Holding Patterns: Detecting Handedness With A Moving Smartphone At Pickup

Détection de la manualité via les capteurs d'orientation du smartphone lors de la prise en main

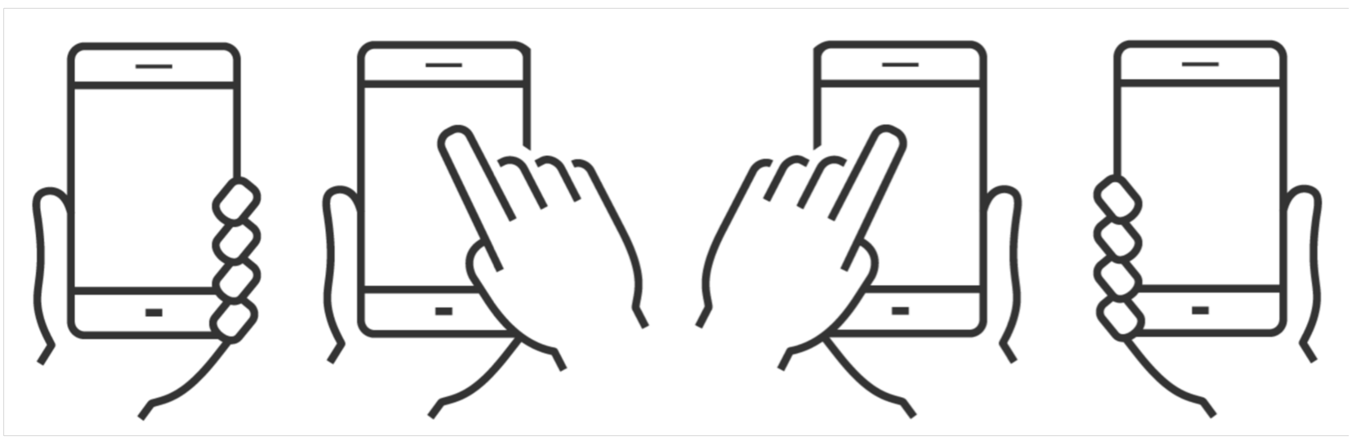
Jeff Avery
University of Waterloo
Waterloo, Canada
jeffery.avery@uwaterloo.ca

Daniel Vogel
University of Waterloo
Waterloo, Canada
dvogel@uwaterloo.ca

Edward Lank
Cheriton School of Computer Science,
University of Waterloo
Waterloo, Ontario, Canada
lank@uwaterloo.ca

Damien Masson
University of Waterloo
Waterloo, Canada
dmasson@uwaterloo.ca

Hanae Rateau
University of Waterloo
Waterloo, Canada
hrateau@uwaterloo.ca



(a) left-single (b) left-cradle (c) right-cradle (d) right-single

Figure 1: Smartphone holding postures: our method detects which hand is holding the phone prior to first interaction.

ABSTRACT

People often switch hands while holding their phones, based on task and context. Ideally, we would be able to detect which hand they are using to hold the device, and use this information to optimize the interaction. We introduce a method to use built-in orientation sensors to detect which hand is holding a smartphone prior to first interaction. Based on logs of people picking up and unlocking a smartphone in a controlled study, we show that a dynamic-time warping approach trained with user-specific examples achieves 83.6% accuracy for determining which hand is holding the phone, prior to touching the screen.

IHM '19, December 10–13, 2019, Grenoble, France

ACM ISBN 978-1-4503-7026-4/19/12…$15.00
https://doi.org/10.1145/3366550.3372253

CCS CONCEPTS

• **Human-centered computing → HCI theory, concepts and models**; **Pointing**; **Visualization techniques**.

KEYWORDS

interaction technique, models, handedness, smartphone, unlock

RÉSUMÉ

En fonction de la tâche et du contexte, les utilisateurs de smartphone ont pour habitude de changer de main pour tenir leur appareil. Idéalement, nous souhaiterions connaître la main utilisée afin d'optimiser l'interaction. A cet effet, nous introduisons une méthode utilisant les capteurs d'orientation intégrés afin de déterminer la main tenant le smartphone avant toute interaction. Nous montrons, par l'analyse des données de participants prenant et déverrouillant leurs smartphones durant une expérience contrôlée, qu'une approche utilisant l'algorithme Dynamic-Time Warping permet d'obtenir une précision de 83.6% afin de détecter la main utilisée.

MOTS CLÉS

techniques d'interaction, modèles

1 INTRODUCTION

Smartphones are often held one-handed with the thumb used for touch input [7]. The shift towards larger phones makes this style of interaction challenging: reaching far corners of a large screen with a thumb while holding a phone in the same hand is uncomfortable and awkward. Researchers have recognized this issue and introduced methods to make single-handed use on large devices easier. For example, Kim et al. [8] introduce Edge and Large touch for interacting with large screens, and Apple's Reachability feature "shifts" the interface down to make the furthest corners of the screen accessible. However, these techniques require explicit activation, which can be inefficient since people switch hands based on what they are doing and what they are holding [7, 8], requiring continual readjustment of their hand posture. A more proactive solution would dynamically recognize how the phone is held and adapt the interface accordingly. Many attempts have been made to optimize interfaces based on detecting handedness, but most require special sensors, or utilize touch data collected during the initial interaction. We feel that the idea mechanism would detect handedness before the user had attempted to interact with the screen.

TouchMe, for example, place additional capacitive sensors along the sides of the case to determine hand position [2]. In Pre-Touch [5], a self-capacitance touchscreen is used to leverage aspect of the touch happening before an actual contact. Similar to our work, their prototype can detect the hand holding the phone prior to the first interaction. They also show how this information can be used to augment interfaces in the context of one-handed interaction. Our approach differs in that it uses information already available in modern smartphones and does not need specific hardware. ContextType detects which hand is holding the phone only after a firm tap on the side of case [3]. GripSense [4] combines touch characteristics like capacitive pressure with internal motion sensor thresholds to distinguish between holding a phone with the left hand, the right hand, or both hands with 84.3% accuracy. Park et al. [12] extend GripSense with accelerometer data to improve accuracy to 87.7%. However, GripSense requires 4 to 5 taps or swipes to achieve these levels of accuracy, which takes time, and is only valid for that specific interaction.

Lochtefeld et al. use DTW on swipe data captured during device unlock to achieve an impressive 98.5% accuracy in detecting handedness, but are restricted to working with data captured as part of the unlock swipe gestures [9]. There is a recent move towards using biometric unlock mechanisms to unlock devices (e.g. gaze to unlock on recent Apple iPhones, or fingerprint identification). If we wish to support these mechanisms, we need to consider approaches that do not rely on touch-screen gestures to identify handedness.

We contribute a method to detect which hand is holding a smartphone by analyzing the built-in orientation sensor stream as the phone is lifted and unlocked, before the user touches the screen. Our insight is that the movement of the phone as the user removes it from their pocket can be recast as an implicit motion gesture [13]. We use dynamic-time warping (DTW) trained on these "pre-unlock" motions performed by the same person. With this setup, we can detect whether a phone is held in the left or right hand with 83.6% accuracy, prior to the user touching the screen. Our method is as accurate as GripSense's holding hand detection, and we show that two training examples may be sufficient to achieve this level of accuracy, so user-specific configuration in a real deployment is practical. While our method cannot detect a change of hand after unlock, it could be combined with GripSense that predicts the holding hand after a few taps or swipes, effectively detecting when the phone is held in a different hand.

2 SMARTPHONE HOLDING PATTERNS

Before describing our data gathering experiment and detection method, we review Hoober's field survey of smartphone holding postures [7] since it defines hand posture terminology and characterizes real world occurrence.

Hoober classified 1,333 observations of people using smartphones in public into 5 holding postures. We change

Hoober's class names to emphasize which hand is holding the phone and what style of hold is used (Figure 1): held in the left hand with left thumb input ("left-single"), held in the right hand with right thumb input ("right-single"), cradled in the left hand with right finger input ("left-cradled"), cradled in the right hand with left finger input ("right-cradled"), and two-handed with two thumb input ("two-handed").

Hoober found that most interactions were single-handed (49%) or cradled (36%), with people using the two-handed posture predominantly for typing (15%). Given a strong mapping of two-hands to typing [10] and its relatively low occurrence, we focus only on single-handed holding postures. For single-handed postures, 67% were right-single and 33% were left-single. This does not correlate with the estimated proportion of left-handed individuals in a population (between 3% to 12% [10]), suggesting some level of non-dominant hand interaction. Hoober, and Ng et al. [11], both observed people holding their smartphone with different hands based on the context (e.g. when opening a door, they may wish to use their dominant hand).

3 DATA COLLECTION

Each time a person grasps their smartphone and unlocks it, there is a predictable series of events: the power switch is pressed, the phone is lifted up to face the user, and a touch screen unlock widget is moved. Our hypothesis is that the motion during this "pre-unlock" sequence varies depending on which hand is used to hold the phone, and systematic differences can be detected using a classifier. The goal of this experiment is to collect example motion data from the accelerometer, gravity, and orientation sensors in the period between power-on and unlock. To our knowledge, no previous work has explored this unlocking motion. Truong et al. [15] augment the unlock gesture itself and Hintze et al. [6] examine unlocked and locked usage focusing on touch screen interactions before and after unlocking.

Participants

16 people were paid $2 to participate (8 females, 1 left-handed, mean age 23.8 years, SD 6.1). All had prior experience with smartphones and using swipe to unlock.

Apparatus

The experiment was conducted using a Nexus 5 phone running Android 5.02. A custom Android application was written to simulate the appearance, position, and behaviour of standard unlock widgets, but with greater instrumentation. All sensor data streams were logged at the highest available sample rate (100 Hz). From an initial pilot with 6 participants, we observed similar motion data regardless whether a person is sitting or standing, whether the phone starts in a pocket or laying on table. This observation would need further investigation, but it suggests that our results should generalize to all contexts even if the collected data is limited to one of these conditions. For this reason, the smartphone was placed face on a table down beside participants, prior to starting each task.

Task

The experimental task required participants to pick up the smartphone from the table and unlock it using a specific hand posture. Before each trial, the phone was locked and placed on a table beside them. Three seconds later, a chime rang prompting the participant to pick up the phone, turn on the power, and swipe an unlock slider. This completed the trial, and the process repeated (Figure 2).

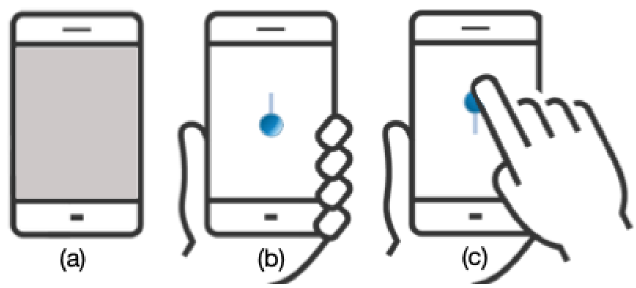


Figure 2: Experimental task. From a resting position (a), the participant picked up the phone and pressed the power button (b), then used the unlock widget to unlock the phone (c).

The unlock slider was displayed in one of ten combinations of position and direction representing the most common configurations of current devices (Figure 3). The unlock slider target was a 15 mm diameter circle with a 15 mm long slider track, similar to standard unlock widgets. Based on the initial target position, the slider was either centered in the display, or positioned 8 mm from the bot-tom, top, or side to approximate positions of unlock widgets on commercial devices.

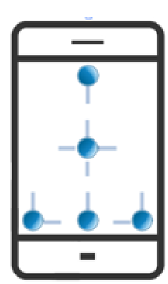


Figure 3: Unlock slider positions. Sliders were presented in one of ten common unlock widget positions.

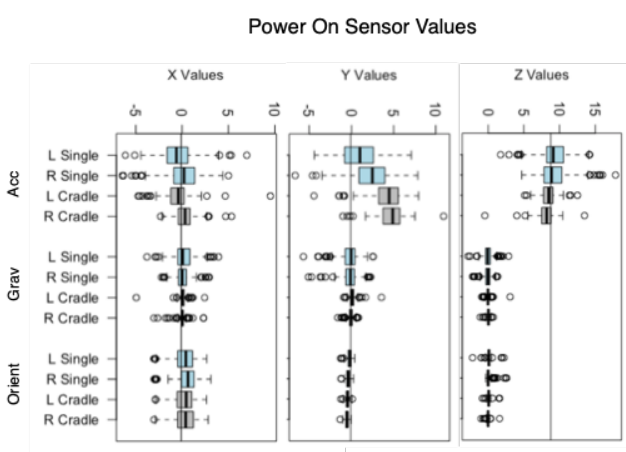
Design

The independent variables are the HAND used to hold the phone (LEFT or RIGHT) and the type of HOLD used (SINGLE or CRADLE) which combine into a hand holding POSTURE factor with four levels: LEFT-SINGLE, LEFT-CRADLE, RIGHT-SINGLE, RIGHT-CRADLE (Figure 1). The dependent variables were X, Y, and Z values for three sensor data streams: accelerometer (ACC), gravity (GRAV) and orientation (ORIENT). These values were logged from power-on until the unlock widget was fully engaged. Each participant completed all 10 positions and directions of the unlock widget with each POSTURE in one contiguous block. The order of POSTURE was a 4 x 4 Latin-square. The data-gathering portion of the study was approximately 15 minutes per participant.

Results

In total, 640 power-on to unlock sequences were logged (40 per participant, 10 per POSTURE per participant). The logs revealed that sensors sometimes needed to "warm up" after power-on (values were all 0 otherwise), so we use the first stable X, Y, Z values updated after power-on (usually within 100 ms of power-on). The mean time of logged sequences was 1403 ms (SD 736).

Distribution of Sensor Readings at Power-on and Unlock. We examined the distribution of each component for each sensor reading at power-on (the start of the movement) and unlock (the end of the movement). If a sensor reading was sufficient to classify postures, we would expect to see a visual separation of the values by hand posture when the sensor readings were plotted (Figure 4). Although we found no clear separation between LEFT and RIGHT HANDS, or SINGLE and CRADLED HOLDS, GRAV-Y showed some minimal separation by HAND. Previous work has essentially used thresholds on values like these to detect hands, so we explore this possibility in the next section.



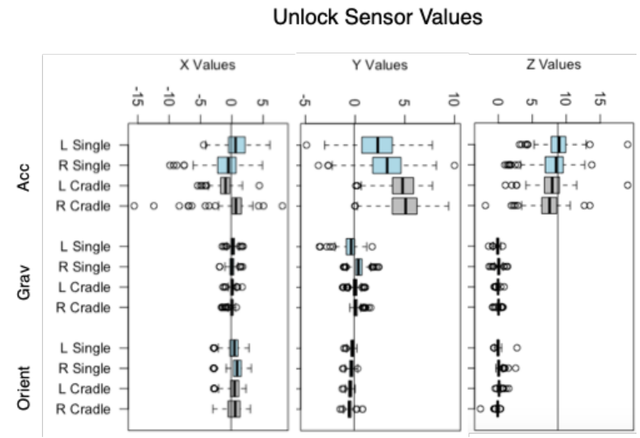


Figure 4: Key sensor component readings at power-on and unlocking for each POSTURE.

Sequence of Sensor Readings from Power-on to Unlock. Visual inspection of the ORIENT X, Y, and Z sensor values during a typical unlock sequence suggests a subtle, but distinguishable pattern (see a comparison of single trials example in Figure 5). The Y and Z values converge for LEFT-SINGLE, but diverge for RIGHT-SINGLE. The X and Z values are offset but similar for RIGHT-CRADLE, but notice-able diverge at the beginning of the trial for LEFT-CRADLE. These kinds of observations suggest pre-unlock motion sequences differ, according to which hand is holding the device. These are the types of patterns we leverage for classification.

4 CLASSIFICATION MODELS

We test two methods to detect which hand is holding a phone using the motion data from our experiment; simple thresholds, based on measuring sensor values at key points, and dynamic-time warping, based on the full sequence of sensor values between power on and device unlock.

Simple Thresholds

Our first attempt at classification looked at the distribution of sensor values at power-on and unlock, to determine if any single set of sensors could be used to classify hand POSTURE at a given point-in-time. We tested simple thresholds for each sensor value (ACC-X, ACC-Y, . . .). Thresholds were determined as a “midpoint” between hand POSTURE distributions. Specifically, the mean of the third-quartile of the lowest factor, and the first quartile of the highest factor. This threshold was used to classify the POSTURE using two-way cross validation, and the analysis was repeated using each sensor’s X, Y, and Z. This failed to produce useful results: the best POSTURE detection was below chance. Collapsing SINGLE and CRADLED postures together to detect which

Figure 5: Orientation sensor values during a typical unlock sequence: each graph represents a single trial.

HAND is holding the phone (i.e. LEFT or RIGHT hand regardless of HOLD) improved accuracy, but the results were poor, ranging from near chance (51.8% for ORIENT X) to 62.5% (for ACC X). For this reason, simple thresholds are deemed insufficient for HAND or POSTURE classification.

Dynamic Time Warping

In a typical pre-unlock sequence, we observed possible characteristic patterns depending on the POSTURE used (Figure 5). To classify sensor values over time, we use dynamic-time warping (DTW), a nearest-neighbor pattern-matching algorithm commonly used in speech [14] and gesture recognition [1, 2]. Using DTW, we compare a sequence of sensor values against labelled model data representing a known hand posture; the closest matching path returned from DTW is used to classify the hand posture.

We implemented DTW using Euclidean distances for each measurement in our calculations. One potential issue using DTW is that motion sequences vary in length; unlocking a phone can take a variable amount of time, so paths contain a variable number of points. To make paths easier for DTW to match, we first normalize them by interpolating the data over a 3 second window sampled at 100Hz. Normalizing to 3 seconds maintains a reasonable representation of all sequences (since the mean sequence time plus two standard deviations is 2.9 seconds.)

To determine what combination of sensor readings to use, we ran DTW with each sensor and each combination of X, Y, and Z values (i.e. testing ACC X, ACC Y, …, ACC XYZ, ORIENT X, ORIENT Y, …, ORIENT XYZ, GRAV X, GRAV Y, …). The best sensor combination, ORIENT X,Y,Z is used in subsequent results.

Generic Classification. A generic classifier would require no per-participant configuration in a real system. To test if a generic classifier is possible, we split our data evenly into across training and test sets regardless of participant. For each task in the training set, we generated sequences according to the combination of sensor readings being tests (e.g. ACC-X, ORIENT-XYZ, …). This provided us with a set of paths for each task that we could use as a model. Using DTW, we found the closest matching gesture and the POSTURE associated with this match was used for classification.

We tested using two-way cross validation, and generated confusion matrices for a four-class model (treating each POSTURE separately), as well as a binomial classifier (col-lapsing HOLD, so classes were LEFT and RIGHT hand only). The highest recognition rates were found using ORIENT Z (see four-class results in Table 1). Accuracy for a specific hand posture is low, with LEFT-CRADLE having the highest recognition rate at 46.0%. If we group single and cradled holds together, recognition rates improve to 55.3% for LEFT hand, and 79.4% for RIGHT hand detection, for a mean recognition rate of 67.4%. This is an improvement over using simple thresholds, but falls short of techniques like GripSense [4].

Our informal observations during the data gathering suggest individuals may have distinct pre-unlock motions, which reduces the feasibility of a generic model and indicates that a machine learning approach trained on the user would be

Table 1: General Model: Observed vs. Predicted Accuracy for Left, Right {L,R}, Single, Cradled {S,C} Postures

	LS	LC	RS	RC
Left-Single (LS)	**31.1**%	19.8%	24.5%	24.5%
Left-Cradle (LC)	14.0%	**46.0**%	20.0%	20.0%
Right-Single (RS)	9.5%	11.1%	**39.7**%	39.7%
Right-Cradle (RC)	9.5%	11.1%	39.7%	**39.7**%

Table 2: User-Specific Model: Observed vs. Predicted Accuracy for Left, Right {L,R}, Single, Cradled {S,C} Postures

	LS	LC	RS	RC
Left-Single (LS)	**63.0**%	15.2%	10.9%	10.9%
Left-Cradle (LC)	10.4%	**60.4**%	14.6%	14.6%
Right-Single (RS)	1.3%	5.9%	**46.4**%	46.4%
Right-Cradle (RC)	1.3%	5.9%	46.4%	**46.4**%

more appropriate. To test this possibility, we explore the use of user-specific models.

User-Specific Classification. We split data for each participant evenly into training and test sets, and used two-way cross validation to assess each participant's individual performance. We repeated this for every participant, and calculated the mean for each cell in our confusion matrix (see four-class results in Table 2).

Hand POSTURE detection is no greater than 63.0% (for LEFT-SINGLE hand detection) and would require further investigation. However, focusing on HAND detection alone greatly increases accuracy. By combining SINGLE and CRADLED classes, accuracy improves to 74.5% for LEFT hand and 92.8% for RIGHT hand, an overall accuracy of 83.6%. This is comparable to holding hand detection accuracy rates reported by GripSense [4].

A per-user model requires training data for each user. We examined how varying the size of the training set would impact our results. We re-ran our tests using DTW with leave-n-out cross-validation, and training set sizes of n = 2 to 8 samples. A training set size of 5 samples was optimal, with other sizes varying in accuracy between 76% (n=2) and 77% (n=8). This suggests as few as two training samples could achieve reasonable accuracy.

5 DISCUSSION

These are encouraging accuracy rates, but what are the implications for actual deployment?

The first consideration is what does it really mean to detect which hand is holding the phone when we do not know if the thumb of the same hand is used (single) or if the finger of the other hand is used (cradle). We argue that an interface compensating for single-handed use does not mean the interface becomes unusable while the phone is cradled and the other hand's finger is used. In fact, people often switch between single and cradle while continuing to hold the phone with the same hand [7]. The main implication is that interface adjustment techniques should be designed with this in mind. For example, if the system knows which hand is holding the phone, a menu of commands can be positioned on the correct side to be near that hand's thumb, but the rest of the interface may be designed to be used with both single and cradle holds.

Our attempt at creating a generic classifier was unsuccessful mainly because individuals have distinct pre-unlock motions. Therefore, a deployed system would require some kind of training and configuration stage where the user demonstrates a set of unlock motions with different postures to train the classifier. This would only need to be done once. Alternately, a system could collect per-unlock motion data and then ask the user to verify what their posture was. This would gradually build up a set of unlock motion paths. Since two examples of each posture provides 76% accuracy, this process would not be too arduous.

An issue with any detection is addressing classification errors. Even 84% accuracy means an incorrect classification about 1 in 7 times. If the goal is to adjust the interface for a particular hand, an error means it would be in a suboptimal state. One approach is to use the DTW distance metric to determine the detection confidence, to help decide when an interface adjustment should be made. A second approach is to turn this implicit pre-unlock gesture into an explicit one. Users could move their phone with a particular flourish to simultaneously distinguish between hands and trigger inter-face adjustment. This is related to error correcting motion gestures like double-flip [13]. In future work, we plan to investigate this and similar refinements.

Finally, we believe this type of pre-unlock detection is highly complementary to post-unlock detection techniques like GripSense. Our method cannot detect changes of hand after unlock, but on a basic level, our method could be used to condition a prior probability for the GripSense method to increase combined detection accuracy. This would benefit holding hand detection, but also other kinds of detection supported by GripSense since an a priori estimate of what hand is holding the phone can be exploited as a constraint.

6 CONCLUSION

We contribute a method to detect which hand is holding a smartphone by analyzing the built-in orientation sensor stream as the phone is lifted and unlocked, prior to first interaction, without the use of external sensors. Our insight is that we can treat the recognition problem as an implicit motion gesture [13], and use dynamic-time warping (DTW) to classify hand based on lift-and-unlock motions per-formed by the same person. With this setup, we can detect whether a phone is held in the left or right hand with 83.6% accuracy. Our method can be used standalone and achieve effective results with 2 to 5 training samples collected during during typical device usage, making user configuration and training in a real deployment feasible.

ACKNOWLEDGMENTS

Funding for this research was provided by the Natural Science and Engineering Research Council of Canada and the National Research Council of Canada. Experiments described in this paper were reviewed by the Office of Research Ethics at the University of Waterloo. We thank participants for volunteering for our study.

REFERENCES

[1] Donald J. Bemdt and James Clifford. 1994. Using Dynamic Time Warping to Find Patterns in Time Series.

[2] Alexander De Luca, Alina Hang, Frederik Brudy, Christian Lindner, and Heinrich Hussmann. 2012. Touch Me Once and I Know It's You!: Implicit Authentication Based on Touch Screen Patterns. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 987–996. https://doi.org/10.1145/2207676.2208544

[3] Mayank Goel, Alex Jansen, Travis Mandel, Shwetak N. Patel, and Jacob O. Wobbrock. 2013. ContextType: Using Hand Posture Information to Improve Mobile Touch Screen Text Entry. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, New York, NY, USA, 2795–2798. https://doi.org/10.1145/2470654.2481386

[4] Mayank Goel, Jacob Wobbrock, and Shwetak Patel. 2012. GripSense: Using Built-in Sensors to Detect Hand Posture and Pressure on Commodity Mobile Phones. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology (UIST '12)*. ACM, New York, NY, USA, 545–554. https://doi.org/10.1145/2380116.2380184

[5] Ken Hinckley, Seongkook Heo, Michel Pahud, Christian Holz, Hrvoje Benko, Abigail Sellen, Richard Banks, Kenton O'Hara, Gavin Smyth, and William Buxton. 2016. Pre-Touch Sensing for Mobile Interaction. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 2869–2881. https://doi.org/10.1145/2858036.2858095

[6] Daniel Hintze, Rainhard D. Findling, Muhammad Muaaz, Sebastian Scholz, and René Mayrhofer. 2014. Diversity in Locked and Unlocked Mobile Device Usage. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication (UbiComp '14 Adjunct)*. ACM, New York, NY, USA, 379–384. https://doi.org/10.1145/2638728.2641697

[7] Steven Hoober. 2013. How Do Users Really Hold Mobile Devices? http://www.uxmatters.com/mt/archives/2013/02/how-do-users-really-hold-mobile-devices.php

[8] Sunjun Kim, Jihyun Yu, and Geehyuk Lee. 2012. Interaction Techniques for Unreachable Objects on the Touchscreen. In *Proceedings of the 24th Australian Computer-Human Interaction Conference (OzCHI '12)*. ACM, New York, NY, USA, 295–298. https://doi.org/10.1145/2414536.2414585

[9] Markus Löchtefeld, Phillip Schardt, Antonio Krüger, and Sebastian Boring. 2015. Detecting Users Handedness for Ergonomic Adaptation of Mobile User Interfaces. In *Proceedings of the 14th International Conference on Mobile and Ubiquitous Multimedia (MUM '15)*. ACM, New York, NY, USA, 245–249. https://doi.org/10.1145/2836041.2836066

[10] Luke Mastin. 2013. Handedness Statistics. www.rightleftrightwrong.com/statistics.html

[11] Alexander Ng, John Williamson, and Stephen Brewster. 2015. The Effects of Encumbrance and Mobility on Touch-Based Gesture Interactions for Mobile Phones. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '15)*. ACM, New York, NY, USA, 536–546. https://doi.org/10.1145/2785830.2785853

[12] Chanho Park and Takefumi Ogawa. 2015. A Study on Grasp Recognition Independent of Users' Situations Using Built-in Sensors of Smartphones. In *Adjunct Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15 Adjunct)*. ACM, New York, NY, USA, 69–70. https://doi.org/10.1145/2815585.2815722

[13] Jaime Ruiz and Yang Li. 2011. DoubleFlip: A Motion Gesture Delimiter for Mobile Interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, New York, NY, USA, 2717–2720. https://doi.org/10.1145/1978942.1979341

[14] H. Sakoe and S. Chiba. 1978. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 26, 1 (February 1978), 43–49. https://doi.org/10.1109/TASSP.1978.1163055

[15] Khai N. Truong, Thariq Shihipar, and Daniel J. Wigdor. 2014. Slide to X: Unlocking the Potential of Smartphone Unlocking. In *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 3635–3644. https://doi.org/10.1145/2556288.2557044