Improving Discoverability and Expert Performance in Force-Sensitive Text Selection for Touch Devices with Mode Gauges

Alix Goguey, Sylvain Malacria, Carl Gutwin

HAL Id: hal-01714195
https://inria.hal.science/hal-01714195

Submitted on 21 Feb 2018

Improving Discoverability and Expert Performance in Force-Sensitive Text Selection for Touch Devices with Mode Gauges

Alix Goguey[1], **Sylvain Malacria**[2] **& Carl Gutwin**[1]
[1]University of Saskatchewan, Canada, [2]Inria, France
alix.goguey@usask.ca, sylvain.malacria@inria.fr, gutwin@cs.usask.ca

ABSTRACT

Text selection on touch devices can be a difficult task for users. Letters and words are often too small to select directly, and the enhanced interaction techniques provided by the OS – magnifiers, selection handles, and methods for selecting at the character, word, or sentence level – often lead to as many usability problems as they solve. The introduction of force-sensitive touchscreens has added another enhancement to text selection (using force for different selection modes); however, these modes are difficult to discover and many users continue to struggle with accurate selection. In this paper we report on an investigation of the design of touch-based and force-based text selection mechanisms, and describe two novel text-selection techniques that provide improved discoverability, enhanced visual feedback, and a higher performance ceiling for experienced users. Two evaluations show that one design successfully combined support for novices and experts, was never worse than the standard iOS technique, and was preferred by participants.

Author Keywords

Touch interaction; text selection; force; mobile device.

ACM Classification Keywords

H.5.2 Information interfaces (e.g. HCI): User interfaces

INTRODUCTION

Text selection on touch-based devices can be a difficult task for users. The difficulties arise because the letters and words are small targets and fingers are large, causing both targeting inaccuracy and visual occlusion. As a result, selection using direct touch alone is too difficult – so current mobile platforms attempt to overcome the problem with several additions, including magnifier views, selection handles, word snapping, or even AI-based inference of the user's intended selection (Android's "Smart Text" selection). Even with these additions, however, many users find mobile text selection to be difficult and frustrating. For example, in a 2016 Reddit thread[1], numerous users complained about being unable to select accurately, with comments such as:

- "I am constantly struggling to select text in iOS"
- "Oh god, this happens to me every time. Whenever I select something to copy, I always have to select it twice"
- "Another issue is that I clearly selected the whole thing from the first letter of the first word to the last letter of the last word but somehow part of the first words will be gone"
- "This is exactly why I haven't been able to use my iPad to replace my secondary laptop (for basics like email)."

With the introduction of force-sensitive screens on recent smartphones (such as the latest Apple's iPhones) yet another augmentation has been added to text selection. The iOS text-selection technique makes use of force (the user performs a sequence of hard presses) as a way to switch between word-level, sentence-level, and paragraph-level selection. However, the force commands in the current technique are difficult to discover, and have no visual feedback or feedforward to tell users what selection mode has been or will be chosen. In this paper we report on our investigation of the task of mobile text selection, and report both a design framework for text selection as well as two novel interaction techniques that provide improved discoverability, smoother transitions to expertise thanks to enriched feedback and a higher performance ceiling once users are experienced.

Our design framework focuses on several factors that must be balanced in a task such as mobile text selection: easy discoverability for novices, support for a smooth transition from novice to expert behaviour, high performance ceiling for experts, and fluid switching between text selection modes. We introduce *ForceSelect* and *ForceSelect High-Pressure*, two novel touch-based text selection techniques that were designed by balancing the factors of this framework. Like the existing iOS method, the new techniques also use time, selection handles, multiple views, and force-dependent selection modes - but also provide clear visual feedback and feedforward using novel "mode gauges" that do not require additional screen real estate, make the techniques' capabilities more obvious and simplifies the use of the techniques for different kinds of selection tasks.

We compared these two novel techniques to the standard iOS method in two studies - one that examined performance on a variety of selection tasks, and one that examined the discoverability of each technique's capabilities. Our evaluations

---

[1] *"Text selection on iOS has become extremely frustrating"* – http://ns.inria.fr/mjolnir/forceselect/iosreddit/

showed that our techniques were substantially more discoverable than the iOS method. They also showed that ForceSelect was faster than the standard method for various kinds of selection, was never worse than the other techniques, and was preferred by a majority of participants.

Overall, our work presents the following contributions: the design and implementation of 2 novel force-based interaction techniques using mode gauges; the evaluation of how they improve expert performance; the evaluation of how they improve discoverability (evaluation of interaction techniques in HCI often focuses only on expert performances without considering critical usability criteria like learnability and discoverability). Our research shows that some common interactive tasks are more complex than they initially seem, with substantial variability and subtlety – and that it is important to think about underlying principles of interaction (such as affordances, transitions, and performance ceilings) when designing for these tasks. Our design of the ForceSelect technique can serve as an example of how several degrees of freedom can be successfully combined in a single high-power technique, without sacrificing the requirements of novice or occasional users.

BACKGROUND ON TOUCH-BASED TEXT SELECTION

A text selection typically represents a range of text selected by a user, characterized by a start endpoint (the first character of the range) and a finish endpoint (the last character of the range). On a desktop computer, text selection is typically performed with a simple dragging operation that defines the whole range. However, text selection on touch-based devices is not as simple as a direct drag operation because by default, drag is used for scrolling. As a result, touch-based text selection usually occurs in three phases: a dedicated input that will switch the system to text-selection mode, pre-selection of a default range, and adjustment of the selection by manually adjusting the handles attached to each endpoint.

Switching to text-selection mode

Desktop interfaces allow text selection through several different means (*e.g.* keyboard, shift+click or dragging with the mouse [1, 17]), and touch-based text selection can also be triggered using different methods. However, since by default drag input scrolls the view on a touch-based system, switching to text-selection mode requires a dedicated operation that usually depends on the current application and the type of widget the user is interacting with. For instance, for an editable text field on iOS (*e.g.* the Notes app), text-selection mode can be enabled in four different ways:

- Double-tapping on a word (which pre-selects the word and displays a menu)
- Long-pressing anywhere in the text (which displays a menu) and tapping the "Select" menu item (which preselects the word under the long press)
- Force-pressing on a word (which preselects the word)
- Force-pressing on the keyboard to enter caret-positioning mode, then force-pressing again to select the word under the caret.

Note that for the latter two operations, the user can adjust the pre-selection in one "continuous" operation before finger lift-off by moving the finger on the display. However, for a non-editable text-field on iOS (*e.g.* a webpage or PDF document), double-tapping is reserved for zooming operations, so text-selection mode can only be enabled by performing a long-press, which preselects the word under the press. The user can again adjust the pre-selection before lift-off by force-pressing to display the handles and then moving the finger on the display. On Android, double-tapping or long-pressing pre-selects the word located under the finger and positions handles on each side of it.

Initial range of the selection

As described above, most existing solutions select by default the word located under the finger that was used for switching to text selection mode (except for the force-based iOS methods that enter a different mode and allow adjustment of the selection before lift-off). These techniques allow after finger lift-off to expand the initial selection range either forwards or backwards from the initial touch - but not both (this requires lifting the finger from the screen to reach for the other endpoint). The TextView widgets of some Android versions support a 2-finger long-press action that switches to text selection mode and uses the location of the two fingers as the initial endpoints of the selection. While this selection method allows users to rapidly select a large range, it is difficult to accurately position the fingers at a character level. The latest version of Android also introduced *Smart Text Selection* [2] that uses machine learning to infer the user's intended selection. Chang et al. introduced the concept of intentional uncertain selection [4], which consists of defining an overall selection range around a touch point, depending on the force applied to the display. While this approach is useful for tasks that do not require hard boundaries, it is not adapted to common activities such as copying an address or cutting/pasting a paragraph. Scriboli [14] is a selection mechanism for pen-based computers that relies on a lasso tool for circling the content that the user wants to select, and a pigtail gesture to display a contextual menu for performing operations on the selection. However, this is not adapted to touch interaction as it requires the selection mode to be enabled by default.

Adjusting endpoints

On iOS and Android devices, once the initial selection has been defined, users can adjust its endpoints by moving the associated handles. Adjusting a handle is typically performed with a conventional drag gesture. Handles were developed because of the difficulty of selecting individual letters on a touch device - previous work suggests that a minimum size of 9mm is required for correct finger acquisition [21], and a character on a smartphone screen can be as small as 1mm. In addition, the take-off strategy [22, 25] was designed to alleviate the problem of finger occlusion. Take-off places a cursor slightly above the user's finger when they touch the screen. Users can then drag the cursor to the desired region. However, take-off breaks the direct mapping of touch-based

---
[2]Android O Smart Text Selection – **http://ns.inria.fr/ mjolnir/forceselect/androidsmartselect/**

devices; therefore, Vogel and Baudisch introduced Shift [28], which creates a callout showing a copy of the screen area occluded by the finger, as well as a pointer representing the selection point of the finger. Callouts have been adopted by mobile operating systems such as iOS and Android - the callout view appears when switching to text-selection mode and when adjusting the handles, so users can select at the character level. Fix and Slide [27] proposes an alternative to the callout to overcome the occlusion problem by using a movable background to adjust the endpoints. Users place the first endpoint by tapping on the screen and then move the background by touching and dragging. Another proposal for simplifying range adjustment is to automatically snap the selection to word boundaries [18], which makes it easier to select at the word level, but makes it virtually impossible to select at the character level. Finally, for situations where the endpoints need to expand beyond the edge of the display, Antoine et al. proposed an autoscroll technique that uses force-sensing to scroll the view while adjusting endpoint locations [2].

Other text selection techniques
Researchers have also introduced text-selection techniques that differ substantially with the conventional implementations that can be found in commercial products. As an example, Fuccella et al. [8, 9] proposed the use of discrete gestures executed on the soft keyboard to perform various actions including moving the caret or performing selections. While this approach offers benefits, it supports only word-level selection (sub-word-level are achieved through refinements of the primary selection) and is not compatible with gestural keyboards. BezelSwipes [23] have been proposed as a way to specify the endpoints of a text selection using swipe gestures from the edge of the touchscreen, thus requiring two swipes to specify a text selection. However, this method has not been fully evaluated in the context of text selection and it is unclear how endpoint adjustment is performed. Bezel-Copy [5], a copy-paste technique, builds on the idea of bezel swiping. It uses a bezel swipe gesture to determine a rough area of interest in the document, usually a sentence. Chosen text is magnified in a new panel to enable fast and precise selection. This method has several limitations, as it pops up a new window that might disrupt the user, does not allow the selection to be expanded beyond the bounds of the display, and does not provide any visual affordances for the bezel-swipe mechanism, potentially making it difficult to discover. Finally, Push-Push [11, 12] proposes to use a force threshold in order to determine when a touchscreen is *touched* or *pressed hard*, thus allowing the user to perform press actions to position the endpoints of a text selection.

DESIGN CONSIDERATIONS
The motivation of our work comes from the inherent limitations of state-of-the-art text selection techniques on touch-based devices. While force-based text-selection exists, we argue that current designs suffer from several limitations, leading to the following design considerations derived from design principles known in HCI such as Nielsen's heuristics and Shneiderman's golden rules [20, 26].

Design consideration 1: Supporting discoverability

It is important for infrequent users that the system exposes the features of the technique, especially the mapping of the device's force-sensing capabilities to text-selection granularity. Design cues (*i.e.* feedforward and affordances) should invite the user to explore the possibilities of the technique, and should not hide information about the interaction mechanisms. For example, force-based selection in iOS enables sentence-level and paragraph-level selection features, but these are hidden from the user – most users discover this feature by luck, through social suggestion or by reading online documentation. In contrast, the visual affordances of handles mean that interactions with handles are easily discoverable.

Design consideration 2: Favor novice-to-expert transition
Once aware of the possibilities of the technique, the user needs to improve with it. The rehearsal hypothesis [15, 10], which states that novices should carry out selections in the same way that experts do (although with different feedback), can be followed so novices build up experience with the actions needed for expert use. This design consideration intends that users will improve with practice – often following the power law of practice [19], where users will move from an *initial performance* phase, to a *extended learnability* phase, and eventually to an *ultimate performance* phase [6, 24].

Design consideration 3: High performance ceiling
Another important property is that the technique should offer a high performance ceiling (ultimate level of performance achievable once the user is expert). One of the problem observed in standard techniques is that only a single strategy is supported (*e.g.* selecting a word and dragging handles), which caps the performance ceiling. The use of force is an example of an augmentation that raises the ceiling – for example, because users can now select a whole paragraph without moving the finger.

Design consideration 4: Fluid mode switching
Finally, expert techniques should ideally offer different features that are better suited for different tasks. Those features could be reached using mode switching mechanisms. Going from one mode to another should be a fluid and easy task (*e.g.* switching between absolute and relative pointing using simple unistroke gestures [7]). In iOS, changing the granularity of selection (*e.g.* from word to sentence level) is achieved through a pattern of force touches, but with timing that is difficult for users to discover.

TECHNIQUE DESIGN
In the following sections, we describe the interaction mechanisms of FORCESELECT (FS) and FORCESELECT HIGH-PRESSURE (HP), two novel text selection techniques that make use of the force-sensing capabilities of modern smartphones, with enhanced visual feedback using a "mode gauge" visualization (figure 1). We also describe NORMAL (N), the iOS pressure-based text selection technique.

FORCESELECT **design**
FORCESELECT maps the force sensed by the touchscreen to the granularity of text selection: the higher the force, the more text is selected (figure 2). A long press on the screen displays
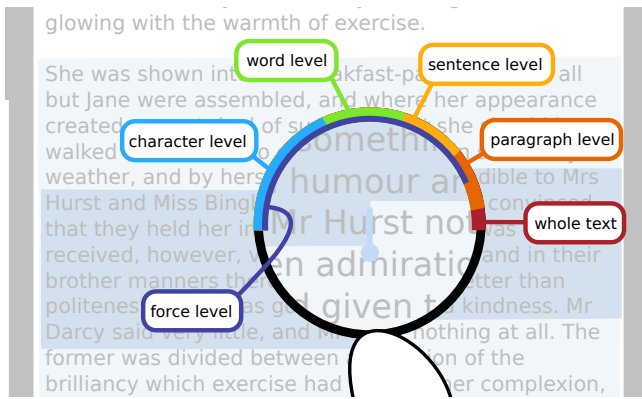
Figure 1: Close-up of the "mode gauge". Also, notice in the background two types of text highlighting: dark highlighting covers between both handles and light highlighting acts as a feedforward of which portion of text will be selected if the user released her finger (here the whole paragraph).
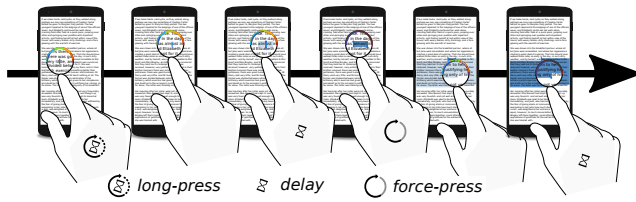

Figure 2: Example of text selection using FORCESELECT. The user performs a long-press that displays the callout magnifier. Keeping the force in the character level, the user adjusts its position by moving her finger. She then presses harder to start the manipulation of the second cursor. If she releases her finger while the force is at the sentence level of the "mode gauge", she will select the whole paragraph. If she releases her finger while the force is at the character level, she will only select what is between the two cursors.

a callout magnifier view above the finger. The callout contains a zoomed representation of the text located under the initial touch. The top semicircle around the magnifier shows a *mode gauge* whose background is segmented into five colored sections. The gauge represents the amount of force applied by the finger. If the force remains in the light-pressure region of the gauge (colored blue), moving the finger moves a cursor which is snapped to the closest character. Pressing stronger (moving the gauge's indicator above the blue region) fixes the initial cursor and starts the manipulation of a second cursor. Returning to the light-force region fixes the second cursor and enters a clutch mode: the magnifier can be moved around without changing the cursors' positions. Pressing stronger again captures the closest cursor for manipulation, until the clutch mode is entered again.

Text selection occurs when lifting the finger from the screen. At all times, the text between both cursors is highlighted. However, the selected text depends on the last level of force: if the finger is lifted off while the gauge is in the blue region, the selected text will be what is between the cursors. If the finger is lifted with the gauge in the green region, the selected text will span from the word containing the first cursor to the word containing the last cursor. Similarly, lifting the finger while in the yellow region will select at the sentence level, while in the orange region will select at the paragraph level, and in the red region will select the whole text. At all times, a lighter highlight informs the user of exactly which text would be selected if she released her finger (figure 1

If a selection is already on the screen, long-pressing near a handle will enter the clutch mode described above. However, in this case lifting the finger will only affect the handle that is currently being manipulated – for example, lifting off while in the red region while manipulating the end handle will select from the first cursor to the end of the text.

FORCESELECT HIGH-PRESSURE **design**
FORCESELECT HIGH-PRESSURE (figure 3) is a text selection technique similar to FORCESELECT. A long press on the

screen triggers the same gauge-augmented magnifier. The semantics behind the gauge colors remain the same; the difference lies in the selection trigger. Whereas with FORCESELECT, the selection depends on the force applied right before the finger was lifted-off, the selection with FORCESELECT HIGH-PRESSURE relies on spatio-temporal stability. If the force remains in the gauge's blue light-pressure region, moving the finger moves a cursor which is snapped to the closest character. Maintaining a position and pressing harder for 300ms will select text containing the cursor, with extends depending on the current force level on the gauge: a word is selected when in the green region, a sentence when in the yellow region, a paragraph when in the orange region, and the whole text when in the red region.

Once an initial selection has been made, the gauge background (except the red region) is grayed out to inform the user that the selection is locked and she has entered finger-clutch mode. While in the clutch mode, she can move the magnifier around without affecting the selection or the cursors' positions. To leave the clutch mode, the user presses stronger (*i.e.* reaching the red region of the gauge). Once unclutched, she can manipulate the closest cursor and update only that end of the selection as described above.

NORMAL **design**
The NORMAL technique is the current text selection on iOS, and uses force to enable certain features (figure 4). After a long press on the screen a magnifier appears (with no mode gauge). The user can move the magnifier around, snapping a cursor to the nearest character of the text. Without lifting the finger, the user can increase and decrease the force applied (simulating a physical *press* action) which results in selecting the word located below the finger and displaying two cursors right before (respectively after) the first (last) character of the word. Moving the magnifier around snaps one of the two cursors to the nearest word while the other cursor stays at its initial position. If the user performs another *press* action, the cursors disappear and moving the magnifier highlights the word under the finger. Additional presses on the screen cycle through these two modes. Three rapid presses enable a

Figure 3: Example of text selection using FORCESELECT HIGH-PRESSURE**. The user performs a long-press that displays the callout magnifier. Keeping the force in the character level, the user adjusts its position by moving her finger. She then holds the force in the word level of the "mode gauge", locks the selection and enters the clutch mode. When force-pressing to the whole text level of the "mode gauge", she un-clutches the selection and updates it.**

similar snapping mode at the sentence level, and four enables snapping at the paragraph level.

At any time, the user can lift her finger to confirm the current selection and convert the cursors in *handles*. Then, she can grab any of these handle and drag them at the character level or press the *Select All* button of a contextual menu that pops-up above the selected text.

EVALUATING EXPERT PERFORMANCE

Our first evaluation assessed the performance ceilings of FORCESELECT (FS), FORCESELECT HIGH-PRESSURE (HP) and NORMAL (N) with different text-selection tasks.

Apparatus

The experiment was conducted on a iPhone 6S running iOS 10.3.3. Input was provided through the touch screen (which has force-estimating capability). Experiment software was written in Objective-C with the CocoaTouch API.

Experimental focus and rationale

In order to better compare the core elements of the interactions described above, we equalized some of the design features of the three techniques: for all techniques, tapping on the screen resets the text selection, and each technique is triggered using a long press, to use direct touch manipulation and fix the magnifier behavior. However, other triggers could have been implemented as well, such as using a double tap as a trigger, or using indirect touch (*e.g.* using the iOS software keyboard as a trackpad). In addition, to avoid confounding effects and to focus on text selection performance, we disabled the scrolling mechanism of the system, disabled the software keyboard, and used the same size and zoom level for all three magnifier views.

Experimental method

Task and stimulus.
Participants were instructed to select a section of text as quickly and accurately as possible in a full-screen view displaying a long text passage. The target section of text was displayed using a red font. The study used a think-aloud protocol [16] and participants were encouraged to comment at any time during the experiment.
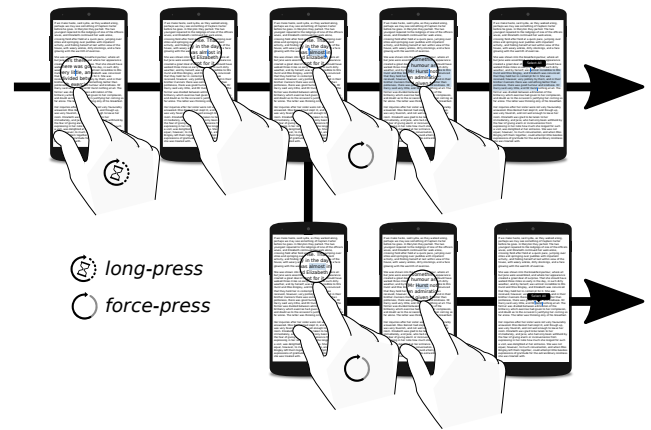

Figure 4: Example of text selection using NORMAL**. The user performs a long-press that displays the callout magnifier. The user adjusts its position by moving her finger. She then can force-press and manipulate the selection snapping the cursors at the nearest word, or force-press again and only highlight the word located below her finger.**

Procedure and design.
Participants completed the tasks with all three techniques: FS, HP and N (order balanced). For each technique, participants completed a practice phase followed by a test phase. During the practice phase, participants were asked to select some text with the given technique in a dedicated "playground" application. During this practice, the experimenter explained how the current technique worked and participants were invited to ask additional questions. Once the participant felt comfortable with the technique, the experimenter started the application used for the test phase. For each technique, the experiment consisted in a succession of 11 BLOCKS: 9 REPBLOCK followed by 2 MIXBLOCK. Each REPBLOCK is focused on one of the 9 different sections of text used as target for the experiment (we later refer to these section as *Tasks*). In each block, participants started with 2 training trials (advertised as such) in order to test and refine their own strategies for each task, and then completed 8 repetitions of the same task. By repeating the same task in these blocks, we intended that participants were able to reach expert performance more quickly. Each MIXBLOCK presented each of the 9 tasks once in a random order. The mixture of tasks was intended to test a more realistic usage setting where people use text selection more infrequently and in with different ranges of selection.

Tasks always involved a sequence of consecutive letters and punctuation marks, but differed in terms of size (*i.e.* the number of characters to select) and in terms of where the first and last character were positioned in relation to a word, sentence or paragraph. The 9 TASKS were chosen to cover the variety of selections that can be required in real-world use:

- *Sub-word*: selecting 4 letters inside a 10-letter word (*c.f.* figure 5a);

- *Word and char*: selecting 2 words and half of an 8-letter word (*c.f.* figure 5b);

- *Four words*: selecting 4 words (*c.f.* figure 5c)
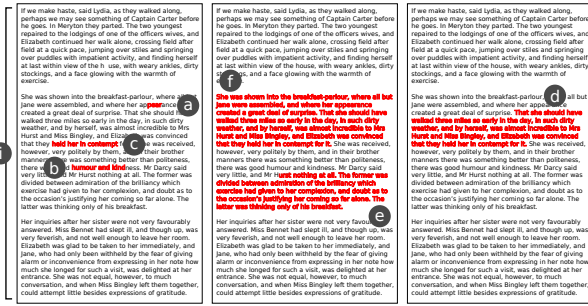
Figure 5: TASKS **used in the experiments: a)** *Sub-word*, **b)** *Word and char*, **c)** *Four words*, **d)** *Sentence*, **e)** *Char to paragraph*, **f)** *Two sentences*, **g)** *Paragraph*, **h)** *Two paragraphs* **and i)** *Whole text*.

- *Sentence*: selecting one sentence spanning 5 lines (*c.f.* figure 5d);

- *Char to paragraph*: selecting from the middle of a word to the end of a paragraph (*c.f.* figure 5e);

- *Two sentences*: selecting 2 sentences spanning 7 lines located at the beginning of a paragraph (*c.f.* figure 5f);

- *Paragraph*: selecting an entire paragraph spanning 15 lines (*c.f.* figure 5g);

- *Two paragraphs*: selecting 2 paragraphs spanning 26 lines (*c.f.* figure 5h);

- *Whole text*: selecting the whole text which spans 37 lines (*c.f.* figure 5i).

The goal behind this panel of tasks was to assess whether or not selections with mixed granularity were feasible (from a user standpoint) using our techniques, and if the techniques added value for conventional tasks. The order of REPBLOCK was randomized for each technique. Once the 2 MIXBLOCK were completed, the participant answered a NASA-TLX questionnaire [13]. The experiment then moved on to the next technique. Once all techniques had been completed, the participant was asked which technique she felt was the slowest/fastest, which was the most/least accurate, and which technique she preferred the most/least. The experimenter then opened the interview asking what were the motives that led to those choices and steered the conversation to discuss the design of each technique. Participant were also asked which features of each technique they would have discovered on their own. The interview ended with open comments. In total, the experiment lasted approximately 1h15 (including about 30min of debriefing). Participants were allowed to take breaks between blocks.

Design.
The experiment used a $3 \times 9 \times 2$ within-subject design with the factors *Technique* (FS, HP, N), *Task* (Sub-word, Word and char, Four words, Sentence, Char to paragraph, Two sentences, Paragraph, Two paragraphs, Whole text) and *Block type* (REPBLOCK, MIXBLOCK). The order of *Technique* was counterbalanced across participant using a Latin Square, and the order of *Task* was randomized across participants as a way to counterbalance across subjects given the large number of
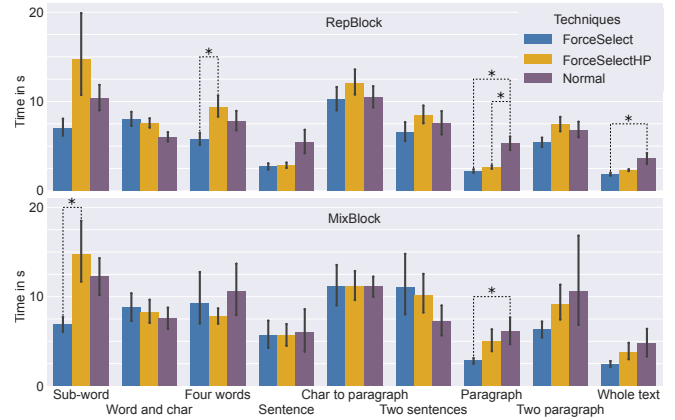
Figure 6: Average completion time in ms for each TASKS, **each** TECHNIQUES **and both** REPBLOCK **and** MIXBLOCK. **Error bars correspond to the 95% CI. * represents significant differences.**

tasks. MIXBLOCK were always carried out after all REPBLOCK had been completed. In total, each participant performed 3 TECHNIQUES × (9 REPBLOCK × 8 REPETITIONS + 2 MIXBLOCK × 9 TASKS) = 270 correct trials that were kept for analysis. For each of these trial, we logged the completion time, as well as the raw input data.

Participants
We asked 12 participants (25.6 mean age, 4.4 std dev, all right-handed, 1 female, recruited at the local university campus) to complete our study. In the following sections, we will refer to these participants as P_1 to P_{12}. All were daily users of touchscreen, 5 owned an iOS device, and one had already played a game relying on a force-sensing touchscreen.

Results
The aim of the analysis is to compare TECHNIQUES for two settings: one where the users are experts (*i.e.* the analysis focused on the REPBLOCK blocks) and one that corresponds to occasional use with different kinds of selections (*i.e.* the analysis focused on the MIXBLOCK blocks). The mean times of all conditions are summarized in figure 6. The grand means for REPBLOCK were: FS: $\bar{m}$=5.5s, sd=4.8s; HP: $\bar{m}$=7.5s, sd=9.9s; N: $\bar{m}$=7.1s, sd=6.0s). The grand means for MIXBLOCK were: FS: $\bar{m}$=7.1s, sd=5.9s; HP: $\bar{m}$=8.4s, sd=5.8s; N: $\bar{m}$=8.5s, sd=7.0s).

The following quantitative analyses report repeated-measures ANOVAs using the *ez* package of *R*. When significant effects are found, we report post-hoc pairwise t-tests with the Holm adjustment method. All qualitative analyses report Friedman tests for repeated measurements using the *scipy.stats* package of *Python*. T-tests for the means of two independent samples of scores are conducted when significant effects are found.

Expert behaviors (repeated blocks)
Since we were interested in expert performance, we first tested if there was a learning effect across REPETITIONS for each of the TASKS in REPBLOCK. For the TASK *Sub-word*, *Word and char*, *Paragraph* and *Whole text* there was no effect of REPETITION on time (all $F_{7,84} < 1.9$ and all $p > 0.07$).

For the TASKS *Four words*, *Sentence*, *Char to paragraph*, *Two sentences* and *Two paragraphs* we found a significant main effect of REPETITION (all $F_{7,84} > 2.3$ and all $p < 0.04$). However, the weak Generalized Eta-Squared ($all \eta^2 < 0.06$) and the lack of significance in post-hoc analysis (all $p > 0.2$) suggested that learning effect were minimal, and that participants had (as intended) reached a reasonable level of expertise with the tasks. We therefore keep all the REPETITIONS for all TASKS in the subsequent analysis of REPBLOCK.

We then tested the effect of TECHNIQUES on time in the REP-BLOCK blocks, aggregating all trials using the mean. For the TASK *Sub-word*, *Sentence*, *Char to paragraph*, *Two sentences* and *Two paragraphs* there was no effect of TECHNIQUES on time (all $F_{2,24} < 3.2$ and all $p > 0.05$).

For the TASK *Word and char* we found a significant main effect of TECHNIQUE ($F_{2,24}$=4.0, p<0.04 and η^2=0.25). However post-hoc analysis did not show any difference due to high standard deviation (all p>0.1; FS: $\bar{m}$=8.0s, sd=3.1s; HP: $\bar{m}$=7.7s, sd=2.1s; N: $\bar{m}$=6.0s, sd=1.5s).

For the TASK *Four words* we found a significant main effect of TECHNIQUE ($F_{2,24}$=4.5, p<0.03 and η^2=0.27). Post-hoc analysis revealed that FS was significantly faster than HP (p<0.3; FS: $\bar{m}$=5.7s, sd=2.3s; HP: $\bar{m}$=9.5s, sd=4.0s; N: $\bar{m}$=7.8s, sd=4.0s).

For the TASK *Paragraph* we found a significant main effect of TECHNIQUE ($F_{2,24}$=16.3, p<0.0001 and η^2=0.58). Post-hoc analysis revealed that N was significantly slower than both FS and HP (all p<0.001; FS: $\bar{m}$=2.2s, sd=0.6s; HP: $\bar{m}$=2.6s, sd=0.5s; N: $\bar{m}$=5.4s, sd=2.7s).

For the TASK *Whole text* we found a significant main effect of TECHNIQUE ($F_{2,24}$=4.4, p<0.03 and η^2=0.27). Post-hoc analysis revealed that N was significantly slower than FS (p<0.02; FS: $\bar{m}$=1.8s, sd=0.6s; HP: $\bar{m}$=2.3s, sd=0.4s; N: $\bar{m}$=3.7s, sd=2.7s).

Occasional use (mixed blocks)
We then tested the effect of TECHNIQUES on time in the MIXBLOCK blocks, aggregating both trials using the mean. For the TASK *Word and char*, *Four words*, *Sentence*, *Char to paragraph*, *Two sentences*, *Two paragraphs* and *Whole text* there was no effect of TECHNIQUES on time (all $F_{2,24}$ <3.2 and all p>0.05).

For the TASK *Sub-word* we found a significant main effect of TECHNIQUE ($F_{2,24}$=8.4, p<0.002 and η^2=0.41). Post-hoc analysis revealed that FS was significantly faster than HP (p<0.02; FS: $\bar{m}$=6.9s, sd=2.0s; HP: $\bar{m}$=14.7s, sd=8.5s; N: $\bar{m}$=12.2s, sd=5.2s).

For the TASK *Paragraph* we found a significant main effect of TECHNIQUE ($F_{2,24}$=5.4, p<0.02 and η^2=0.31). Post-hoc analysis revealed that N was significantly slower than FS (p<0.01; FS: $\bar{m}$=2.8s, sd=0.6s; HP: $\bar{m}$=5.0s, sd=2.6s; N: $\bar{m}$=6.1s, sd=3.5s).

Subjective effort and preferences
We summarize participant responses for both the NASA-TLX questionnaire and the rankings in table 1. For the

	FS	HP	N	χ^2	p
Mental demand	4.8 (2.1)	6.8 (1.2)	5.8 (1.8)	8.2	**0.02**
Physical dem.	4.3 (1.8)	5.2 (2.1)	5.9 (2.0)	6.3	**0.04**
Temporal dem.	4.5 (1.7)	5.5 (2.1)	5.7 (2.1)	5.8	0.06
*Performance	7.2 (1.2)	6.5 (1.7)	6.0 (1.8)	4.9	0.09
Effort	5.3 (1.7)	5.4 (1.7)	6.3 (1.5)	3.9	0.14
Frustration	3.3 (1.7)	4.6 (2.1)	6.6 (1.8)	17.8	**<0.001**
Speed	1.3 (0.6)	2.1 (0.5)	2.7 (0.6)	12.2	**<0.005**
Accuracy	1.6 (0.6)	1.8 (0.7)	2.7 (0.7)	8.2	**0.02**
Preference	1.4 (0.6)	1.9 (0.5)	2.7 (0.8)	9.5	**<0.01**

Table 1: Summary of the Friedman test for the NASA-TLX questionnaire and the rankings. Results of the 3 first columns show the mean (and standard deviation). For the NASA-TLX items (top), marks range from 1 to 10. Lower marks are better, except for *. For the rankings item (bottom), marks range from 1 to 3. Lower marks are better.

NASA-TLX, we found a significant difference for: *mental demand* (HP was found to be more demanding than FS, p=0.02), *physical demand* (however no post-hoc differences were found) and *frustration* (N was found to be more frustrating than both FS and HP, all p<0.03). For the rankings, we found a significant difference for: *speed* (FS was found to be faster than HP which was faster than N, all p<0.03), *accuracy* (N was found to be less accurate than both FS and HP, all p<0.005) and *preference* (preference for N was lower than for both FS and HP, all p<0.02).

Summary
Overall the results of study 1 show that the new techniques were significantly faster for some tasks (that required selecting a large number of characters), and that the standard technique was never faster than the new techniques. This suggests that although FS and HP are designed to support expert use, they do not diminish casual user performance. Participant preferences and comments were strongly in favor of the new techniques, with most people preferring FS. The standard deviation for both HP and N was also higher than for FS, suggesting that performance may vary more across users for those technique (whereas FS seems to be more consistent), but also that users may have still been learning the new techniques, as discussed above.

Finally, we want to note that differences in the *Whole text* task may be due to the fact that some participants, even though aware of the *select all* button in N's contextual menu, did not use that feature. Therefore we hypothesize that the significant differences does not account fully for better performances but rather reveals an affordance issue.

EVALUATING DISCOVERABILITY
We conducted a follow-up study to assess the discoverability of the force/selection range mappings of FS, HP and N.

Method, procedures and tasks
We recruited 6 participants (26.5 mean age, 3.2 std dev, all right-handed, 2 females, 2 of them self-reporting that they "struggle with technology"). None of them participated in the previous study. All were daily users of touchscreen, two were users of iOS devices, and none had used a force-sensitive

screen before. In the following sections, we will refer to these participants as P_{13} to P_{18}.

The experiment used the same apparatus as the previous study (iPhone 6S and experimental software implemented in Objective-C and Cocoa). For each technique, participants were presented with the same playground application used in the previous study, for an exploration phase. Participants were only informed that the touch-screen had force sensing capabilities, that tapping on the screen would reset the selection, and that a long press would trigger the selection mode. However, unlike in previous study, no information was given regarding how the techniques worked or how the force applied to the touchscreen is mapped the range of text selected. Participants were invited to notify the experimenter once they felt comfortable with the technique. After notifying the experimenter, the participant was asked to complete 2 MIXBLOCK blocks, similar to the previous study, in order to test the techniques across a variety of tasks. Once the blocks were finished, the experimenter asked the participant to describe how the technique worked in her own words. After that, the experiment moved on to the next technique. All participants experimented all techniques, and all orders of presentation were tested (one participant in each order). Once the experiment had been completed with all techniques, the participant was asked to answer which technique she felt was slowest/fastest, most/least accurate, most/least preferred, and easiest/hardest to discover, and to describe the rationale behind her choices. Finally, if the participant did not find all the features of a technique, the experimenter demonstrated them and asked the participant if, in her opinion, she would have discovered them on her own given time. Overall, the experiment lasted approximately 50 minutes per participant.

Results and observations

Although we did not ask people to work quickly, it is interesting to note the different amounts of time that participants needed to become comfortable with the different techniques (that is, the duration of the exploration phase). It took on average 3.2 minutes for participants to become comfortable with the FS technique (4 minutes if it was the first technique); for HP, the average times were 6.5 minutes (9.5 minutes if first), and for N, the average times were 4.7 minutes (8 minutes if first).

For FS, all participants discovered the main force-based principle of the technique, but four people did not discover that long pressing back on the screen would refine the selection of one handle only (*i.e. physical clutch*). In the interview, all of these participants suggested that they would have found this functionality eventually.

For HP, all participants but one discovered the principle of the technique but three did not discover the physical clutching mechanism (*i.e.* lifting off the finger and long-pressing back on the screen); two others discovered the physical clutching mechanism but not the high-pressure clutch (*i.e.* keeping the finger in contact and pressing to the red region to unlock the gauge). One participant had a very different mental model of this technique, and managed to finish the task without noticing that the selection would lock after holding both position

and force. Nonetheless, he suggested that with a bit more time he "would have picked that up".

For N, all participants discovered that simulating a button press while in contact with the screen would highlight the word underneath the finger. However, two participants triggered this feature but then couldn't remember how they did it (both had N as the first technique). After few minutes, the experimenter reminded the participants that the screen had force sensing capabilities. One of the participants could not reproduce word highlighting consistently, and used *select all* as an entry point to refine the selection with the handles. Two of the six participants discovered the one word only highlight feature: one figured it out while she was explaining the technique to the experimenter, and the other discovered it out of frustration (tapping rapidly to get the initial highlight). No participant discovered the sentence or paragraph snapping features. All participants suggested during the interview, however, that they would have stumbled upon these features and eventually figure them out.

Summary

Overall, our observations show that participants were quite fast in exploring the different techniques, with N taking the longest. There was an obvious difference in the discoverability of the force mapping: even though N was easy to understand (one participant felt comfortable after only a minute and a half), some participants entered a local optimum thinking that there were no other possibilities (confirmed by their own explanations of the technique). It was surprising that participants explored FS for less time but were still able to figure out the range of possibilities. They took more time for HP, but this is mainly due to one participant taking 17 minutes to thoroughly explore it (this participant found all of the features of the technique). If this participant is removed from the sample, the mean exploration time for HP is 4.4 minutes.

PARTICIPANT COMMENTS (BOTH STUDIES)

Overall there was a strong preference for the new techniques (15 of 18 people chose either FS or HP as their preferred method), and particularly for FS (13 participants of 18). During the interview, it was clear that a major part of this preference was the feedback and feedforward provided by the mode gauge. For example, P_2 said "feedback on the force level is important"; P_4 said "I could have figured out the mechanism of [FS] and [HP] thanks to the color"; P_5 said "The control of the force is much easier than expected with the gauge"; and P_8 said "I liked [FS] because the gauge was helpful for force feedback."

Participants' comments also suggest that they found FS more intuitive than the other techniques: P_3 said "I preferred [FS] because there was room for error correction. Out of the 3 technique, it was the most accurate and intuitive."; P_4 said ""[FS] is easy to get your head around it"; P_6 said "I liked the sense of control in [FS]"; and P_{11} said ""[FS] is easy to handle, it felt intuitive for me".

Participants comments also suggested that the low preference for N arose from three issues. First, people were frustrated with the accuracy needed when releasing the handles: P_1 said

"The character precision level is difficult and frustrating"; P_5 said "the exit of handles was difficult to get right", P_{11} "It needs a lot of effort to complete a task because of the accuracy needed when you release". Second, the timing of presses needed to enable sentence and paragraph snapping was difficult to discover and achieve: P_4 said "[N] needs too much precision for the rhythm for the sentence/paragraph and is really frustrating at the beginning"; P_5 said "It took too long to get the rhythm"; and P_6 said "It was difficult to get the timing, I had to think more and it was easier to get frustrated." Third, people found the lack of force feedback when circling through the modes annoying: P_2 said "If [N] would have had feedback it would definitely be better"; P_5 said "I wished it had a feedback on the amount of force"; and P_6 said "[N] lacked feedback".

When participants did express a preference for N, it appeared to arise from their existing familiarity with the technique: P_{10} said "It was similar to the iOS technique, it's easy to get and I prefer to drag handles. But I didn't get the cycle between word/sentence/paragraph selection, it caused me to press hard and was tiring"; and P_{16} said "it was the most familiar". One participant also found the N technique simpler, saying "[N] is easier to use.". However, two of the three participants who preferred N also stated that they might change their preference after further use: for example, P_5 said "[N] is easier to use but in the long term [FS] would be better. For the accuracy [FS] was better", and P_{10} said "The release [while using FS] was better because I controlled the selection. The concept is hard to get, but once you get it, it's better than [N]".

The main problems raised for HP involved the time pressure induced by the locking mechanism, which sometimes was perceived as an error (*e.g.* P_5 said "I often ended up selected more than I wanted"). Participants also had positive comments about this technique however: they liked that the locking mechanism preserved the current selection, and gave them time to refine the selection at their own pace (*e.g.* P_2 said "It feels nice when the selection is locked down because you have plenty of time to think of the next move. The next move needs to be planned though because it locks the new selection down quickly").

Only one participant ranked FS last, but stated "maybe because it was the first technique and I was not familiar with the color"; this may also be why he liked [HP] the most (as he said, "because [he] was accustomed").

Finally, comments suggested that some participants did not feel completely expert while using the new techniques: P_2 said "If people are using [FS and HP] techniques a lot, they would definitely get faster than with [N]"; P_9 said "I would have got [FS and HP] eventually but not as quickly as [N]"; and P_{11} said "Maybe after you use [HP] a lot you get used to it but at first it was not easy to handle". These comments suggest that the new techniques may perform better with continued experience.

DISCUSSION

Overall both FS and HP were well received as technique enabling expert behavior: P_{11} "I hope to see those new techniques on laptops. Not really for smartphone though. I do too few text selection for them to be useful. But on the laptop with pressure, yes", P_{15} "I feel like you can become good with [FS] and [HP]. I would have taken the time to learn it, like I do when I'm getting a new device. You know those times when you get a new gadget and starts fiddling with it.", P_{17} "[FS] would be great for text selection, I'm doing that everyday on my smartphone. It would be even more useful on a touch screen used in a work context. It's because of that that I'm not using a touch laptop.".

Both FS and HP were designed to enrich interaction mechanisms that already exist (*e.g.* use of the same screen real estate, use of the same magnifier). Therefore, they could be easily integrated in current system.

Results explanation and generalization

After compiling our observations on both experiments, we identified various points that might account for the difference in performances. In situations where FS and HP performed better, we argue that the feedback provided by the gauge helped to build a mental model of the force-based mechanisms, and thus understand the current state of the interaction. The possibility to quickly access the different levels of entities (*i.e.* word, sentence, paragraph, whole text) and the fact that all of them were accessed the same way was also beneficial. The last advantage of our techniques over classical ones is that they require less precision when the finger is released from the touchscreen: both capitalize on the last stable text being selected for the final selection. When it comes to situations where there were no performance improvement, we argue that it might come from the different approach our techniques use compared to classical ones. We sometimes saw participants selecting the text as they would do with a classic technique: starting the selection on top of the first character to end it on top of the last one. If users don't take advantage of the granular selection, our techniques will not provide more efficiency. Finally, the notion of closest cursor might not have been intuitive for participants. Our definition of the closest cursor was based on the number of characters between the center of the magnifier and the cursors. However, from our observations it seemed that the closest cursor was sometime interpreted as the closest in term of physical distance.

Given our new approach of text selection compared to classical touch-based selection techniques, we want to be careful regarding the generalization of our results. Although the second experiment shows that users can discover and improve relatively quickly with our techniques, some training or familiarization remain required. For very casual use, if users are not willing to fiddle a bit with both FORCESELECT techniques, they might be frustrated and perform poorly compared to more familiar mechanisms. However, FS and HP remain compatible with the classic *3 phases* text selection approach described in the *Background* section, which would probably mitigate users' frustration. Moreover, if FS and HP trigger their curiosity or if they perform a lot of text selections and therefore practice with the techniques, we argue that our results will stand and are likely to improve over time.

Limitations and further work

To design the mapping between the force and the granularity of text selection, we relied on feedback gathered during informal studies as well as a trial and error iterative design process. The blue part size was tuned to capture casual touches. The upper limit was set so that the force required to pass the threshold was performed by an explicit action of the user. We reduced the size of the red part since maintaining a "high pressure" is relatively easy. We then divided the remaining pressure range into equal parts. Although we have not seen participants struggling to control that level of granularity, we do not claim that the chosen mapping is optimal. Further research should be done to investigate the best mapping as well as the number of levels one can easily control with a force-sensing smartphone.

During our experiments, handle-manipulation (as performed with the classic 3 steps technique) was disabled on purpose with FS and HP in order to focus our study on the expert mechanism itself. This makes the techniques harder overall (or at the very least more frustrating to use). However, enabling handle-dragging along with our gauge mechanism is feasible and can ease novice use of the technique.

Although our second study focused on the discoverability of our techniques by novices, and our our first study focused on the ultimate performance one would reach as an expert, we did not investigate in detail the steps involved in progressing from a novice to an expert level. Further studies focusing on this transition should be conducted to better assess the amount of time and training needed.

Another limitation of this work is that it requires a touch screen with force-sensing capabilities. While alternatives such as contact area have been previously used to replace force-sensing [3], it can difficulty be applied in our case because text selection requires high precision finger positioning. Unfortunately, the centroid of the contact area shifts when the users increase (or decrease) the size of the contact area. However, force is one of several possible linear degrees of freedom that could be added to an interaction, and further studies could investigate text selection with other 1D augmentations that could replace pressure on non-force-sensitive devices (*e.g.* a second finger touching the screen to provide an additional linear input, or the tilt of the device).

Our research is a first step toward force-based expert interaction techniques for text selection. In the future, we plan to investigate two main further research directions. First, we want to investigate the use of indirect pointing devices. On iOS, the software keyboard can be turned into a trackpad by force-pressing on it. Being force sensitive, this "temporary trackpad" could be used as input for our text selection techniques, raising new challenges such as the need to clutch to move the pointer. The same principle could also be studied on laptop force-sensing trackpads, such as the Apple Force Touch trackpad. Second, we focused here on text selection techniques, but at the core of our design framework are principles that could be applied to many different types of selection mechanism where the selection granularity can be adjusted using pressure. We therefore want to adapt this concept to other types of selection, such as the "magic wand" selection in Photoshop, or item selection in a file explorer.

CONCLUSION

Text selection on touch devices can be a difficult task, and even with several augmentations (*e.g.* callout magnifiers, selection handles, or word / sentence / paragraph snapping), the task can still frustrate users. The introduction of force-sensitive touchscreens has added another enhancement to text selection (using pressure for different selection modes); however, these modes are difficult to discover and many users continue to struggle with accurate selection. In this paper we reported on an investigation of the design of touch-based and pressure-based text selection mechanisms, and describe a design framework and two novel text-selection techniques that provide improved discoverability and enhanced visual feedback using a "mode gauge" visualization, and have design features intended to support a higher performance ceiling for experts as well as smooth transitions from novice use. We carried out two studies to evaluate the new techniques. The first study showed that the new techniques (and one in particular, called FS) performed well compared to the standard iOS technique, were never worse than the standard method, and were strongly preferred by participants. A second study that focused on discoverability showed that the gauge visualization was instrumental in helping users explore the range of functionality provided by the new techniques, and that the pressure features of the standard technique were difficult to discover. Our work shows the value of careful attention to design details and underlying principles such as affordances, discoverability, and transitions - even for seemingly-simple interactions like text selection.

REFERENCES

1. J. Aceituno, S. Malacria, P. Quinn, N. Roussel, A. Cockburn, and G. Casiez. 2017. The design, use, and performance of edge-scrolling techniques. *International Journal of Human-Computer Studies* 97 (2017), 58 – 76. DOI: **http://dx.doi.org/10.1016/j.ijhcs.2016.08.001**

2. Axel Antoine, Sylvain Malacria, and Géry Casiez. 2017. ForceEdge: Controlling Autoscroll on Both Desktop and Mobile Computers Using the Force. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 3281–3292. DOI: **http://dx.doi.org/10.1145/3025453.3025605**

3. Mathias Baglioni, Sylvain Malacria, Eric Lecolinet, and Yves Guiard. 2011. Flick-and-brake: Finger Control over Inertial/Sustained Scroll Motion. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems (CHI EA '11)*. ACM, New York, NY, USA, 2281–2286. DOI: **http://dx.doi.org/10.1145/1979742.1979853**

4. Joseph Chee Chang, Nathan Hahn, and Aniket Kittur. 2016. Supporting Mobile Sensemaking Through Intentionally Uncertain Highlighting. In *Proceedings of the 29th Annual Symposium on User Interface Software*

and Technology (UIST '16). ACM, New York, NY, USA, 61–68. DOI:
http://dx.doi.org/10.1145/2984511.2984538

5. Chen Chen, Simon T. Perrault, Shengdong Zhao, and Wei Tsang Ooi. 2014. BezelCopy: An Efficient Cross-application Copy-paste Technique for Touchscreen Smartphones. In *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces (AVI '14)*. ACM, New York, NY, USA, 185–192. DOI:
http://dx.doi.org/10.1145/2598153.2598162

6. Andy Cockburn, Carl Gutwin, Joey Scarr, and Sylvain Malacria. 2014. Supporting Novice to Expert Transitions in User Interfaces. *ACM Comput. Surv.* 47, 2, Article 31 (Nov. 2014), 36 pages. DOI:
http://dx.doi.org/10.1145/2659796

7. Clifton Forlines, Daniel Vogel, and Ravin Balakrishnan. 2006. HybridPointing: Fluid Switching Between Absolute and Relative Pointing with a Direct Input Device. In *Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology (UIST '06)*. ACM, New York, NY, USA, 211–220. DOI:http://dx.doi.org/10.1145/1166253.1166286

8. Vittorio Fuccella, Poika Isokoski, and Benoit Martin. 2013. Gestures and Widgets: Performance in Text Editing on Multi-touch Capable Mobile Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, New York, NY, USA, 2785–2794. DOI:
http://dx.doi.org/10.1145/2470654.2481385

9. Vittorio Fuccella and Benoît Martin. 2017. TouchTap: A Gestural Technique to Edit Text on Multi-touch Capable Mobile Devices. In *Proceedings of the 12th Biannual Conference on Italian SIGCHI Chapter (CHItaly '17)*. ACM, New York, NY, USA, Article 21, 6 pages. DOI:
http://dx.doi.org/10.1145/3125571.3125579

10. Carl Gutwin, Andy Cockburn, Joey Scarr, Sylvain Malacria, and Scott C. Olson. 2014. Faster Command Selection on Tablets with FastTap. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 2617–2626. DOI:
http://dx.doi.org/10.1145/2556288.2557136

11. Jaehyun Han, Sunggeun Ahn, and Geehyuk Lee. 2014. Push-push: A Two-point Touchscreen Operation Utilizing the Pressed State and the Hover State. In *Proceedings of the Adjunct Publication of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST'14 Adjunct)*. ACM, New York, NY, USA, 103–104. DOI:
http://dx.doi.org/10.1145/2658779.2658797

12. Jaehyun Han and Geehyuk Lee. 2015. Push-Push: A Drag-like Operation Overlapped with a Page Transition Operation on Touch Interfaces. In *Proceedings of the*

28th Annual ACM Symposium on User Interface Software & Technology (UIST '15). ACM, New York, NY, USA, 313–322. DOI:
http://dx.doi.org/10.1145/2807442.2807457

13. Sandra G Hart and Lowell E Staveland. 1988. Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. *Advances in psychology* 52 (1988), 139–183.

14. Ken Hinckley, Xiaojun Bi, Michel Pahud, and Bill Buxton. 2012. Informal Information Gathering Techniques for Active Reading. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 1893–1896. DOI:
http://dx.doi.org/10.1145/2207676.2208327

15. Gordon Kurtenbach and William Buxton. 1994. User Learning and Performance with Marking Menus. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '94)*. ACM, New York, NY, USA, 258–264. DOI:
http://dx.doi.org/10.1145/191666.191759

16. Clayton Lewis. 1982. *Using the" thinking-aloud" method in cognitive interface design*. IBM TJ Watson Research Center.

17. Sylvain Malacria, Jonathan Aceituno, Philip Quinn, Géry Casiez, Andy Cockburn, and Nicolas Roussel. 2015. Push-Edge and Slide-Edge: Scrolling by Pushing Against the Viewport Edge. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 2773–2776. DOI:
http://dx.doi.org/10.1145/2702123.2702132

18. Motoki Miura and Kenji Saisho. 2014. A Text Selection Technique Using Word Snapping. *Procedia Computer Science* 35 (2014), 1644–1651.

19. A. Newell and P. S. Rosenbloom. 1993. The Soar Papers (Vol. 1). MIT Press, Cambridge, MA, USA, Chapter Mechanisms of Skill Acquisition and the Law of Practice, 81–135.
http://dl.acm.org/citation.cfm?id=162580.162586

20. Jakob Nielsen and Rolf Molich. 1990. Heuristic Evaluation of User Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '90)*. ACM, New York, NY, USA, 249–256. DOI:
http://dx.doi.org/10.1145/97243.97281

21. Pekka Parhi, Amy K. Karlson, and Benjamin B. Bederson. 2006. Target Size Study for One-handed Thumb Use on Small Touchscreen Devices. In *Proceedings of the 8th Conference on Human-computer Interaction with Mobile Devices and Services (MobileHCI '06)*. ACM, New York, NY, USA, 203–210. DOI:http://dx.doi.org/10.1145/1152215.1152260

22. R. L. Potter, L. J. Weldon, and B. Shneiderman. 1988. Improving the Accuracy of Touch Screens: An Experimental Evaluation of Three Strategies. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '88)*. ACM, New York, NY, USA, 27–32. DOI: http://dx.doi.org/10.1145/57167.57171

23. Volker Roth and Thea Turner. 2009. Bezel Swipe: Conflict-free Scrolling and Multiple Selection on Mobile Touch Screen Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09)*. ACM, New York, NY, USA, 1523–1526. DOI: http://dx.doi.org/10.1145/1518701.1518933

24. Joey Scarr, Andy Cockburn, Carl Gutwin, and Philip Quinn. 2011. Dips and ceilings: understanding and supporting transitions to expertise in user interfaces. In *Proc. CHI*. 2741–2750. http://doi.acm.org/10.1145/1978942.1979348

25. Andrew Sears and Ben Shneiderman. 1991. High Precision Touchscreens: Design Strategies and Comparisons with a Mouse. *Int. J. Man-Mach. Stud.* 34, 4 (April 1991), 593–613. DOI: http://dx.doi.org/10.1016/0020-7373(91)90037-8

26. Ben Shneiderman. 2010. *Designing the user interface: strategies for effective human-computer interaction*. Pearson Education India.

27. Kenji Suzuki, Kazumasa Okabe, Ryuuki Sakamoto, and Daisuke Sakamoto. 2016. Fix and Slide: Caret Navigation with Movable Background. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '16)*. ACM, New York, NY, USA, 478–482. DOI: http://dx.doi.org/10.1145/2935334.2935357

28. Daniel Vogel and Patrick Baudisch. 2007. Shift: A Technique for Operating Pen-based Interfaces Using Touch. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '07)*. ACM, New York, NY, USA, 657–666. DOI: http://dx.doi.org/10.1145/1240624.1240727