Learning, Forgetting, and Relearning for Keystroke- and Mouse-driven Tasks: Relearning is Important

Jong Wook Kim and **Frank E. Ritter**

The Pennsylvania State University

June 11, 2013

Human-Computer Interaction (in press).

Author Note. At the time of study, the first author was at the Pennsylvania State University and is now a research scientist in the Department of Psychology at the University of Central Florida. The data presented in this manuscript is based on the first author's dissertation study and expanded.

Acknowledgments. This research was sponsored by grants from the Division of Human Performance Training, & Education at the Office of Naval Research, under Contracts W911QY-07-01-0004 and N00014-10-1-0401. The views and conclusions contained in this article are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Government or the Pennsylvania State University. Comments from several anonymous reviewers have helped improve this paper.

Corresponding Author

Jong Wook Kim, PhD

Department of Psychology
University of Central Florida
Orlando, FL

Email: jong.kim@ucf.edu
jongkim626@gmail.com
Phone: +1 (407) 312-8272

Document Statistics **(skill retention data47.doc)**

Tables: 4
Figures: 9
Word Count: 12,960 (abstract: 274, references: 1,554)

Abstract

This paper investigates performance change arising through learning, forgetting, and relearning. Participants learned a spreadsheet task with either keystroke-driven (keyboard, *n* = 30) or mouse-based menu-driven (mouse, *n* = 30) commands. Their performance confirmed the power law of practice. The keyboard users learned to complete the task faster than the mouse users on the last learning session (Day 4). At a 6-day retention interval, the mouse users were observed to forget more—they took more time to complete the task than the keyboard users. Interestingly, the participants in the two modality groups showed no significant differences in their forgetting under the retention of 12 and 18 days. With additional practice, the mouse group users with the 6-day retention relearned more—they significantly reduced the time to complete the task in comparison to the keyboard group. The experiment with these two modality groups helps understand why people may choose to use a mouse-driven graphical user interface (GUI) rather than a keystroke-driven interface—people choosing to use a mouse-based menu-driven interface may not need to use a knowledge-in-the-head strategy but knowledge-in-the-world, and may be doing so because this strategy provides better relearning, rather than because it is faster or easier initially or because it is better for learning or forgetting. This helps provide a richer explanation of why menu-driven interfaces (knowledge-in-the-world) are more ubiquitous, and suggests when they can be replaced, e.g., where use is infrequent but often enough that forgetting does not occur. Our results provide preliminary suggestions for choosing optimal training strategies and supporting these strategies in terms of the three stages of learning and forgetting.

Keywords: Learning, Forgetting, Relearning, Performance change, ACT-R

Contents

1 Introduction

Understanding how performance change, in terms of learning, forgetting, and relearning after disuse of task knowledge, can play an important role in designing interactive interfaces and instructional systems (e.g., Fisk & Hodge, 1992; Johnson, 1981; Shiffrin & Schneider, 1977; Shute & Gawlick, 1995; Stefanidis, Korndorffer, Markley, & Sierra, 2006). While learning is studied as well as forgetting, there is, however, little research on the full time course of learning, forgetting, and relearning across time with regard to the user's interaction with a system. It has been argued that performance during acquisition is an imperfect indicator of later performance, and, therefore, to better examine and support performance, acquisition and retention should not be separated (Schmidt & Bjork, 1992). We agree with Schmidt and Bjork that retention can help us to better understand the efficacy of training and skill decay after disuse of a system (e.g., Olson & Olson, 1990). Furthermore, it is valuable to have empirical data including both learning and forgetting (and relearning) because this type of data can be useful to test theories such as unified theories of cognition (e.g., Newell, 1973, 1990).

The goal of this paper is to answer how different types of task knowledge are learned and retained, and how the user interacts with the task via different modalities. It is reasonable to hypothesize that the interactive modality with a mouse-based menu-driven interface rather than a keystroke-driven interface could provide better performance (e.g., less task completion time) in learning and provide better retention for later performance only if we assume that most people seek to use a mouse-based menu-driven interface. If this is not true, what scientific accounts can explain the greater prevalence of menu-driven interfaces, often realized as GUIs (e.g., Shneiderman, 1983)?

In this article, we investigate performance change on a task with two input modalities—a keystroke-driven interface requiring keystroke commands (the Keyboard group) and a mouse-based menu-driven interface (e.g., GUI) requiring the menu-driven commands (the Mouse group). For this comparison, we not only look at learning the task knowledge but also forgetting and relearning as well. In the next section, we describe the relevant theoretical accounts in terms of learning, forgetting, and relearning. Then, we report our empirical findings and their implications to the design of everyday things that we use in this digital era (e.g., Norman, 1988).

2 Performance Change through Learning and Forgetting

How does performance change occur through learning, forgetting, and relearning? This section describes theoretical accounts of performance change based on the ACT-R theory. We chose to use the ACT-R architecture (Anderson et al., 2004) as a theoretical basis because it is one of the most widely used cognitive architectures representing human cognitive performance, and because it provides theoretical mechanisms for representing learning and forgetting.

2.1 Learning in ACT-R

There is a theoretical consensus about the stages of learning reviewed in Kim, Koubek, and Ritter (2013). Many theories propose a three-stage process of learning as shown in Figure 1: (a) the first stage to acquire declarative knowledge to perform a procedural task—that is, enough knowledge to generate behavior using the declarative knowledge structure, like following a script or a recipe, (b) the second stage for consolidating the acquired knowledge, and (c) the final stage for tuning the knowledge toward overlearning. Fitts (1964) labeled these three stages of learning: *early*, *intermediate*, and *late* phases. Anderson (1982) developed a theory of cognitive skill acquisition with three stages of *declarative*, *transitional*, and *procedural*, corresponding to

Fitts' three stages of learning. Rasmussen (1986) proposed a framework pertaining to skilled performance that is differentiated by knowledge-based, rule-based, and skill-based. VanLehn (1996) also described the three stages of cognitive skill based on Fitts' framework.

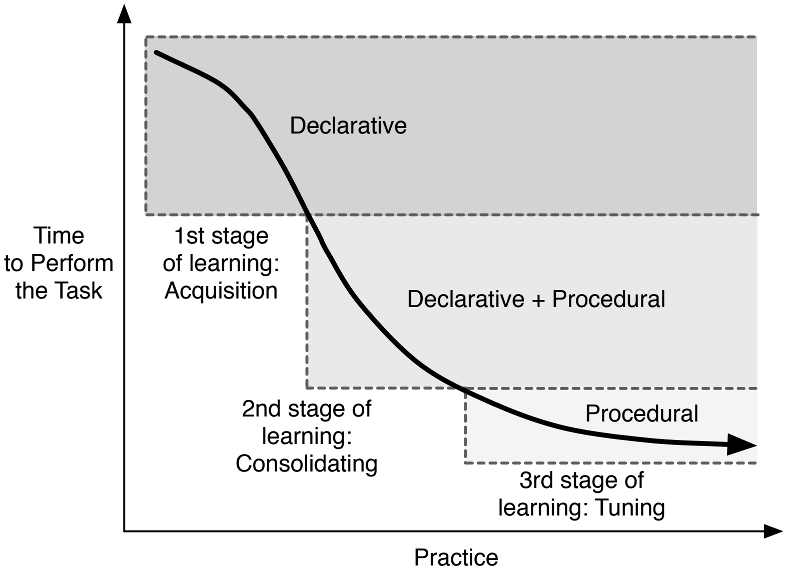


Figure 1. Performance change in the three stages with declarative, mixed, and procedural representation of knowledge, which is based on the theory of Fitts (1964), Anderson (1982), Rasmussen (1986), and VanLehn (1996).

In the first stage (declarative stage), an individual learns task knowledge from instructions. Acquiring information is first supported through initial encoding of facts about task knowledge. Then, in the second stage (declarative + procedural), the acquired task knowledge is interpreted to produce behavior. Through a mechanism called knowledge compilation[1] (e.g., Anderson, 1982; Jones, Ritter, & Wood, 2000; Neves & Anderson, 1981; Rosenbloom & Newell, 1987), the acquired task knowledge is converted to a procedural form with appropriate practice. After knowledge compilation, further tuning of the knowledge occurs in the third stage,

[1] Knowledge compilation is called chunking in the Soar architecture and proceduralization in other theories.

producing speedup of the knowledge application process, which is referred to as the procedural stage.

The ACT-R theory describes human learning as a three-stage process through a distinctive classification of task knowledge representation: declarative and procedural knowledge. In ACT-R (Anderson et al., 2004), declarative knowledge is represented as a propositional network of facts consisting of chunks, and procedural knowledge is represented as production rules. Each production rule has a condition/action statement and specifies when a cognitive act should take place. Procedural knowledge refers to knowledge that is displayed in behavior such as steps and sequences of how to do a task. Based on the construct of declarative and procedural knowledge, ACT-R provides symbolic and sub-symbolic learning mechanisms: an activation mechanism (Anderson et al., 2004) and a production rule learning mechanism (Anderson, 1982, 1987; Anderson, Conrad, & Corbett, 1989; Taatgen & Anderson, 2002). The effects of retrieval on declarative memory items and the learning process of production rules can account for performance change.

This learning behavior generally follows a regularity known as the power law of practice, where the time to complete a task speeds up with practice according to a power function and the amount of improvement decreases as practice continues (e.g., Anderson, Fincham, & Douglass, 1999; Card, English, & Burr, 1978; Delaney, Reder, Staszewski, & Ritter, 1998; Newell & Rosenbloom, 1981; Ritter & Schooler, 2001; Seibel, 1963). This learning curve describes an important aspect of behavior, providing a mathematical account of a general learning rate (Rosenbloom & Newell, 1987), and supports building models of learning complex tasks that can be decomposed into smaller components (Lee & Anderson, 2001).

2.2 An Example of Skill Learning

To describe learning in more detail, we present a simple example of learning how to type (e.g., Anderson, 1993). Learning typing has the same structure as many similar skills where applying declarative knowledge leads to procedural knowledge. When learning to type, individuals generally first memorize the layout of the keyboard declaratively and learn to use the keyboard procedurally. This is at least one way; there are others.

Practicing typing enables individuals to memorize the keyboard layout and to type faster with practice. Over time (several months or more), individuals generally and slowly lose their declarative knowledge of the keyboard's layout because they do not use it but retain their procedural typing skills. Thus, once fully learned, few individuals can recall key positions declaratively (e.g., retrieving where a key is), but rather rely more on their proceduralized task knowledge (e.g., imagining typing a letter and seeing where their finger goes).

This example illustrates how individuals use both declarative and procedural knowledge in memory, and how the kinds of memory used can be both dependent and independent at different stages of knowing a task. In addition, it suggests that procedural knowledge can be more robust than declarative knowledge. In the first stage, individuals depend almost exclusively on declarative memory elements to perform the task—this initial stage is both cognitively intensive and slow. In the second stage, individuals begin to rely more heavily upon procedural memory elements, but still rely on their declarative knowledge of the keyboard (*q* is above *a*, for instance) for some hard to learn or rarely practiced keys. Finally, as individuals evolve into experts, they shift entirely or almost entirely to using their procedural memory. In addition, the transition from a primarily declarative to a procedural representation of the keyboard is associated with a reduced need for knowledge maintenance—lack of practice may result in slower typing speeds but not an entire loss of task knowledge.

Early experimental work by Posner (1973) showed that procedural memory can be more robust than declarative memory. In Posner's experiment, skilled typists were asked to label a diagram of a standard keyboard. He reported that the skilled typists had difficulty in recalling a visual location of a letter from the standard keyboard (declarative memory), whereas the skilled typists could type the letters in a few seconds without errors. Skilled typists exhibit greater precision of typing movements and lower variability of interkey intervals (e.g., Rabbitt & Banerji, 1989; Salthouse, 1986). This example supports the view that declarative location knowledge can be degraded with time while procedural knowledge can remain relatively robust against decay (Ericsson & Simon, 1993; also see Stefanidis, Korndorffer, Markley, & Sierra, 2006 for an example of procedural memory not decaying), suggesting as well that long-term retention can be improved when declarative knowledge turns into procedural knowledge. In the next section, we describe relevant theoretic accounts of forgetting so that we can integrate learning, forgetting, and relearning into these three stages.

2.3 An Integrated Understanding of Learning and Forgetting

Figure 2 represents learning and the corresponding forgetting in the three stages based on our summary (Kim, Ritter, & Koubek, 2013). The main continuous line indicates continuous practice. Dashed lines indicate periods of no practice, with solid lines showing later learning (and relearning).

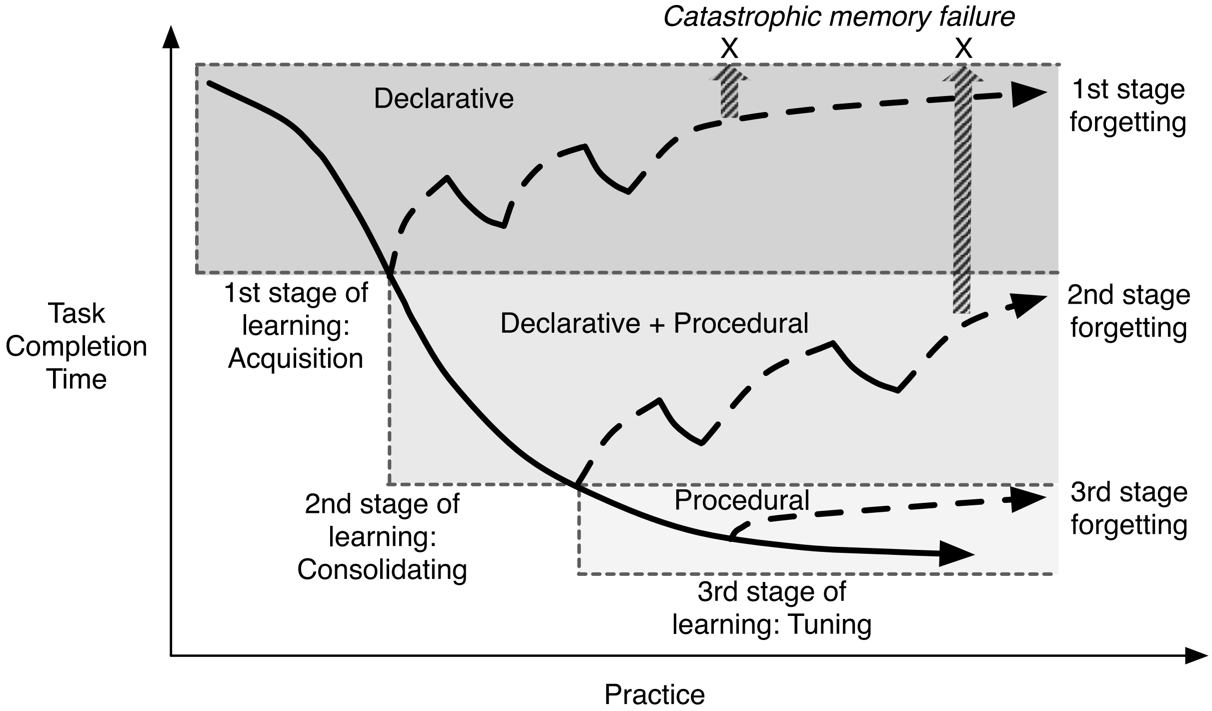


Figure 2. A graph describing a theory of task knowledge retention, showing the three stages of learning, and also the effects of forgetting and relearning. The solid lines are the learning and relearning curves. The dashed lines are forgetting curves from each stage. At each stage, the learning and forgetting rates are different.

It is interesting to consider what kinds of knowledge are forgotten in each stage. In the first stage, declarative knowledge would be degraded without practice, as shown in Figure 2. In the second stage, declarative knowledge is degraded and procedural knowledge could be degraded (but it is not degraded in current cognitive architectures, Chong, 2004). In the third stage, similar to the second stage, both declarative and procedural forms of knowledge could be degraded. For clarity, we describe each stage distinctively but the three stages are continuous, and complex tasks could have subtasks in each of these stages.

The First Stage: Declarative. For this first stage of learning and forgetting, task knowledge in declarative memory degrades with lack of use, perhaps catastrophically as indicated by the X's in Figure 2, leading to inability to perform the task. In this stage, learning and forgetting are explained by an activation mechanism in ACT-R (e.g., Pavlik & Anderson, 2005). ACT-R's activation equation, Eq. (1), proposes that a declarative memory item receives an increment of strength when it is practiced, but that the increment decays as a power function of time. Thus, with lack of use, the strength of declarative memory declines, and decreased memory strength leads to response time increasing and accuracy decreasing. In Eq. (1), n is the number of presentations for a memory item i, t_j is the time since the j^{th} presentation, and d is the decay parameter. ACT-R assumes a noise component (ε) that follows a logistic distribution and is controlled by a parameter s.

$$A_i = \ln(\sum_{j=1}^{n} t_j^{-d}) + \varepsilon(s) \qquad \text{Eq. (1)}$$

ACT-R also addresses response accuracy (Eq. 2) and latency (Eq. 3). A memory item in declarative memory is retrieved only if the activation value of that item is above threshold, τ in Eq. (2), the probability of recall. The retrieval latency equation, Eq. (3), represents how quickly an item is retrieved from memory (F is a latency factor parameter). The time to complete a task, therefore, decreases as a power function of the number of practice trials (e.g., Anderson, Fincham, & Douglass, 1999).

$$\mathrm{P}(i) = \frac{1}{1 + e^{\frac{\tau - A_i}{s}}} \qquad \text{Eq. (2)}$$

$$T = Fe^{-A} \qquad \text{Eq. (3)}$$

The Second Stage: Associative. In the second stage of learning, task knowledge is represented with a mix of declarative and procedural memory. Thus, forgetting in this stage should be considered as a dual process in both declarative and procedural memory. A declarative memory item is forgotten with lack of use or practice, which can lead to missed steps in sequential tasks. This phenomenon can be represented using the activation mechanism. In addition, a production rule could also be forgotten[2] and selection of a production rule should be controlled in terms of its history of usage based on linking declarative memory items. In the first and second stage, catastrophic memory failure (where a user could not perform the task) can occur if a declarative memory item is not fully activated (i.e., below the retrieval threshold value) and is needed to proceed. In this mixed stage, training should be provided to keep declarative knowledge active and also to support further proceduralization.

The Third Stage: Procedural. In the third stage of learning, task knowledge is available in both declarative and procedural forms, but practice has compiled knowledge into a procedural form of knowledge and procedural knowledge predominantly drives performance. We describe this type of task knowledge as *proceduralized task knowledge*. With lack of use, declarative knowledge may be degraded. Nevertheless, individuals can still perform the task—if all the knowledge is proceduralized and thus not forgotten with time. Less well-known task knowledge that is infrequently used, like recovery from unusual errors, may be degraded. This type of task knowledge would require knowledge retrieval from declarative memory unless it is proceduralized. To reach this final stage, practice for proceduralization should be provided for all tasks. It also suggests that training should occur until trainees cross the thresholds, noted as dashed horizontal lines in Figure 2.

[2] Most theories, including ACT-R, do not include this as a possibility (Chong, 2004).

2.4 Summary

There is a consensus view in the literature that learning occurs in three stages. Also, we recognize that more research studies are necessary to investigate the mechanisms of forgetting in the three stages. Decay of declarative knowledge is predicted by most architectures. On the other hand, decay of procedural knowledge is not predicted by most cognitive architectures. Chong (2004) reviews several existing architectures and summarizes that they do not model or predict procedural knowledge decay. For example, Soar (Newell, 1990), ACT-R (Anderson et al., 2004), and EPIC (Kieras & Meyer, 1997) do not include procedural knowledge decay. This could be an area where a cognitive architecture does not take a theoretical stance, but it appears to be taken explicitly in Soar (e.g., Newell, 1990, p. 164) and in ACT-R (e.g., Anderson, 1993, p. 18).

Interestingly, we also have noticed that little research has been conducted to provide possible mechanisms of decay in procedural memory. That is, in procedural memory where ACT-R compiles production rules to show speedup effects of practice, the compiled production rules cannot be forgotten to represent decreased performance (e.g., task completion time increases when not used). Thus, it may be worth exploring the production rule learning mechanism in ACT-R (i.e., change in the utility value of productions to model procedural memory decay), adding a parameter to the current architecture, or extending the existing architectural mechanisms. In addition, it will be necessary to build a model to test this claim and remove this architectural limitation if it is one.

In the next section, we present our empirical investigation of how two types of task knowledge are learned and forgotten in a laboratory setting. In this investigation, we sought to identify performance change from learning to forgetting in two different input modalities (i.e., the Keyboard and Mouse groups).

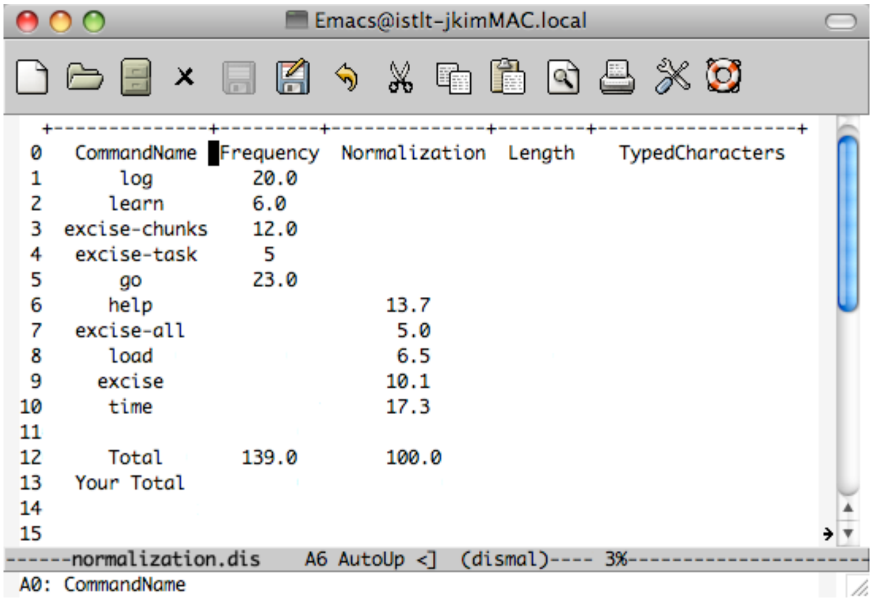
3 The Task and Performance Predictions

Research studies with text editing tasks have provided important insights on human information processing and task performance. For example, Card, Moran, and Newell (1983) studied how a user's task knowledge interacts with computer-based systems focusing on text editing tasks. Similarly, Singley and Anderson (1989) investigated the transfer of cognitive skills using a text editing task. As an extension of such studies, we created a set of novel spreadsheet tasks to measure performance change characterized by learning, forgetting, and relearning over time. We explain, in this section, the task, the two modality groups based on the different task components (e.g., keystroke- and mouse-driven performance), an example analysis of microgenetic learning and forgetting, and a baseline prediction of performance time.

3.1 The Dismal Spreadsheet Task

The task was created in the Dismal[3] spreadsheet (see Figure 3) that was initially developed to analyze process models and sequential behavioral data (Ritter & Larkin, 1994; Ritter & Wood, 2005). The Dismal spreadsheet task in this study consists of fourteen sequential subtasks shown in Table 1.

[3] Also see http://acs.ist.psu.edu/dismal/dismal.html



```
 +--------------+---------+-------------+--------+----------------+
0   CommandName  Frequency  Normalization  Length    TypedCharacters
1      log          20.0
2     learn         6.0
3  excise-chunks    12.0
4   excise-task      5
5      go           23.0
6     help                      13.7
7   excise-all                   5.0
8     load                       6.5
9    excise                     10.1
10    time                      17.3
11
12    Total        139.0        100.0
13  Your Total
14
15
------normalization.dis    A6 AutoUp <]  (dismal)---- 3%-------------------
A0: CommandName
```

Figure 3. The Dismal spreadsheet. Dismal runs under Emacs. In the spreadsheet, there are blank cells in five columns that users fill in.

Table 1. *The fourteen subtasks in the Dismal spreadsheet task.*

(1) Open a file, named *normalization.dis* under the *experiment* folder
(2) Save as the file with your initials
(3) Calculate and fill in the *Frequency* column (B6 to B10)
(4) Calculate the total frequency in B13
(5) Calculate and fill in the *Normalization* column (C1 to C5)
(6) Calculate the total normalization in C13
(7) Calculate the *Length* column (D1 to D10)
(8) Calculate the total of the *Length* column in D13
(9) Calculate the *Typed Characters* column (E1 to E10)
(10) Calculate the total of the *Typed Characters* column in E13
(11) Insert two rows at cell A0
(12) Type in your name in A0
(13) Fill in the current date in A1 using the command *dis-insert-date*
(14) Save your work as a printable format

In the Dismal spreadsheet task, the A (first) column (“Command Name”) has ten different names of computer commands (A1 to A10). The B column (“Frequency”) has frequencies of each command listed in the A column. There are default values of each frequency (B1 to B5). Participants calculate frequencies of each command from B6 to B10, using Eq. (4).

Normalized frequencies are listed in the C column (“Normalization”). While the cells (C6 to C10) are of default values of normalized frequencies, participants need to calculate the blank cells of C1 to C5, using Eq. (5). In the D column (“Length”), participants need to calculate the length of each command with a function given in the instructions. The typed characters in the E column are calculated by multiplying a command name’s frequency by its length. The totals of each column (B13, C13, D13, and E13) are to be calculated. Then, participants need to insert two rows at the first row and type their name and the date by using a Dismal command. Finally, the last step is to save the work as a printable version.

$$\text{Normalization} = \frac{(\text{Frequency} \times 100.0)}{\text{Total frequency}} \qquad \text{Eq. (4)}$$

$$\text{Frequency} = \frac{(\text{Normalization} \times \text{Total freqeuency})}{100.0} \qquad \text{Eq. (5)}$$

3.2 The Two Modality Groups

The knowledge that participants learn in the Dismal spreadsheet task ranges from simple motor skills based on retrieval of declarative knowledge to problem-solving skills of calculating frequency and normalization. The Dismal spreadsheet task supports different interaction modalities for an HCI study. In our study, all participants were grouped into one of two groups by input modality and task knowledge: (a) *the keyboard group* and (b) *the mouse group*. Some participants in the keyboard group are only allowed to use *keystroke-driven commands*. They were instructed not to use *menu-driven commands* with a mouse. Other participants in the mouse group used a vertical mouse (an Evoluent™ vertical mouse) with *menu-driven commands* and a keyboard to type letters, numbers, and symbols.

For example, participants were trained to achieve the goal in the first subtask, *Open a File*, following these instructions:

(a) The Keyboard Group: The keyboard group learned the procedures of the task (how-to-do-it knowledge) and a number of Emacs keystroke commands (e.g., `C-x C-f`, `C-b`, etc.). When opening a file, participants needed to retrieve knowledge from memory—such as the keystroke command to open a file, "`C-x C-f`"[4]. Also, when moving around the cells in the spreadsheet, they were only allowed to use corresponding keystroke commands (e.g., `C-f` for moving right, `C-b` for moving left, and `C-p` for moving up).

(b) The Mouse Group: The mouse group learned the procedures of the task (how-to-do-it knowledge) shown in Table 2 and how to use the vertical mouse—When moving around the cells, they were only allowed to use the vertical mouse.

Both the keyboard and mouse group participants needed to retrieve arithmetic equations from memory—such as equations to calculate frequency and normalization. While the mouse group participants relying on the knowledge-in-the-world could minimize the number of declarative memory items, the keyboard group participants were required to memorize all the declarative memory items. Table 2 summarizes the task knowledge by the two modality groups.

[4] `C-x` indicates holding down the control key while pressing `x`.

Table 2. *Keystrokes and mouse actions for the first subtask.*

Subtask	Group	Keystrokes	Mouse actions
Open File	Keyboard	Press `C-x C-f` Type in `normalization.dis`↲	none
	Mouse	none	Point & click on `File` Point & click on `Open File` Point & click on `experiment` Point & click on `normalization.dis`↲

Note: The symbol ↲ indicates pressing the return key.

In this study, a vertical mouse was chosen because it provides new motor skills to learn and retain, and because the learning data can provide a clearer learning effect than using a well-practiced, traditional mouse. A quantitative analysis on this concern is presented in Section 3.4. This vertical mouse requires a neutral hand position, compared to the pronated hand position of a traditional mouse, and requires different muscle activity in the forearm and wrist (e.g., Chen & Leung, 2007; Gustafsson & Hagberg, 2003). Also, it is reported that providing instruction (learning) is needed to enhance the ergonomic benefits arising from the alternative mouse (Houwink, Oude Hengel, Odell, & Dennerlein, 2009).

3.3 Predicting the Total Task Performance Time

It is useful when analyzing the results to have a baseline task predictions. The predictions can help describe and differentiate the two interfaces, and help understand learning, forgetting, and relearning. A KLM-GOMS analysis (Card, Moran, & Newell, 1983) provides a baseline prediction.

The task completion times from the two modality groups were computed, assuming error-free expert performance. The model includes primitive physical-motor operators (*K* – keystroke, *P* – pointing, *H* – homing, and *D* – drawing), mental operators (*M*), and system response time

(*R*), as shown in Eq. (6). There were no lines drawn ($T_D = 0$) in this task. In the interest of simplicity and because of relatively fast response times, we ignored the system response time ($T_R = 0$).

$$T_{execute} = T_K + T_P + T_H + T_D + T_M + T_R \qquad \text{Eq. (6)}$$

For time predictions of the Dismal spreadsheet task we used three physical-motor operators (*K*, *P*, and *H*) and a mental operator (*M*). We used the default time of homing and of mental operators.

During the mental preparation time (T_M), participants mentally prepare what to press, what to retrieve from memory, and what to do for the next step. We followed the existing heuristic rules for determining the use of mental preparation (Card, Moran, & Newell, 1983, p. 265) and used the default time, 1.35 s. We placed a mental operator in front of all pointing activities (pointing to a menu item) and all key-press activities (pressing a keystroke command). To complete the first subtask of *Open File*, theoretically, participants in the keyboard group need 3 mental operators, and participants in the mouse group need 5 mental operators. We assumed that a mouse user's hand location was initially on the mouse and the hand to mouse time was, thus, 0. The homing time (T_H) for hand movements between different physical devices was 0.4 s.

To calculate the keystroke time (T_K) that we know varies across individuals, we computed the time from the first keystroke to the last in the first subtask for both modalities. The average keystroke time ranged from 0.95 s/keystroke on the first day of training to 0.47 s/keystroke on the last day of training. If we refer to the keystroke time in Card et al. (1983, p. 264), our data indicate the participants' keystroke speed resided between the worst typist, 1.20 s, and the speed of average non-secretary typist, 0.28 s. We decided to use 0.47 s for the T_K parameter as an expert performance. Shift and control keys were counted as separate keystrokes.

For the positioning time (T_P), we computed the time when the cursor begins to move until when the mouse button is pressed for selection. For example, the first subtask (*Open File*) consists of five mouse movements: (a) from *center* to *File*, (b) from *File* to *Open File*, (c) from *Open File* to *Experiment*, (d) from *Experiment* to *Normalization*, and (e) from *Normalization* to *Choose*. The average time of mouse positioning ranged from 5.78 s to 2.12 s, which was recorded on the first day of training and the last day of training, respectively. We chose to use 2.12 s for the T_P parameter as expert performance. This positioning time is larger than the default time (1.10 s) that was used in the KLM-GOMS analysis (see Card, Moran, & Newell, 1983, p. 264), and can of course vary based on target sizes, which are relatively small in this task.

Table 3 describes how many operators were used for the task in terms of each modality group. The predicted task completion time for users in the keyboard group was 666.67 s, and the predicted task completion time for users in the mouse group was 797.14 s. The vertical mouse played a role in increasing the task completion time. If we assume participants use a regular mouse, T_P = 1.10, the predicted task completion time would be 672.70 s.

Table 3. *KLM-GOMS analysis on the Dismal spreadsheet task by the two modality groups.*

	Keystroke-driven		Menu-driven (vertical)		Menu-driven (regular)	
Operators	Operators	Time (s)	Operators	Time (s)	Operators	Time (s)
Keystroke (K)	981	461.07	540	313.20	540	313.20
Positioning (P)	0	0	122	258.64	122	134.20
Homing (H)	1	0.40	57	22.80	57	22.80
Mental (M)	152	205.20	150	202.50	150	202.50
Predicted Time		666.67		797.14		672.70

This analysis predicts that the keystroke-driven interface would take less task completion time (i.e., around 130 s less) than the menu-driven interface. It can also be argued that the efficacy of a menu-driven interface is not so dramatically greater to lead to choosing to use a menu-driven interface, compared to a keystroke-driven interface. On the contrary, it is predicted that the menu-driven interface using a regular mouse has a similar predicted task completion time (i.e., only 6 s more) than that of the keystroke-driven interface, suggesting that a menu-driven interface is efficient. However, this comparison is based on a large number of assumptions. The two predictions are at least close to each other, almost but not quite within +/-10%. Furthermore, with knowledge disuse, the variance in user performance would increase, but this is not predicted by the KLM-GOMS framework. We will seek to resolve these prediction conflicts and limitations, and to provide a test of these predictions empirically.

3.4 Does the Vertical Mouse Provide New Motor Skills to Learn?

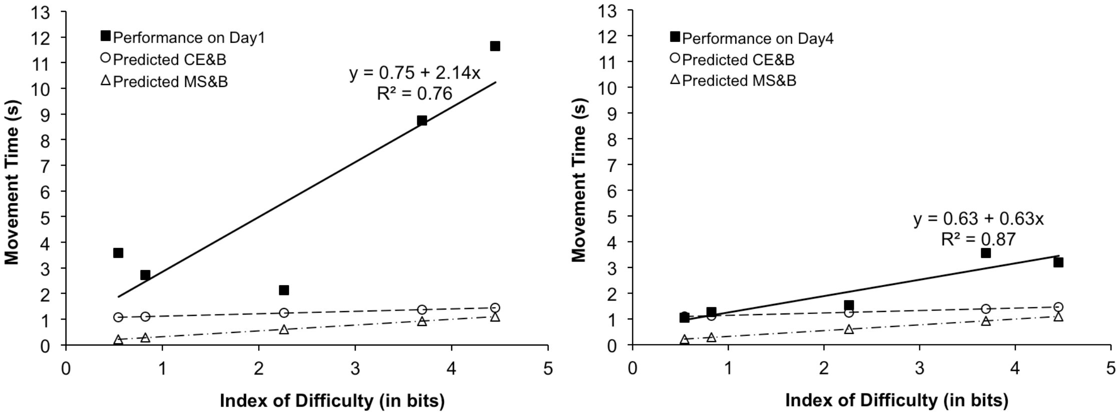
The KLM-GOMS analysis leads to a question of why the mouse users are predicted to have a longer task completion time. To further investigate this, the index of difficulty (*ID*) of the first subtask (*Open File*) was analyzed from the 30 mouse users' data. We chose to use Eq. (7) proposed by MacKenzie (1989) for this analysis because it prevents *ID* in Eq. 7 from being negative—some of the movements in our experiment are small, resulting in a negative time prediction using other formulas.

$$MT = a + b\log_2(A / W + 1) = a + bID \qquad \text{Eq. (7)}$$

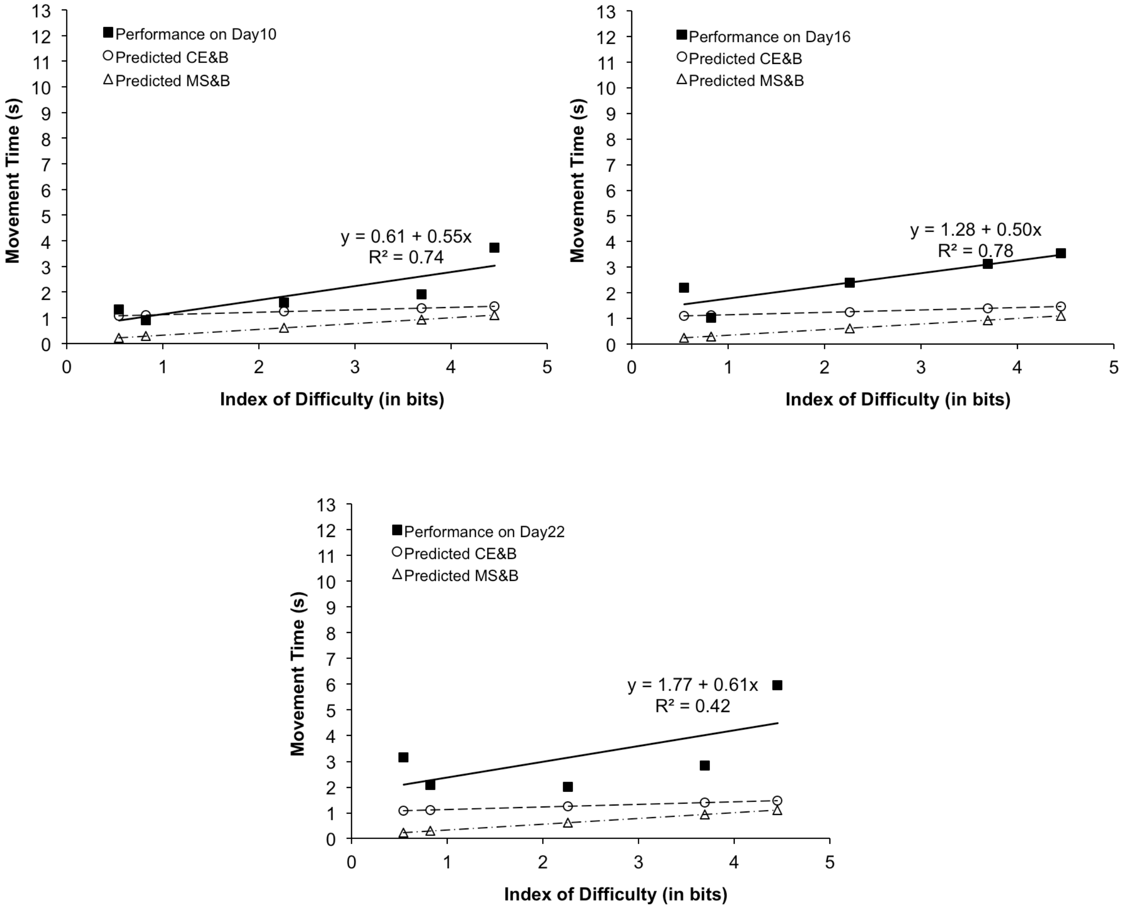
The first subtask (*Open File*) consists of mouse activities of pointing to five menu items—the *File* menu item, the *Open File* menu item, the *Experiment* folder, the file named *normalization.dis*, and the *Choose* button. For example, the *File* menu item from the home location that is the center of the task screen, around x = 840, y = 525 pixels. The amplitude (A) is 877 pixels, that is, the distance from home to the object (*File*). The target width (W) is 42 pixels.

To examine learning with the vertical mouse, we examined the first subtask of the mouse participants. Figure 4 shows the mouse movement time versus index of difficulty in completing the first subtask (*Open File*) during the learning sessions (Day 1 and Day 4) and the forgetting sessions (Day 10, Day 16, and Day 22). No participants (*n* = 30) had prior experience using the vertical mouse.

Figure 4(a) shows that the vertical mouse is harder to use than a regular mouse used in the previous studies because the measured times are greater than the predicted mouse times. In the later learning stage shown in Figure 4(a), the participants seemed to have learned the new motor skill, showing the movement time is getting closer to the other predicted times (e.g., Card, English, & Burr, 1978; MacKenzie, 1992; MacKenzie, Sellen, & Buxton, 1991) through changes in both the intercept and the index of difficulty constants. Figure 4(b) shows increase in the mouse movement time by different retention intervals. Consequently, we conclude that the vertical mouse is novel enough to study learning and forgetting.



(a) Learning sessions: User performance on Day 1 and 4.



(b) Forgetting sessions: User performance on Day 10, 16, and 22

Figure 4. The average movement time of the vertical mouse versus index of difficulty. The linear model “Predicted CE&B” is $MT = 1.03 + 0.096\ ID$, $R^2 = 0.83$ by Card, English, and Burr (1978). The linear model denoted by “Predicted MS&B” is $MT = 0.107 + 0.223\ ID$, $R^2 = 0.98$ by MacKenzie, Sellen, and Buxton (1991).

4 Method

We conducted an experiment to investigate how participants learn the Dismal spreadsheet task and retain the acquired task knowledge to perform the task after a period of skill disuse. Each participant had four learning sessions and was tested on one of three retention schedules.

4.1 Participants

Students at the Pennsylvania State University (N = 78) were recruited for this experiment and received monetary compensation for participation. The first 6 participants were the pilot participants—based on their performance data, we chose retention intervals (i.e., a 6-day, 12-day, and 18-day) to obtain the data explaining the progression of performance degradation without catastrophic memory failure. 12 participants could not complete the multiple sessions because of a personal time conflict (e.g., an unexpected job interview). A total of 60 (42 males, 18 females, aged 18-35) completed all of the sessions including learning, forgetting, and relearning. No participants had previous experience with the Dismal spreadsheet and this task. All the mouse group participants reported no experience with the vertical mouse. No participants reported experience using Dismal or Emacs and its keystroke commands.

4.2 Materials

Participants used a Macintosh computer with a keyboard or a vertical mouse and keyboard. The task completion times, keystrokes, mouse clicks (pressed and released), and mouse movements (e.g., xy coordinates of mouse locations in pixels) were recorded by the Recording User Input (RUI) keystroke logger (Kukreja, Stevenson, & Ritter, 2006; Morgan, Cheng, Pike, & Ritter, 2013). During the experiment we resolved a problematic issue of data-gathering on public machines (e.g., in a classroom). University policies should and Penn State's policy does prohibit installing application software that can obtain a user's identifying information (e.g., a login id or password) on public machines. This concern prohibits the RUI keystroke logger from being installed on classroom computers. Thus, we installed RUI and a shell script program on a memory stick, providing a way to efficiently and safely use RUI on public machines (Kim & Ritter, 2007).

4.3 Design

We examined two independent factors on learning. The first factor was input modality—the keystroke-driven modality (the keyboard group, K) and the menu-driven modality (the mouse group, M). The second factor was retention interval (R)—a 6-day retention interval (R6), 12-day retention interval (R12), and 18-day retention interval (R18). The retention interval (R) variable indicates a period of knowledge and skill disuse between the last learning (or practice) on Day 4 and the first return day for the forgetting test. With these retention intervals, we did not observe catastrophic memory failures, where a participant could not perform the task. We only observed a nonlinear increase of the task completion time. We did not include the retention interval of 32 days (around 1 month) because we observed a user's catastrophic memory failure with a one-month retention interval in a pilot study.

All participants were randomly assigned to the 6 conditions (10 participants per condition) shown in Table 4—the combination of the three retention intervals (i.e., R6, R12, and R18), and the two modality conditions (K and M). During the retention period participants were asked not to do mentally rehearsal or practice the task.

Table 4. *Learning and testing schedules.*

Group	Modality	Retention	Learning				Testing (and relearning opportunity)		
			Day 1	Day 2	Day 3	Day4	Day 10	Day 16	Day 22
Group 1	K	K-R6	X	X	X	X	X	X	
Group 2	M	M-R6	X	X	X	X	X	X	
Group 3	K	K-R12	X	X	X	X		X	X
Group 4	M	M-R12	X	X	X	X		X	X
Group 5	K	K-R18	X	X	X	X			X
Group 6	M	M-R18	X	X	X	X			X

4.4 Procedure

Participants in all 6 groups completed a series of learning sessions for four consecutive days from Day 1 to Day 4. In the learning session, participants used the study booklet to learn the task knowledge. The duration of each study session was less than 30 minutes. After participants studied the booklet, they performed the Dismal spreadsheet task. During this learning session, they had access to the study booklet. For example, on Day 1, participants had a maximum of 30 minutes to study the given spreadsheet task and then performed the Dismal spreadsheet task. On Days 2 to 4, participants were allowed to refresh their acquired knowledge from Day 1, using the study booklet, and then performed the task.

After completing the learning session, participants returned for a later measure of forgetting with a 6-day, 12-day, or 18-day time interval. In these sessions participants completed a test session only; that is, participants completed the Dismal spreadsheet task without the aid of the study booklet. Participants returned for a test on Day 10 (Group 1 and 2), on Day 16 (Group 3 and 4), and on Day 22 (Group 5 and 6). All participants were instructed to stop the experiment if they could not perform the task—i.e., they could not retrieve how-to-do-it task knowledge. In this case, the experimenter recorded the performance of a participant as “incomplete due to catastrophic memory failure”.

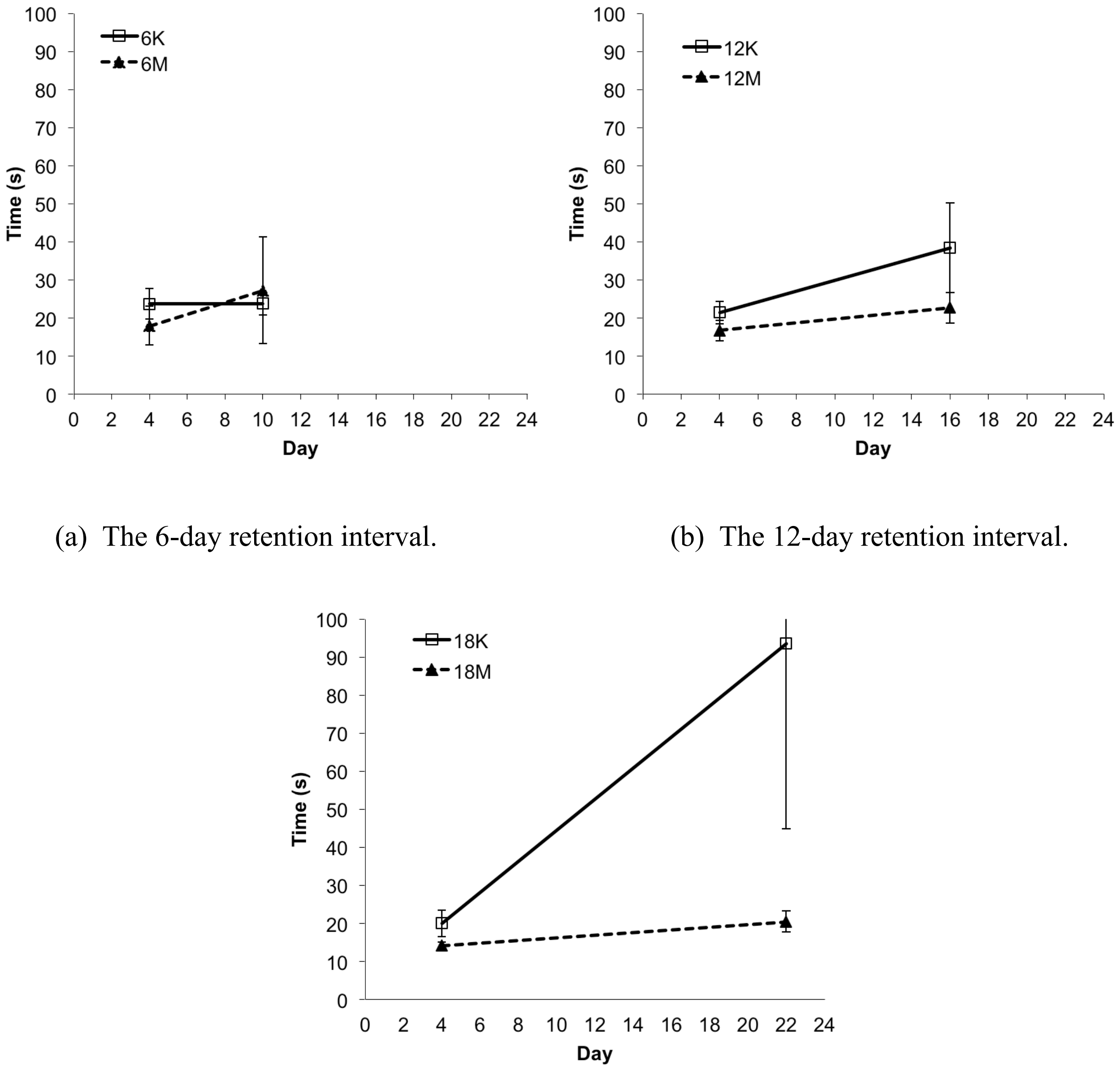
After the forgetting test, participants in Group 1, 2, 3, and 4 were asked to return for an additional test after an additional 6-day retention interval. They were in the R6 and R12 groups including both keystroke-driven and menu-driven participants.

5 Results

All of the participants completed the task without catastrophic memory failures. We break the analysis into three main sections: learning, forgetting, and relearning. But we first start by testing if the main manipulation of key- vs. menu-driven interface was effective.

5.1 Testing the Basic Manipulation

We observed different task completion times for the first subtask by the retention intervals and by the modality groups as shown in Figure 5. Under the 6-day retention condition, the mouse group showed more increase in time to complete the task than the keyboard group (52% increase for the mouse group, and 0.4% for the keyboard group). Under the 12- and 18-day retention intervals, the keyboard group showed more increase in time to complete the task than the mouse group. The percent increase for the keyboard group is 79% under the 12-day retention, and 368% under the 18-day retention. The percent increase for the mouse group is 37% under the 12-day retention, and 45% under the 18-day retention.



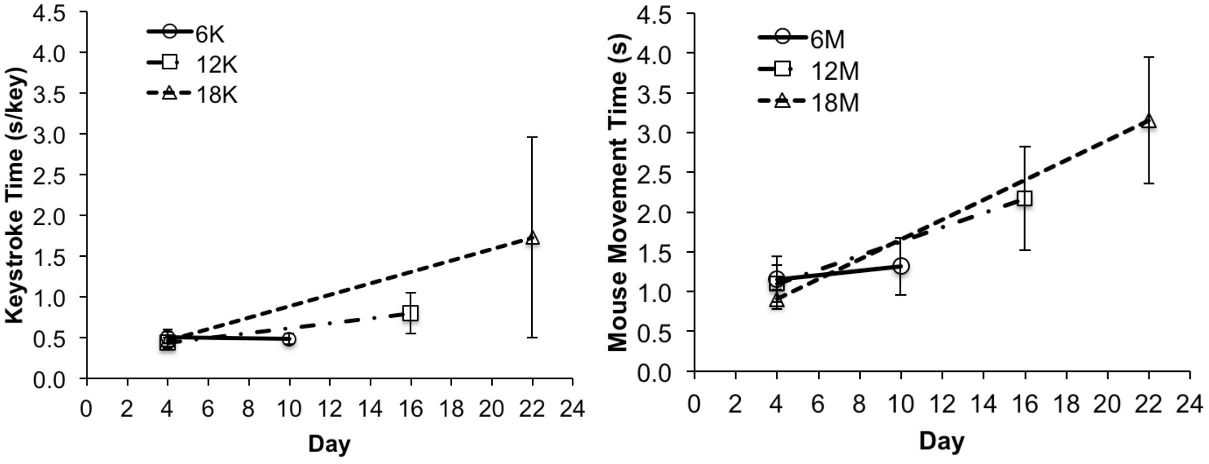
(a) The 6-day retention interval.

(b) The 12-day retention interval.

(c) The 18-day retention interval.

Figure 5. The task completion time of the first subtask (*Open File*) by the two modality groups. The graph shows the mean task completion times (*n* = 10) and the SEM.

Based on this, we analyzed the keystroke time and the mouse movement time as a way to identify what has been learned and forgotten. In completing the first subtask, the keyboard group needs 33 key presses and the mouse group moves the mouse pointer to menu items (e.g., File, Open File, a folder, a file, Choose) and makes clicks 122 times. We observed that the keystroke and mouse movement time for the first interaction (moving the mouse from *File* to *Open File*) increased as the retention intervals increase, shown in Figure 6.



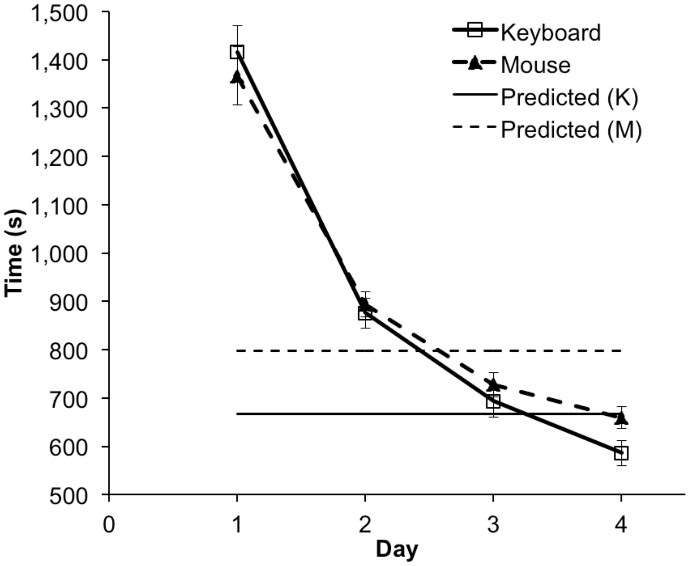
(a) Keystroke times.

(b) Mouse movement times.

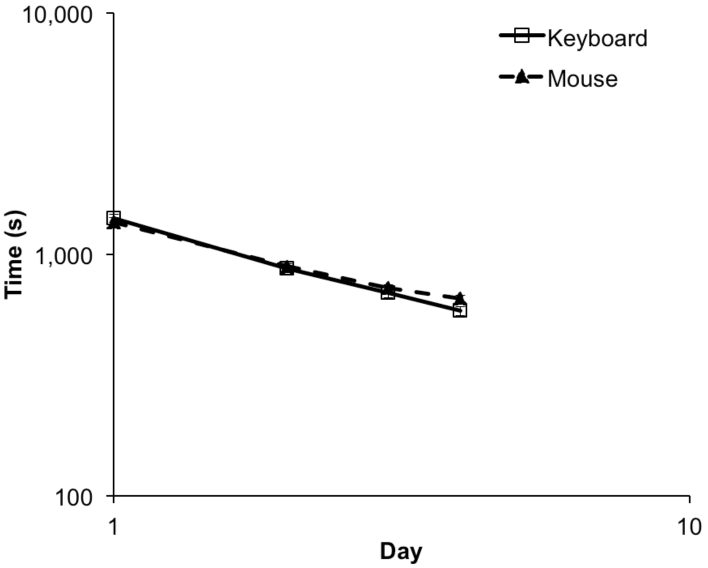
Figure 6. Microgenetic changes of the keystroke and mouse movement time for the first interaction by retention interval, 10 participants for each group. Error bars are standard errors of the mean (SEM). Due to skill disuse, the variance in performance increases.

5.2 Learning

The two modality groups showed similar learning performance as shown in Figure 7 — practice leads to faster task performance for both groups. After Day 2, it was observed that there was a crossover effect on the performance between the keyboard and mouse groups. The keyboard group participants were slower to complete the task than the mouse group participants at the first learning session on Day 1 (1,415 s for the keyboard group and 1,366 s for the mouse group), but gradually became faster at the last learning session on Day 4 (585 s for the keyboard group and 659 s for the mouse group). That is, the difference of the task completion time by the two modality groups was approximately 74 s on Day 4, which is statically significant as shown in a follow-up test.



(a) Task completion times for four learning sessions (linear plot).



(b) The log-log plot of learning data.

Figure 7. Learning performance of the keyboard and mouse modality groups. Error bars indicate the SEM. Straight lines in Figure 7 (a) are KLM-GOMS predictions from Table 3.

After completing all the learning sessions, both the keyboard and mouse groups appeared to reach the expert level in terms of the KLM-GOMS predictions (667 s for the keyboard group and 797 s for the mouse group), shown in Figure 7 (a). Figure 7 (b) shows the mean task completion time on a log-log scale (Newell & Rosenbloom, 1981), confirming the power law of learning ($y = 396.9x^{-0.6}$, R^2 = 0.99, for the keyboard group; $y = 1{,}338.1x^{-0.5}$, R^2 = 0.99, for the mouse group).

A mixed ANOVA was conducted to assess statistical significance on the task performance by the four-day learning sessions and the two modality groups. The within subjects variable was the four-day learning session, and the between subjects factor was the two modality groups (K and M). The Shapiro-Wilk test (a test for normality) revealed that some data sets (i.e., the mouse group on Day 1 and Day 3, and the keyboard group on Day 2 and Day 3) had non-normal distributions. So, we performed a log transformation of the data, and they became normally distributed. Because the repeated measures variable had four levels (four days of learning sessions), we also checked the sphericity assumption. The results of Mauchly's sphericity test revealed that the variable violated the sphericity assumption. Thus, we analyzed the data with the Huynh-Feldt correction (modifying the degrees of freedom). We found a significant interaction effect between the modality and the learning sessions, $F(2.33, 134.96) = 4.85, p < .05$, η_p^2 = .08.

We compared a limited number of pairs of means by the modality groups and the learning sessions. The assumption of equal variances was met. Our follow-up test of contrasts revealed that the task completion times by the modality groups were not significantly different from Day 1 to Day 3, $t(58) = .68$, $p >$ .05 for Day 1, $t(58) =$ -0.59, $p >$ .05 for Day 2, and $t(58) = -1.10, p > .05$ for Day 3. Interestingly, we found that, on Day 4, the task completion time of the keyboard group was significantly different from that of the mouse group, $t(58) = -2.3$, $p <.05$.

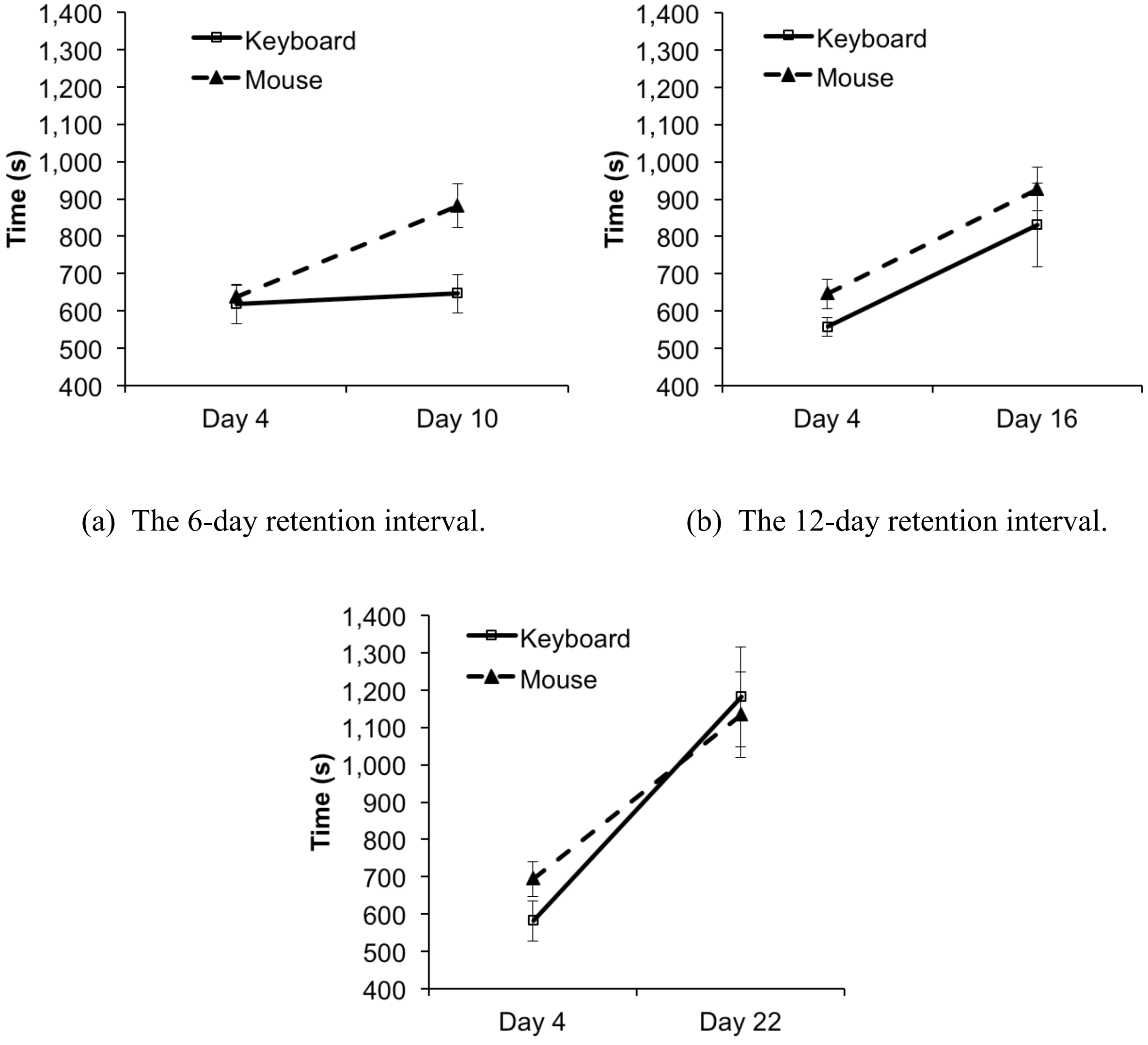
5.3 Forgetting

The forgetting performance measured in time was examined with a 2×3×2 mixed ANOVA (test sessions [the last learning, the first return] × retention intervals [6, 12, 18] × modality [K, M]). The Shapiro-Wilk test revealed that the task completion times were non-normally distributed. We, therefore, conducted a log transformation of the data, and the data became normally distributed. The sphericity assumption was met in that the repeated measures variable consisted of only two levels (the epsilons are 1). In all cases, catastrophic memory failure was not observed, in which participants could not complete the task, but the task completion time increased for all retention intervals. We only found a significant interaction effect between the retention intervals and the test sessions, $F(2, 54) = 6.76, p < .05, \eta_p^2 = .2$.

To further investigate, we computed contrasts to compare the task completion times of the keyboard and mouse groups at each of three retention intervals. The assumption of equal variance was met for both the last learning and the first return session. Unlike the analysis of learning ($n = 30$ for each modality group), we divided all the groups both by the retention intervals and by the modality groups, which leads to 10 participants for each data point in Figure 8. The task completion times on the last learning session (Day 4) for each retention interval and each modality group were not significantly different; for the 6-day retention, $t(54) = - 0.57$, $p = .57$, for the 12-day retention, $t(54) = - 1.39$, $p = .17$, and for the 18-day retention, $t(54) = - 1.90, p = .06$.

At the 6-day retention interval, the mouse group participants showed a statistically significant greater task completion time than the keyboard group, t $(54) = -2.27$, $p < .05$, as shown in Figure 8 (a). The task completion time on Day 10 was 646 s for the keyboard group and 882 s for the mouse group, a difference of around 236 s. This indicates that knowledge and skill degradation in the mouse group was triggered earlier than the keyboard group under the 6-day retention interval.

All other task completion times on the first return sessions (Day 16 and 22) were not significantly different between the keyboard and mouse groups. As shown in Figure 8 (b) and (c), they showed similar forgetting patterns. Interestingly, concerning learning, at the last practice session (Day 4) the keyboard users completed the task faster than the mouse group users, but, under the 18-day retention interval, the users with the keyboard group took a longer time to complete the task than the mouse group shown in Figure 8 (c). The keyboard group took around 48 s longer to complete the task, which might indicate another possible crossover effect for forgetting between the modality groups. The keyboard group seemed to take longer to retrieve knowledge from memory as the retention interval increased.



(a) The 6-day retention interval.

(b) The 12-day retention interval.

(c) The 18-day retention interval.

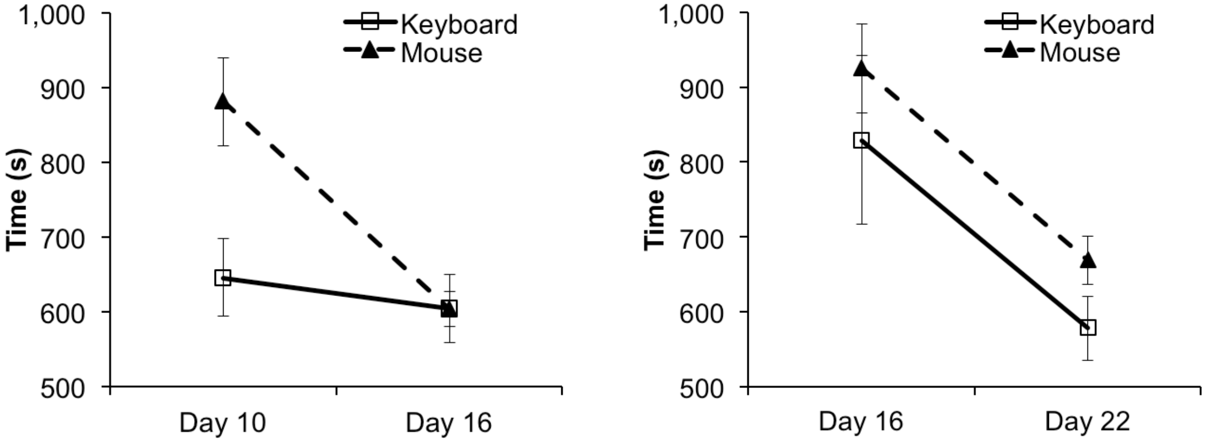
Figure 8. Performance on the last learning session and the first return for the forgetting measure after a retention interval (*n* = 10 per point). The error bars indicate the SEM.

5.4 Relearning

The additional test data was gathered with a 6-day lag after the first return sessions at each retention interval (a 6- or 12-day retention). Data were examined with a 2×2×2 mixed ANOVA (test sessions [the first return, the additional return] × retention intervals [6, 12] × modality [K, M]). The Shapiro-Wilk test revealed that a group of task completion times was non-normal. We, therefore, conducted a log transformation of the data, and the data became normally distributed. The sphericity assumption was met in that the repeated measures variable consisted of only two levels (the epsilons are 1).

The first return sessions for the forgetting test served as an opportunity for participants to relearn how to perform the task. Thus, we analyzed any effect of this relearning from the previous test. For example, some participants (Group 1 and 2) attended the learning sessions from Day 1 to Day 4, returned on Day 10 for the forgetting test, and returned again on Day 16 for the relearning test, shown in Figure 9 (a). Similarly, other participants (Groups 3 and 4) attended the learning session from Day 1 to Day 4, returned on Day 16 for the forgetting test (i.e., the 12-day retention interval), and returned again on Day 22 for the relearning test, shown in Figure 9 (b).

As noted, at the 6-day retention interval, the task completion time of the mouse group increased more than that of the keyboard group. The forgetting test on Day 10 seemed to serve as a relearning opportunity because the task completion time of the mouse group decreased to the keyboard group task completion time on Day 16, shown in Figure 9 (a). Under the 12-day retention, both the keyboard and mouse group showed similar increase in the task completion time, and the two groups similarly decreased in their task completion times.



(a) The 6-day retention for forgetting and the 6-day retention for relearning (Group 1: Keystroke-driven, and Group 2: Menu-driven).

(b) The 12-day retention for forgetting and the 6-day retention for relearning (Group 3: Keystroke-driven, and Group 4: Menu-driven).

Figure 9. The performance on the first return day and an additional return day. The error bars indicate SEM.

A mixed ANOVA analysis revealed whether there were significant differences in performance of the relearning groups. Significant interaction effects were found between the sessions (the first return and the additional return) and the modality groups, $F(1, 36) = 5.43$, $p < .05$, $\eta_p^2 = .13$. Furthermore, it was also found that there was a significant interaction between those factors by the retention intervals for the first return, $F(1, 36) = 4.92, p < .05$, $\eta_p^2 = .12$.

To further examine the statistical significance of differences among the factors, we examined simple effect contrasts to compare some preselected pairs of means of interest. For example, we compared the means of the task completion times on Day 10 and Day 16, and the means on Day 16 and Day 22 by two modality groups. The assumption of homogeneity was violated ($p < .05$), and, thus, our contrasts analysis did not assume equal variances.

Under the 6-day retention the task completion times by the two modality groups were significantly different only on Day 10, $t(16.6) = -3.0$, $p < .05$, but not significantly different on

Day 16. That is, the first 6-day retention interval caused the mouse group users to show more performance degradation (i.e., increase in task completion time), but the test session for the forgetting measure on Day 10 served as an opportunity for the mouse group users to have a faster recovery. This indicates that the mouse group users can quickly relearn the task knowledge compared to the keyboard group under a shorter retention (a 6-day) interval. Other task completion times under the 12-day retention did not show statistically significant differences on Day 16 and Day 22.

6 Discussion and Conclusions

We investigated how performance change occurs with learning, forgetting, and relearning from the perspective of how a user learns to perform a task (i.e., the Dismal spreadsheet task) and of how the acquired knowledge is maintained over time. Both the keystroke-driven and the menu-driven modality users learned to perform the Dismal spreadsheet task. These two modality groups progressed through learning in a similar pattern shown earlier in Figure 7, following the power law of learning.

At the beginning, the two modality groups showed learning patterns following the power law of practice, but, interestingly, on the last learning session (Day 4), the keyboard group users took reliably less time to complete the task than the mouse group users. The other reliable differences include that under the 6-day retention the forgetting and relearning performance were significantly different by the modality groups (shown in Figure 8 and 9).

Based on these results, after noting potential limitations, we discuss several theoretically relevant topics that can improve our fundamental understanding of human-computer interaction:

(a) a theoretically derived design principle for human-computer interaction of where to keep task knowledge, (b) the learning stages, and (c) a framework for a model of cognitive performance.

6.1 A Design Rationale: To Keep Knowledge in the Head or in the World?

Is there any scientific reason why people prefer to use a mouse-based menu-driven interface? We presume that the reason menu-driven systems are ubiquitous is because everyday objects should be designed for ease of use, and a menu-driven interface actually provides ease of use. In reality, GUIs may have been widely adopted by system designers to improve ease of use because many academic interface designers have argued for the superiority of menu-driven interfaces over keystroke-driven interfaces (e.g., Sears & Shneiderman, 1994; Shneiderman, 1983). But, are they easier to use? The KLM-GOMS predictions and the data on this task show that the keystroke interface is learnable and also faster to use under uninterrupted and continuous practice training curves. Why do people seem to prefer mouse driven interfaces then?

Norman (1988) introduced the terms, *knowledge-in-the-head* and *knowledge-in-the-world*, providing a fundamental design principle for an interactive system—i.e., placing the knowledge in the world might be helpful in reminding, than placing the knowledge in the head. As an extended theoretical account of this approach, Gray and his associates coined a term, the *soft constraints hypothesis* (Gray, Sims, Fu, & Schoelles, 2006), providing an understanding of how cognitive resources—cognition, perception, and motor modules—are allocated for interactive behavior based on a rational analysis approach (e.g., Anderson, 1991; Oaksford & Chater, 1994).

The soft constraints hypothesis assumes that interactive routines that consist of cognition, perception, and motor operations are selected to minimize performance cost measured in time. The soft constraints hypothesis proposes that the tradeoff between interaction-intensive and memory-intensive strategies in the memory system depends on factors including the time required to encode an item in memory, the time required to retrieve the item from memory, and the probability of retrieving the encoded item from memory (if retrievable). As a counterpart of the soft constraints hypothesis, the *minimum memory hypothesis* suggests that people favor strategies that minimize memory load. Wilson (2002) argues that people tend to reduce cognitive workload by putting it onto the environment because of the limited capacity of human information processing (e.g., the limited capacity of attention or memory). Cary and Carlson (1999) also support the view that people tend to minimize working memory demands in problem-solving routines—the limited capacity of working memory would restrict possible solution paths in a problem-solving task. The minimum memory hypothesis does not, however, account for performance costs that are sensitive to time.

In our investigation, the keyboard group users learned the task knowledge with greater reliance on knowledge retrieval from memory; this task interface relied more on *knowledge-in-the-head*, a memory intensive strategy. The mouse users learned the task knowledge with greater reliance on interaction with the environment; this task relied more on *knowledge-in-the-world*, an interaction-intensive strategy.

Based on our data, the interaction-intensive and memory-intensive strategies did not produce significantly different patterns of learning in the Dismal spreadsheet task except for the performance on Day 4. Interestingly, as noted earlier, the keyboard users completed the task faster than the mouse users on the last learning session (Day 4). This result indicates that practice can change performance response—users with the memory-intensive strategy had a faster task time than those with the interaction-intensive strategy in a later stage of learning. Gray and his associates (2006) observed in their experiments “a progressive switch from more interaction-intensive to more memory-intensive strategies as information access costs [in the interface] increased”. This phenomenon of switching from one strategy to another suggests that the interaction- and memory-intensive strategies could produce different performance responses in terms of the progression of the learning stage—a novice or an expert might choose different strategies.

The mouse group users (i.e., the interaction-intensive strategy) not only conserved their memory use but also quickly reduced their task time. That is, under a short-term retention interval (i.e., a 6-day retention in our experiment), the newly learned motor or cognitive skills using the vertical mouse seemed to be degraded more than the motor skills using a keyboard—or, less or different learning of the menu knowledge. In the mean time, the memory retrieval performance seems not to be much degraded as we can see the keyboard group users retained relatively more after this short-term retention interval. We also observed that the mouse group users showed a greater increase in task completion time with a 6-day retention interval after the learning session than the keyboard group users. However, the mouse group users recovered their performance by completing the test session for a later measure after the 6-day retention interval, which was approximately the same level as the keyboard users as shown in Figures 8 (a) and Figure 9 (a). Furthermore, when it comes to forgetting, there might be a stage when the keyboard and mouse groups produce approximately similar (nearly identical) performance (i.e., the 12- and 18-day retention intervals in our experiment). The (re)learning of the menu-driven interface users after forgetting arose partly because they had forgotten, but it might also be the case that this relearning is pleasant and leads the users to remember learning on the interface (see Cacioppo & Petty, 1982). Disentangling these concepts is too complex to do with our data, but we note them.

In a theoretical sense, there may be a period in learning where a skill performed with procedural and motor skill knowledge is more robust against decay than the equivalent declarative memory representation. This indicates that the menu-driven modality can help the user to relearn how to perform a task. That is, in our everyday interaction, the menu-driven modality can allow the user not only to perform the task as a novice but also to complete a task even after an arbitrarily long time lag of knowledge disuse. This implication may help explain why menu-driven interfaces (e.g., GUIs) are well accepted in interactive system designs even though they are slower. That is, they are used not because they are more learnable, which has often been argued, but because they are more re-learnable. The curves also show that relearning is also occurring and is important, not just basic learnability or usability.

It is reasonable to view interaction style preference as a result of memory activation (see Weber & Johnson, 2006). That is, the priming effect (i.e., shorter reaction time, decrease in task completion time, and greater likelihood of memory retrieval) helps memory accessibility, leading to the preference. A strong relationship between memory activation and selection of a product has been noted before (e.g., North, Hargreaves, & McKendrick, 1997).

The ubiquitous preference for GUIs suggests that the users prefer perceptual-motor effort to memory retrieval. That is, the users would choose this minimum memory strategy (i.e., knowledge-in-the-world that can be also represented as a GUI) in their task completion, even though memory retrieval saves time, and they have been instructed to complete the task as quickly as possible (e.g., Ballard, Hayhoe, Pook, & Rao, 1997).

In the meantime, Gray and his associates (Gray & Fu, 2004; Gray, Sims, Fu, & Schoelles, 2006) argue that a user would choose one set of interactive routines (i.e., a pattern of cognitive, perceptual, and motor operations) over another as a cost-benefit tradeoff, serving as soft constraints that are only sensitive to the expected utility in time. For example, more practice can support keystroke-driven users to have faster task completion times, suggesting that people would prefer to use the keystroke-driven interface, but only if there is a long enough learning period. The soft constraints hypothesis can also explain why some expert users prefer not to use a menu-driven interface but keystrokes, not only are they faster but they stay that way if relearning is not necessary.

Some expert users might favor keystroke-driven interfaces relying on knowledge in-the-head, but their forgetting could be greater if the knowledge is not proceduralized. In our experiment we could not see this effect clearly, but the trend in performance is suggestive—we observed that the keyboard group users' task completion time gets relatively slower (but not significantly) after an 18-day retention interval as shown in Figure 8(c). If we assume the power law of forgetting (e.g., Anderson & Schooler, 1991), retrieval of knowledge would be degraded in a nonlinear pattern as the retention interval increases. Consequently, the memory-intensive strategy may be forgotten more with longer retention intervals. When it comes to the soft constraints hypothesis, people would prefer to use an interaction-intensive interface requiring less knowledge retrieval because the keystroke-driven modality requires more information processing capacity of knowledge retrieval, taking longer to complete the task. At the least, keystrokes should be able to be relearned through the interface.

This lesson can be applied to various industrial sectors where learning by human operators is important. For example, these results will be applicable to surgical task knowledge learning. Medical students would progress through a learning (and forgetting) curve to reach expertise as shown in Figure 2. In the meantime, they might lose what they have learned from memory and they might want to conserve memory, favoring an interaction-intensive strategy in their task rather than a recall-intensive strategy. On the other hand, if they interact often and interaction time is important, supporting a recall-based strategy or both strategies would be important. Supporting both strategies is often possible in interfaces, for example, through menus and keystroke commands. These results suggest that during system design optimal strategies and how to support their use should be given consideration with respect to learning and forgetting. These results also provide some preliminary suggestions for the learning and forgetting intervals for training and use.

6.2 Potential Limitations of Our Study

Our study has several limitations. We tried to clearly distinguish the two modality groups, but the tasks in the modality groups remain a bit tangled—The menu-driven task requires both input using a mouse as well as keystrokes (for textual input). The menu-driven modality is done with a vertical mouse. It could be redone with keystroke-driven menus and also with a normal mouse and a trackpad.

The current study consists of only one study task including learning, forgetting, and relearning with three retention intervals. More tasks could be examined. Even though the study includes learning, forgetting, and relearning sessions, further sessions with different periods would help explore these effects including relearning across different practice schedules.

Also, if we could obtain results with much longer retention intervals, the study could provide different effects and an enhanced understanding. Obtaining retention data is expensive but it is worth obtaining to improve our understanding of learning. We also would like to note that different forms of interaction styles (e.g., voice or movement) could produce different results.

6.3 The Learning Stages

The data in this study can be compared back to the learning theory summarized in the introduction. The data over the course of the learning sessions roughly follow an expected learning curve noted in Figure 1. The forgetting and relearning sessions also seem consistent with the theory shown in Figure 2.

The early performance data does not match the KLM-GOMS predictions because the participants are relatively novices at the task. But, with practice over 4 trials, they match fairly well the KLM-GOMS predictions.

With the learning curve and the KLM-GOMS analysis in mind, we can suggest that performance after 4 trials is likely to improve further. We can also see that that performance is not a monolithic construct at that point. That is, there are 14 subtasks, with over 800 sub-actions in each interface. The participants are unlikely to have each subtask at the same level of learning. This learning will vary by the component and the learner. The resulting times represent sum of the distribution of how well they have learned each of the subtasks. Some components will be in Stage I, some in Stage II, and some in Stage III. To understand this effect, of learning varying by subtask, we will have to see further analyses and perhaps even further measures such as fMRI.

6.4 A User Model Framework: From KLM-GOMS to ACT-R and Beyond

The KLM-GOMS framework predicts the expert performance of the keystroke-driven and the menu-driven users, which can be observed in the later stage of learning (Figure 2). The keystroke-driven users are predicted to exhibit a slightly faster performance than the menu-driven users. In our data, the average task completion time for the keyboard group users was faster than the mouse group users on the last learning session (on Day 4). However, the participants in the later stage of learning took much less time to complete the Dismal task than the KLM-GOMS predictions. We also saw the KLM-GOMS predictions match the data after the learning trial on Day 3. We did not examine or control for previous knowledge, although users reported no experience with the interface. So, for similar tasks with similar amounts of transfer, 3 repeated practice trials might be a good heuristic for creating experts for comparison to real users with a less regular practice schedule, and then the KLM-GOMS analysis on this task predicts too long a time if the user has additional practice.

The retention curves in this study suggest that novices will be well predicted for trials 3-5 and then be faster than the predictions, depending on the training interval and task. Also, these results suggest that the KLM-GOMS framework may better predict experts with breaks in practice rather than those continuously performing a task—The KLM-GOMS prediction is a constant and the curve shows that users will produce a curve as they continue to practice with experts in a band.

There are several possible explanations for the over-prediction. One explanation is that the discrepancies between the KLM-GOMS framework and the data might be ascribed to the timing characteristics of the KLM-GOMS operators. In Table 3, the number of mental operators (*M*) is 152 ms for the keystroke-driven modality and 150 ms for the menu-driven modality; these numbers are close to each other. In our analysis, we assumed each menu choice requires one mental operator. The number of keystroke operators (*K*) from the two modalities is quite different; the number of keystroke operators is 981 for the keystroke-driven modality and 540 for the menu-driven modality. The timing characteristics of these operators might vary in terms of learning and forgetting. Also, other positioning (*P*) and homing (*H*) operators might be affected by learning and forgetting.

It is assumed that a time-consuming mental operation precedes each command. The mental operator (*M*) is interpreted as the time to retrieve a chunk of information from long-term memory into working memory. In ACT-R, this retrieval time is governed by the activation equation, Eq. (1), and the time to complete a task, therefore, follows a power function of the number of trials of practice, which KLM-GOMS does not.

It may be possible to extend the KLM-GOMS framework in an attempt to make some initial predictions of learning. These two interaction modalities have different numbers of mental operators and other operators in KLM-GOMS. If we assume that the mental operators will speed up with practice (not that the KLM-GOMS framework currently predicts this), then a very simple prediction is that an interface with more mental operators should get greater speed up with practice than the one with less. It has been reported that fewer mental operators might be involved in choosing a menu in a hierarchical menu system—some assume that a series of menu choices would involve only one mental operator (Lane, Napier, Batsell, & Naman, 1993). In our task, the first subtask, *Open File*, requires 5 mental operators for the menu-driven modality but requires 3 mental operators for the keystroke-driven modality. Thus, if we replace the 5 mental operators with the one mental operator for the first subtask because this menu selection task is fully anticipated as a cognitive unit (see Card, Moran, & Newell, 1983, p. 268), the skilled performance in the later stage of learning would take less time to do that subtask (*Open File*). This approach would move the KLM-GOMS framework towards ACT-R.

We acknowledge the limitation that our result is relatively weak evidence for this extension of the KLM-GOMS framework to predict learning, but a longer and deeper analysis across more interfaces and across subtasks (where there may be a finer view of the data) may help us explore how to model novice to expert transitions, and to understand where and how fast learning influences interfaces, and if KLM-GOMS is a reasonable place to start. Also, in this perspective, it might be necessary to differentiate the different types of mental operations when it comes to learning (e.g., Olson & Olson, 1990) and forgetting as well.

References

Anderson, J. R. (1982). Acquisition of cognitive skill. *Psychological Review, 89*(4), 369-406.

Anderson, J. R. (1987). Skill acquisition: Compilation of weak-method problem solutions. *Psychological Review, 94*(2), 192-210.

Anderson, J. R. (1991). Is human cognition adaptive? *Behavioral and Brain Sciences, 14*, 471-517.

Anderson, J. R. (1993). *Rules of the mind*. Hillsdale, NJ: Lawrence Erlbaum.

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of mind. *Psychological Review, 111*(4), 1036-1060.

Anderson, J. R., Conrad, F. G., & Corbett, A. T. (1989). Skill acquisition and the Lisp tutor. *Cognitive Science, 13*, 467-707.

Anderson, J. R., Fincham, J. M., & Douglass, S. (1999). Practice and retention: A unifying analysis. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 25*(5), 1120-1136.

Anderson, J. R., & Schooler, L. J. (1991). Reflections of the environment in memory. *Psychological Science, 2*(6), 396-408.

Ballard, D. H., Hayhoe, M. M., Pook, P. K., & Rao, R. P. N. (1997). Deictic codes for the embodiment of cognition. *Behavioral and Brain Sciences, 20*, 723-767.

Cacioppo, J. T., & Petty, R. E. (1982). The need for cognition. *Journal of Personality and Social Psychology, 42*(1), 116-131.

Card, S. K., English, W. K., & Burr, B. J. (1978). Evaluation of mouse, rate-controlled isometric joystick, step keys, and text keys for text selection on a CRT. *Ergonomics, 21*(8), 601-613.

Card, S. K., Moran, T. P., & Newell, A. (1983). *The psychology of human-computer interaction*. Hillsdale, NJ: Lawrence Erlbaum.

Cary, M., & Carlson, R. A. (1999). External support and the development of problem-solving routines. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 25*(4), 1053-1070.

Chen, H.-M., & Leung, C.-T. (2007). The effect on forearm and shoulder muscle activity in using different slanted computer mice. *Clinical Biomechanics, 22*, 518-523.

Chong, R. S. (2004). Architectural explorations for modeling procedural skill decay. In M. Lovett, C. Schunn, C. Lebiere & P. Munro (Eds.), *Proceedings of the Sixth International Conference on Cognitive Modeling*. [online edition] Mahwah, NJ: Lawrence Erlbaum.

Delaney, P. F., Reder, L. M., Staszewski, J. J., & Ritter, F. E. (1998). The strategy-specific nature of improvement: The power law applies by strategy within task. *Psychological Science, 9*(1), 1-7.

Ericsson, K. A., & Simon, H. A. (1993). *Protocol analysis: Verbal reports as data* (Revised ed.). Cambridge, MA: The MIT Press.

Fisk, A. D., & Hodge, K. A. (1992). Retention of trained performance in consistent mapping search after extended delay. *Human Factors, 34*(2), 147-164.

Fitts, P. M. (1964). Perceptual-motor skill learning. In A. W. Melton (Ed.), *Categories of human learning* (pp. 243-285). New York: Academic Press.

Gluck, K. A., Gunzelmann, G., Hudlicka, E., Gratch, J., & Ritter, F. E. (2006). Modeling the impact of cognitive moderators on human cognition and performance. In *28th Annual Conference of the Cognitive Science Society*. Vancouver, BC, Canada.

Gray, W. D., & Fu, W.-T. (2004). Soft constraints in interactive behavior: The case of ignoring perfect knowledge in-the-world for imperfect knowledge in-the-head. *Cognitive Science, 28*, 359-382.

Gray, W. D., Sims, C. R., Fu, W., & Schoelles, M. J. (2006). The soft constraints hypothesis: A rational analysis approach to resource allocation for interactive behavior. *Psychological Review, 113*(3), 461-482.

Gustafsson, E., & Hagberg, M. (2003). Computer mouse use in two different hand positions: Exposure, comfort, exertion, and productivity. *Applied Ergonomics, 34*, 107-113.

Houwink, A., Oude Hengel, K. M., Odell, D., & Dennerlein, J. T. (2009). Providing training enhances the biomechanical improvements of an alternative computer mouse design. *Human Factors, 51*(1), 46-55.

Johnson, S. L. (1981). Effect of training device on retention and transfer of a procedural task. *Human Factors, 23*(3), 257-272.

Jones, G., Ritter, F. E., & Wood, D. J. (2000). Using a cognitive architecture to examine what develops. *Psychological Science, 11*(2), 93-100.

Kieras, D. E., & Meyer, D. E. (1997). An overview of the EPIC architecture for cognition and performance with application to human-computer interaction. *Human-Computer Interaction, 12*(4), 391-438.

Kim, J. W., & Ritter, F. E. (2007). Automatically recording keystrokes in public clusters with RUI: Issues and sample answers. In D. S. McNamara & J. G. Trafton (Eds.), *Proceedings of the 29th Annual Cognitive Science Society* (p. 1787). Austin, TX: Cognitive Science Society.

Kim, J. W., Ritter, F. E., & Koubek, R. J. (2013). An integrated theory for improved skill acquisition and retention in the three stages of learning. *Theoretical Issues in Ergonomics Science, 14*(1), 22-37.

Kukreja, U., Stevenson, W. E., & Ritter, F. E. (2006). RUI: Recording user input from interfaces under Window and Mac OS X. *Behavior Research Methods, 38*(4), 656-659.

Lane, D. M., Napier, H. A., Batsell, R. R., & Naman, J. L. (1993). Predicting the skilled use of hierarchical menus with the keystroke-level model. *Human-Computer Interaction, 8*, 185-192.

Lee, F. J., & Anderson, J. R. (2001). Does learning a complex task have to be complex? A study in learning decomposition. *Cognitive Psychology, 42*(3), 267-316.

MacKenzie, I. S. (1989). A note on the information-theoretic basis for Fitts' law. *Journal of Motor Behavior, 21*, 323-330.

MacKenzie, I. S. (1992). Fitts' law as a research and design tool in human-computer interaction. *Human-Computer Interaction, 7*, 91-139.

MacKenzie, I. S., Sellen, A., & Buxton, W. (1991). A comparison of input devices in elemental pointing and dragging tasks. In *Proceedings of the CHI '91 Conference on Human Factors in Computing Systems* (pp. 161-166). New York, NY: ACM.

Morgan, J. H., Cheng, C.-Y., Pike, C., & Ritter, F. E. (2013). A design, tests, and considerations for improving keystroke and mouse loggers. *Interacting with Computers, 25*(3), 242-258.

Neves, D. M., & Anderson, J. R. (1981). Knowledge compilation: Mechanisms for the automatization of cognitive skills. In J. R. Anderson (Ed.), *Cognitive skills and their acquisition* (pp. 57-84). Hillsdale, NJ: Lawrence Erlbaum.

Newell, A. (1973). You can't play 20 questions with nature and win: Projective comments on the papers of this symposium. In W. G. Chase (Ed.), *Visual information processing* (pp. 283-308). New York: Academic Press.

Newell, A. (1990). *Unified Theories of Cognition*. Cambridge, MA: Harvard University.

Newell, A., & Rosenbloom, P. S. (1981). Mechanisms of skill acquisition and the law of practice. In J. R. Anderson (Ed.), *Cognitive skills and their acquisition* (pp. 1-55). Hillsdale, NJ: Lawrence Erlbaum.

Norman, D. A. (1988). *The design of everyday things*. New York: Basic Books.

North, A. C., Hargreaves, D. J., & McKendrick, J. (1997). In-store music affects product choice. *Nature, 390*(6656), 132.

Oaksford, M., & Chater, N. (1994). A rational analysis of the selection task as optimal data selection. *Psychological Review, 101*(4), 608-631.

Olson, J. R., & Olson, G. M. (1990). The growth of cognitive modeling in human-computer interaction since GOMS. *Human-Computer Interaction, 5*, 221-265.

Pavlik, P. I., & Anderson, J. R. (2005). Practice and forgetting effects on vocabulary memory: An activation-based model of the spacing effect. *Cognitive Science, 29*, 559-586.

Posner, M. I. (1973). *Cognition: An introduction*. Glenview, Illinois: Scott, Foresman and Company.

Rabbitt, P., & Banerji, N. (1989). How does very prolonged practice improve decision speed? *Journal of Experimental Psychology: General, 118*(4), 338-345.

Rasmussen, J. (1986). *Information processing and human-machine interaction: An approach to cognitive engineering*. New York: Elsevier.

Ritter, F. E., & Larkin, J. H. (1994). Developing process models as summaries of HCI action sequences. *Human-Computer Interaction, 9*, 345-383.

Ritter, F. E., & Schooler, L. J. (2001). The learning curve. In W. Kintch, N. Smelser & P. Baltes (Eds.), *International Encyclopedia of the Social and Behavioral Sciences* (Vol. 13, pp. 8602-8605). Amsterdam: Pergamon.

Ritter, F. E., & Wood, A. B. (2005). Dismal: A spreadsheet for sequential data analysis and HCI experimentation. *Behavior Research Methods, 37*(1), 71-81.

Rosenbloom, P., & Newell, A. (1987). Learning by chunking: A production system model of practice. In D. Klahr, P. Langley & R. Neches (Eds.), *Production System Models of Learning and Development*. Cambridge, MA: MIT Press.

Salthouse, T. A. (1986). Perceptual, cognitive, and motoric aspects of transcription typing. *Psychological Bulletin, 99*(3), 303-319.

Schmidt, R. A., & Bjork, R. A. (1992). New conceptualizations of practice: Common principles in three paradigms suggest new concepts for training. *Psychological Science, 3*(4), 207-217.

Sears, A., & Shneiderman, B. (1994). Split menus: Effectively using selection frequency to organize menus. *ACM Transactions on Computer-Human Interaction, 1*(1), 27-51.

Seibel, R. (1963). Discrimination reaction time for a 1,023-alternative task. *Journal of Experimental Psychology, 66*(3), 215-226.

Shiffrin, R. M., & Schneider, W. (1977). Controlled and automatic human information processing: II. Perceptual learning, automatic attending, and a general theory. *Psychological Review, 84*(2), 127-190.

Shneiderman, B. (1983). Direct manipulation: A step beyond programming languages. *IEEE Computer, 16*(8), 57-69.

Shute, V. J., & Gawlick, L. A. (1995). Practice effects on skill acquisition, learning outcome, retention, and sensitivity to relearning. *Human Factors, 37*(4), 781-803.

Singley, M. K., & Anderson, J. R. (1989). *The transfer of cognitive skill*. Cambridge, MA: Harvard University.

Stefanidis, D., Korndorffer, J. R., Markley, S., & Sierra, R. (2006). Proficiency maintenance: Impact of ongoing simulator training on laparoscopic skill retention. *Journal of the American College of Surgeons, 202*(4), 599-603.

Taatgen, N. A., & Anderson, J. R. (2002). Why do children learn to say "Broke"? A model of learning the past tense without feedback. *Cognition, 86*(2), 123-155.

VanLehn, K. (1996). Cognitive skill acquisition. *Annual Review of Psychology, 47*, 513-539.

Weber, E. U., & Johnson, E. J. (2006). Constructing preferences from memory. In S. Lichtenstein & P. Slovic (Eds.), *The construction of preference* (pp. 397-410). New York, NY: Cambridge University Press.

Wilson, M. (2002). Six views of embodied cognition. *Psychonomic Bulletin & Review, 9*(4), 625-636.