stack and paste them in your stack. Create your own objects in your stack so that it looks like the example stack. You have 20 minutes for this task. Just do the best you can.
APPENDIX E
Debriefing Form

The experiment in which you participated examines the effect of training methods on immediate performance, retention, and transfer of knowledge to more complex tasks. It would seem that training someone to use a graphical program would be most effective if you show them how to do something while explaining what you are doing and why. However, research has suggested that people who learn with animation may simply mimic the procedures. This research was conducted to see if people would be able to retain information longer if it were presented in a redesigned animated format that discourages simple mimicry (i.e., one that provides more general concepts, as well as procedures).

There were three conditions differing only in the format of the instructions given to the subject: Text only, Animation plus spoken text, and a Choice condition. The subjects in the Text condition were presented with the task procedures in a written format on the screen. The subjects in the Animation condition watched the task being performed in a “movie” on the screen that displayed the procedures while they listened to a voice describing basic HyperCard concepts. The subjects in the Choice condition could choose between a text or animated presentation of the procedures. It was expected that the subjects who were trained with the animation would perform better than the Text subjects on the test during the 1st session and would retain this advantage so that they also performed better on the procedural and transfer tests during the 2nd session.

The goal of this experiment is to discover if training with animation is more conducive to learning tasks in a graphical programming language (i.e., HyperCard) than the usual use of text instructions, as well as to assess users' preferences for methods of training. Hopefully, the data collected from this experiment will answer
some of the questions regarding the utility of training people with “visual” demonstrations of procedures. If you have any questions about this research and the results please feel free to call me. Thank you for your participation.
Tables Of Means For All Measures

Group performance across sessions for each task (1-7).

**TABLE 4.** Average performance of animation group across sessions (sec)

<table>
<thead>
<tr>
<th>Task</th>
<th>Training Session</th>
<th>First Test</th>
<th>Second Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>79.182 (98.156)</td>
<td>58.727 (45.180)</td>
<td>40.273 (38.139)</td>
</tr>
<tr>
<td>2</td>
<td>84.636 (53.407)</td>
<td>36.636 (43.732)</td>
<td>18.000 (37.963)</td>
</tr>
<tr>
<td>3</td>
<td>77.636 (84.160)</td>
<td>51.636 (43.768)</td>
<td>28.182 (19.778)</td>
</tr>
<tr>
<td>4</td>
<td>70.091 (57.074)</td>
<td>78.273 (60.107)</td>
<td>88.545 (63.331)</td>
</tr>
<tr>
<td>5</td>
<td>76.636 (51.276)</td>
<td>42.000 (30.466)</td>
<td>23.636 (9.069)</td>
</tr>
<tr>
<td>6</td>
<td>105.545 (74.395)</td>
<td>64.818 (68.104)</td>
<td>28.364 (12.151)</td>
</tr>
<tr>
<td>7</td>
<td>233.636 (160.895)</td>
<td>267.364 (111.413)</td>
<td>186.909 (133.060)</td>
</tr>
</tbody>
</table>

*Note.* Standard deviations are in parentheses

**TABLE 5.** Average performance of text group across sessions (sec)

<table>
<thead>
<tr>
<th>Task</th>
<th>Training Session</th>
<th>First Test</th>
<th>Second Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>48.364 (45.229)</td>
<td>57.000 (48.754)</td>
<td>53.909 (48.486)</td>
</tr>
<tr>
<td>2</td>
<td>41.909 (20.545)</td>
<td>11.909 (9.148)</td>
<td>31.818 (50.123)</td>
</tr>
<tr>
<td>3</td>
<td>38.727 (84.160)</td>
<td>29.909 (10.653)</td>
<td>41.000 (30.226)</td>
</tr>
<tr>
<td>4</td>
<td>44.909 (22.915)</td>
<td>81.273 (46.819)</td>
<td>66.273 (40.359)</td>
</tr>
<tr>
<td>5</td>
<td>49.091 (30.550)</td>
<td>33.818 (13.768)</td>
<td>31.727 (20.977)</td>
</tr>
<tr>
<td>6</td>
<td>94.909 (155.251)</td>
<td>40.545 (15.807)</td>
<td>31.909 (16.544)</td>
</tr>
<tr>
<td>7</td>
<td>69.636 (56.747)</td>
<td>257.545 (93.040)</td>
<td>238.000 (135.683)</td>
</tr>
</tbody>
</table>

*Note.* Standard deviations are in parentheses

Group performance for all tasks (averaged) across sessions.

**TABLE 6.** Average performance for all tasks across sessions (sec)

<table>
<thead>
<tr>
<th>Session</th>
<th>Animation</th>
<th>Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>103.909 (102.154)</td>
<td>55.364 (67.652)</td>
</tr>
<tr>
<td>First Test</td>
<td>85.636 (96.742)</td>
<td>73.143 (89.352)</td>
</tr>
<tr>
<td>Second Test</td>
<td>59.130 (81.011)</td>
<td>70.662 (91.485)</td>
</tr>
</tbody>
</table>

*Note.* Standard deviations are in parentheses
Group performance for identical tasks 1-3 (averaged) across sessions.

**TABLE 7.** Average performance for identical tasks across sessions (sec)

<table>
<thead>
<tr>
<th>Session</th>
<th>Animation</th>
<th>Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>63.394 (44.398)</td>
<td>42.455 (31.021)</td>
</tr>
<tr>
<td>First Test</td>
<td>49.000 (43.836)</td>
<td>32.939 (34.039)</td>
</tr>
<tr>
<td>Second Test</td>
<td>28.818 (33.356)</td>
<td>42.242 (43.473)</td>
</tr>
</tbody>
</table>

*Note.* Standard deviations are in parentheses

Group performance for similar tasks 4 & 5 (averaged) across sessions.

**TABLE 8.** Average performance for similar tasks across sessions (sec)

<table>
<thead>
<tr>
<th>Session</th>
<th>Animation</th>
<th>Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>64.227 (50.914)</td>
<td>41.818 (25.961)</td>
</tr>
<tr>
<td>First Test</td>
<td>60.136 (50.070)</td>
<td>57.545 (41.520)</td>
</tr>
<tr>
<td>Second Test</td>
<td>56.091 (55.250)</td>
<td>49.000 (36.024)</td>
</tr>
</tbody>
</table>

*Note.* Standard deviations are in parentheses

Group performance for different tasks 6 & 7 (averaged) across sessions.

**TABLE 9.** Average performance for different tasks across sessions (sec)

<table>
<thead>
<tr>
<th>Session</th>
<th>Animation</th>
<th>Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>First Test</td>
<td>166.091 (137.347)</td>
<td>149.045 (128.740)</td>
</tr>
<tr>
<td>Second Test</td>
<td>107.636 (122.820)</td>
<td>134.955 (141.495)</td>
</tr>
</tbody>
</table>

*Note.* Standard deviations are in parentheses

Group performance for transfer task.

**TABLE 10.** Average performance for transfer task (110 points possible)

<table>
<thead>
<tr>
<th>Animation</th>
<th>Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transfer Task</td>
<td>32.182 (18.798)</td>
</tr>
</tbody>
</table>

*Note.* Standard deviations are in parentheses