

One-Touch Pose Detection on Touchscreen Smartphones

Karsten Seipp

Goldsmiths, University of London, UK
k.seipp@gold.ac.uk

Kate Devlin

Goldsmiths, University of London, UK
k.devlin@gold.ac.uk

ABSTRACT

We present a technique that allows distinguishing between index finger and thumb input on touchscreen phones, achieving an average accuracy of 82.6% in a real-life application with only a single touch. We divide the screen into a virtual grid of 9mm² units and use a dedicated set of training data and algorithms for classifying new touches in each screen location. Further, we present correlations between physical and digital touch properties to extend previous work.

ACM Classification Keywords

H.5.2 User Interfaces:: Graphical user interfaces

Author Keywords

Finger detection; Input detection; Touch; Mobile

INTRODUCTION AND PREVIOUS WORK

To maximise the usability of a GUI for one-handed interaction, researchers have devised a set of approaches comprising enlarged buttons or curved interfaces. However, these often need manual activation or even lock the user to a static GUI once they have configured the application, making it hard to switch between single-handed and bimanual use. Thus, a flexible system for input mode detection is required that allows input pose detection and ultimately interface adaptation on the fly – ideally using a single tap when starting an app.

Some researchers have attempted to detect interaction context using contact-sensitive hardware attached to a device [3, 7, 14] or changes in resonance when vibrating an object [9]. Others have used computer vision [1, 5] to process the shape of a touch, or measured differences in impedance to distinguish between users [4]. However, such approaches do not work on commodity smartphones without additional hardware. Therefore, some researchers have explored the potential of motion and touch sensors intrinsic to modern devices. Wang and Ren [12] provide a detailed description of the finger's physical properties, and how these may be used to enrich the digital dialogue, and use a finger's landing process on the screen of a tabletop to differentiate between left and right hand [11]. On smartphones, researchers use a device's motion sensors to infer where on screen a touch may have occurred [15, 8] and thereby illustrate how different

screen regions possess a distinctive motion profile. Combining changes in device motion and touch size is GripSense [2], which uses the spatiotemporal differences of touch and motion properties to distinguish between different hand postures in an average of four to five interaction steps [p.551]. Yet, as a premise for dynamic input mode detection this may be inadequate in cases where users only interact with a small area of the screen. What is needed is an approach that allows the detection of mode of operation within a single interaction step – ideally by performing only a simple touch of the screen. To address this, we examine whether the limited data provided by screen and motion sensors can be used to infer whether the phone is operated one-handedly via the thumb or with two hands, using the index finger of the one hand while holding it in the other.

INITIAL DATA COLLECTION

To map out the characteristics of a human touch in all areas of a 4.7" smartphone screen, we collected data from 27 participants (8 F, mean age 22.3, SD 2.94, 1 left-handed, 1 ambidextrous). For this we divided the screen of a HTC Sensation XE phone running Android 4.03 into a grid of 6x10 units, each unit with a physical dimension of about 9x9mm, derived from the ideal target size suggested by Parhi et al. [10]. To facilitate data collection, each grid unit was populated by a button. Buttons were highlighted at random to signify the screen area users had to touch (Fig. 1). Users had to complete a total of eight rounds, operating the device with one hand using the left and right thumb, and holding it in either hand, operating it with the index finger of the other, while walking and sitting. Each round started with a full grid of targets which vanished from the display when touched (Fig. 1). After touching the target grid unit, the next one was highlighted with a delay of one second. Errors, such as pressing the wrong button, had no effect, as only the highlighted grid unit was "active" and responded to touch events. The sitting condition was performed in an office, the walking condition on the pavement of a quiet street at about 2.5 km/h, measured via an app.

As a human touch has properties such as duration, location and size, we decided to record the following properties to describe the digital touch: *Grid ID*, *Touch Size Mean*, *Number of Touches*, *Touch Time*, *Offset* on the X/Y axis, the difference between first and last touch point of a touch event on the X/Y axis (*Diff X/Y*) as well as the rotation of the device over three axes during the touch (*Gyro X, Y, Z*) and before and during the touch (*Gyro All X, Gyro All Y, Gyro All Z*). Compared to previous work classifying input pose and hand, we introduce touch duration, number of touch points, and differentiation of device movement *during* and *before and during* a touch. Following [12] and [2], the accelerometer was not recorded.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.

ITS '15, November 15-18, 2015, Funchal, Portugal.

© 2015 ACM. ISBN 978-1-4503-3899-8/15/11...\$15.00

DOI: <http://dx.doi.org/10.1145/2817721.2817739>

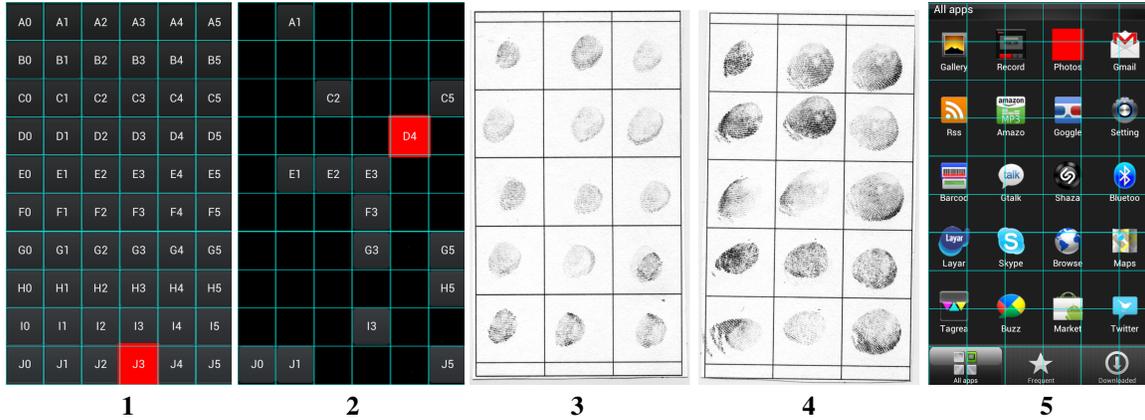


Figure 1. 1: Screen of the initial data collection with an overlaid grid. The button/screen area to be tapped is red. 2: Once a screen area had been tapped and touch data collected, the target was removed. 3+4: Tap shapes of the left index finger and left thumb on the device, recorded on paper using ink. 5: Application: Users have to launch an application by tapping the red icon. The grid, in which touch data has been collected in 1 and 2, is overlaid to illustrate the principle of attributing a new touch on the screen to a unit of the grid.

Physical Data

In addition to the digital properties, we recorded the length of index finger and thumb of both hands by measuring the distance from each finger’s base (the bottom of the proximal phalanx) to its tip (the end of the distal phalanx). To ascertain the physical size of the area of the fingertip that might be used for a full-contact touch, we measured width and height of the participants’ fingerprints. We asked them to press the top of their finger, ranging from the bottom of the distal phalanx right up to the nail bed, onto an ink pad and roll their finger to cover all parts of the skin that could potentially come into contact with the display. We then asked them to position their finger on a parallel line to the tabletop and lower it onto a sheet of paper on the table, without rolling it to either side. Width and height of the print on paper were measured in cm. To examine the shape of the physical touch area in different parts of the screen, a sheet of paper was attached to the phone’s display. The sheet replicated the layout of the button grid, but buttons were combined into groups of four, creating 15 zones (Fig. 1). This procedure is similar to Katre’s [6] for measuring the thumb’s shape in different parts of the screen.

INITIAL EVALUATION

A set of ANOVAs together with a visualisation in the *Weka Explorer* showed *Touch Size Mean* and *X*, *Y*, and *Z* gyroscope amplitudes as suitable for pose classification. *Number of Touches*, *Touch Time*, *X* and *Y* gyroscope amplitudes and *Offset X* were shown as suitable for hand classification.

The mean values of the digital touch properties for the thumb often exceeded those of the index finger by several orders of magnitude. The thumb’s physical touch ellipse was an average of 26.2% wider and 17.5% higher than that of the index finger, which resulted in the thumb’s *Touch Size Mean* property being an average of 39.8% larger. Further, touches of the left hand seemed to be slightly larger in size than those of the right. This suggests that touches with the non-dominant hand are less precise and dexterous than those performed with the dominant hand, resulting in a “sloppier” touch with increased *Touch Size Mean*, *Touch Time*, and gyroscope movement.

The strongest physical factor impacting the digital touch properties seemed to be the limited dexterity and reach of the thumb, mainly caused by the thumb’s total length, which was 16.6% shorter than that of the index finger. This may be the cause of a prolonged contact time with the display when touching (*Touch Time* +26.1% in comparison to the index finger), and found its strongest manifestation in the greatly increased gyroscope amplitudes, caused by the comparatively strong device movement when reaching for a target outside the thumb’s natural movement arc, supporting the findings of Goel et al. [2]. This held true for all three gyroscope axes, but was especially visible for the rotation around the Z-axis before and during the touch, where the amplitude was an average of 235.1% higher for the thumb. The data of the walking condition showed the same trends but amplified, possibly due to higher device movement or reduced dexterity, likely to be caused by a higher cognitive load while walking [13]. Table 1 shows the degree of correlation between the average values of physical and digital properties in the sitting condition.

Digital property	Physical property	Sp. rho	p
<i>No. of Touches</i>	Touch shape width (index f.)	.423	.028
<i>Gyro Y Ampl.</i>	Limb length (thumb)	.464	.015
<i>Gyro Z Ampl.</i>		.533	.004
<i>Gyro X All Ampl.</i>		.576	.002
<i>Gyro Y All Ampl.</i>		.644	< .001
<i>Gyro Z All Ampl.</i>		.632	< .001
<i>Gyro X All Ampl.</i>	Touch shape width (thumb)	.479	.012

Table 1. Correlations of digital and physical properties, measured in Spearman’s rho (Sp. rho). Degree of significance given as p.

Table 1 suggests that the ergonomic characteristics *limb length* and *touch shape width* of the thumb are related to the change in the digital touch values. The thumb’s fixation to the hand holding the device and the resulting limited area of movement may cause a greater degree of device motion together with a flatter connection angle of thumb and screen, as also found by Goel et al.[2] and Katre [6]. The shortage of statistically significant correlations between physical and digital properties for the index finger indicates that distinctions between finger and thumb input (based solely on correlations between their average physical and digital properties) may

Mode	App. A	App. B
P (S)	84.1	82.1
P (W)	78.7	80.8
H (S)	61.8	64.5
H (W)	60.9	63.7

Table 2. Mean classification accuracy in % derived from the cross-validation for input pose (P) and hand (H) when using App. A and App. B, sitting (S) and walking (W).

Mode	App. A	App. B
P (S)	82.2	82.5
P (W)	73.8	73.9
H (S)	58.6	62.5
H (W)	60.8	62.9

Table 3. Mean classification accuracy for input pose (P) and hand (H) in % when using App. A and App. B on the verification data set of the second study, sitting (S) and walking (W).

Mode	App. A	App. B
P (S)	83.1	82.3
P (W)	77.3	75.2
H (S)	61.6	62.2
H (W)	62.3	61.9

Table 4. Mean accuracy for App. A and App. B in % for classifying input pose and hand, while sitting (S) and walking (W), derived from the cross-validation and the verification.

Config.	S	W
App. A	82.6	71.1
App. B	81.9	72.3

Table 5. The average pose classification accuracy of App. A and App. B in % when embedded into the example application, while sitting (S) and walking (W).

not be made sweepingly for the whole screen, but rather require a subdivision of the screen to detect differences in the values of each with reference to a spatial location.

Analysis Using Machine Learning

The initial data collection created a set of reference values for each of the 60 grid positions for each of the poses in the two conditions. For further examination, we applied various machine learning algorithms to the data set of each grid point individually to examine their effectiveness for classifying input pose and hand in this region: Three *K Nearest Neighbour* algorithms with a configuration of $K=1$, $K=3$ and $K=5$; one *Random Forest* algorithm using 12 attributes of the touch event (all but the *DiffXY* properties, which were unsuitable) and *Weka's J48* classifier in standard configuration:

- **KX**: weka.classifiers.lazy.IBk -K X -W 0 -A weka.core.neighboursearch.LinearNN Search -A weka.core.EuclideanDistance -R first-last
- **J48**: weka.classifiers.trees.J48 -C 0.25 -M 2
- **RF12**: weka.classifiers.trees.RandomForest -I 10 -K 12 -S 1 -num-slots 1

The algorithms were chosen in order to explore whether pose and hand can be determined based on a degree of similarity of their overall characteristics (*KNN*) or based on the values of certain key properties in relation to others (*J48*, *Random Forest*). To avoid overfitting, we validated each model for each target position with a stratified tenfold cross-validation. Using this approach we defined the classification performance of each algorithm in each of the 60 grid units while sitting and walking. From this, we created a classification accuracy map of the screen for both input pose and hand, while sitting and while walking. For a detailed breakdown of grid position and algorithm performance, please see the additional material. All data was evaluated using *Weka*.

As shown by previous work [2, 15], touch properties may vary greatly between different screen areas. Therefore, using a single algorithm to evaluate all grid points may not be adequate. Although the *RF12* algorithm provided the best overall accuracy (82.7%), its performance was sometimes shadowed by the various *NN* algorithms, depending on screen position. To address this, we developed two approaches:

Approach A uses the highest-ranking algorithm in each of the 60 grid units determined by the cross-validation to classify new data.

Approach B uses the three highest-ranking algorithms (determined by the cross-validation) in each of the 60 grid units in a majority voting process to classify new data. In the case of equal performance, preference was given to higher K over lower ones, and to decision trees (*J48*, *RF12*) over *NN*. Applying these approaches to our classification accuracy map

suggests a potential average classification accuracy of input pose and hand as defined in Table 2. **While one-tap input pose classification (index finger/thumb input) seems possible, hand classification (left/right) does not.**

Model Verification

To verify the cross-validated model performance for each screen location, we conducted a second study with 10 new users (5 F, mean age: 33.8, SD: 3.97, 2 left-handed, 1 ambid.) using the same design as the first study. The new data points for each grid unit were then evaluated in *Weka* using the cross-validated models (Tab. 3). Using the verification data set, the average accuracy for input pose and hand classification differs slightly from the accuracy predicted by the cross-validation (Tab. 2), but, as differences are rather small, the results suggest the models may be considered stable under the study conditions. Therefore we suggest the classification accuracy to be derived from the mean performance of cross-validation and verification (Tab. 4).

REAL-LIFE APPLICATION

In order to adapt the interface or interpretation of input events to the input pose of the user, it ought to be determined using only a single touch – ideally at the start of the application, so as not to surprise the user with a sudden change. To assess our technique's performance for this purpose, we mimicked the layout of a typical application grid on a HTC Sensation XE using Android 4.03, showing a total of 23 icons, including three action buttons at the bottom of the screen. At random, one of the elements was highlighted in red (Fig. 1), indicating to the user to tap it. Once an icon had been tapped, it vanished. After a delay of one second the next item was highlighted. Only "active" icons responded to a user's touch and errors could not be made. Fourteen new users (8 F, mean age: 35.1, SD: 4.98, 1 left-handed, 2 ambid.) performed eight rounds, operating the device with one hand using the left and right thumb, and holding it in either hand, operating it with the index finger of the other. The tasks were performed in two conditions: While sitting in an office and while walking on a busy street at approximately 2.5 km/h. The study was counterbalanced by input pose and mobility (sitting/walking).

The classification of incoming touches works as follows: The screen is divided into a virtual grid of 9mm^2 sized squares, corresponding to the grid of the initial data collection. Each square presents a screen area for which the best algorithm (*Approach A*) or the best three algorithms (*Approach B*) for pose classification have been determined earlier in the cross-validation. When a new touch occurs on the screen, its *XY*

coordinate is matched to a position in the grid into which dimensions and coordinates it fits. Once the corresponding grid unit has been determined, the new touch data is evaluated against the previously collected reference data for this grid unit and screen area using the *Weka* library implemented into the application. It is important to understand that grid units do not have to be actual buttons. Rather, they are invisible, laid over the interface to serve as points of reference from which to draw comparison data and algorithm configuration (Fig. 1). Hitting a unit off-centre is of little relevance, as the offset property is not important for pose classification.

Results

The performance of *Approach A* and *Approach B* for classifying the user's input pose using a single touch on the screen in our example application are as described in Table 5. The accuracy rates are close to the ones given by the cross-validation and verification, showing that one-tap input pose classification is possible with a promising degree of accuracy in a real-life application, but reduced by about 10% when walking, due to greater device movement. However, hand detection is not reliable with a single touch and was therefore not evaluated.

COMPARISON TO PREVIOUS WORK

Wang et al. [11] use the angle between first and last touch point to classify a user's hand on tabletops. We applied their approach to our data and achieved an average of 68.8% accuracy. This is likely due to the unfulfillable prerequisite of expecting users to perform an "oblique" touch for the algorithm to work best. Still, this is preferable to our approach (62.3%) when aiming to classify the user's hand. A comparison to GripSense [2] is twofold: For differentiating between index finger and thumb input, *Approach A* provided an average accuracy of 82.6% compared to GripSense's 84.3%, but with the advantage of using up to only a fifth of the interactions and without spatial and procedural constraints. For distinguishing between input via the left and right hand, GripSense has 85.4% accuracy (when the device is used with the thumb) and is 23% more accurate than our technique, but still needs multiple touches. Yet, it has to be considered that GripSense's accuracy was evaluated using a different set of applications and that its power consumption may be lower due to its simpler algorithm, complicating a direct comparison.

CONTRIBUTION

We have shown how physical and digital finger and touch properties relate to each other (Tab. 1) and thus extended the work of [12] and [2]. Further, we have presented an approach that allows a device to differentiate between index finger and thumb input with an average accuracy of up to 82.6% using only a *single touch*. This is possible by recording touch and movement data for 60 screen regions on a 4.7" smartphone and evaluating new touches by attributing these to a given reference unit in the grid with a specified algorithm. This is an advance over previous work, which uses spatiotemporal differences in touch and movement data from *multiple touches* [2], yielding a similar degree of accuracy. Yet, accuracy is susceptible to device movement (walking) and hand detection is not possible with a single tap. We therefore suggest

to pair our technique with a hardware-based approach, such as HandSense [14], for maximum effectiveness. While this paper has shown the potential of the technique, future work will examine the scalability of the approach, whether it can be used to differentiate between other poses and fingers, and aim to limit the effect of walking on accuracy.

REFERENCES

1. Cao, X., Wilson, A. D., Balakrishnan, R., Hinckley, K., and Hudson, S. E. Shapetouch: Leveraging contact shape on interactive surfaces. In *Proc.TABLETOP'08*, 129–136.
2. Goel, M., Wobbrock, J., and Patel, S. Gripsense: using built-in sensors to detect hand posture and pressure on commodity mobile phones. In *Proc.UIST'12*, 545–554.
3. Harrison, B. L., Fishkin, K. P., Gujar, A., Mochon, C., and Want, R. Squeeze me, hold me, tilt me! an exploration of manipulative user interfaces. In *Proc.CHI'98*, 17–24.
4. Harrison, C., Sato, M., and Poupyrev, I. Capacitive fingerprinting: Exploring user differentiation by sensing electrical properties of the human body. In *Proc.UIST'12*, 537–544.
5. Holz, C., and Baudisch, P. Fiberio: A touchscreen that senses fingerprints. In *Proc.UIST'13*, 41–50.
6. Katre, D. One-handed thumb use on smart phones by semi-literate and illiterate users in india. vol. 316 of *IFIP Advances in Information and Communication Technology*. Springer, 189–208.
7. Kim, K.-E., Chang, W., Cho, S.-J., Shim, J., Lee, H., Park, J., Lee, Y., and Kim, S. Hand grip pattern recognition for mobile user interfaces. In *Proc.IAAI'06*, 1789–1794.
8. Miluzzo, E., Varshavsky, A., Balakrishnan, S., and Choudhury, R. R. Tapprints: Your finger taps have fingerprints. In *Proc.MoSys'12*, 323–336.
9. Ono, M., Shizuki, B., and Tanaka, J. Touch&activate: Adding interactivity to existing objects using active acoustic sensing. In *Proc.UIST'13*, 31–40.
10. Parhi, P., Karlson, A. K., and Bederson, B. B. Target size study for one-handed thumb use on small touchscreen devices. In *Proc.MHCI'06*, 203–210.
11. Wang, F., Cao, X., Ren, X., and Irani, P. Detecting and leveraging finger orientation for interaction with direct-touch surfaces. In *Proc.UIST'09*, 23–32.
12. Wang, F., and Ren, X. Empirical evaluation for finger input properties in multi-touch interaction. In *Proc.CHI'09*, 1063–1072.
13. Wilson, G., Brewster, S. A., Halvey, M., Crossan, A., and Stewart, C. The effects of walking, feedback and control method on pressure-based interaction. In *Proc.MHCI'11*, 147–156.
14. Wimmer, R., and Boring, S. HandSense: discriminating different ways of grasping and holding a tangible user interface. In *Proc.TEI'09*, 359–362.
15. Xu, Z., Bai, K., and Zhu, S. TapLogger: inferring user inputs on smartphone touchscreens using on-board motion sensors. In *Proc.WISEC'12*, 113–124.