A Three-Step Interaction Pattern for Improving Discoverability in Finger Identification Techniques

Alix Goguey[1], Géry Casiez[2], Daniel Vogel[3], Fanny Chevalier[1], Thomas Pietrzak[2] & Nicolas Roussel[1]

[1]Inria Lille, [2]University of Lille, France, [3] University of Waterloo, Canada

{alix.goguey, fanny.chevalier, nicolas.roussel}@inria.fr, {gery.casiez, thomas.pietrzak}@lifl.fr, dvogel@uwaterloo.ca
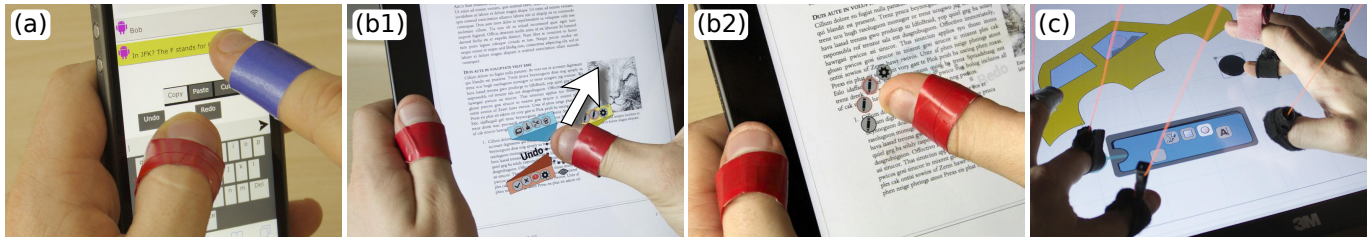
Figure 1: Our three-application demo: (a) SMS application: the thumb can be used to feed-forward the current mapping between individual fingers and commands; here, the index triggers the *copy* command. (b) PDF annotation application: (b1) thumb showing the different subsets of commands available for selection; after sliding the thumb in the corresponding direction, (b2) updated feed-forward corresponding to the top-right set. (c) Vectorial drawing application demo: non-dominant thumb and index select a command mapping enabling the dominant ring finger to draw an ellipse.

ABSTRACT

Identifying which fingers are in contact with a multi-touch surface provides a very large input space that can be leveraged for command selection. However, the numerous possibilities enabled by such vast space come at the cost of discoverability. To alleviate this problem, we introduce a three-step interaction pattern inspired by hotkeys that also supports feed-forward. We illustrate this interaction with three applications allowing us to explore and adapt it in different contexts.

Author Keywords

Multi-touch; finger identification; shortcuts; command selection; parameter control; direct manipulation

ACM Classification Keywords

H.5.2 Information interfaces and presentation: User interfaces

INTRODUCTION

Finger identification is a way to increase the multi-touch input space. Significant previous work has tackled the technical sensing problem with various approaches including inferring finger identity based on geometric relationships between touch contact points [6], using an overhead camera to track bare hands [4] or fingers with coloured rings [7], wearing gloves with fiducial markers [5], recognizing fingerprints [2], and forearm electromyography [1]. It seems inevitable that one day finger identification will be a standard feature of consumer multi-touch devices.

Robust finger identification sensing is on the horizon [1, 2], but its implications for interaction have only been considered in isolated point designs where chords were mapped to commands. Marquardt *et al.* built an exploratory toolkit for tabletops [5] and explored a number of interaction technique ideas to emphasize the toolkit's expressiveness. However, these explorations did not address the problem of discoverability brought by finger identification: discriminating fingers on a touch screen theoretically enables the access to up to 1023 finger combinations, a too large of a set to choose from for one who is not aware of the fingers-to-command mapping.

To address the discoverability problem associated with this large input space, we propose a three-step interaction pattern inspired by the combined use of keyboard shortcuts and a mouse in traditional graphical applications where 1) the non-dominant hand activates a mode with a key, 2) the dominant hand manipulates the mouse, and 3) the non-dominant hand optionally constrains the manipulation with a modifier key.

In our case, on *step 1*, a set of (*selector*) fingers determines a command mapping for another set of (*trigger*) fingers. On *step 2*, *trigger* fingers select a command and manipulate its parameters in a single stroke (Figure 1-c). On *step 3* (optional), *selector* fingers can constrain the manipulation. This three-step pattern only requires users to remember which fingers are in which one of the *selectors* set or the *triggers* set. More importantly, the separation between *selectors* and *trig-*

gers allows the introduction of on-demand feed-forward: the command mapping can be visualized on-screen as a crib sheet if the user pauses after the first step. This on-demand feed-forward makes it easy for novice users to browse the available commands, without hindering expert performance.

Depending on the context (*e.g.* smartphone, tablet, tabletop), the crib sheet can take different forms and be invoked in different ways by one or more *selectors* fingers. It may be triggered by a chord (Figures 1-a, 1-c) or a selection in a marking menu [3], for example (Figures 1-b1, 1-b2). We prototyped several applications to illustrate this three-step pattern in different contexts, that we describe in the following.

PROOF OF CONCEPT

We developed three applications: an SMS editor on a smartphone, a PDF annotation application on a tablet, and a vectorial drawing application on a tabletop system. For the smartphone and the tablet, considering the devices' form factors and the chosen applications, we assigned the *selectors* and *triggers* to a single hand and used only the thumb as a *selector*. For the tabletop application, we used the non-dominant hand's fingers as *selectors* and the dominant ones as *triggers*.

Hardware and low-level software configuration

Our focus is not on the technical problems of finger identification, but to experiment with our approach, we need to reliably identify which fingers are in contact with the display.

For this demonstration, we chose to attach rings tagged with two-dimensional barcodes to the user's fingers and to track their 2D position using a 640×480px RGB camera running at 20Hz and computer vision code. This technique can reliably track up to 6 fingers (Figure 2). The tracked positions are transformed into the display reference frame using an homography computed from a short calibration procedure. Our software makes it possible to generate custom touch events with hand and finger IDs in Qt applications running on Android or iOS devices.
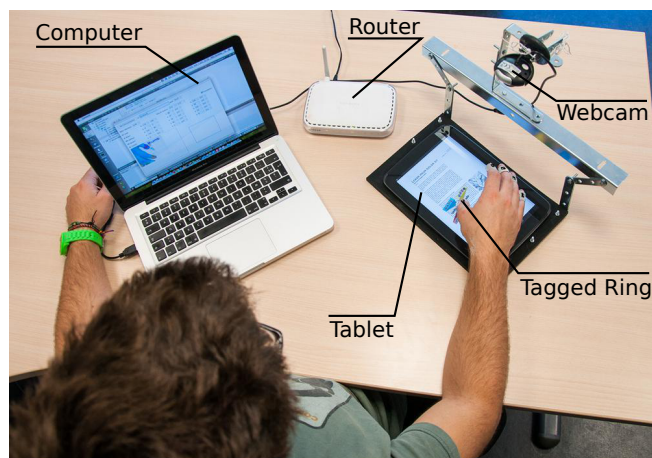

Figure 2: Demonstration setup: the user wears tagged rings that are tracked by the computer using the webcam; the tablet sends the touch inputs to the computer via the router; the computer then merges both webcam and tablet information and identifies the touches which are sent back to the tablet via the router.

Applications

The SMS writing application (Figure 1-a) uses our three-steps pattern to support commands such as copy/cut/paste, spelling corrections and undo/redo, which are traditionally accessed on smartphones through contextual menus, dwell or gestures (*e.g.* shaking the device to undo). Considering the relatively low number of commands, we used one *selector* finger only for command selection – enabling two mappings of *trigger* fingers: with or without the *selector*.

For the PDF annotation application, we identified 19 common commands in our informal observations. Considering this number and inherent relationships between the commands, we decided to break down this set into three subsets that can be selected by moving the thumb in different directions: when the thumb crosses a threshold distance represented by the dotted circle in Figure 1-b1, the corresponding subset is selected (e.g., all different ink tools). Once a subset is selected, the feed-forward is updated to show the four tools associated to the remaining *trigger* fingers (Figure 1-b2). Some commands can also be constrained using the thumb (e.g., freeform vs. by-line highlighting).

For the vectorial drawing application, we identified 26 common commands. We grouped these commands into seven categories comprising at most five commands each. Chord combinations of three fingers of the non-dominant hand are used to select command mappings (i.e. categories), while the five fingers of the dominant hand are used to trigger the commands (Figure 1-c). Shape creation tools can be constrained in several ways (*e.g.* aspect ratio, snapping) and those constraints can be combined.

NOTE

The demonstration will be interactive. We will give people a take away set so they can try at the conference and at home.

REFERENCES

1. Benko, H., Saponas, T. S., Morris, D., and Tan, D. Enhancing input on and above the interactive surface with muscle sensing. In *Proc. ITS*, ACM (2009), 93–100.

2. Holz, C., and Baudisch, P. Fiberio: a touchscreen that senses fingerprints. In *Proc. UIST*, ACM (2013), 41–50.

3. Kurtenbach, G., and Buxton, W. User learning and performance with marking menus. In *Proc. CHI*, ACM (1994), 258–264.

4. Malik, S., Ranjan, A., and Balakrishnan, R. Interacting with large displays from a distance with vision-tracked multi-finger gestural input. In *Proc. UIST*, ACM (2005), 43–52.

5. Marquardt, N., Kiemer, J., Ledo, D., Boring, S., and Greenberg, S. Designing user-, hand-, and handpart-aware tabletop interactions with the TouchID toolkit. In *Proc. ITS*, ACM (2011), 21–30.

6. Wagner, J., Lecolinet, E., and Selker, T. Multi-finger chords for hand-held tablets: Recognizable and memorable. In *Proc. CHI*, ACM (2014), to be published.

7. Wang, J., and Canny, J. FingerSense: augmenting expressiveness to physical pushing button by fingertip identification. In *CHI EA*, ACM (2004), 1267–1270.