BackXPress: Using Back-of-Device Finger Pressure to Augment Touchscreen Input on Smartphones

Christian Corsten[†] **Bjoern Daehlmann**[††] **Simon Voelker**[†] **Jan Borchers**[†]

RWTH Aachen University
52074 Aachen, Germany
[†]{corsten, voelker, borchers}@cs.rwth-aachen.de, [††]bjoern.daehlmann@rwth-aachen.de

ABSTRACT

When people hold their smartphone in landscape orientation, they use their thumbs for input on the frontal touchscreen, while their remaining fingers rest on the back of the device (BoD) to stabilize the grip. We present *BackXPress*, a new interaction technique that lets users create BoD pressure input with these remaining fingers to augment their interaction with the touchscreen on the front: Users can apply various pressure levels with each of these fingers to enter different temporary "quasi-modes" that are only active as long as that pressure is applied. Both thumbs can then interact with the frontal screen in that mode. We illustrate the practicality of BackXPress with several sample applications, and report our results from three user studies: Study 1 investigated which fingers can be used to exert BoD pressure and found index, middle, and ring finger from both hands to be practical. Study 2 revealed how pressure touches from these six fingers are distributed across the BoD. Study 3 examined user performance for applying BoD pressure (a) during single touches at the front and (b) for 20 seconds while touching multiple consecutive frontal targets. Participants achieved up to 92% pressure accuracy for three separate pressure levels above normal resting pressure, with the middle fingers providing the highest accuracy. BoD pressure did not affect frontal touch accuracy. We conclude with design guidelines for BoD pressure input.

ACM Classification Keywords

H.5.2. User Interfaces: Input Devices and Strategies (e.g., mouse, touchscreen)

Author Keywords

Back-of-Device; pressure; bimanual input; smartphone

INTRODUCTION

Interaction on mobile devices, such as smartphones and tablets, is usually done by touching the frontal screen that serves both for input and output. In recent years, research has not only looked into input at the front, but also at the back of such mobile devices. This back-of-device interaction (BoDI) complements classic touch interaction at the front in two ways: First, it mitigates the occlusion problem, since the finger touching from behind does not occlude the visual output at the front [35, 1]. Second, it enables the use of more fingers for interaction, since when holding the device in portrait orientation, usually only thumb and/or index finger interact at the front, and in two-handed landscape orientation, only the thumb(s) can interact at the front. In contrast, more fingers are available to touch at the back, e.g., for text input [4, 26].

CHI 2017, May 06 - 11, 2017, Denver, CO, USA

ISBN 978-1-4503-4655-9/17/05...$15.00
DOI: http://dx.doi.org/10.1145/3025453.3025565

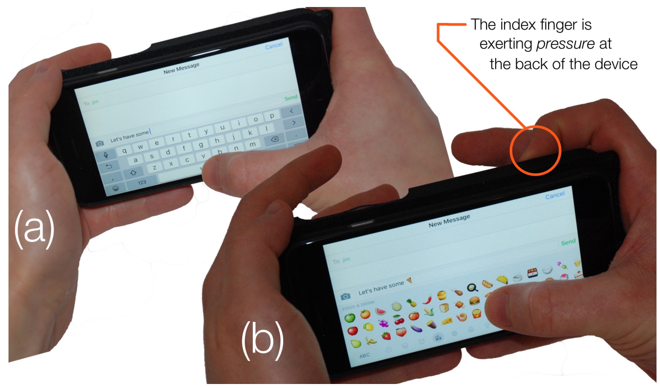

Figure 1. Emoji insertion with BackXPress: (a) The user is typing a message on a smartphone keyboard using both thumbs. (b) By applying back-of-device pressure, the keyboard switches to emoji layout. Maintaining the pressure and tapping an emoji appends it to the message. Releasing pressure reverts to the text keyboard, and the user can continue typing immediately.

So far, most BoDI research has only used the location information of touches at the back [35, 1, 27]. However, touch input has more properties than its location [34]. One of these properties is pressure. Pressure input on touchscreens has been studied since the 1980s [5]. Even commercial products, such as Apple's iPhone 7, embed pressure input into the UI, e.g., by letting the user peek into content while applying pressure against the touchscreen. In both examples, the user applies pressure *transiently* to enter different "quasi-modes" that revert upon pressure release—hence, no need for displaying a back button. While transient pressure has been investigated for input at the front [17], it has not been explored for BoDI.

In this paper, we introduce a new interaction technique called *BackXPress* that utilizes transient pressure at the back of landscape-oriented smartphones to complement touch input at the front. Landscape orientation is convenient, e.g., for gaming, interaction with media, such as photos and videos, or two-thumb typing. Compared to BoDI in portrait orientation, landscape orientation provides more stability in holding the device—especially important when applying pressure—and has the advantage of having up to eight fingers available for pressure input at the BoD and two thumbs for touching at the front. An application example for BackXPress is insertion of emoji characters during text entry on a smartphone (Fig. 1): When BoD pressure is applied, the keyboard could switch to show emojis, with each combination of finger and pressure level showing a different category, e.g., smileys or animals. Tapping an emoji with the thumb would insert it, and subsequent pressure release could automatically flip back to the regular keyboard to continue typing immediately. With BackXPress, selecting a mode does not occlude the content, since the user's fingers interact at the back. In addition, there is no need for moving the finger to a back button since quasi modes are exited upon pressure release. Furthermore, the display of controls or menus to access the quasi modes could be omitted for users who are familiar with the finger-to-menu mappings.

This paper investigates this new interaction technique with two studies on device grip and finger placement for BoD pressure control, and an experiment on users' performance using BackXPress for two use cases: Applying BoD pressure during single touches at the front, and for up to 20 seconds while touching multiple consecutive frontal targets. Using a correction model, pressure accuracy lay above 92% for three separate pressure levels above normal resting pressure. We conclude with design guidelines for this new interaction technique.

RELATED WORK

BackXPress combines **(1)** BoDI on mobile touchscreens such as smartphones and tablets, and **(2)** pressure input on such devices. We review the related work for both fields below.

Back-of-Device Interaction (BoDI)

BoDI improves interaction with mobile devices in various ways. Initial BoDI addressed target occlusion by the finger while touching the screen at the front. E.g., LucidTouch [35] video-captured finger input at the BoD to display the finger silhouette *behind* the target on the frontal screen. NanoTouch [1] solved the occlusion problem for screens as small as 0.3".

BoDI also solves reachability issues, i.e., when not all areas at the front can be touched easily, by sensing finger positions at the BoD. For one-handed smartphone usage, Bergstrom et al. [2] predicted the area on the touchscreen that the thumb can reach by—among other parameters—locating the index finger at the back. Löchtefeld et al. [16] proposed to use the thumb to interact with the lower part of the touchscreen at the front, whereas the index is used to interact at the back as it can reach the upper part of the device easier. Yang et al. [40] used a similar separation of input but added BoD cursor control to avoid the fat finger problem [28]. Hakoda et al. [11] attached a small photo reflector to the back of a smartphone that, when tapped by the index finger, moved the content shown on the touchscreen towards the area reachable by the thumb. Mohd Noor et al. [20] improved touch input accuracy by predicting where a touch will land on the screen based on how fingers are grasping the BoD. BoDI also helps to reduce shoulder surfing for passwords entered on smartphones [9].

The examples above address one-handed BoDI in portrait orientation. In contrast, holding the device with two hands provides a more stable grip and has the benefit that up to eight fingers can be used for interacting with the BoD. BrailleTouch [30] enabled blind users to type braille characters with six fingers on a flipped smartphone, and Buschek et al. [4] and Schoenleben et al. [26] combined a sandwiched tablet with finger pose estimation for typing with eight fingers at the back and two thumbs at the front. BeyondTouch [41] detected up to four different tap locations at the back of a landscape-held smartphone, e.g., to watch and navigate videos. HaptiCase [7] added tactile landmarks to the back of a smartphone that users sense with their fingers for more accurate eyes-free touch input. Shen et al. [27] used BoD multi-touch on a sandwiched smartphone to control virtual 3D objects.

BackXPress also exploits the expressiveness of being able to use multiple fingers at the BoD in two-handed landscape orientation. Unlike the examples presented above, our technique allows each finger to express multiple states per finger by pressure, thus without moving them.

Pressure Input

Pressure input makes interaction with mobile devices more efficient. Brewster et al. [3] used binary pressure on a resistive touchscreen as modifier to type lowercase and uppercase letters without having to reach for a shift key. Forcetap [13] distinguished gentle taps from strong taps with 90% accuracy based on accelerometer data. The simulated pressure was used to quickly pop up context menus or enable magnification for more accurate acquisition of small touch targets. ForceDrag [14] used pressure to toggle a dragging mode on smartphones.

Pressure also adds richer expressiveness to gestures on touchscreens. Davidson et al. [8] and Qiu et al. [22] used the pressure property of a touch to push virtual objects along the z-dimension, and Force Gestures [12] combined normal and tangential force with existing touchscreen input to obtain a richer gesture set. Pressure also enables expressive input even when the fingers are in a static pose, e.g., when holding the device. Spelmezan et al. [31] attached a continuous pressure-sensitive button to the side of a smartphone to control interface widgets by the thumb. Two buttons at the side enabled bidirectional gesturing, such as scrolling and pinching, without finger movement, so that device grip was maintained [32]. Wilson et al. [36] investigated multi-digit pressure performance on a smartphone using multiple force sensing resistors (FSR) around the device. Index finger, ring finger, and the combined use of ring and little finger as well as index and middle finger handled pressure input most accurately, with errors rates as low as 3.2%. For tablets, one hand usually just holds the device. To allow bi-manual interaction, McLachlan et al. [17] added an FSR to the bezel that is covered by the thumb of the hand holding the tablet. Pressure selected different entries from a menu, while the index finger of the dominant hand touched the screen to interact in that mode. Maintaining pressure while tapping with the dominant hand reached 96% accuracy, but dropped by 10% when executing sliding gestures.

To our knowledge, only Steward et al. [33] have previously looked into BoD pressure input. They showed that users could control nine different pressure levels with the index finger by pushing against the BoD to move a cursor along a horizontal line. BackXPress is the first technique that combines BoD pressure input with frontal touch on mobile touchscreens to allow multi-digit, multi-level input in a static finger pose.

Transfer Function and Selection

Pressure based interaction techniques range in the mapping from sensor values to input values, called *transfer function*, as well as pressure space and number of targets within that space, selection of pressure, feedback modality, and context of use.

Transfer functions vary from linear [23], to quadratic [6], to sigmoid [25]. Steward et al. [33] compared these and stated that a linear mapping worked best. This way, users can control eight to ten pressure targets on a 3–10 N pressure range in a stationary context with visual feedback [6, 19, 33, 38, 17]. Walking, however, significantly increases error, selection time, and subjective workload of pressure input [37]. Target pressure is typically selected either upon crossing a threshold [3, 13], maintaining pressure for a particular dwell time (usually 1 second), or upon quick release of pressure [23, 38, 37]. Each of these techniques has certain drawbacks [17]: Thresholding is limited to two states, dwell time slows down the interaction and does not allow the user to linger on a state longer than the dwell time [17, 3], and quick release causes selection errors [37].

Transient Pressure

An alternative to these methods is to decouple pressure selection from its control, e.g., by spitting the two across both hands [17, 18]. Pressure can then be modeled as transient [17]: The user applies pressure to traverse different pressure targets. Upon release, targets are revisited in reversed order until the original state is returned. This combination of *natural inverse* and *bounce back* [10] "encourages the exploration of unfamiliar options and assures the user that errors can be undone" [17]. When the desired pressure value is found, the user can then explicitly perform its selection, e.g., with the other hand.

BackXPress also decouples pressure selection from its control by letting the user control transient pressure at the back and selecting the current pressure target by tapping at the front. Hence, BackXPress also does natural inverse and bounce back, thus enabling users to easily explore and undo pressure control.

INTERACTION TECHNIQUE: BackXPress

When the user is holding her smartphone with two hands in landscape orientation, up to eight fingers are in contact with the BoD, whereas both thumbs can interact with the touchscreen at the front (Fig. 2). Unlike existing BoDI, *BackXPress* does not use fine-grained location information of input at the back. Technically, BackXPress divides the BoD into different pressure-sensitive areas, one for each finger. This way, the user does not need to move the finger and can maintain a stable grip and still communicate rich input: Each finger can apply individual transient pressure, resulting in a large number of input states, depending on how many different pressure levels each finger can control. Each finger and pressure level combination enters a "quasi-mode" as long as the pressure is maintained. The user then interacts in that mode using both thumbs at the front. Upon pressure release, the mode is exited, and according to *natural inverse*, previously passed modes are quickly traversed in reversed order.

BackXPress targets at two different use cases: (1) **1-Tap** and (2) **N-Tap**. In 1-Tap, the user maintains pressure shortly to tap a single item on the touchscreen in the currently active mode, whereas in N-Tap, the user maintains pressure over a longer duration to tap multiple consecutive targets at the front. Below, we present three application examples for 1-Tap and N-Tap. All examples are also explained in our video figure.

Application Examples

Text Input

(1-Tap): During text entry, the user may want to insert an emoji. Each finger could control a different emoji category, e.g., pressing the left ring finger would display animals. Usually, not all emojis from one category fit on the screen altogether. Different pressure levels could be used to flip trough different pages. Tapping an emoji by thumb inserts it, and releasing pressure restores the regular keyboard layout to continue text input.
(N-Tap): The user might need to enter a few word in a language different from the rest of the text. To activate the keyboard that brings the right layout and auto-correction dictionary, the user exerts BoD pressure to flip through her installed keyboards that are shown on top of the standard layout. When the right keyboard appears, the user maintains pressure and types the word with both thumbs. When pressure is released, the standard layout flips back, and the user can continue typing immediately in her natural language.

Gaming

(1-Tap): In mobile point-and-"click" adventure games, the user frequently engages with an on-screen menu to pick an action, such as *speak* or *walk*, or select an object from the character's inventory. Once selected, the user then taps on the screen to interact with the game context in that mode. With BackXPress, the user could flip through the different actions using pressure from one finger, whereas another finger presses to select an object. E.g., to speak to a character in the game scene, the user would apply pressure until *speak* is selected and then tap on the character with the thumb to start the conversation.

Hotel Finding

(1-Tap): An application could show nearby hotels on a map of a location at interest. BackXPress could enable the user to explore hotels and experiment with star ratings and price ranges. Pressure on the left side at the BoD controls the hotel ratings between one and five stars. More pressure pushes the rating down. On the right hand side, the user could then press to control the price range. Pressing harder shrinks the price. Tapping on a "show results" button allows the user to release pressure and browse all matching hotels in a list.

As all examples show, with BackXPress the user does not need to move her fingers back and forth between on-screen menus and a back button. The user can maintain a stable grip and still access multiple quasi-modes directly.

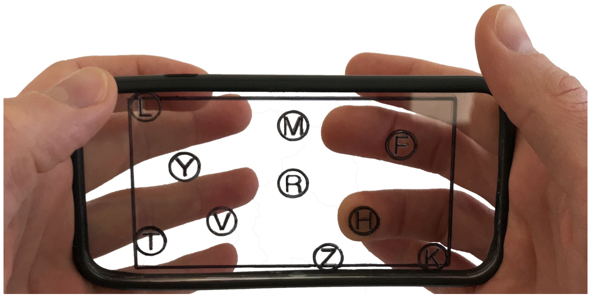

Figure 2. Exemplary hand and finger pose for exerting BoD pressure on the 4.7" prototype from Study 1. Unlike the little fingers, index, middle, and ring fingers are usually located behind the device. Both thumbs can reach the entire frontal screen represented by a black rectangle.

STUDY 1: BoD PRESSURE FINGER CANDIDATES

To find out with which fingers users can comfortably exert pressure at the BoD, we conducted a small user study. We asked 12 participants (aged 19–31, M = 22.92, SD = 3.34, two females, all right-handed) to press at ten different locations on the back of 4.7" and 5.5" smartphone mockups that were made from acrylic with rounded edges and an enclosing bumper (Fig. 2). Users held the mockups in portrait and in landscape orientation. Since the devices were transparent, users could see the targets from the front but also check that their finger position at the back actually landed on a target. Targets were read out by a computer voice, and once a target had been pressed, the instructor initiated the next target. Users were not limited in how to grasp the device and which fingers and hands to use to press the targets. Targets were randomized and repeated once after all targets had been pressed. In total, each user did 2 × 2 × 10 × 2 (DEVICE SIZE × ORIENTATION × TARGET × REPETITION) = 80 trials. Afterwards, users were asked to state which hands and fingers they considered feasible for BoD pressure input.

Results

In portrait orientation, almost all users used the right hand, whereas only half of the participants considered also the left hand suitable for BoD pressure input. For the 4.7" mockup, the right index finger was chosen by eleven subjects, but only seven users used the left index finger. Other fingers did not exceed a count of four, with the little finger just a count of one. These results were almost similar for the 5.5" device (Fig. 3). During the study, we observed that eight participants preferred to use just one hand to both grasp the portrait mockups and exert pressure. Using this grip, however, users had to frequently tilt and re-grasp the device to stabilize their grip for reaching lower targets. When two hands were used, fingers snagged. Therefore, we conclude that portrait orientation is rather impractical for BoD pressure.

For both landscape form factors, users applied the typical hand posture that people use when interacting with a smartphone in landscape orientation: Both hands hold the device, with the thumbs resting above the front, and the remaining fingers resting at the back to stabilize the device. When users exerted BoD pressure for a target, they tended to gently push the hands against the device side to obtain additional stabilization. For the 4.7" device, nobody opted for the little finger to exert BoD pressure; index and middle finger were most preferred (ten

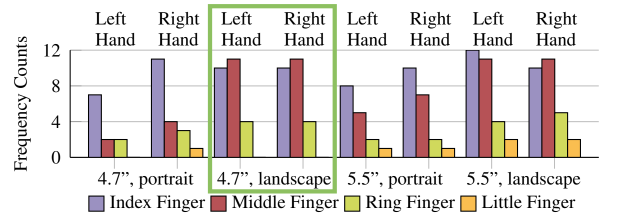

Figure 3. Frequency counts for the fingers used for exerting BoD pressure. Users preferred the small device held with two hands in landscape orientation (green box).

resp. eleven counts), followed by the ring finger (four counts). These results were the same for both hands and similar for the 5.5" form factor. When asked which device size users preferred for BoD pressure input, eight out of twelve voted for the 4.7" landscape form factor. Based on all these results, we decided to focus on a 4.7" landscape form factor for index, middle, and ring finger BoD pressure input for our next studies.

STUDY 2: BoD PRESSURE FINGER POSE

We now wanted to understand at what areas users place their index, middle, and ring finger at the BoD to exert pressure. We sandwiched two iPhones to the back of each other—a common technique to read input from both, back and front [33, 27, 26, 39]. At the front, we used an Apple iPhone 6, whereas at the back a pressure-sensitive iPhone 6S was used. Pressure sensitivity was set to "firm", giving a range of pressure values from 0 to $\frac{480}{72} \approx 6.67$ in steps of $\frac{1}{72}$ using a linear transfer function. Apple does not provide any information about how these values translate to Newtons, but experiments [21] hint at a maximum close to 4 N. Both devices share the same 138 × 67 mm^2 width and height with a screen size of 104 × 58 mm^2, resulting in a diagonal of 4.7" at a 16:9 aspect ratio. Users were asked to hold the device in landscape orientation using the typical posture observed in Study 1. The pressure-sensitive side was facing away from the user. The touchscreen at the front was divided into six equally-sized areas. Each area represented a target that was to be pressed from behind using one of the six fingers that were located roughly behind the target. E.g., the target at the upper left area was to be pressed with the left index finger. We tested three different pressure levels: low (1.50–2.50), medium (3.84–4.84), and high (6.17–6.67). We ignored pressure below the lowest level to disambiguate resting fingers from pressure input. According to Apple's SDK [15], a value around 1.0 represents a normal tap, whereas values beyond that level represent intentional pressure input. Although the range for the highest pressure level was half as small as for medium and low, it actually had infinite target width: users could always apply more pressure than the device could sense to achieve the highest level. By default, a target had a light gray color. Users had to exert pressure with the corresponding finger at the back until the target turned green. If users applied too much pressure, they left the requested pressure range and the target turned red; when pressure was too low, the target turned blue. As soon as the user stayed within the correct pressure level for 1s, the next trial was shown. Each participant did 6 × 3 × 3 (FINGER × PRESSURE LEVEL × REPETITION) = 54 trials. FINGER was randomized. Once all fingers were tested, a REPETITION was

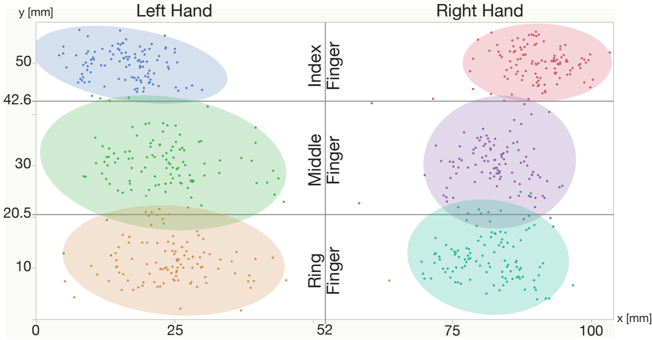

Figure 4. Distribution of BoD pressure touches from Study 2 split across hands and fingers. Ellipses denote 95% coverage. Gray bars are heuristic divides to identify the finger that performed a pressure touch.

done, and after three repetitions in total, the next PRESSURE LEVEL was chosen. To get acquainted with the system, users performed test trials beforehand.

Overall, 12 participants (aged 22–57, M = 41.83, SD = 13.09, five females, two left-handed) participated. Users' finger lengths were M = 58.33 mm (SD = 4.64) for the thumb, M = 75.42 mm (SD = 4.34) for the index finger, M = 81.75 mm (SD = 4.97) for the middle finger, and M = 76.00 mm (SD = 5.59) for the ring finger. Eight users regularly used a smartphone with a screen diagonal of about 5", and four subjects used a smartphone with a screen sized between 3.5" and 4".

Results

Figure 4 shows the distribution of pressure touches at the back. As can be seen, touches from both hands almost never spread across the center of the x-axis, thus both sides can be distinguished in software by cutting that axis in half. One participant (aged 50, right-handed, female) tended to exert pressure by the left hand closer to the center of the x-axis than all other users, such that the touch ellipses for the left hand are wider than for the right hand. Regarding the spread in y-direction, pressure touches of adjacent fingers of each hand showed some overlap. Touches from the index finger had the smallest y-spread, whereas middle finger touches spread equally up and down from the center of the y-axis, and touches performed by the ring finger did not usually occur below 5 mm on that axis. We calculated the effective height [29] for each finger area by multiplying the standard deviation of the pressure touches y-direction with a factor of 4.133, such that 96% of the touches would be contained within a finger area. This height was similar for the same finger types of each hand. Still, these heights had a slight overlap between adjacent fingers, such that we took the center of these overlays as separators between fingers (Fig. 4, gray lines). These separators serve as a heuristic to identify from which finger pressure is exerted at the back, which we used in our next study.

STUDY 3: PRESSURE DYNAMICS & TAPPING ACCURACY

Our next goal was to understand the usability of BackXPress for our two use cases 1-Tap and N-Tap: 1-Tap addressed the question of *"How accurately can users apply BoD pressure for a single tap at the front of a smartphone while holding it with two hands in landscape orientation?"*. N-Tap addressed the question of *"How accurately can users maintain BoD pressure over time while tapping several targets in sequence?"*. Both tasks are inspired by McLachlan et al. [17], who called them *Targeting* and *Maintaining*. We applied some modifications to these tasks to match the particular features of BackXPress. Upfront, we expected that BoD pressure would increase upon tapping, since this pushes the device against the fingers at the back, that are in a static position and hence repel that additional force. We also expected BoD pressure to affect frontal tapping negatively. For N-Tap, we expected the number of frontal taps to be smaller than for tapping without BoD applying pressure.

Experimental Design

Apparatus

We reused the devices from Study 2. This time, however, we improved on the perceived device thickness that our users from Study 2 found unnaturally thick. We added a 3D-printed case with stand-away sides that were just 7 mm thick (Fig. 5).

Participants

18 participants (aged 19–35, M = 25.06, SD = 4.58, seven females, three left-handed) participated in this study. Users' finger lengths were similar to those from Study 2. Thumb: 57.61 mm (SD = 5.53), index finger: 75.72 mm (SD = 5.34), middle finger: 81.06 mm (SD = 5.48), ring finger: 76.78 mm (SD = 5.45). All subjects regularly used a smartphone with a screen size of about 5". Half of them started with *1-Tap*.

Task 1: 1-Tap

In 1-Tap, users were asked to apply a certain pressure at the BoD and then tap a touch target on the frontal touchscreen. When pressure was applied, visual feedback was given: A cursor highlighted the current pressure level on a menu visualized by a vertical color bar (Fig. 5). Each color represented a certain pressure range. Applying more pressure moved the cursor up, less moved it down. The color bar was displayed at the left and right on the frontal touchscreen such that it was visible even when it was hidden by the thumb tapping.

A white vertical line next to the pressure menu indicated which of the six fingers to use for pressure navigation: The back was split into 3×2 regions, one for each finger, based on the results from Study 2. If, e.g., the finger indicator appeared on the left hand side next to the the upper third of the menu, then the user had to use her index finger of the left hand. Touch targets were displayed on the frontal screen one at a time as a crosshair. The crosshair color matched the requested pressure range from the colored menu item. A circle around the crosshair visualized the currently applied BoD pressure by color: A color match between crosshair and circle signaled the participant that the desired pressure range was reached. Tapping the touch target completed the trial: As soon as finger contact with the frontal touchscreen was detected, pressure sensing was locked, such that the color bar cursor would not move anymore. This *pressure lock* was used to facilitate pressure maintenance upon tapping and to avoid that pressure changed during selection. Upon finger release, pressure lock was disabled again. Users were asked to touch the crosshair as accurately as possible.

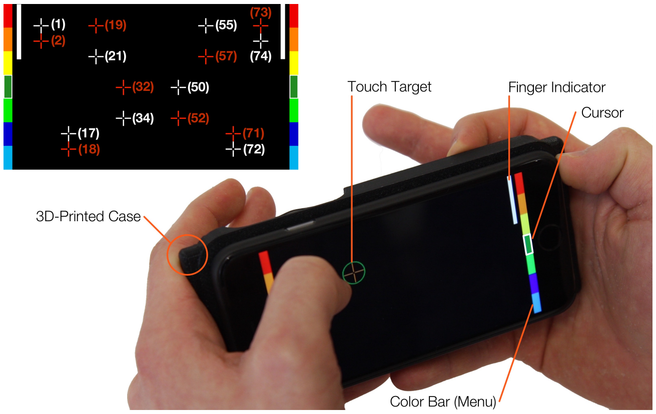

Figure 5. Top: Distribution of TOUCH TARGETs for Study 3. Targets were arranged on an invisible 9×9 grid with a 12.5% offset from the corners of a 1334×750 px area. White crosses show targets for 1-Tap. Red crosses are mirrored along the y-axis and were used for N-Tap. Bottom: Prototype with UI with 7 menu items for Study 3. The user has to exert BoD pressure with the right index finger, which indicated by the white vertical bar. Currently, the 4th item (dark green) is selected, which is indicated by both a white rectangular cursor around the item and a color-matching circle around the touch target. The touch target is shown as a cross matching the target pressure (here: orange).

Variables

1-Tap had four **independent variables**: MENU SIZE, PRESSURE LEVEL, FINGER, and TOUCH TARGET. FINGER referred to the finger to be used to exert pressure at the back and had six levels: index, middle, and ring finger of each hand. MENU SIZE split the pressure range between 1 and 6 equidistantly into 3, 5, or 7 discrete items. We did not exploit the full pressure range up to ≈6.67, to avoid infinite target width, that would make controlling pressure easier for the last menu item by simply pressing as hard as possible. For practical application, however, an infinite target width should be considered. We also added a baseline condition to contrast the user's pure touch performance with the influence of BoD pressure. Hence, this condition only involved tapping at the front and ignored pressure input completely. PRESSURE LEVEL had three values: 1.83, 3.50, and 5.17—these were the centers of the items for MENU SIZE 3. Users did not have to reach these exact values, but just stay within the corresponding menu item. For MENU SIZE 3, all three items were tested, for MENU SIZE 5 items 1, 3, and 5 were tested, and for MENU SIZE 7, items 2, 4, and 6 were tested. This way, not all items for a MENU SIZE were tested, but it ensured that an equal number of measurements per MENU SIZE was obtained—an approach that has also been applied by [24, 37, 38, 17]. TOUCH TARGET had eight levels. These were the targets that users had to tap at the front using their thumbs. The targets were positioned on an invisible 9×9 grid (Fig. 5, top). These locations were chosen such that users had to touch at different regions in proportion to the finger exerting BoD pressure, e.g., the thumb being close to that finger or far away, since such constellations could affect pressure maintenance differently.

MENU SIZE was counterbalanced using a Latin Square. FINGER, PRESSURE LEVEL, and TOUCH TARGET were counterbalanced altogether to ensure that the user had to release pressure after each trial, which is typical for 1-Tap interactions.

After all trails for a MENU SIZE were tested, the participant was given a one minute break to relax her arms and fingers. A left TOUCH TARGET was always succeeded by a right one and vice versa to balance the usage of the thumbs. Half of the participants started at the left. Users were not forced to use a particular thumb for a TOUCH TARGET, but they usually tapped targets on each side with the corresponding thumb.

The **dependent variables** were the *Time* (ms) between a target appeared and the user releasing her finger from it; *Success* (yes/no) whether the touch target was selected with the correct item from the pressure menu; *Touch Error* (mm) as Euclidean distance between the center of a given touch target and the user's touch at the front; and *Pressure Range* (relative, on a normalized scale from 0–1) between the minimum and maximum BoD pressure values sensed between the user touching and releasing the frontal touchscreen. *Success* and *Pressure Range* were not measured for the baseline condition, since it did not involve any pressure measurement. Likewise, FINGER and PRESSURE LEVEL were inexistent in the baseline condition, since these variables referred to BoD pressure execution.

1-Tap had 3×6×3×8 (MENU SIZE × FINGER × PRESSURE LEVEL × TOUCH TARGET) = 432 pressure trials. For comparison, users performed 3×8 = 24 baseline trials, hence 1-Tap had 456 trials per participant in total. A trial ended as soon as the user lifted the thumb off the touchscreen, independent from whether pressure was within the correct range for PRESSURE LEVEL or how accurately the TOUCH TARGET was hit. Before data was recorded, users had 24 test trials.

Results – 1-Tap

We first present the results for data compatible with the baseline condition, followed by data from BoD pressure conditions.

Baseline Data

Time. We conducted a two-way repeated measures ANOVA on the log-transformed *Time* data. There was a significant main effect for MENU SIZE ($F_{3,8156} = 245.05$, $p < .0001$). Tukey-HSD post hoc pairwise comparisons revealed significant differences across all MENU SIZEs ($p < .01$, each). *Time* increased with the number of menu items (Fig. 6, left). There was no main effect for TOUCH TARGET and there were no interaction effects.

Touch Error. A two-way repeated measures ANOVA on *Touch Error* showed a significant main effect for TOUCH TARGET ($F_{7,8156} = 70.45$, $p < .0001$). Tukey-HSD post hoc pairwise comparisons revealed that *Touch Error* for targets 1, 17, 21, and 74 was significantly lower compared to all other targets ($p < .05$, each) (Table 1, left). However, the difference of up to 1.10 mm between these two target groups is relatively small and practically negligible. There was no main effect for MENU SIZE and no interaction effect, indicating that BoD pressure had no influence on frontal tapping accuracy. On average, *Touch Error* was 2.47 mm (95% CI: [2.44; 2.50] mm).

BoD Pressure Data

Time. We ran a four-way repeated measures ANOVA on the log-transformed *Time* data. Apart from the significant main ef-

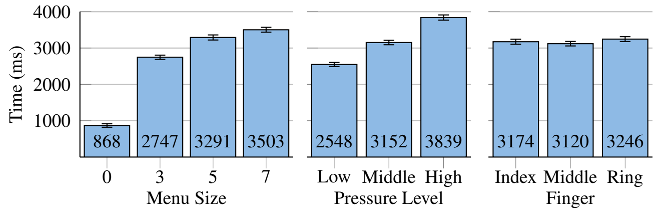

Figure 6. ***Time*** **data for 1-Tap split by MENU SIZE, PRESSURE LEVEL, and FINGER. MENU SIZE 0 was the baseline condition. Error bars denote 95% CI.**

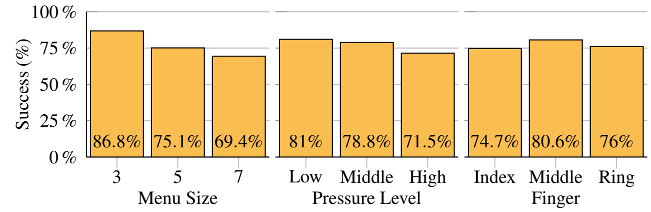

Figure 7. ***Success*** **for 1-Tap split by MENU SIZE, PRESSURE LEVEL, and FINGER.**

fect for MENU SIZE (cf. baseline analysis), there was a significant main effect for PRESSURE LEVEL ($F_{2,7540} = 160.77$, $p < .0001$). Tukey-HSD post hoc pairwise comparisons showed significant differences across all levels ($p < .0001$, each), increasing from lowest to highest (Fig. 6, middle). This was to be expected since users navigated pressure upwards starting from zero, hence higher pressure levels were further away and required more time. There was also a significant main effect for FINGER $F_{2,7540} = 6.57$, $p < .01$). Tukey-HSD post hoc pairwise comparisons revealed that exerting BoD pressure with the ringer fingers took significantly more time compared to index and middle fingers ($p < .05$, both) (Fig. 6, right). There was also a MENU SIZE × FINGER interaction effect ($F_{4,7540} = 2.64$, $p < .05$), that confirmed that users controlled BoD pressure significantly faster for a combination of fewer menu items with lower pressure values compared to more menu items paired with higher pressure values.

Touch Error. We ran a four-way repeated measures ANOVA on *Touch Error*. Apart from the significant main effect for TOUCH TARGET (cf. baseline analysis), there were no further main effects or interaction effects.

Success. We ran a Cochran's Q tests on the dichotomous *Success* data. There was a significant main effect for MENU SIZE ($Q(2) = 232.66$, $p < .0001$), PRESSURE LEVEL ($Q(2) = 75.89$, $p < .0001$), FINGER ($Q(2) = 29.38$, $p < .0001$), and TOUCH TARGET ($Q(7) = 30.44$, $p < .0001$). Post hoc pairwise comparisons for MENU SIZE were all significant ($p < .0001$, each), with MENU SIZE 3 leading to the highest *Success* (86.8%, Fig. 7, left). Post hoc pairwise comparisons for PRESSURE LEVEL showed that the highest level led to significantly lower *Success* compared to the medium and lowest level ($p < .0001$, both) (Fig. 7, middle). Post hoc pairwise comparisons for FINGER revealed that BoD pressure exertion by middle finger led to significantly higher *Success* compared to index and ring finger ($p < .0001$, both) (Fig. 7, right). Post hoc pairwise comparisons for TOUCH TARGET showed that outer targets 1, 72, and 74 led to significantly higher *Success* (79.9–82.3%) compared to center target 34 (73.7%) ($p < .0001$, each). Interestingly, *Success* for center target 50 (76.9%), was not significantly different from outer targets.

Pressure Range. A four-way repeated measures ANOVA on the normalized *Pressure Range* data showed a significant main effect for FINGER ($F_{2,7540} = 7.48$, $p < .001$) and TOUCH TARGET ($F_{7,7540} = 3.99$, $p < .001$). Tukey-HSD post hoc pairwise comparisons between different FINGERs showed that *Pressure Range* for the index finger was significantly higher compared to middle and ring finger ($p < .05$, both) (Table 1, left). However, the difference of ≈1.00% is practically negligible. For TOUCH TARGET, Tukey-HSD post hoc tests revealed that *Pressure Range* for targets 21 and 72 was significantly higher compared to targets 74 and 1 ($p < .05$, each), but, again, at a difference of ≈1.00% practically negligible. On average, *Pressure Range* was 6.44% (95% CI: [6.20; 6.68]%).

Discussion – 1-Tap

Not surprisingly, *Success* decreased with MENU SIZE. BoD-pressure control with selection at the front worked best for three menu items (86.6%). When only the middle finger was used, a success rate of 89.5% was achieved. We believe that the middle fingers performed best, because they are located at the horizontal center of the device, where applying pressure hardly tilts the device towards the user. Furthermore, users were more successful when controlling low pressure. Overall, *Success* was fairly acceptable, but bi-manual touch input and pressure control at the front of a tablet reached the same *Success* for 10 menu items [17]. We are confident, however, that potential learning effects and using an infinite target width for the last item of each MENU SIZE will lead to a higher *Success* since then the user could just press as hard as possible to reliably reach the last menu item.

Time increased with MENU SIZE even though the same PRESSURE LEVELS had to be acquired. Some users stated that pressure control for MENU SIZES 5 and 7 "[...] was a bit annoying" since the cursor was jumping a lot. Consequently, participants needed more time to carefully control pressure. Reaching for highest PRESSURE LEVEL was also considered negatively: "It takes longer to reach high pressure.". Not surprisingly, BackXPress was slower than plain touching at the front (baseline condition), since the timings for BackXPress included navigation time for the menu. Still, BackXPress was slower than [17], where users controlled pressure for seven to ten items at the front of a tablet within the same time. Hence, we conclude that pressure navigation at the BoD is more demanding than at the front.

Upon tapping, BoD pressure increased, but it dropped back to almost the same value as soon as the thumb was lifted off the front. Overall, this change in pressure was small (6.29% for MENU SIZE 3, cf. Table 1), which is about half the force applied during a normal tap on a touchscreen. Still, it makes sense to apply the pressure lock technique, because when pressure is applied near a pressure range boundary, the cursor might still jump to the adjacent bar, confusing the user. *Touch Error* was not affected by BoD pressure, which is good and in-line with previous findings from [17].

1-Tap

		Menu Size				Pressure Level			Finger		
		Baseline	3	5	7	Low	Medium	High	Index	Middle	Ring
Touch Error [mm]	M	2.19	2.42	2.48	2.55	2.52	2.47	2.47	2.50	2.49	2.46
	SD	1.15	1.24	1.23	1.31	1.27	1.27	1.24	1.31	1.24	1.24
Pressure Range [%]	M	N/A	6.29	5.76	7.27	5.77	6.46	7.09	7.22	6.29	5.82
	SD	N/A	9.94	9.13	12.86	12.36	10.65	9.04	11.30	10.85	10.11

N-Tap

		Menu Size				Pressure Level			Finger			Time Block			
		Baseline	3	5	7	Low	Medium	High	Index	Middle	Ring	0–5 s	5–10 s	10–15 s	15–20 s
Touch Error [mm]	M	2.26	2.20	2.24	2.28	2.23	2.26	2.22	2.26	2.22	2.23	2.18	2.27	2.28	2.19
	SD	1.99	1.19	1.14	1.19	1.18	1.12	1.22	1.19	1.21	1.12	1.11	1.26	1.17	1.10
Pressure Range [%]	M	N/A	2.04	1.76	2.36	1.50	2.16	2.70	2.24	1.86	2.04	2.60	1.89	1.85	2.10
	SD	N/A	4.15	3.60	5.20	4.03	4.59	4.29	4.14	4.22	4.58	5.25	4.07	3.92	4.39

Table 1. ***Touch Error*** **and** ***Pressure Range*** **data for 1-Tap (left) and N-Tap (right) split by MENU SIZE, PRESSURE LEVEL, FINGER, and TIME BLOCK.**

Overall, for interactions like 1-Tap, we recommend a MENU SIZE of three items, and using the pressure lock technique. If only up to six different modes are needed, let users use their middle fingers to enter those modes.

Task 2: N-Tap

N-Tap was similar to 1-Tap, but users had to maintain PRESSURE LEVEL for 20 seconds while tapping TOUCH TARGETS. Users were instructed to tap as many TOUCH TARGETs as possible, but not at the cost of maintaining the correct PRESSURE LEVEL or tapping accuracy. As in 1-Tap, pressure lock was only active per tap and not the entire 20 seconds, since, in reality, the device would not know whether the user intends to tap only once or a sequence of TOUCH TARGETs for a fixed PRESSURE LEVEL. We used the same UI as for 1-Tap to visualize the task. Each trial ended after 20 seconds.

Variables

The **independent variables** MENU SIZE, PRESSURE LEVEL, and FINGER were the same as for 1-Tap. Again, we added a baseline condition that involved just tapping at the front for 20 seconds. In addition, TIME BLOCK assigned users' taps on touch targets to one of the four equally-sized time ranges: 0–5, 5–10, 10–15, and 15–20 seconds, based on the time when the device registered first contact with the touchscreen. This was done to analyze how users' performance developed over time. In contrast to 1-Tap, TOUCH TARGETS were mirrored along the vertical axis (Fig. 5, top) to avoid learning effects. MENU SIZE was counterbalanced and FINGER and PRESSURE LEVEL were counterbalanced together using a Latin Square. The participant was given a one minute break to relax her arms and fingers until the next MENU SIZE was chosen. TOUCH TARGET was completely randomized, but left and right targets always alternated. Half of the participants started at the left.

Dependent variables were *Touch Error* and *Pressure Range* as defined for 1-Tap. In addition, *Tap Count* measured how many touch targets users tapped within the 20 seconds. We did not investigate *Time*, as each trail lasted exactly 20 seconds. Again, *Success* and *Pressure Range* were not measured for the baseline condition, and FINGER and PRESSURE LEVEL did not exist for this condition.

In total, N-Tap had $3 \times 6 \times 3$ (MENU SIZE $\times$ FINGER $\times$ PRESSURE LEVEL) = 54 pressure trials. For comparison, users performed three baseline trials, resulting in 57 trials per participant in total, each lasting 20 seconds. Users had three test trials to get familiar with the task before data recording started.

Results – N-Tap

Again, we first present the results for baseline condition compatible data, followed by data from BoD pressure conditions.

Baseline

Touch Error. A three-way repeated measures ANOVA revealed a significant main effect for TOUCH TARGET ($F_{7,11602} = 14.00$, $p < .0001$). Post hoc Tukey-HSD pairwise comparisons showed that upper targets 2, 19 and 73 led to a significant lower *Touch Error* compared to center targets 52, 32, and 57 ($p < .01$, each). However, the difference between these two target groups of up to 1.00 mm is relatively small and practically negligible. There were no significant main effects for MENU SIZE and TIME BLOCK and there were no interaction effects. On average, *Touch Error* was 2.24 mm (95% CI: [2.22; 2.26] mm) (Table 1, right).

Tap Count. A Friedman test showed a significant main effect for MENU SIZE ($\chi^2(3) = 157.92$, $p < .0001$). *Tap Count* decreased with increasing MENU SIZE (Fig. 8, left). Post hoc pairwise comparisons were all significant ($p < .0001$, each) except between MENU SIZEs 5 and 7. There was also a significant main effect for TIME BLOCK ($\chi^2(3) = 1376.47$, $p < .0001$). Post hoc pairwise comparisons were all significant except between TIME BLOCKs 5–10s and 10–15s. *Tap Count* was lowest within the first five seconds, then increased and stabilized from five to 15 seconds, but slightly decreased afterwards, possible due to fatigue. There was also a MENU SIZE $\times$ TIME BLOCK interaction effect ($\chi^2(9) = 2.76$, $p < .005$). Any TIME BLOCK for MENU SIZE 0 was faster than any TIME BLOCK from MENU SIZEs 3, 5, and 7. Furthermore, for these MENU SIZEs, *Tap Count* was similar and lowest within the first five seconds (Fig. 8, right).

BoD Pressure Data

Touch Error. We ran a five-way repeated measures ANOVA on *Touch Error*. Apart from the significant main effect for TOUCH TARGET (cf. baseline analysis), there were no further main effects or interaction effects.

Tap Count. In addition to the significant main effects for MENU SIZE and TIME BLOCK (cf. baseline analysis), a Friedman test showed a significant main effect for PRESSURE LEVEL ($\chi^2(2) = 618.06$, $p < .0001$). *Tap Count* decreased with increasing PRESSURE LEVEL (Fig. 8, middle). Post hoc pairwise comparisons were all significant ($p < .0001$, each).

Success. We ran a Cochran's Q test on the dichotomous *Success* data. There was a significant main effect for MENU SIZE ($Q(2) = 181.81$), PRESSURE LEVEL ($Q(2) = 117.03$), FINGER ($Q(2) = 17.33$), and TIME BLOCK ($Q(3) = 64.47$), $p < .0001$, each. *Success* decreased with increasing MENU SIZE (Fig. 9, left). Post hoc pairwise comparisons for MENU SIZE were significant between MENU SIZEs 3 and 5 and between 3 and 7 ($p < .0001$, each), as well as between 5 and 7 ($p < .05$). Pairwise comparisons for PRESSURE LEVEL were all significant ($p < .05$, each). *Success* decreased with increasing

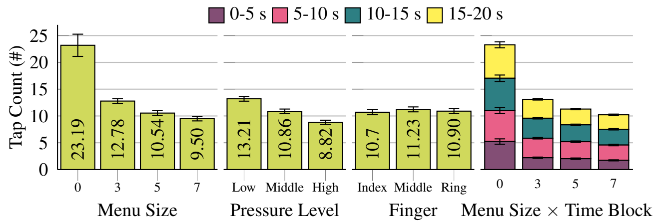

Figure 8. ***Tap Count*** **for N-Tap split by MENU SIZE, PRESSURE LEVEL, FINGER, and MENU SIZE × TIME BLOCK. Error bars denote 95% CI.**

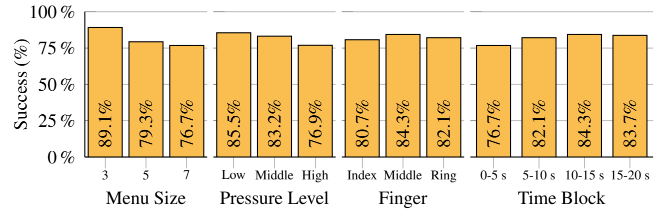

Figure 9. ***Success*** **for N-Tap split by MENU SIZE, PRESSURE LEVEL, FINGER, and TIME BLOCK.**

PRESSURE LEVEL (Fig. 9, middle) indicating that higher BoD pressure was more difficult to maintain. Pairwise comparisons for FINGER showed that BoD pressure exerted by the middle finger led to significantly higher *Success* compared to ring and index finger ($p < .05$, each) (Fig. 9, middle). Regarding TIME BLOCK, *Success* was significantly lower within the first five seconds compared to the remaining time slots ($p < .0001$, each), indicating that users needed some time to get familiar with pressure navigation (Fig. 9, right). There was also a significant main effect for TOUCH TARGET ($Q(7) = 45.51$, $p < .0001$), but without a clear pattern.

Pressure Range. A five-way repeated measures ANOVA on the normalized *Pressure Range* data showed a significant main effect for PRESSURE LEVEL ($F_{2,10305} = 27.72$, $p < .0001$) and TIME BLOCK ($F_{3,10380} = 21.04$, $p < .0001$). Tukey-HSD pairwise comparisons revealed that *Pressure Range* for the highest PRESSURE LEVEL was significantly higher compared to the medium and low levels ($p < .0001$, both) (Table 1, right). However, the difference was only 1–2% and therefore practically negligible. Tukey-HSD pairwise comparisons between different TIME BLOCKs were significant between 0–5 seconds and all other blocks ($p < .005$, each), but the difference was below 1% (Fig. 1, right). There was also an interaction effect for MENU SIZE × FINGER ($F_{4,10305} = 2.53$, $p < .05$), but post hoc pairwise comparisons were not significant. On average, *Pressure Range* was 2.04% (95% CI: [1.96; 2.13]%).

Discussion – N-Tap

Like for 1-Tap, *Success* dropped with MENU SIZE, and reached up to 89.1% for MENU SIZE 3 (Fig.9, left). *Success* was about 5% lower within the first five seconds compared to the remaining 15 seconds, suggesting that users were getting familiar with the task and that with regular use, results would be the same across the entire interaction time of 20 seconds. As for 1-Tap, BoD pressure had no impact on *Touch Error*. Maintaining BoD pressure, however, slowed down the interaction significantly, far more than we expected. Ideally, the user would acquire the desired pressure level once and then consecutively touch at the front, which users can do quickly, as the baseline condition showed. This, however, would not lead to a drop of about 50% in frontal taps for MENU SIZE 3 compared to the baseline condition. This could mean that users needed to re-adjust the cursor after tapping—especially for the highest PRESSURE LEVEL—or they were more careful to not change the pressure, which would slow down the interaction for both cases.

Basically, we could not compare our N-Tap results with those from [17], since our measurements and PRESSURE LEVELS were different. Only our medium PRESSURE LEVEL compares to the 2 N level from [17]. Here, users deviated by 0.15 N (*SD* = 0.12 N), whereas with BackXPress, users deviated by 0.22 N (*SD* = 0.20 N) on average, hence, more. This indicates, again, that controlling and maintaining pressure at the BoD is more difficult than at the front.

Overall, performance for N-Tap was similar to the results from 1-Tap. Therefore, we give the same recommendations for both tasks, i.e., using a MENU SIZE of three items, with preferring the use of the middle fingers. However, in both tasks, we intentionally did not use an infinite target width for the last item of each MENU SIZE. In practice, using an infinitive target width, however, will likely increase *Success* for both 1-Tap and N-Tap and also *Tap Count*, since the user could then press as hard as possible to quickly reach the last menu item.

IMPROVING DYNAMIC PRESSURE PERFORMANCE

For both 1-Tap and N-Tap, pressure control and selection follow a clear sequential pattern: The user first applies BoD pressure to enter the desired mode, then selects a target on the touchscreen. We were interested in seeing how pressure changed between these two events. Therefore, we looked into our data streams that continuously sampled BoD pressure. To compare pressure curves across different taps, we set the time at which the thumb started hitting the touchscreen, denoted as *Contact Event*, as zero. Figure 10 shows an example. As can be seen, pressure stabilized up to 750 ms before a Contact Event, which we interpreted as that the user had controlled to the desired menu item and was about to touch the screen. Note that the pressure values were towards the lower end on the correct pressure range. This is likely due to the fact that users navigated pressure from below and were about to tap as soon as the UI indicated that the correct item was reached. For each Contact Event, we calculated the mean pressure from *t* ms before the event until 0 ms, the time of contact. Although

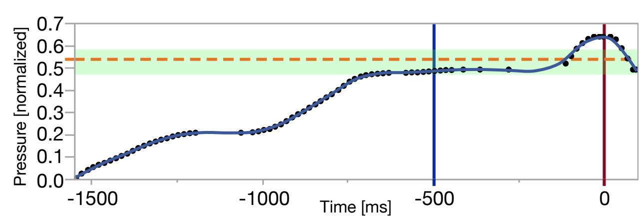

Figure 10. Pressure curve for a single trial for ***1-Tap*** **from one user using the ring FINGER. The green area shows the pressure range for the medium PRESSURE LEVEL for MENU SIZE 7. At the time the user tapped at the front (red line), she exceeded that range. Averaging pressure over 500 ms (blue line) before the tap, led to a better estimate (orange line) that lay within the pressure range.**

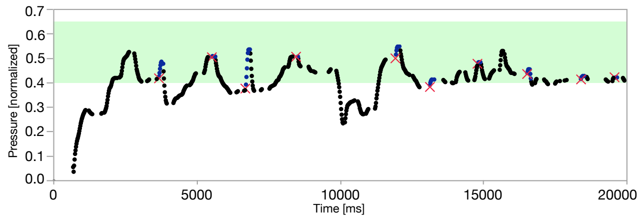

Figure 11. Typical 20s pressure curve for N-Tap from one user using the middle FINGER. The green area shows the pressure range for the medium PRESSURE LEVEL for MENU SIZE 3. Each red cross indicated when the thumb hit the screen. Blue dots indicate that the finger was still in contact with the touchscreen. The user's pressure fluctuated over time, but mostly stayed withing the pressure range.

in Figure 10 the pressure exceeded the pressure range about 100 ms before the tap, averaging until 0 ms lead to better results because it helped pushing the average pressure a little bit up, compared to the values before 100 ms, which were, as stated above, at the lower end of the pressure range. Figure 11 shows a typical pressure curve for N-Tap. Using this averaged pressure upon touch target selection, we could increase *Success* across all MENU SIZEs, with a value of $t = -500$ ms for 1-Tap and a value of $t = -700$ ms for N-Tap leading to the best results. Pairwise McNemar tests between the same MENU SIZEs for 1-Tap and N-Tap confirmed that averaged pressure lead to significantly higher *Success* compared to the original values from our study ($p < .0001$, each). This way, *Success* reached 92.4% for N-Tap with MENU SIZE 3 and increased by about 10% for MENU SIZEs 5 and 7 (Fig. 12). Using a parameter of $t = -500$ ms for N-Tap showed no significant difference for *Success* compared to when choosing $t = -700$ ms. Hence, a combined value of $t = -500$ ms will lead to a significant increase in success for both 1-Tap and N-Tap.

DESIGN GUIDELINES

Based on our results, we give the following design guidelines:

1. Use a landscape-oriented device that allows both thumbs to reach the entire frontal touchscreen. Expect up to three fingers (index, middle, and ring finger) from both hands for BoD pressure control (Study 1). Use the pressure sensing area for each finger according to Figure 4 (Study 2).
2. Use *Pressure Lock*: When users touch at the front, BoD pressure involuntarily increases. Since the user has controlled pressure *before* selecting a touch target, stop pressure sensing while the finger is in contact with the screen. Release *Pressure Lock* as soon as the thumb is lifted (Study 3).
3. Use a 500 ms pressure history to increase success: Each time when the user's thumb hits the touchscreen to perform a selection, use an averaged pressure value over the last 500 ms instead of a single pressure value (Study 3).
4. Use three, or if absolutely necessary, five pressure levels above normal resting pressure: We obtained a 92% success for three menu items, and 87% for five items (Study 3).
5. Prefer middle fingers: If only a few modes are needed, let users control pressure with the middle fingers since they do this more accurately compared to the index or ring fingers. Do not consider using the little fingers (Study 3).

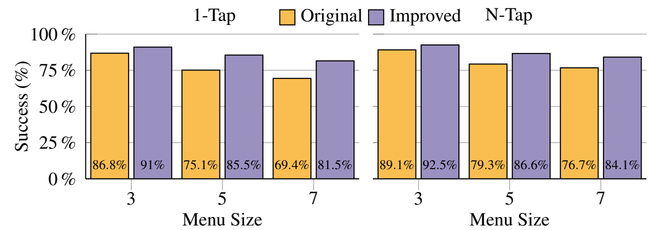

Figure 12. Comparison of original vs. improved *Success* by MENU SIZE for 1-Tap (left) and N-Tap (right).

LIMITATIONS AND FUTURE WORK

We used a 4.7" sandwiched prototype to measure both touch input at the front and pressure input at the back. While this is a common approach applied in HCI research, both device thickness (19 mm) and weight (308 g) are higher than we would expect from a future commercial device. We reduced perceived device thickness using a 3D-printed case (Fig. 5), but this added about 7 mm in width to each device side. Furthermore, the pressure sensor of our prototype was limited to a range of 0–4 N. However, most studies in HCI on pressure input used sensors with a similar range. Finally, we only investigated frontal tapping but no thumb movement, e.g., for dragging, while maintaining BoD pressure. Using *Pressure Lock*, however, we expect no difference to 1-Tap: As soon as the thumb touches the screen, pressure changes are ignored. Only upon release, the current pressure likely mismatches the locked pressure. We will investigate this in future work. In addition, we would like to investigate simultaneous multi-finger BoD pressure input to combine two or more "quasi-modes", as in our hotel finding application, and we would like to compare BackXPress to other existing quasi-mode techniques.

CONCLUSION

We presented *BackXPress*, a new interaction technique that lets users create back-of-device (BoD) pressure input to augment their two-handed interaction on touchscreens of landscape-oriented smartphones. In Study 1 and 2 we learned that index, middle, and ring finger of both hands can reliably apply pressure at the BoD. With BackXPress, users can apply various pressure levels with each of these fingers to enter different "quasi-modes" that are only active as long as that pressure is applied. The thumbs of both hands then interact with the frontal touchscreen in that mode. We provided application scenarios, such as multi-language text entry, that benefit from BackXPress. In Study 3 we tested the practicability of applying BoD pressure during single touches at the front and frontal touch input for 20 seconds. Using a 500 ms history of averaged sampled pressure values *before* the user touches the screen led to a significant improvement in users' pressure selection. With three pressure levels above normal resting pressure, pressure accuracy was more than 92%. BoD pressure did not affect tapping accuracy at the front. We concluded with design guidelines for our interaction technique.

ACKNOWLEDGMENTS

This work was funded by the German B-IT Foundation. We thank Florian Busch, Ravi Kanth, and Oliver Nowak for editing the video figure.

REFERENCES

1. Patrick Baudisch and Gerry Chu. 2009. Back-of-Device Interaction Allows Creating Very Small Touch Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09)*. ACM, New York, NY, USA, 1923–1932. DOI: http://dx.doi.org/10.1145/1518701.1518995
2. Joanna Bergstrom-Lehtovirta and Antti Oulasvirta. 2014. Modeling the Functional Area of the Thumb on Mobile Touchscreen Surfaces. In *Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 1991–2000. DOI: http://dx.doi.org/10.1145/2556288.2557354
3. Stephen A. Brewster and Michael Hughes. 2009. Pressure-Based Text Entry for Mobile Devices. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '09)*. ACM, New York, NY, USA, 9:1–9:4. DOI: http://dx.doi.org/10.1145/1613858.1613870
4. Daniel Buschek, Oliver Schoenleben, and Antti Oulasvirta. 2014. Improving Accuracy in Back-of-Device Multitouch Typing: A Clustering-based Approach to Keyboard Updating. In *Proceedings of the 19th International Conference on Intelligent User Interfaces (IUI '14)*. ACM, New York, NY, USA, 57–66. DOI: http://dx.doi.org/10.1145/2557500.2557501
5. William Buxton, Ralph Hill, and Peter Rowley. 1985. Issues and Techniques in Touch-Sensitive Tablet Input. In *Proceedings of the 12th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '85)*. ACM, New York, NY, USA, 215–224. DOI: http://dx.doi.org/10.1145/325334.325239
6. Jared Cechanowicz, Pourang Irani, and Sriram Subramanian. 2007. Augmenting the Mouse with Pressure Sensitive Input. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '07)*. ACM, New York, NY, USA, 1385–1394. DOI: http://dx.doi.org/10.1145/1240624.1240835
7. Christian Corsten, Christian Cherek, Thorsten Karrer, and Jan Borchers. 2015. HaptiCase: Back-of-Device Tactile Landmarks for Eyes-Free Absolute Indirect Touch. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 2171–2180. DOI: http://dx.doi.org/10.1145/2702123.2702277
8. Philip L. Davidson and Jefferson Y. Han. 2008. Extending 2D Object Arrangement with Pressure-Sensitive Layering Cues. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology (UIST '08)*. ACM, New York, NY, USA, 87–90. DOI:http://dx.doi.org/10.1145/1449715.1449730
9. Alexander De Luca, Marian Harbach, Emanuel von Zezschwitz, Max-Emanuel Maurer, Bernhard Ewald Slawik, Heinrich Hussmann, and Matthew Smith. 2014. Now You See Me, Now You Don't: Protecting Smartphone Authentication from Shoulder Surfers. In *Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 2937–2946. DOI: http://dx.doi.org/10.1145/2556288.2557097
10. Masitah Ghazali and Alan Dix. Knowledge of Today for the Design of Tomorrow. In *Proceedings of the 2nd International Design and Engagibility Conference (IDEC '05)*. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.563.6831&rep=rep1&type=pdf
11. Hiroyuki Hakoda, Yoshitomo Fukatsu, Buntarou Shizuki, and Jiro Tanaka. 2015. Back-of-Device Interaction Based on the Range of Motion of the Index Finger. In *Proceedings of the Annual Meeting of the Australian Special Interest Group for Computer Human Interaction (OzCHI '15)*. ACM, New York, NY, USA, 202–206. DOI: http://dx.doi.org/10.1145/2838739.2838812
12. Seongkook Heo and Geehyuk Lee. 2011a. Force Gestures: Augmented Touch Screen Gestures Using Normal and Tangential Force. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems (CHI EA '11)*. ACM, New York, NY, USA, 1909–1914. DOI: http://dx.doi.org/10.1145/1979742.1979895
13. Seongkook Heo and Geehyuk Lee. 2011b. Forcetap: Extending the Input Vocabulary of Mobile Touch Screens by Adding Tap Gestures. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services (MobileHCI '11)*. ACM, New York, NY, USA, 113–122. DOI:http://dx.doi.org/10.1145/2037373.2037393
14. Seongkook Heo and Geehyuk Lee. 2012. ForceDrag: Using Pressure as a Touch Input Modifier. In *Proceedings of the 24th Australian Computer-Human Interaction Conference (OzCHI '12)*. ACM, New York, NY, USA, 204–207. DOI: http://dx.doi.org/10.1145/2414536.2414572
15. Apple inc. 2017 (accessed January 6, 2017). iOS9 UIKit API Reference. https://developer.apple.com/reference/uikit/uitouch. (2017 (accessed January 6, 2017)).
16. Markus Löchtefeld, Christoph Hirtz, and Sven Gehring. 2013. Evaluation of Hybrid Front- and Back-of-Device Interaction on Mobile Devices. In *Proceedings of the 12th International Conference on Mobile and Ubiquitous Multimedia (MUM '13)*. ACM, New York, NY, USA, Article 17, 4 pages. DOI: http://dx.doi.org/10.1145/2541831.2541865
17. Ross McLachlan, Daniel Boland, and Stephen Brewster. 2014. Transient and Transitional States: Pressure as an Auxiliary Input Modality for Bimanual Interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 401–410. DOI: http://dx.doi.org/10.1145/2556288.2557260

18. Ross McLachlan and Stephen Brewster. 2015. Bimanual Input for Tablet Devices with Pressure and Multi-Touch Gestures. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '15)*. ACM, New York, NY, USA, 547–556. DOI: http://dx.doi.org/10.1145/2785830.2785878

19. Sachi Mizobuchi, Shinya Terasaki, Turo Keski-Jaskari, Jari Nousiainen, Matti Ryynanen, and Miika Silfverberg. 2005. Making an Impression: Force-Controlled Pen Input for Handheld Devices. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems (CHI EA '05)*. ACM, New York, NY, USA, 1661–1664. DOI: http://dx.doi.org/10.1145/1056808.1056991

20. Mohammad Faizuddin Mohd Noor, Andrew Ramsay, Stephen Hughes, Simon Rogers, John Williamson, and Roderick Murray-Smith. 2014. 28 Frames Later: Predicting Screen Touches from Back-of-Device Grip Changes. In *Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 2005–2008. DOI: http://dx.doi.org/10.1145/2556288.2557148

21. R. Kevin Nelson. 2017 (accessed January 6, 2017). Exploring Apple's 3D Touch. https://medium.com/@rknla/exploring-apple-s-3d-touch-f5980ef45af5#.pijkgrkw0. (2017 (accessed January 6, 2017)).

22. Siyuan Qiu, Lu Wang, and Laikuan Wong. 2016. Pressure-Based Touch Positioning Techniques for 3D Objects. In *Proceedings of the 20th ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games (I3D '16)*. ACM, New York, NY, USA, 199–200. DOI: http://dx.doi.org/10.1145/2856400.2876010

23. Gonzalo Ramos, Matthew Boulos, and Ravin Balakrishnan. 2004. Pressure Widgets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '04)*. ACM, New York, NY, USA, 487–494. DOI:http://dx.doi.org/10.1145/985692.985754

24. Gonzalo A. Ramos and Ravin Balakrishnan. 2007. Pressure Marks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '07)*. ACM, New York, NY, USA, 1375–1384. DOI: http://dx.doi.org/10.1145/1240624.1240834

25. Xiangshi Ren, Jibin Yin, Shengdong Zhao, and Yang Li. 2007. The Adaptive Hybrid Cursor: A Pressure-Based Target Selection Technique for Pen-Based User Interfaces. In *Proceedings of the 2007 IFIP TC15 International Conference on Human-Computer Interaction (INTERACT'07)*. Springer, Berlin, Heidelberg, 310–323. DOI:http://dx.doi.org/10.1007/978-3-540-74796-3_29

26. Oliver Schoenleben and Antti Oulasvirta. 2013. Sandwich Keyboard: Fast Ten-Finger Typing on a Mobile Device with Adaptive Touch Sensing on the Back Side. In *Proceedings of the 15th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '13)*. ACM, New York, NY, USA, 175–178. DOI: http://dx.doi.org/10.1145/2493190.2493233

27. Erh-li Early Shen, Sung-sheng Daniel Tsai, Hao-hua Chu, Yung-jen Jane Hsu, and Chi-wen Euro Chen. 2009. Double-side Multi-touch Input for Mobile Devices. In *CHI '09 Extended Abstracts on Human Factors in Computing Systems (CHI EA '09)*. ACM, New York, NY, USA, 4339–4344. DOI: http://dx.doi.org/10.1145/1520340.1520663

28. Katie A. Siek, Yvonne Rogers, and Kay H. Connelly. 2005. Fat Finger Worries: How Older and Younger Users Physically Interact with PDAs. In *Proceedings of the 2005 IFIP TC13 International Conference on Human-Computer Interaction (INTERACT'05)*. Springer-Verlag, Berlin, Heidelberg, 267–280. DOI: http://dx.doi.org/10.1007/11555261_24

29. R. William Soukoreff and I. Scott MacKenzie. 2004. Towards a Standard for Pointing Device Evaluation, Perspectives on 27 Years of Fitts' Law Research in HCI. *Int. J. Hum.-Comput. Stud.* 61, 6 (Dec. 2004), 751–789. DOI:http://dx.doi.org/10.1016/j.ijhcs.2004.09.001

30. Caleb Southern, James Clawson, Brian Frey, Gregory Abowd, and Mario Romero. 2012. An Evaluation of BrailleTouch: Mobile Touchscreen Text Entry for the Visually Impaired. In *Proceedings of the 14th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '12)*. ACM, New York, NY, USA, 317–326. DOI:http://dx.doi.org/10.1145/2371574.2371623

31. Daniel Spelmezan, Caroline Appert, Olivier Chapuis, and Emmanuel Pietriga. 2013a. Controlling Widgets with One Power-up Button. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology (UIST '13)*. ACM, New York, NY, USA, 71–74. DOI:http://dx.doi.org/10.1145/2501988.2502025

32. Daniel Spelmezan, Caroline Appert, Olivier Chapuis, and Emmanuel Pietriga. 2013b. Side Pressure for Bidirectional Navigation on Small Devices. In *Proceedings of the 15th International Conference on Human-computer Interaction with Mobile Devices and Services (MobileHCI '13)*. ACM, New York, NY, USA, 11–20. DOI:http://dx.doi.org/10.1145/2493190.2493199

33. Craig Stewart, Michael Rohs, Sven Kratz, and Georg Essl. 2010. Characteristics of Pressure-Based Input for Mobile Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*. ACM, New York, NY, USA, 801–810. DOI: http://dx.doi.org/10.1145/1753326.1753444

34. Feng Wang and Xiangshi Ren. 2009. Empirical Evaluation for Finger Input Properties in Multi-touch Interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09)*. ACM, New York, NY, USA, 1063–1072. DOI: http://dx.doi.org/10.1145/1518701.1518864

35. Daniel Wigdor, Clifton Forlines, Patrick Baudisch, John Barnwell, and Chia Shen. 2007. Lucid Touch: A See-Through Mobile Device. In *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology (UIST '07)*. ACM, New York, NY, USA, 269–278. DOI: http://dx.doi.org/10.1145/1294211.1294259

36. Graham Wilson, Stephen Brewster, and Martin Halvey. 2013. Towards Utilising One-handed Multi-digit Pressure Input. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems (CHI EA '13)*. ACM, New York, NY, USA, 1317–1322. DOI: http://dx.doi.org/10.1145/2468356.2468591

37. Graham Wilson, Stephen A. Brewster, Martin Halvey, Andrew Crossan, and Craig Stewart. 2011. The Effects of Walking, Feedback and Control Method on Pressure-Based Interaction. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services (MobileHCI '11)*. ACM, New York, NY, USA, 147–156. DOI:http://dx.doi.org/10.1145/2037373.2037397

38. Graham Wilson, Craig Stewart, and Stephen A. Brewster. 2010. Pressure-Based Menu Selection for Mobile Devices. In *Proceedings of the 12th International Conference on Human Computer Interaction with Mobile Devices and Services (MobileHCI '10)*. ACM, New York, NY, USA, 181–190. DOI: http://dx.doi.org/10.1145/1851600.1851631

39. Katrin Wolf, Christian Müller-Tomfelde, Kelvin Cheng, and Ina Wechsung. 2012. PinchPad: Performance of Touch-Based Gestures While Grasping Devices. In *Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction (TEI '12)*. ACM, New York, NY, USA, 103–110. DOI: http://dx.doi.org/10.1145/2148131.2148155

40. Xing-Dong Yang, Edward Mak, Pourang Irani, and Walter F. Bischof. 2009. Dual-Surface Input: Augmenting One-Handed Interaction with Coordinated Front and Behind-the-Screen Input. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '09)*. ACM, New York, NY, USA, Article 5, 10 pages. DOI: http://dx.doi.org/10.1145/1613858.1613865

41. Cheng Zhang, Anhong Guo, Dingtian Zhang, Caleb Southern, Rosa Arriaga, and Gregory Abowd. 2015. BeyondTouch: Extending the Input Language with Built-in Sensors on Commodity Smartphones. In *Proceedings of the 20th International Conference on Intelligent User Interfaces (IUI '15)*. ACM, New York, NY, USA, 67–77. DOI: http://dx.doi.org/10.1145/2678025.2701374