Towards Utilising One-Handed Multi-Digit Pressure Input

Graham Wilson
Stephen Brewster
Glasgow Interactive Systems Group
School of Computing Science
University of Glasgow
Glasgow G12 8QQ, UK
first.last@glasgow.ac.uk

Martin Halvey
Glasgow Caledonian University
School of Engineering and Built
Environment,
Glasgow, G4 0BA, UK
Martin.halvey@gcu.ac.uk

Copyright is held by the author/owner(s).

CHI 2013 Extended Abstracts, April 27–May 2, 2013, Paris, France.
ACM 978-1-4503-1952-2/13/04.

Abstract

This paper explores the potential uses of pressure input from multiple digits (i.e., all 4 fingers and the thumb) of one hand squeezing a mobile device: multiple digits may provide multiple inputs. The potential advantages for mobile interaction include freeing the second hand for other tasks, and providing access to multiple functions simultaneously. A range of possible interactions is discussed including the benefits and challenges posed by complex pressure-based input on mobile devices. An example usage scenario is described and tested: pressure-based input for simultaneous zooming and rotating in a map task. Results suggest multi-digit pressure input may be a useful means of interaction with mobile devices.

Author Keywords

Pressure input; one-handed input; mobile interaction

ACM Classification Keywords

H.5.2. Information interfaces and presentation: User Interfaces – Haptic I/O

Introduction

One-handed interaction with mobile devices is common and desirable, for when the individual requires the use of one hand for other tasks, such as carrying objects. There are a number of research papers exploring alternative means of controlling interface elements with one hand. With the ubiquity of touchscreen devices, most interfaces require input on the screen, which occludes the content being interacted with. Because the same hand must grip the device and also provide input, most research interfaces use only one or two digits for input, most commonly the index finger and thumb, or use physical movement of the device, such as tilting, for input. The remaining digits are used only for maintaining device stability and play no part in input.

Simple tasks, such as targeting/menu interaction or panning maps/web-pages, can be carried out individually one-handed, but more complex tasks, or controlling more than one axis, such as zooming and rotating content, often require two hands: one to hold the device and one to carry out a multi-touch gesture. When holding a mobile device one-handed, all five digits may be in contact around its edges, so there is an opportunity to provide multiple inputs from several digits simultaneously. Capacitive sensing around the device can be used to detect the location of digits and be used to change the current mode or function [1]. However, capacitive input is generally limited to binary input: either contact is made or it is not.


Figure 1: Representative one-handed grip of mobile device (right-handed).

Pressure-based input may offer more possibilities, as it provides a range of continuous control. Pressure-based squeezing around the device could provide multiple continuous inputs from several digits simultaneously: referred to here as Multi-Digit Pressure Input (MDPI). By doing so, more than a single channel or axis of input may be controlled with a single hand and without obscuring the screen. This paper describes several potential uses for MDPI on mobile devices. While there may be benefits to this type of interaction, this paper also discusses potential issues that may need to be addressed if such interfaces are to be useful. As an illustration of MDPI, we present initial results from a study that compared one-handed pressure input to two-handed multi-touch gestures for zooming and rotating in a map task.

Related work

A number of research papers on one-handed mobile interaction have focused on improving accuracy or reducing the complexity of existing functionality, such as targeting [5], or shortcut commands [8], through pre-set gestures, either performed on-screen with the thumb or through physical movement of the device [3]. But placing thumbs on the screen covers the content being interacted with, and moving the device makes content more difficult to see. More than one axis, such as both vertical and horizontal touchscreen/tilting movement, may be accessible during the same action, but only for a single task, such as panning, or zooming. As only a single digit is used for input, it is difficult to control or access more than one function at one time.

Pressure input on mobile devices is highly accurate when using a single digit [6, 10] and also when squeezing with different combinations of digits in concert [11]. One-handed pressure input is possible, by pressing into the screen [2, 6] or tangentially across the screen [2]. However, only a single digit is generally used for input (usually the thumb) and input is still provided onto the screen, occluding content. Also, existing mobile pressure interfaces only tend to control a single axis or task at a time, such as zooming [6], linear targeting [10] or scrolling [2]. Some interfaces that have taken input from multiple digits around a device simultaneously have merely summed the total

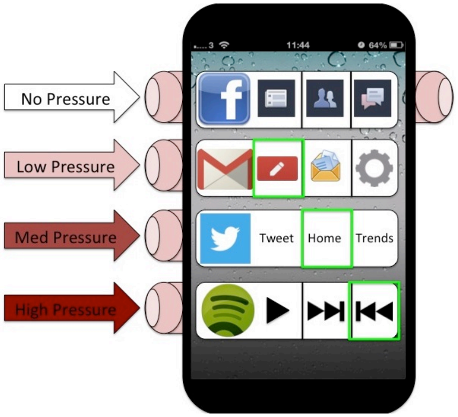

Figure 2: Multi-digit pressure menu interface. Each digit accesses a different app/menu.

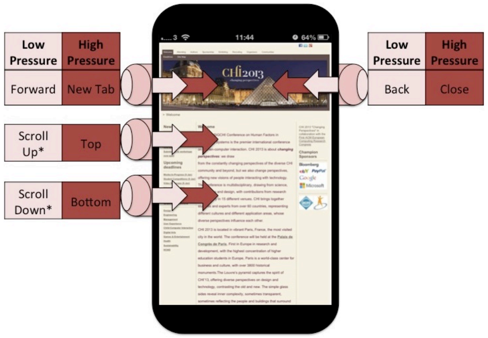

Figure 3: Multi-digit pressure navigation interface. Each digit controls different functions. Scrolling up and down is continuous and velocity-based. Adapted from Heo & Lee [2].

input, and not utilised individual digits [4]. Wilson *et al.* [11] tested how precisely each digit, and various 2 to 5-digit combinations, on one hand could apply pressure to the sides, back and top of a mobile phone during a linear targeting/menu interaction. They showed that every digit could apply pressure accurately, but not equally. While not providing active input to the system, capacitive sensors around the edges of a mobile device have been used to detect hand posture [1]. Back-of-device interaction [13] avoids occlusion issues but input is generally limited to one digit during one-handed use.

Potential Uses

Pressure provides continuous and dynamic input. It is suited to velocity control, where, for example, the speed of scrolling is controlled by how much pressure is applied. However, it has also been used successfully in positional control, where the position of a cursor (e.g. in a menu) is controlled by pressure. Assuming a device can detect pressure across its entire surface, each digit (or digit combination) in contact could therefore theoretically control one direction of movement or traversal, one menu/sub-menu or task/sub-task. The discussion here works from a hypothetical interaction scenario where the mobile device is held in portrait, as shown in Figure 1. The palm and fingers provide opposing forces to grip the device and also provide one channel of pressure input each. While mobile devices are held in landscape for certain functions, it is considered a less common one-handed orientation.

Menus

Pressure input with a single digit or sensor has commonly been used for linear targeting, a way of accessing a menu or similar widget [7]. In MDPI, each digit could access a different menu, within or across applications, providing quicker access to a wider range of common functions (see Figure 2).

Navigation

Heo & Lee [2] developed a set of force gestures for navigating webpages and e-books, based on normal (into the device) and tangential (across device) force applied to the screen. While error rates were low, some gestures were confused with each other and normal touchscreen functions. Using MDPI around the sides of a device would leave the screen visible and could remove potential input confusion. Functions such as forward/back, top/bottom and scrolling can be controlled using light and heavy presses (Figure 3).

Movement/Traversal

Extending navigation to movement within applications such as maps and webpages, MDPI could be well suited to accessing more than a single axis at one time, including scrolling, zooming and rotating. Control over all three simultaneously would be unlikely, given the number of available digits, but scrolling + zooming, or zooming + rotation could be combined. Zooming and rotation are common tasks on mobile applications, but they require two hands to use: one to hold the device and one to carry out a multitouch gesture. Two-handed multitouch has the benefit of providing control over 4 degrees of freedom at one time: in maps, for example, zooming, rotation and x-y translation can be achieved simultaneously. While this may not be possible with MDPI, it has the benefit of accessing 2 degrees of freedom with only 1 hand without covering the screen.

Pilot test: Map Zooming & Rotation

An initial pilot test was run to gain impressions about the potential usefulness of MDPI. Force-sensing

Figure 4: Nexus One phone in a protective case with force sensing resistors attached around its edges. The sensors used during the map task are circled.

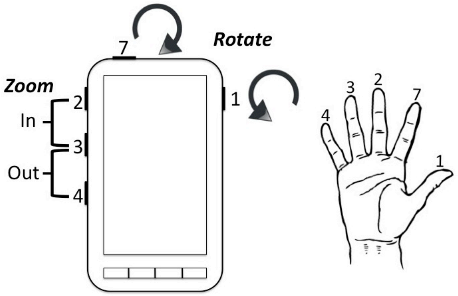

Figure 5: Sensors used for pressure-based controls with relative function and digit used for input. Sensor 7 provides input form the top of the device.

resistors (FSRs) were attached around the edges of a mobile phone and input from multiple digits was compared to standard multitouch zoom and rotate gestures. There were three tasks: *Rotation* (rotating the map to a target angle), *Zooming* (zooming the map to a target zoom level) and both zooming and rotating combined (*Combination*). Participants altered the angle and/or zoom level to the target before leaving the map stationary for a three second dwell confirmation time. There were 6 target zoom levels (in 15% increments) and 6 target angles (30°, 60°, 180°, 240° and 300°). The 6 combination targets paired zoom level 1 with angle 1, zoom level 2 with angle 2 and so on.

Apparatus

Eight FSRs (Interlink Electronics model 400FSR) with linearized output [10] were attached to the body of a Nexus One mobile phone in the configuration shown in Figure 4, for right-handed input (based on the layout from Wilson *et al.* [11]). Only those five sensors shown in Figure 5 (circled in Figure 4) were used for the evaluation. These differ from the example grip in Figure 1; pressing on the top of the device with the index finger (sensor 7) was more precise than pressing from the left-hand side [11]. A PC sent sensor values to the Nexus One GUI over USB.

Multi-Digit Input and Multitouch Gestures

The design of multi-digit pressure input was based on how accurately the different digits could apply pressure to a mobile phone [11]. The FSRs needed to be placed in locations that were easily reachable without repositioning the hand or device. They also had to be in positions that could provide opposing forces so that the phone could be held securely and squeezed freely with the same hand. It needed to be possible to perform both rotation and zooming at the same time, so the same digit could not be used for both zooming and rotating. The pressure-based controls are shown in Figure 5. This design provided somewhat compartmentalized controls, with the rotation and zooming inputs being physically separate from each other. A pressure space (range of detectable pressure) of 3.5N per digit was used [11], and the mappings of pressure-to-movement were based on initial testing.

Rotation: For standard touchscreen gestures, rotation occurs by placing two fingers onscreen and rotating them in unison. For the pressure-based controls, pressing on sensor 1 with the thumb rotated the map anticlockwise and pressing on sensor 7 with index finger rotated clockwise. In both cases, the speed of rotation increased as the amount of force applied increased, at a rate of 2-200°/sec (57.14° per N/sec).

Zooming: Multitouch zooming was controlled through inward, or outward, pinch gestures with two fingers. For the pressure controls, pressing sensors 2 & 3 with middle and ring fingers together zoomed in and pressing sensors 3 & 4 with ring and little zoomed out. Increasing the amount of force increased the speed of zooming, between 1-20% per sec (2.8% per N/sec).

Participants & Experimental Procedure

Twelve participants (6 male, 6 female) aged between 21 and 63 (mean 29.1) took part. Due to the positioning of the sensors, all were required to be right-handed and each was paid £10. The within-subjects study was split into two by Control Method: one half using pressure-based controls first and one half using touch controls first, with the order counterbalanced. In the pressure condition, participants held the device in

Task	Control	Error (%)	MT (sec)	Workload (0-21)
Overall	MDPI	7.01	5.95	7.29
	Multi Touch	6.24	6.49	8.81
Rotation	MDPI	0.52	5.21	-
	Multi Touch	0.38	4.58	-
Zooming	MDPI	11.48	3.24	-
	Multi Touch	11.54	2.69	-
Combined	MDPI	9.07	9.67	-
	Multi Touch	6.86	12.66	-

Table 1: Mean Error, Movement Time (MT) and Overall Workload for each Control Method, and Error and MT for each Task.

their right hand, in the way illustrated in Figure 5. During the touch condition, they held the phone in their left hand and gestured with the right. Within each Control Method condition, the participant completed two blocks, each consisting of 36 trials: 12 trials of each of the three tasks, selecting all 6 targets twice, in a random order. The first block of each Control Method started with 9 practice tasks, three of each type.

The Independent Variables were *Control Method* (Multitouch, Pressure), *Block* and *Task* (Rotation, Zooming, Combined). The Dependent Variables were *Error* (distance between zoom level/angle and target level/angle, expressed as % of zoom/rotation space), *MT* (movement time from first input to last input in each task) and *Subjective Workload* (NASA TLX).

Results

Due to data violating normality assumptions the Error and MT data were analysed using a 2 x 2 x 3 Mixed Model REML with participant as a random factor, following the ART procedure [12]. Note that MT figures do not include the three-second confirmation time. Mean Error, MT and Workload values are shown in Table 1. A paired samples T-test of the NASA TLX workload ratings showed no effect of Control Method on overall subjective workload (t(12) = 1.022, p>0.05).

There was a significant main effect of Control method on Error ($F_{1,131}$=11.743, p=0.001): Multitouch (mean = 6.24%) produced lower average Error than Pressure (7.01%). There was a significant main effect of Control Method on MT ($F_{1,131}$ = 36.111, p<0.001). Pressure produced faster targeting times (mean = 5.95s) than Multitouch (6.49s). There was a significant effect of Task on both Error ($F_{2,262}$ = 1492.021, p<0.001) and MT ($F_{2,262}$=696.252, p<0.001). In both cases each task differed significantly from the other (p<0.001). Rotation had the lowest error (mean=0.44%) followed by Combination (7.91%) and Zooming (11.52%). Concerning MT, Zooming was fastest (mean=2.95s) followed by Rotation (4.85s) and Combination (10.85s).

A significant interaction effect was found between Control Method and Task for both Error ($F_{2,262}$=13.526, p<0.001) and MT ($F_{2,262}$=26.877, p<0.001). For Error, Multitouch was more accurate for both Rotation and Combination, but less accurate for zooming. For MT, Multitouch was faster for Rotation and Zooming but Pressure was faster for Combination.

Initial Discussion

The pilot results were encouraging and, overall, indicate that one-handed pressure-based input could provide concurrent control over two axes or tasks at one time. As might be expected, multitouch input was slightly more accurate (apart from zooming), but pressure input was slightly faster during the combined task. A further encouraging result is that there were no significant differences in subjective workload ratings between the two control methods. Task time was high, but precision was encouraged. However, there are some important issues to take into consideration when evaluating the usefulness of such an interface.

Issues and Future Research Opportunities

Learning: Pressure interfaces will require learning, as the abstract mappings may not be as natural as touch and multitouch gestures. However, performance and workload ratings were comparable between the two control methods in the pilot study, which suggests learning occurred quickly.

Input Interference: Because the same digits are holding the device stable as well as providing input, there is likely to be some interference, as force is required to hold the device. Recent research has suggested that the first 0.6N of detectable pressure should not be used for interaction, due to inadvertent changes in force, especially when walking [9].

Input Limitation: While our pilot interface controlled zooming and rotating, x-y translation of the map would still require on-screen input. Future work will test whether MDPI is limited to controlling two dimensions bi-directionally and how translation could be integrated.

Individual Differences: The maximum force that can be comfortably applied varies between people, as does hand size. Future research should test if 1) how the device is held and 2) where the digits grip the phone influence how pressure input can be applied.

Other Applications: Multi-digit control over multiple menus, navigation of operating systems and other interfaces will also be investigated, particularly when the user is walking and encumbered, such as by carrying bags or other objects.

Conclusions

This paper explored the potential uses of pressure input from multiple digits of one hand squeezing a mobile device, to provide multiple potential inputs. A range of possible interactions was discussed including the benefits and challenges posed by complex pressure-based input on mobile devices. An initial pilot study suggested that multi-digit pressure input may be a suitable means of controlling two dimensional input, specifically bi-directional zoom and rotation in maps.

REFERENCES

1. Chang, W., Kim, K.E., Lee, H., Cho, J.K., Soh, B.S., Shim, J.H., Yang, G.H., Cho, S.J. and Park, J. Recognition of Grip-Patterns by Using Capacitive Touch Sensors. *Proc. IEEE ISIE*, pp 2936-2941.
2. Heo, S. and Lee, G. Force Gestures: Augmenting Touch Screen Gestures with Normal and Tangential Forces. *Proc. UIST 2011*, pp 621-626.
3. Hinckley, K. and Song, H. Sensor Synaesthesia: Touch in Motion, and Motion in Touch. *Proc. CHI 2011*, pp 801-810.
4. Hoggan, E., Stewart, C., Haverinen, L., Jacucci, G. and Lantz, V. Pressages: Augmenting Phone Calls with Non-Verbal Messages. *Proc. UIST 2012*, pp 555-562.
5. Karlson, A. and Bederson, B. One-handed touchscreen input for legacy applications. *Proc. CHI 2008*, pp 1399-1408.
6. Miyaki, T. and Rekimoto, J. GraspZoom: Zooming and scrolling control model for single-handed mobile interaction. *Proc. MobileHCI 2009*, pp 81-84.
7. Ramos, G., Boulos, M. and Balakrishnan, R. Pressure widgets. *Proc. CHI 2004*, pp 487-494.
8. Roudaut, A., Lecolinet, E. and Guiard, Y. MicroRolls: expanding touch-screen input vocabulary by distinguishing rolls vs. slides of the thumb. *Proc. CHI 2009*, pp 927-936.
9. Stewart, C., Hoggan, E., Haverinen, L., Salamin, H. and Jacucci, G. An Exploration of Inadvertent Variations in Mobile Pressure Input. *Proc. MobileHCI 2012*, pp 35-38.
10. Stewart, C., Rohs, M., Kratz, S. and Essl, G. Characteristics of Pressure-Based Input for Mobile Devices. *Proc. CHI 2010*, pp 801-810.
11. Wilson, G., Hannah, D., Brewster, S. and Halvey, M. Investigating One-Handed Multi-Digit Pressure Input for Mobile Devices. *Proc. CHI 2012 Extended Abstracts*, pp 1727-1732.
12. Wobbrock, J., Findlater, L., Gergle, D. and Higgins, J. The Aligned Rank Transform for Nonparametric Factorial Analyses Using Only Anova Procedures. *Proc. CHI 2011*, pp 143-146.
13. Wobbrock, J., Myers, B. and Aung, H. The performance of hand postures in front-and back-of-device interaction for mobile computing*. Int. J. Human-Computer Studies*, 2008. 66, pp 857-875.