Simple Touch Prediction with Built-In IMUs

Benedict Steuerlein
University of Stuttgart
Stuttgart, Germany
st111340@stud.uni-stuttgart.de

Felix Bühler
University of Stuttgart
Stuttgart, Germany
st117123@stud.uni-stuttgart.de

noch ein bis zwei doofe keywords finden

ABSTRACT

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium. Integer tincidunt. Cras dapibus. Vivamus elementum semper nisi. Aenean vulputate eleifend tellus. Aenean leo ligula, porttitor eu, consequat vitae, eleifend ac, enim. Aliquam lorem ante, dapibus in, viverra quis, feugiat a, tellus. Phasellus viverra nulla ut metus varius laoreet. Quisque rutrum. Aenean imperdiet. Etiam ultricies nisi vel augue. Curabitur ullamcorper ultricies nisi. Nam eget dui. Etiam rhoncus. Maecenas tempus, tellus eget condimentum rhoncus, sem quam semper libero, sit amet adipiscing sem neque sed ipsum. Nam quam nunc, blandit vel, luctus pulvinar

CCS CONCEPTS

• **Human-centered computing** → **User studies**; **Ubiquitous and mobile devices**;

KEYWORDS

Touch prediction; Smartphone; sensors; regression; deep learning; .

FIS'18, July 2018, Stuttgart, Germany

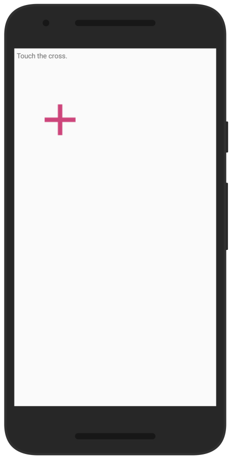

Figure 1: Touch task with one cross displayed.

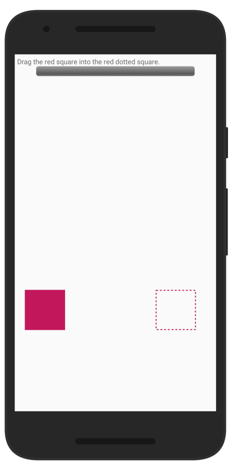

Figure 2: Fitts Law task with a progress bar displaying the current progress.

ACM Reference Format:
Benedict Steuerlein and Felix Bühler. 2018. Simple Touch Prediction with Built-In IMUs. In *Proceedings of Fachpraktikum Interaktive Systeme (FIS’18).* ACM, New York, NY, USA, 6 pages.

1 INTRODUCTION

Touch is the preferred input method on smartphones today. Current research and manufacturers are constantly trying to improve and enhance the interaction on smartphones. Enhancing smartphones with new, rich interaction methods allows users to operate their phone faster and more accurate, thereby increasing the usability and user-experience. One way can be the extension of interactable space with interaction possibilities on the back of device (BoD) [2, 8, 10]. E.g. unlocking the phone by using gestures on the backside, or explicit touches on the backside to reach for unreachable targets on the front side may be possible applications of this technique. However, some of these solutions do not conform with the form factors and weights of ordinary phones [3, 4, 14]. Another way to extend interaction is the introduction of additional touch gestures and touch recognition on touchscreens [6, 7, 9]. Previous work tried predicting touch positions on the touchscreen based on sensor data from either built-in or additional sensors [11, 12].
Accurately predicting touch positions could offer many use-cases:

Preloading Content: Preloading certain content that a user might request in the near future, e.g. a web page, reduces the waiting time and thus improves the user experience. Removing the latency and thereby getting rid of the delay between certain actions greatly enhances the usability of a system. However, when navigating on websites and preloading content from small links a high and precise accuracy is required. Inaccurate prediction would require to preload more data around the predicted point.

Highlighting Objects: When navigating through folders, predicted future positions of a touch can be used to highlight information about the to be touched folders. Navigation through a picture gallery on the phone can slightly enlarge the to be touched pictures to give a small preview, maybe showing picture information of where and when the picture was taken. The accuracy required for this approach does not necessarily have to be very high as elements are usually larger than small weblinks.

Despite its novelties, accuracy of some of these solutions has been low. In this paper, motivated by related work and its unexploredness, we present a machine learning model for predicting touches on smartphones using only the phones’ built-in sensors.

Likewise, traditional touch events fire at the moment one makes contact with the digitizer, yet the genesis of the grasping or aiming movement comes much earlier, and originates away from the screen itself.

2 RELATED WORK

This paper explores the applicability of touch prediction on smartphones with built-in sensors. We considered related work that extends or improves everyday interaction and therefore looked into work that tries to adapt touch interaction in the field of mobile interaction.

An addition to the interaction enhancement through e.g. BoD elements could be the integration of the internal sensors of smartphones, as modern telephones are shipped with them by default. Goel et al. [5] presented *GripSense*, a system that can imply pressure and infers hand postures on phones based on inertial sensor measurements by the gyroscope with an accuracy of 84.3%. To reduce the noticeable latency of continuous motion on touchscreens, Le et al. [11] introduced *PredicTouch*, a system consisting of three external IMUs attached to the wrist, the finger, and to a stylus in order to predict where users will continue their motion on touchscreens in the near future. Using a combination of IMU and a multi-layer feedforward neural network for regression, which was trained on touch coordinates with preceding data from the IMU, they were able to accurately predict touches 33ms and 66ms into the future. Additionally, user's throughput for finger input was increased by 15% and 17% for stylus input. Support was found for Banks et al. [1] approach of using self-capacitance touchscreen displays as a means for mobile interaction. Self-capacitive touchscreens capture fingers and their distance from the screen before touching it. Their contribution covers the applicability of multi-touch hover and grip, which is enabled by self-capacitance touchscreens, in common interaction scenarios. Motivated by overcoming the lack of input modalities, Mohd Noor et al. [12] presented *28 Frames Later*, a system that predicts future touch positions on smartphones. Based on grip data they gathered from a total of 24 capacitive sensors built inside the BoD and on the laterals while performing touches on the touchscreen they built a machine learning model that was able to predict touch positions $200ms$ before the actual touch with an offset of $18mm$ to the actual touch position. However their system required the 24 built on sensors which is not feasible for ordinary usage.

Some of the presented work require additional features for their approach to work. This includes, for example, having additional sensors mounted to the telephone or having special touch-sensitive displays. The normal use of today's smartphones does not support this kind of interaction elements. However, the use of internal sensors is a promising solution for this limitation as smartphones nowadays are equipped with them as standard. Confirmation of this can be found in the approaches mentioned above. We present a combination of a neural network in combination with the internal sensors to predict where one will touch in the near future.

Device	Release	Weight (g)	Diagonal (in)	Height (cm)	Width (cm)	Depth
S3 Mini	2012	113	4.	12.16	6.3	0.99
S4	2013	130	5.	13.7	7.0	0.79
Nexus 5X	2015	136	5.2	14.7	7.26	0.79
Nexus 6	2014	184	6.	15.93	8.3	1.01

Table 1: Data about the smartphones that were used in the study.

3 DATA COLLECTION STUDY

We conducted a data collection study to gather IMU data while performing touches on a smartphone. Our collected data set consists of 6 smartphone sensors that were sampled while participants performed successive touches on the smartphones front side. For our data collection study we used a repeated-measures design with one independent variable: PHONE, which was counterbalanced using Latin Balanced squares. The total amount of conditions was: PHONE = 4.

Apparatus

Our dataset was generated using four different sized smartphones on which participants had to perform a certain amount of touches (for further information see Section 3). The phones we used were a Samsung S3 Mini, a Samsung S4, a Google Nexus 5X, and a Motorola Nexus 6. For more technical details about the used devices see Table 1. Our used phone sizes range from 4′′ (S3) to 6′′ (N6). Using phones of these sizes we were able to cover the sizes of everyday smartphones, including some high-end devices and create a generalizable machine-learning model.

Tasks

For our data collection study participants had to touch points displayed as crosses in a 16×9 grid on the touchscreen (see Figure 1). To achieve a high variance, we randomized the positions of the crosses within all the cells. To avoid sequential effects, we randomized the order in which the crosses were displayed. There were a total of 3 repetitions, resulting in a total of $16 \times 9 \times 3 = 432$ touches on one device.

Between two touches our study participants had to perform a simple *Fitts’ Law task* (see Figure 2). Here participants had to drag a filled rectangle into a dashed contour of a rectangle. This task was mainly implemented to reset the participants grip to the bottom half of the device. Because a previous shifted grip of the hand to the upper half of the phone influences the recorded sensor data when reaching for the next target in the lower half and vice versa.

Procedure

Participants were either invited within the course *FIS’18* or orally. All appointments were discussed orally. After participants have arrived they signed a consent form. We then continued measuring their hand length. We asked the participants to take a seat on a chair without armrests and explained the study procedure and its sense. We started carrying out the study and handed out the first phone accordingly to the balanced Latin Square order. After participants finished the tasks (see Section 3) on the first phone, we asked them if they need a short recovery break and then continued with the next phones. Additionally, we allowed participants to rest and put away the phone during the *Fitt’s*

	S3M 0ms	**S4 0ms**	**N5X 0ms**	**N6 0ms**
RF	28.11	34.76	36.38	41.4
DT	28.81	35.58	37.03	42.56
KNN	28.67	34.61	36.36	42.91
GP	53.35	66.83	69.53	79.06
	S3M 33ms	**S4 33ms**	**N5X 33ms**	**N6 33ms**
RF	27.91	34.29	36.05	40.45
DT	28.39	35.1	35.03	41.52
KNN	28.41	34.31	35.92	42.13
GP	52.89	66.82	69.52	79.05
	S3M 66ms	**S4 66ms**	**N5X 66ms**	**N6 66ms**
RF	27.54	33.58	35.24	40.18
DT	27.9	32.74	34.67	39.54
KNN	28.1	33.99	35.93	41.62
GP	44.65	64.83	67.88	76.18

Table 2: Average euclidean distances (mm) for baseline regressors from scikit-learn[1].

Law task because we specifically deal with this task in our preprocessing step (see Section 4). The study duration was 54 minutes on average.

Participants

We invited 20 right-handed fellow students as participants (15 male, 5 female). Their age ranged between 21 and 27 ($M = 24.25$, $SD = 1.58$). We measured the hand length of participants. The size was measured from the tip of the middle finger to the wrist crease with fingers stretched out. Hand lengths ranged from $16.0cm$ to $21.3cm$ ($M = 19.3cm$, $SD = 1.47cm$). Our measured data covers samples from the 5th and 95th percentile of the anthropometric data reported in previous work [13].

4 RESULTS

Data Set & Preprocessing

Look at Table 2 We recorded a total of 1079 minutes of sensor data from the *touch* and *Fitt's Law task* task. Due to the high sampling rate of the sensors we first removed occurring duplicates by keeping the last sensor value and timestamp. We then up-sampled the sensors to 333.33 Hz, resulting in 1 sample given every 3ms. Finally, we saved 100 samples before each touch resulting in a total of 3.456.000 samples.

Up-sampling the sensors to 1 sample every 3ms resulted in a discontinuous function for each sensor axis. We have therefore tried applying several smoothing procedures.

Neural Network Structure

[2]

5 DISCUSSION

Disuses why it is still not awesome and how this could be improved. Why this is still awesome? Think about: Nobody has done this before.

Limitations

In this paper we focused on one specific use task where users touched randomized targets on smartphones while sitting in on a chair without armrests. The sensor samples generated during our study are specific for this use case and thereby do not cover ordinary smartphone usage and implied phone movement when for example walking or operating the phone in a train.

Limitations of using our machine learning approach

6 CONCLUSION

Two sentences wrap up what you have done. Than report what you achieved.

[1] https://scikit-learn.org/stable/supervised_learning.html#supervised-learning

[2] https://05.jupyter.interactionlab.io/user/beneste/tree/fapra-imu

REFERENCES

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

[2] Patrick Baudisch and Gerry Chu. 2009. Back-of-device interaction allows creating very small touch devices. In *Proceedings of the 27th international conference on Human factors in computing systems - CHI 09*. ACM Press, New York, New York, USA, 1923. https://doi.org/10.1145/1518701.1518995

[3] Christian Corsten, Bjoern Daehlmann, Simon Voelker, and Jan Borchers. 2017. BackXPress. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems - CHI '17*. ACM Press, New York, New York, USA, 4654–4666. https://doi.org/10.1145/3025453.3025565

[4] Alexander De Luca, Emanuel von Zezschwitz, Ngo Dieu Huong Nguyen, Max-Emanuel Maurer, Elisa Rubegni, Marcello Paolo Scipioni, and Marc Langheinrich. 2013. Back-of-device authentication on smartphones. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '13*. 2389. https://doi.org/10.1145/2470654.2481330

[5] Mayank Goel, Jacob Wobbrock, and Shwetak Patel. 2012. GripSense. In *Proceedings of the 25th annual ACM symposium on User interface software and technology - UIST '12*. 545. https://doi.org/10.1145/2380116.2380184

[6] Anhong Guo, Robert Xiao, and Chris Harrison. 2015. CapAuth. In *Proceedings of the 2015 International Conference on Interactive Tabletops & Surfaces - ITS '15*. 59–62. https://doi.org/10.1145/2817721.2817722

[7] Christian Holz, Senaka Buthpitiya, and Marius Knaust. 2015. Bodyprint. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI '15*. 3011–3014. https://doi.org/10.1145/2702123.2702518

[8] Huy Viet Le, Patrick Bader, Thomas Kosch, and Niels Henze. 2016. Investigating Screen Shifting Techniques to Improve One-Handed Smartphone Usage. In *Proceedings of the 9th Nordic Conference on Human-Computer Interaction - NordiCHI '16*. ACM Press, New York, New York, USA, 1–10. https://doi.org/10.1145/2971485.2971562

[9] Huy Viet Le, Thomas Kosch, Patrick Bader, Sven Mayer, and Niels Henze. 2018. PalmTouch: Using the Palm as an Additional Input Modality on Commodity Smartphones. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (2018-04-21) *(CHI'18)*. ACM, New York, NY, USA, 360:1–360:13. https://doi.org/10.1145/3173574.3173934

[10] Huy Viet Le, Sven Mayer, Patrick Bader, and Niels Henze. 2017. A smartphone prototype for touch interaction on the whole device surface. *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services - MobileHCI '17* (2017), 1–8. https://doi.org/10.1145/3098279.3122143

[11] Huy Viet Le, Valentin Schwind, Philipp Göttlich, and Niels Henze. 2017. PredicTouch. In *Proceedings of the Interactive Surfaces and Spaces on ZZZ - ISS '17*. 230–239. https://doi.org/10.1145/3132272.3134138

[12] Mohammad Faizuddin Mohd Noor, Simon Rogers, and John Williamson. 2016. Detecting Swipe Errors on Touchscreens using Grip Modulation. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI '16*. ACM Press, New York, New York, USA, 1909–1920. https://doi.org/10.1145/2858036.2858474

[13] Alan Poston. 2000. Human engineering design data digest. *Department of Defense Human Factors Engineering Technical Advisory Group Washington* (2000), 82. https://www.acq.osd.mil/rd/hptb/hfetag/products/documents/HE_Design_Data_Digest.pdf

[14] Katrin Wolf, Christian Müller-Tomfelde, Kelvin Cheng, and Ina Wechsung. 2012. PinchPad. In *Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction - TEI '12*. ACM Press, New York, New York, USA, 103. https://doi.org/10.1145/2148131.2148155