

Reducing Error Aversion to Support Novice-to-Expert Transitions with FastTap

Alix Goguey, Sylvain Malacria, Andy Cockburn, Carl Gutwin

► **To cite this version:**

Alix Goguey, Sylvain Malacria, Andy Cockburn, Carl Gutwin. Reducing Error Aversion to Support Novice-to-Expert Transitions with FastTap. Actes de la 31e conférence francophone sur l'Interaction Homme-Machine (IHM 2019), Dec 2019, Grenoble, France. pp.1:1-10, 10.1145/3366550.3372247. hal-02381584

HAL Id: hal-02381584

<https://hal.archives-ouvertes.fr/hal-02381584>

Submitted on 26 Nov 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Reducing Error Aversion to Support Novice-to-Expert Transitions with FastTap

Réduire l'Aversion aux Erreurs pour Aider la Transition Novice-Expert avec Fast Tap

Alix Goguey¹, Sylvain Malacria², Andy Cockburn³, Carl Gutwin⁴

¹Université Grenoble Alpes, LIG, France, ²Inria, Univ. Lille, UMR 9189 - CRISTAL, Lille, France,

³University of Canterbury, New Zealand, ⁴University of Saskatchewan, Canada

alix.goguey@univ-grenoble-alpes.fr, sylvain.malacria@inria.fr,

andy@cosc.canterbury.ac.nz, gutwin@cs.usask.ca

ABSTRACT

Expert interaction techniques such as gestures or hotkeys are more efficient than traditional WIMP techniques because it is often faster to recall a command than to navigate to it. However, many users seem to be reluctant to switch to expert interaction. We hypothesize the cause might be the aversion of making errors. To test this, we designed two *intermediate modes* for the FastTap interaction technique, allowing quick confirmation of what the user has retrieved from memory, and quick adjustment if she has made an error. We investigated the impact of these modes and of various error costs in a controlled study (N=36). We found that participants adopted the intermediate modes, that these modes reduced error rate when error cost was high, and that they did not substantially change selection times. However, while it validates the design of our *intermediate modes*, we found no evidence of greater switch to memory-based interaction, suggesting that reducing the error rate is not sufficient to promote expert use of techniques.

CCS CONCEPTS

• **Human-centered computing** → **Human computer interaction (HCI)**.

KEYWORDS

expertise, adjustment, intermediate modes, error aversion

RÉSUMÉ

Les techniques d'interaction expertes comme les vocabulaires gestuels ou les raccourcis clavier sont plus efficaces que les techniques WIMP traditionnelles. Il est en effet plus rapide de se rappeler une commande plutôt que de la retrouver dans des menus. Cependant, la plupart des utilisateurs semblent réticents à passer aux interactions qui se basent sur leur mémoire. Nous pensons que la cause pourrait être due à leur aversion à faire des erreurs. Pour tester cette hypothèse, nous avons conçu deux modes intermédiaires pour la technique d'interaction FastTap, qui permet de rapidement confirmer ce que l'utilisateur s'est rappelé de mémoire, et d'ajuster si une erreur a été faite. Nous avons étudié l'impact de ces deux modes intermédiaires et de différents coûts d'erreur dans une étude contrôlée (N=36). Nous avons trouvé que les participants ont adopté les modes intermédiaires, que ces modes réduisaient le taux d'erreur quand le coût de l'erreur était important, et qu'ils n'ont pas affecté de manière significative les temps de sélection. Cependant, bien que les résultats valident la conception de nos modes intermédiaires, nous n'avons pas trouvé de preuve sur un plus grand passage aux interactions qui se basent sur la mémoire. Cela suggère que réduire le taux d'erreur n'est pas suffisant pour promouvoir l'utilisation experte des techniques.

MOTS CLÉS

expertise, ajustement, mode intermédiaire, aversion aux erreurs

1 INTRODUCTION

Memory-based expert interaction techniques – such as keyboard shortcuts, marking menus, and gestural commands – have been shown to improve selection performance in user interfaces [25]. However, these “expert” methods are not used by all[4], and many users do not transition from the novice method to the expert one. There are several reasons why users may be unwilling to switch (e.g. [6, 12, 21] and described below), but for memory-based techniques, one main problem is user’s aversion to making errors. That is, even if a user remembers a command, they may be reluctant to invoke it with a memory-based method because selecting

IHM'19, December 10–13, 2019, Grenoble, France

© 2019 Copyright held by the owner/author(s). Publication rights licensed to ACM.

This is the author’s version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *Actes de la 31e conférence Francophone sur l'Interaction Homme-Machine (IHM'19), December 10–13, 2019, Grenoble, France.*
<https://doi.org/10.1145/3366550.3372247>.



Figure 1: FastTapAdjust. In novice mode, users press the menu button and wait for the grid to appear, then select a command by tapping on a grid cell while holding the menu button. In expert mode, users tap the menu button and the command’s location with a two-finger chord. Intermediate mode uses the same chord-based invocation, but users can pause and display items around their selection, allowing them to adjust the chord before selecting.

the wrong command may outweigh the benefit of using the faster expert technique.

Researchers have proposed several ways of improving the transition to expertise in user interfaces by encouraging users to switch, either by informing them of how much faster they could perform [23] or by adding artificial delays in order to increase the “cost” of the novice method [16].

There is little work, however, in reducing the user’s reluctance to make errors with memory-based techniques. The quality of global undo functionality is an important element here [7], but few techniques provide specific support for users who have learned an expert technique and are just at the point of deciding whether to invoke a command with the expert method. We believe the user’s aversion to making errors is an important factor in this decision. To test this hypothesis we developed new techniques allowing quick confirmation for users willing to try expert modes, and use FastTap as a usecase. FastTap is a quasi-mode menu system for touch-based devices that displays a hand-sized grid of items when a dedicated button on the lower left corner is pressed [18]. Experienced users can select commands in a rapid chord gesture without waiting for the buttons to appear.

We built “intermediate” selection methods for FastTap, which allow users to try triggering commands without displaying all icons but with an adjustment mechanism that also allows them to fix mistakes before the command is issued (Figure 1).

We are interested in three questions about these intermediate modes. First, we want to know whether they help users move away from novice interaction - that is, whether users choose intermediate modes as a “safer” alternative to the full expert mode, particularly when error costs are high. Second, we want to know whether these intermediate modes really do reduce errors. Third, we want to know whether the intermediate modes influence the eventual adoption of the

full memory-based expert mode. We examined these questions in a controlled experiment that varied the selection technique and the error cost. We created three command-selection environments that induced different error costs. We compared the original FastTap technique to two new variants that provided intermediate adjustment modes in addition to the novice and expert modes. We recorded people’s selection performance and which method they chose to use for each selection.

The results of our study provide several new insights into the design and use of intermediate modes for memory-based expert techniques: intermediate modes were used when available; when the error costs were high, the adjustment capability of the intermediate modes led to significantly lower error rates; results also suggest that the intermediate modes do not penalize performance. However, there was no difference in term of expert mode adoption (usage rates remained approximately equal), which suggests that reducing the cost of making errors is not enough to encourage users to transition to memory-based interaction.

2 RELATED WORK

Interfaces for novices and for experts

Mainstream Windows-Icons-Menu-Pointer (WIMP) interfaces are designed to maximize discoverability for novice users; as Shneiderman states, they are based on the idea of “see and point versus learn and remember” [30]. Although this approach is successful for novices, it has a low performance ceiling – even when users become very familiar with certain commands, they must still execute them using visually-guided hierarchical interface mechanisms.

Several alternate interfaces have been developed that attempt to raise the performance ceiling for experts. For example, researchers have studied keyboard shortcuts [2, 22], command languages [29], gestural commands [2, 3, 10, 11, 20, 27], and spatially-stable grid menus [18, 28]. These techniques typically achieve their efficiency by using human memory (e.g. memory of a gesture’s path or of a command’s spatial

location) rather than visually-guided hierarchical navigation. Several studies of these expert techniques have found them to be substantially faster than traditional WIMP selection [18, 25, 29] – but only after the commands have been learned.

When users are still inexperienced with the commands, they are not yet able to rely on memory alone, and need some form of visual feedback to help them find the correct item. Therefore, all expert techniques have “novice modes” that provide this feedback – for example, users of Marking Menus start out by using a radial menu system, switching to gestures as they memorize the command locations; FastTap provides a grid menu (see Figure 1) for inexperienced users (who may eventually switch to chords once locations are learned).

Transitions from novice to expert use

As users gain experience with an interface, they typically go through three stages [8]: a *cognitive* phase where they rely on visual search to identify commands, an *associative* phase where they begin to remember the commands, and an *autonomous* phase where they can execute commands quickly and without a great deal of conscious attention. WIMP-style selection interfaces often prevent users from achieving automaticity (because selection in these systems always requires visually-guided navigation), whereas memory-based techniques allow users to exploit their increasing experience.

Despite the increased performance of expert interfaces, however, studies have shown that users do not necessarily switch to the faster expert version of a technique [4, 6, 12, 21]. Several reasons for this phenomenon have been proposed, including satisficing (staying with a “good enough” strategy) [6], the *paradox of the active user* (where users prefer to apply existing knowledge to an ongoing task) [6], or the *value of feedback* (where users can prefer the feedback provided by a well-practiced novice method) [21].

In addition to these issues, memory-based expert techniques present a specific challenge to users who are still in the associative phase of learning: at some point, users must “launch out” and try the memory-based version, but their relative lack of experience means that they are more likely to make errors (*i.e.* by retrieving the wrong location or gesture from memory). This was one of the findings in a study of FastTap use in a non-laboratory setting: few of the study’s participants made the transition to the expert technique, and one of the main reasons given was that people did not want to make errors [17].

Although recall errors are part of any memory-based learning process, they can present a major barrier to development of expertise with an interface. Many people are highly averse to making mistakes: in some cases, incorrect command selections may have high costs (*e.g.* pressing “Send” before

a message is complete); and even when the actual consequences of an error are minor, user perception of the costs of an error can be much higher than the reality, due to inherent *negativity bias* in which “negatives loom larger than positives” [19].

Helping users move away from novice methods

Researchers have proposed several methods that attempt to overcome these barriers and increase user adoption of expert interaction techniques. Strategies include using the idea of rehearsal to ease the transition to an expert technique, increasing costs for using the novice technique, reminding users about the expert technique in contexts where it could be used, and reducing the potential cost of errors.

Rehearsal. Techniques such as Marking Menus [20], FastTap [18], and the SHARK keyboard [31] use Kurtenbach’s principle of rehearsal as a way to reduce the performance dip between novice and expert (and thus encourage users to make the transition). The rehearsal principle states that novices should perform selections in the same way that experts do, leading to incidental learning of the expert mode through everyday use [20]. Feedback and guidance appear for novices, but as users become more experienced, these supports can be removed and sequential actions can be chunked [5].

Increasing costs for the novice technique. Grossman and colleagues explored a variety of techniques for accelerating the use of keyboard shortcuts [16]; one of these imposed a time delay whenever the user selected a command using the menu system instead of the shortcut. This technique, while effective, may not be an acceptable method in real-world use.

Raising expert technique’s awareness. Several systems provide in-context information about expert techniques, to reduce the problem of “interface inertia” with novice methods. For example, the Emacs text editor displays equivalent shortcuts when commands are executed using search or the menu system. A similar idea was explored in the Blur system that provided notifications about text commands during WIMP use [29], and in the IconHK system that showed equivalent shortcuts for toolbar buttons on the toolbar, and emphasized the shortcuts using animations [14]. One problem with these systems is that they respond only after the user has already carried out a command using the novice technique, which further reinforces the suboptimal method. A second approach for motivating users was explored by the Skillometers system, which provides a dashboard that shows users how much time they could save if they switched to expert versions of selection techniques [22].

Mitigating error aversion. Although the aversion to make errors appears to be an important problem in the transition to memory-based expert techniques, there is little research that focuses on reducing it. First, standard usability guidelines

suggest several basic strategies for reducing errors (such as providing warnings or confirmation dialogs when critical commands are executed) [24]. Second, systems may provide comprehensive *Undo* functionality, with the rationale that if users trust in their ability to undo any command, they will be more willing to use their memory. It is unclear, however, whether undo can handle all possible error cases (such as incorrectly pressing *Send*).

A third approach provides information about command completion during the initial stages of the command's invocation (*i.e. feedforward*); this information may reassure users that they are going to make an appropriate selection. For example, some systems show "gesture guides" when a user begins a gesture, to indicate what legal gestures can be completed [3, 9]. Similarly, *Arpège* and *FingerCuts* show completion possibilities for multitouch chords [13, 15]. On the desktop, *ExposeHK* [22] exposes keyboard shortcuts beside interface buttons when a modifier key is pressed.

Feedforward techniques have been shown to be effective, but they can only work when commands are invoked with multi-stage actions – that is, the system needs a point in time to provide the feedforward information. In addition, little is known about whether feedforward methods actually provide a bridge to the memory-based version of the technique – that is, whether the feedforward helps or hinders people in memorizing the full command action. Some research suggests that users can come to rely too much on a feedforward guide, reducing their learning (*i.e.* the "guidance hypothesis") [1]. Anderson and Bischof suggest that a gradual reduction in the amount of support provided by feedforward can lead to improved memorization [1].

Finally, a few techniques explicitly allow adjustment of an executed action in similar fashion to our *FastTapAdjust* techniques. For example, text entry fields in Apple's iOS allow users to adjust the positioning of the text caret, using lift-off instead of touch-down to set the final position. Similarly, a targeting technique for moving objects long distances on digital tabletops allows the user to adjust the "flight" of the object using relative movements after it is been "thrown" [26]. The only adjustment method for memory-based selection technique, however, appears to be Kurtenbach's idea of mark confirmation, an adjustment phase in *Marking Menus* [20]: in expert mode, the user can pause during a gesture, which re-displays the radial menu from the novice mode, allowing the user to correct their selection.

3 DESIGN OF AN INTERMEDIATE MODE FOR FASTTAP

Adjustment mechanisms aim to support users in moving away from novice methods by adding an "intermediate" mode between the novice and expert modes. We use *FastTap* [18] as the core technique to test the ideas of adjustment and

intermediate modes. *FastTap* is a grid-based menu system that overlays a grid of items on the screen of a multi-touch tablet (Figure 1). Expertise in *FastTap* relies on the development of spatial memory in the user: commands are laid out in stable locations, and any command can be invoked by tapping its grid cell. Touching a menu button at the bottom corner of the screen with the thumb displays the grid after a 500ms delay. The user then selects a command by touching a grid cell with their finger (while maintaining their touch on the menu button). When the user becomes expert, she can chunk those two steps into a single one by pressing directly the touchscreen with the chord made up of the menu button and the item location [5].

FastTap implements the principle of *rehearsal* [20]: whether novice or expert, users select commands through the same finger positions. The actions remain the same, but when expert, the user does not need to wait for the grid to be displayed (and thus the selection becomes considerably faster). Be it in novice or expert mode, the command is activated as soon as the touchscreen is touched (not released) at the location of the corresponding grid cell and that the menu button is held. This is similar to keyboard shortcuts where commands are activated as soon as the hotkey is pressed (not released).

This, combined with the fact that the grid is not displayed in expert mode, requires the user to be confident in her spatial memory in order to efficiently transition from novice to expert mode. In expert mode, missing a location by a few millimetres can trigger the wrong command, which can be a problem if that command happens to be critical (*e.g.* erase, quit, or send email). By allowing adjustment in this selection paradigm, we introduce an intermediate mode where users can attempt the expert selection but still have the ability to verify and correct a selection before any command is triggered.

The *FastTapAdjust* Mechanism

As illustrated in figure 1, instead of triggering the command when the finger touches down on a grid cell, selection in *FastTapAdjust* happens when the finger lifts off from the surface. This provides an additional interaction state where adjustments can be made. The triggered command is the last grid cell in contact with the finger. In both novice and expert modes, users *tap* the cell of interest (whether the grid is displayed or not), which keeps the time between touchdown and lift-off minimal. However, the change in the triggering mechanism allows a new state and thus an intermediate mode. When a user makes an expert two-finger touch and then pauses before lifting off, the grid cell under their finger and a variable-size neighbourhood around the cell are displayed (with a 100ms delay for the neighborhood cells). The

user can then adjust her finger position and trigger a different command than what she originally touched. If the finger is moved outside of the displayed cells, no command is triggered and the grid is fully displayed (essentially defaulting to the novice-mode presentation). This intermediate mode allows users to try memory-based chord selection, even when their memory of the grid is still imperfect.

The intermediate mode has two main differences that place it between the two existing modes. First, it uses the same invocation mechanism as the expert method, which shifts users all the way to the expert end in terms of invocation. Second, its partial visual feedback moves the user away from the novice mode by narrowing the range of potential selections. Overall, we expect this mode to present users with a lower barrier for leaving the novice method behind, earlier in the learning process. In addition, we expect that the additional feedback of the intermediate mode will help to maintain a low error rate, and that the reduced number of choices will help to maintain fast performance.

4 STUDY

The aim of the study was to assess whether the adjustment mechanism helped users switch away from novice use of FastTap more quickly and with fewer errors. We compared the original FastTap interaction technique to two FastTapAdjust versions. We compared them in a controlled experiment (N=36) where participants were prompted with specific commands to select. For each technique, we tested three error-cost conditions that were induced by the number of attempts that participants could take to get the correct selection: we called these 1-TRY, 2-TRIES and 5-TRIES. Our goal was to study the learning process in environments where triggering a wrong command has different costs: from a high cost (e.g. sending an email instead of discarding it – represented by 1-TRY) through to a very low cost (e.g. similar to changing the current color to red instead of blue in a drawing application – represented by 5-TRIES). When the participant did not select the correct command within the maximum number of allowed attempts, they received visual and auditory feedback (the background turned red and a buzz sound played).

We used a between-subjects design to test the three different techniques: *FastTap* (FT), *FastTapAdjust* with a Moore neighborhood (i.e. the 8 surrounding cells, *FTA-8*), and *FastTapAdjust* with the full grid displayed as the neighborhood (*FTA-all*). The two different versions of *FTA* were implemented to test intermediate modes that were at different stages along the path from novice to expert: *FTA-all* is closer to the novice mode, and *FTA-8* is closer to the expert mode.



Figure 2: Icon sets used for the experiment: (upper-left) training icon set; (upper-right) food items icon set; (lower-left) flags icon set; (lower-right) transportation means icon set.

Procedure

After completing a short demographics survey, participants were shown the technique they would be using in the experiment. They then practiced the technique in a training session using a different set of icons and commands. Once they were comfortable with all forms of the technique, the experiment sessions started. The experiment consisted of 10 blocks of 24 trials for three error cost conditions (1-TRY, 2-TRIES and 5-TRIES, respectively for 0, 1 and 4 permissible errors). Varying the number permissible of errors aims to induce in participants a sense of higher error cost. For each trial, a command stimulus (icon and command name) was displayed at the top of the screen. The participant then selected the corresponding command from the FastTap menu. The commands were organized into a 5x5 grid: within the same row, the commands were semantically linked (e.g. flags from the same continent) but with no apparent order between the items. For each condition, separate sets of icons were used (Figure 2), and the order of the sets was chosen randomly. Trials were timed from the appearance of the stimulus until the target was successfully selected. The number of wrong selections and the number of times the full grid was shown (triggered from the menu button) were also logged. Participants were instructed to complete tasks as quickly and accurately as possible.

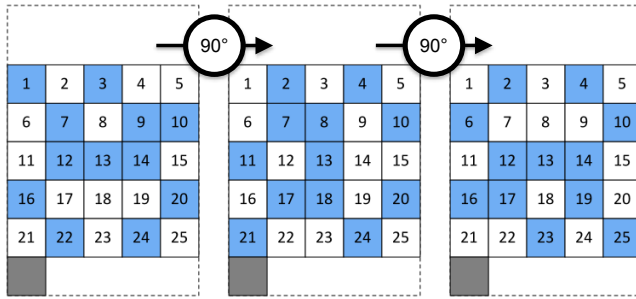


Figure 3: Location sets used for the experiment.

Of the 25 commands in each condition, only 12 were used as stimuli, in order to allow faster development of spatial memory and expertise. Two to three commands were used from each grid row, covering different configurations (corners, sides, interior of the grid). We used three different location sets for the targets, each a 90° rotation of a template set (Figure 3). Each condition used a different set and the order of the sets was chosen randomly. Targets were presented in random order for each block. Between each blocks, participants could take short rest breaks.

Experimental Design

The experiment used a $3 \times 3 \times 10$ between-participants design with factors **TECHNIQUE** (*FT*, *FTA-8* and *FTA-all*), **MAXATTEMPTS** (1-TRY, 2-TRIES and 5-TRIES) and **BLOCK** (1-10). Each combination of factors was repeated 24 times, for a total of 25,920 trials across all participants. The order of the commands was randomized with the constraint that a command could not appear twice in a row. Within each technique, the **MAXATTEMPTS** factor was counter-balanced using a Latin square.

Participants and Apparatus

We recruited 12 participants for each technique – 36 in total (mean age 26 years, SD 6 years; 4 left-handed; 11 males). The study was conducted on a Nexus 7 Android tablet (7-inch screen, 1200×1920 resolution). The software was written in C++, using the Qt framework, and recorded all experimental data including selection times, incorrect selections and touch locations.

Results

To compare the techniques and conditions, we ran statistical analysis on *Time*, *Error* rates, and the proportion of novice, intermediate, and expert selections across blocks.

Selection Time. Overall, mean selection times were similar across the three interfaces, with FastTap at 1986ms (SD 1040ms), FastTapAdjust-8 at 1884ms (830ms), and FastTapAdjust-All at 1822ms (730ms). Mean times were also

similar across the different error cost conditions, with 1-TRY at 1954ms (864ms), 2-TRIES at 1872ms (845ms), and 5-TRIES at 1870ms (928ms).

A mixed-design ANOVA was conducted to compare the effects of **TECHNIQUE**, **MAXATTEMPTS**, and **BLOCK** on *Time*. No significant difference was found for **TECHNIQUE** or **MAXATTEMPTS** ($F_{2,33} = .75, p = .48$ and $F_{2,66} = 1.48, p = .24$), but one was found for **BLOCK** ($F_{1,33} = 591, p < .0001$, see Figure 6). We also found an interaction between **TECHNIQUE** and **BLOCK** ($F_{2,33} = 5.61, p < .01$). As a followup, we conducted a separate ANOVA for each technique; all showed significant effects of **BLOCK** on *Time* (all $F_{1,11} > 152$, all $p < .0001$).

Pairwise comparisons using T-tests and the Bonferroni correction showed significant differences between blocks (in the following $i > j$ means *block i was significantly slower than block j*): for FT $1 > 2 > 3 > 4, 5, 6, 10 > 7, 8, 9$; for FTA-8 $1 > 2 > 3, 4 > 5, 6 > 7, 8, 9, 10$; for FTA-all $1 > 2 > 3 > 4, 5, 6, 7, 8, 9, 10$. However, using a mixed-design ANOVA on each block, we did not find an effect **Technique** on *Time* (all $F_{2,33} < 3.18$, all $p > .05$).

These results show the expected learning effect. They also suggest that introducing the intermediate mode does not present any major barriers to fast interaction, and that time is not strongly influenced by the perception of error cost. The block differences also suggest that participants reached stable performance quicker using the FTA-all technique.

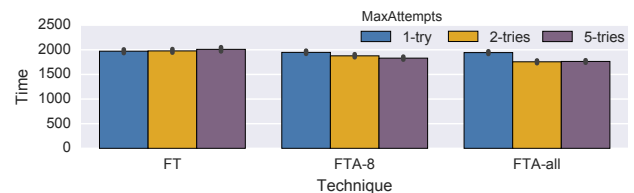


Figure 4: Average times in ms for target selection for *FastTap* and both *FastTapAdjust* versions for the three **MAXATTEMPTS** conditions. Error bars represent the 95% CI.

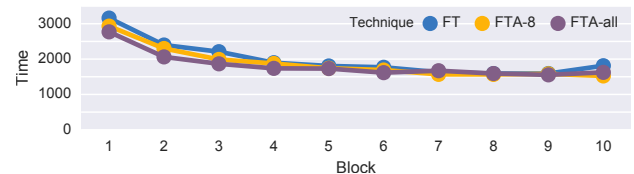


Figure 5: Average times in ms for target selection in each block for *FastTap* and both *FastTapAdjust*. Error bars represent the 95% CI.

Error rates. There were two types of erroneous trials. The first is the definition we gave the participants: an erroneous

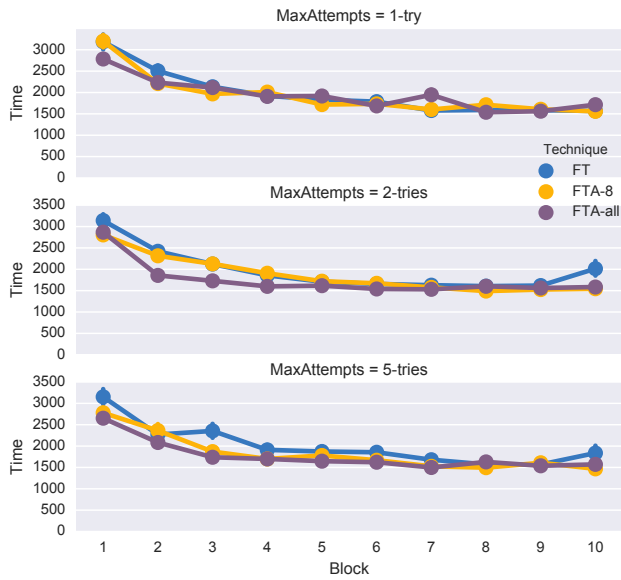


Figure 6: Average times in ms for target selection for *FastTap* and both *FastTapAdjust* versions for the three *MAXATTEMPTS* conditions across the blocks. Error bars represent the 95% CI.

trial is a one where the prompted command was not selected within the allowed maximum attempts. In our analysis, we call this variable *Error*. The second is a trial where the prompted command was not correctly selected on the first attempt. In our analysis, we call this variable *ErrorFirstTry*.

Overall, *Error* rates were similar across the three techniques: *FastTap*'s error rate was 4% (SD 19%), *FastTapAdjust-8* was at 2% (SD 15%), and *FastTapAdjust-all* was at 2% (SD 13%). Rates for the different cost conditions were more variable, with 1-TRY at 7% (26%), 2-TRIES at 1% (9%), and 5-TRIES at 0% (9SD 2%).

A mixed-design ANOVA was conducted to compare the effect of *TECHNIQUE*, *MAXATTEMPTS*, and *BLOCK* on *Error*. A significant difference was found for *MAXATTEMPTS* ($F_{2,66} = 45.51, p < .0001$) but not *TECHNIQUE* ($F_{2,33} = 2.21, p = .13$) or *BLOCK* ($F_{1,33} = .64, p = .43$). We found a significant interaction between *TECHNIQUE* and *MAXATTEMPTS* ($F_{4,66} = 2.67, p < .05$).

Post-hoc analysis using Welch Two-Sample T-Tests with Holm correction revealed significant differences between each *MAXATTEMPTS* condition (all $p < 0.0001$). Post-hoc analysis also revealed significant differences between all (*TECHNIQUE*, *MAXATTEMPTS*) tuples (all $p < 0.05$) except (FT, 2-TRIES) and (FTA-8, 2-TRIES) and all tuples involving 5-TRIES. As shown in Figure 7, these results indicate that when error costs were high, the adjustment mechanism significantly reduced error.

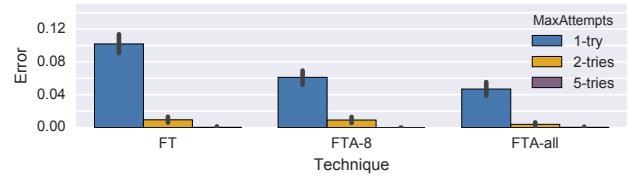


Figure 7: Error rates for *FastTap* and both *FastTapAdjust* versions for the three *MAXATTEMPTS* conditions. Error bars represent the 95% CI.

For *ErrorFirstTry*, the mean error rates (and SD) for the different interfaces were *FastTap* at 10% (SD 29%), *FastTapAdjust-8* at 6% (24%), and *FastTapAdjust-all* at 6% (23%). For the different cost conditions, 1-TRY had a mean rate of 7% (SD 26%), 2-TRIES was at 6% (24%), and 5-TRIES was at 8% (27%).

A mixed-design ANOVA was conducted to compare the effect of *TECHNIQUE*, *MAXATTEMPTS*, and *BLOCK* on *ErrorFirstTry*. No significant effects were found for any of the three factors ($F_{2,33} = 1.95, p = .16$; $F_{2,66} = 2.64, p = .08$; $F_{1,33} = 2.67, p = .11$). We also found no significant interactions. However, Welch Two Sample T-Tests with the Holm correction method revealed significant differences between most of the (*TECHNIQUE*, *MAXATTEMPTS*) tuples (all $p < 0.05$): (FT, 1-TRY) and (FT, 5-TRIES) were significantly more error-prone than the rest of the tuples, and (FTA-all, 1-TRY) and (FTA-all, 2-TRIES) were significantly less error prone than the rest of the tuples except (FTA-8, 5-TRIES) (see Figure 8).

Overall, this result suggests that the selection error cost influences FT users but not FTA users – both versions of FTA performed similarly regardless of how many attempts were allowed.

Moving away from novice: usage mode frequencies. Each trial was categorized as *Novice*, *Intermediate* or *Expert* depending on what occurred during the trial: if the full grid was displayed during a trial, it was categorized as *Novice*; if no feedback was displayed (either the full grid or the adjustment grid), the trial was categorized as *Expert*; otherwise the trial was categorized as *Intermediate*.

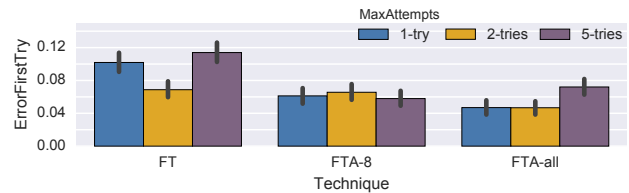


Figure 8: Error rates of the first selection of a trial for *FastTap* and both *FastTapAdjust* versions for the three *MAXATTEMPTS* conditions. Error bars represent the 95% CI.

Participant use of the novice mode decreased over the 10 study blocks. As shown in Figure 9, rate of novice use decreased from 94% in the first block to 34% in the tenth block for FastTap, from 94% to 31% for FastTapAdjust-8 and from 81% to 21% for FastTapAdjust-all. A two-way between-subjects ANOVA was conducted to compare the effect of **TECHNIQUE**, **BLOCK** and **MAXATTEMPTSON** Novice use. A significant difference was found for **BLOCK** ($F_{1,33} = 94.47$, $p < .0001$) but not **TECHNIQUE** ($F_{2,33} = 1.01$, $p = .37$) nor **MAXATTEMPTS** ($F_{2,66} = 2.34$, $p = .10$). We found no significant interaction between the factors (all $F < 1.6$, all $p > .21$).

We also looked at the effect of **BLOCK** on novice use. For all techniques, ANOVA found a significant effect of **BLOCK** (all $F_{1,11} > 30.8$, all $p < .0005$). We used pairwise T-Tests with Holm correction as a post-hoc analysis to compare **BLOCKS**. We focus particularly on the blocks where the novice use drops below 50%, as this level captures the point at which participants started to use the novice method less than intermediate and expert combined.

For FT, the drop happens between **BLOCK**4 (51%) and 5 (48%) - and as these blocks are significantly different between block 3 (at 60%) and block 7 (at 38%) (all $p < .005$), the 50% threshold is likely to be crossed at about block 4. For FTA-8, the drop happens between blocks 2 (63%) and 3 (48%), as blocks 3 and 4 (42%) are significantly different (all $p < .0001$), the 50% threshold is likely to be crossed around block 3. For FTA-all, the drop happens between block 1 (81%) and 2 (47%), with block 2 significantly different from blocks 1 and 3 (all $p < .0001$); the 50% threshold is likely to be crossed around block 2.

This result suggests that introducing the intermediate mode does, as expected, lead to an earlier switch away from use of novice mode.

5 DISCUSSION

The study provides five main findings:

- Both intermediate modes were frequently used by participants (30% of all selections);
- Although novice-mode rate was not significantly different, block-by-block analysis suggests that participants moved away from the novice mode more quickly (at about block 2 for FTA-all, versus about block 4 for FT);
- Selection times were similar for all three versions of the technique, suggesting that adjustment-based intermediate modes do not slow participants down (at least, there is no evidence of a large difference);
- Techniques with intermediate modes had lower error rates when error cost was high, although not when costs were low (and as a result, overall error rates were similar).

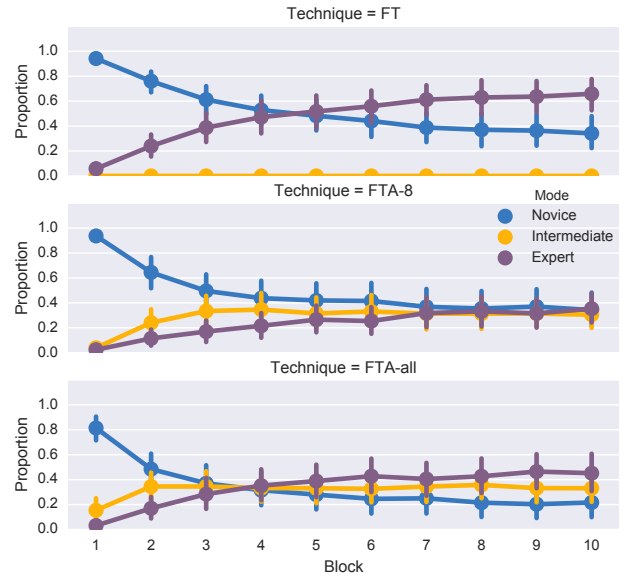


Figure 9: Overall rates of novice, intermediate, and expert use for each technique, by block. Error bars represent the 95% CI.

In the following sections, we consider reasons for these results and issues for the generalization and further study of intermediate modes in memory-based expert techniques.

Enabling “safe” expert selection

In conditions where an adjustment mode was available, participants made similar use of both expert and intermediate modes. As selection times are not significantly different between FTA and FT, and error rates in the high-cost condition are significantly lower using FTA, it suggests that during the associative phase of learning, the intermediate mode behaves more as a “safe” version of the expert selection. The idea of a safe version was even re-invented by a participant who used the regular FT technique: *“I preferred to be on the verge [of novice and expert modes]. In the end even if I knew where the icons were, I positioned my fingers before triggering the command – I’d rather take a quick look at the grid to be sure of the answer and lose no time.”*

Local feedback versus memory

The fact that the intermediate mode does not change completion time, and seems to act more as a “safe” expert mode, is an unexpected outcome. As a post-hoc analysis, we further explored the potential difference in terms of time performances between intermediate and expert selections by looking at the average selection time of the different selection modes (Figure 10), and particularly focus on the last block. For FTA-8,

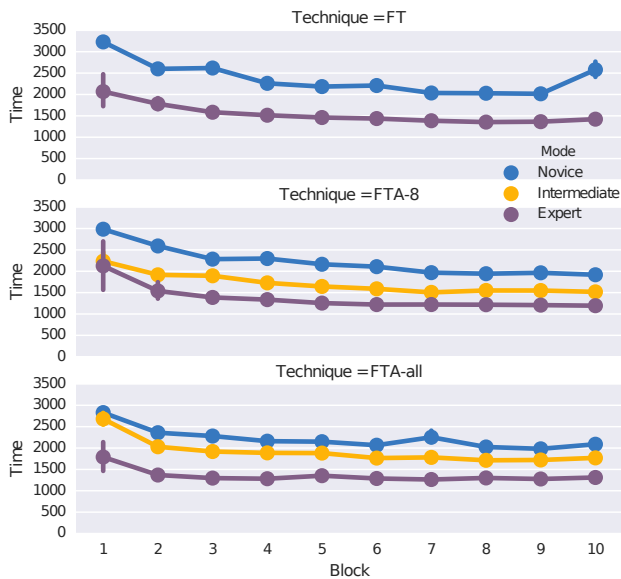


Figure 10: Average selection time for each Mode for FastTap and both FastTapAdjust versions across the blocks. Error bars represent 95% CI.

mean expert-mode selection time was 1194ms, intermediate was 1517ms, and novice was 1918ms. For FTA-all: mean expert-mode selection time was 1311ms, intermediate was 1769ms, and novice was 2089ms.

We conducted ANOVAs for each technique to compare the effect of *Mode* on *Time* on the tenth block (using only participants that used both on the last block). We found significant differences between each pair of modes for FTA-8 (novice-expert, $F_{1,7} = 14.8$, $p < 0.01$; novice-intermediate, $F_{1,7} = 5.68$, $p < 0.05$; intermediate-expert, $F_{1,8} = 11.8$, $p < 0.01$) and for FTA-all (novice-expert, $F_{1,4} = 40.5$, $p < 0.005$; novice-intermediate, $F_{1,5} = 18.8$, $p < 0.01$; intermediate-expert, $F_{1,7} = 23.8$, $p < 0.01$).

We hypothesize that this difference is due to participants relying on memory and visual feedback for the intermediate mode, whereas they only rely on memory for the expert mode. This hypothesis seems to be strengthened when analysing Figure 10, where the intermediate curve is shifted upward toward the novice curve when the neighbourhood is bigger (full grid vs. 8-item neighbourhood).

Reducing the error rate is not enough

Although an intermediate mode seems to reduce the error rate of the FastTap technique, it does not favour the adoption of the memory-based interaction (*i.e.* expert mode). The last block, regardless whether using FT or FTA-8, had a similar rate of novice use. It suggests that reducing the cost of error is not enough to overcome the *error aversion*. However, an interesting trend can be noted for FTA-all: the 10th seems to yield a lower rate of novice use. Arguably FTA-all could be seen

as FT with a 100ms delay instead of 500ms. This reduction of time delay might be a strong candidate to investigate.

The benefit of the intermediate mode seems to arise in encouraging an earlier move away from the novice technique. In both FTA techniques, participants switched away from the novice mode more frequently in early blocks (particularly for FTA-all). The intermediate mode seems to be bridging the gap between novice and expert modes by reducing the cost of switching away from novice mode. The closer to the novice mode an intermediate technique is, the easier the switch can be. However, there is a trade-off between speed and switching away from novice selection. An interesting follow-up study would investigate a dynamic intermediate mode that shrinks the neighbourhood to gradually move users toward expert selection.

Limitations

One limitation of our new intermediate modes could be that participants seemed to remain with these modes rather than switching fully to expert methods. Even though overall performance is not affected, it might translate into a longer transition toward optimal performance. A follow up study investigating the longer-term use of intermediate and expert modes is needed – the ten blocks of our study were not enough to clearly indicate users' eventual choice of which mode to use.

A second limitation could be that our inducement of error cost is different from the real world. In our study, errors were not harmful (for example, unfinished email could be sent by mistake). Debriefings with participants, however, showed that they were being careful to avoid making mistakes.

Finally, our work is also limited to FastTap interfaces and we cannot draw conclusions from this study about other memory-based interaction techniques, typically Marking Menus. Indeed, a recent study conducted by Fruchard et al. [11] suggests that memory-based command selection techniques results in different recall rates whether they rely on directional or positional gestures. Therefore, future work should investigate more carefully this question with other memory-based interaction techniques. A first step could be to compare mode adoption patterns between FastTap and Marking Menus, to test if directional gestures are more prone to adoption of expert behavior.

6 CONCLUSION

A main problem for memory-based expert interaction techniques is that many users are reluctant to move away from novice versions of techniques because of the aversion of making errors. We hypothesized that this aversion could be reduced through the introduction of intermediate modes that allow users to confirm their memory-based retrieval and adjust their selection if they have made an error. We

implemented two adjustment-based intermediate modes for the FastTap grid menu technique, and investigated their performance in a controlled study (N=36). We found that participants quickly made use of the intermediate modes, that these modes reduced error rates when error cost was high, and that they did not substantially affected selection times. Yet, we did not found evidence that it encourages the switch from novice to expert, even though results suggest that the intermediate modes encourage an earlier switch away from novice mode. Overall, our study showed the error reduction benefits of intermediate modes and uncovers strong candidates for further investigating transition between novice and expert methods: reducing time delay to display the menu as well as variable intermediate modes.

REFERENCES

- [1] Fraser Anderson and Walter F. Bischof. 2013. Learning and Performance with Gesture Guides. In *Proc. of CHI*. ACM, 1109–1118. <http://doi.acm.org/10.1145/2470654.2466143>
- [2] Caroline Appert and Shumin Zhai. 2009. Using Strokes As Command Shortcuts: Cognitive Benefits and Toolkit Support. In *Proc. of CHI*. ACM, 2289–2298. <http://doi.acm.org/10.1145/1518701.1519052>
- [3] Olivier Bau and Wendy E. Mackay. 2008. OctoPocus: A Dynamic Guide for Learning Gesture-based Command Sets. In *Proc. of UIST*. ACM, 37–46. <http://doi.acm.org/10.1145/1449715.1449724>
- [4] Suresh K. Bhavnani and Bonnie E. John. 2000. The Strategic Use of Complex Computer Systems. *Hum.-Comput. Interact.* (2000), 107–137. http://dx.doi.org/10.1207/S15327051HCI1523_3
- [5] William Buxton. 1986. Chunking and phrasing and the design of human-computer dialogues. In *IFIP Congress*. 475–480.
- [6] John M Carroll and Mary Beth Rosson. 1987. *Paradox of the active user*. The MIT Press.
- [7] Andy Cockburn, Carl Gutwin, Joey Scarr, and Sylvain Malacria. 2014. Supporting Novice to Expert Transitions in User Interfaces. *ACM Comput. Surv.* (2014), 31:1–31:36. <http://doi.acm.org/10.1145/2659796>
- [8] Paul Morris Fitts and Michael I Posner. 1967. Human performance. (1967).
- [9] Dustin Freeman, Hrvoje Benko, Meredith Ringel Morris, and Daniel Wigdor. 2009. ShadowGuides: Visualizations for In-situ Learning of Multi-touch and Whole-hand Gestures. In *Proc. of ITS*. ACM, 165–172. <http://doi.acm.org/10.1145/1731903.1731935>
- [10] Bruno Fruchard, Eric Lecolinet, and Olivier Chapuis. 2017. MarkPad: Augmenting Touchpads for Command Selection. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. ACM, 5630–5642.
- [11] Bruno Fruchard, Eric Lecolinet, and Olivier Chapuis. 2018. How Memorizing Positions or Directions Affects Gesture Learning?. In *Proceedings of the 2018 ACM International Conference on Interactive Surfaces and Spaces*. ACM, 107–114.
- [12] Wai-Tat Fu and Wayne D Gray. 2004. Resolving the paradox of the active user: Stable suboptimal performance in interactive tasks. *Cognitive science* (2004), 901–935. <http://doi.org/10.1016/j.cogsci.2004.03.005>
- [13] Emilien Ghomi, Stéphane Huot, Olivier Bau, Michel Beaudouin-Lafon, and Wendy E. Mackay. 2013. ArpèGe: Learning Multitouch Chord Gestures Vocabularies. In *Proc. of ITS*. ACM, 209–218. <http://doi.acm.org/10.1145/2512349.2512795>
- [14] Emmanouil Giannakis, Gilles Bailly, Sylvain Malacria, and Fanny Chevalier. 2017. IconHK: Using Toolbar Button Icons to Communicate Keyboard Shortcuts. In *Proc. of CHI*. ACM. <https://hal.inria.fr/hal-01444365>
- [15] Alix Goguy, Daniel Vogel, Fanny Chevalier, Thomas Pietrzak, Nicolas Roussel, and Géry Casiez. 2017. Leveraging finger identification to integrate multi-touch command selection and parameter manipulation. *International Journal of Human-Computer Studies* (2017), 21–36.
- [16] Tovi Grossman, Pierre Dragicevic, and Ravin Balakrishnan. 2007. Strategies for Accelerating On-line Learning of Hotkeys. In *Proc. of CHI*. ACM, 1591–1600. <http://doi.acm.org/10.1145/1240624.1240865>
- [17] Carl Gutwin, Andy Cockburn, and Benjamin Lafreniere. 2015. Testing the Rehearsal Hypothesis with Two FastTap Interfaces. In *Proc. of GL*. Canadian Information Processing Society, 223–231. <http://dl.acm.org/citation.cfm?id=2788890.2788930>
- [18] Carl Gutwin, Andy Cockburn, Joey Scarr, Sylvain Malacria, and Scott C. Olson. 2014. Faster Command Selection on Tablets with FastTap. In *Proc. of CHI*. ACM, 2617–2626. <http://doi.acm.org/10.1145/2556288.2557136>
- [19] Daniel Kahneman. 2011. *Thinking, fast and slow*. Macmillan.
- [20] Gordon Paul Kurtenbach. 1993. *The design and evaluation of marking menus*. Ph.D. Dissertation. University of Toronto.
- [21] David M Lane, H Albert Napier, S Camille Peres, and Anikó Sándor. 2005. Hidden costs of graphical user interfaces: Failure to make the transition from menus and icon toolbars to keyboard shortcuts. *International Journal of Human-Computer Interaction* (2005), 133–144.
- [22] Sylvain Malacria, Gilles Bailly, Joel Harrison, Andy Cockburn, and Carl Gutwin. 2013. Promoting Hotkey Use Through Rehearsal with ExposeHK. In *Proc. of CHI*. ACM, 573–582. <http://doi.acm.org/10.1145/2470654.2470735>
- [23] Sylvain Malacria, Joey Scarr, Andy Cockburn, Carl Gutwin, and Tovi Grossman. 2013. Skillometers: Reflective Widgets That Motivate and Help Users to Improve Performance. In *Proc. of UIST*. ACM, 321–330. <http://doi.acm.org/10.1145/2501988.2501996>
- [24] Rolf Molich and Jakob Nielsen. 1990. Improving a Human-computer Dialogue. *Commun. ACM* (1990), 338–348. <http://doi.acm.org/10.1145/77481.77486>
- [25] Daniel L. Odell, Richard C. Davis, Andrew Smith, and Paul K. Wright. 2004. Toolglasses, Marking Menus, and Hotkeys: A Comparison of One and Two-handed Command Selection Techniques. In *Proc. of GL*. Canadian Human-Computer Communications Society, 17–24. <http://dl.acm.org/citation.cfm?id=1006058.1006061>
- [26] Adrian Reetz, Carl Gutwin, Tadeusz Stach, Miguel Nacenta, and Sriram Subramanian. 2006. Superflick: A Natural and Efficient Technique for Long-distance Object Placement on Digital Tables. In *Proc. of GL*. Canadian Information Processing Society, 163–170. <http://dl.acm.org/citation.cfm?id=1143079.1143106>
- [27] Quentin Roy, Sylvain Malacria, Yves Guiard, Eric Lecolinet, and James Eagan. 2013. Augmented Letters: Mnemonic Gesture-based Shortcuts. In *Proc. of CHI*. ACM, 2325–2328. <http://doi.acm.org/10.1145/2470654.2481321>
- [28] Joey Scarr, Andy Cockburn, Carl Gutwin, and Andrea Bunt. 2012. Improving Command Selection with CommandMaps. In *Proc. of CHI*. ACM, 257–266. <http://doi.acm.org/10.1145/2207676.2207713>
- [29] Joey Scarr, Andy Cockburn, Carl Gutwin, and Philip Quinn. 2011. Dips and Ceilings: Understanding and Supporting Transitions to Expertise in User Interfaces. In *Proc. of CHI*. ACM, 2741–2750. <http://doi.acm.org/10.1145/1978942.1979348>
- [30] Ben Shneiderman. 1981. Direct Manipulation: A Step Beyond Programming Languages (Abstract Only). In *Proc. of CHI*. ACM, 143–. <http://doi.acm.org/10.1145/800276.810991>
- [31] Shumin Zhai and Per-Ola Kristensson. 2003. Shorthand Writing on Stylus Keyboard. In *Proc. of CHI*. ACM, 97–104. <http://doi.acm.org/10.1145/642611.642630>