Finding the Sweet Spot: Analyzing Unrestricted Touchscreen Interaction In-the-Wild

Sven Mayer[1,2], **Huy Viet Le**[2], **Markus Funk**[2,3], **Niels Henze**[4]
[1] Human-Computer Interaction Institute, Carnegie Mellon University, PA, US, info@sven-mayer.com
[2] University of Stuttgart, Stuttgart, Germany, {firstname.lastname}@vis.uni-stuttgart.de
[3] Cerence Inc., Ulm, Germany, markus.funk@cerence.com
[4] University of Regensburg, Regensburg, Germany, niels.henze@ur.de

ABSTRACT

With smartphones being a prime example, touchscreens became one of the most widely used interface to interact with computing systems. Compared to other touchscreen devices, smartphones pose additional challenges as the hand that interacts with the device is commonly used to also hold the device. Consequently, determining how fingers of the hand holding the device can interact with the screen is a non-trivial challenge. A body of recent work investigated the comfortable area in controlled lab studies. This poses limitations as it is based on the assumption that the grips used in the studies are representative for normal smartphone use. In this paper, we extend previous work by providing insights from in-the-wild studies using two different apps that were deployed in the Android App Store. Comparing our results with previous work we confirm that our data fits previously proposed models. Further analyzing the data, we highlight the sweet spot, the position that is touched if the input can be performed on the whole screen.

Author Keywords

Sweet spot; in-the-wild study; mobile device.

CCS Concepts

•**Human-centered computing** → **Human computer interaction (HCI); Touch screens; Interaction techniques;** *User interface design;*

INTRODUCTION

Touchscreens enable designing especially intuitive systems as they combine input and output in a single interface. Over the last decade, touchscreens became one of the most widely used input devices for computing systems. Especially the ongoing success of current mobile devices lead to mass adoption. Compared to other touchscreen devices, such as tabletops and other stationary displays, mobile devices equipped with touchscreens pose additional challenges. Users typically use the hand that interacts with the screen to also hold the device [23].

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 the author(s) 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.

Proceedings of the 2019 ACM International Conference on Interactive Surfaces and Spaces (ISS '19) November 10–13, 2019, Daejeon, Republic of Korea

© 2019 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ISBN 978-1-4503-6891-9/19/11. . . $15.00

DOI: http://dx.doi.org/10.1145/3343055.3359705

Recent work showed how holding a smartphone in one hand and using the same hand to interact with the device restricts the movement of the hand's fingers [3, 20, 21]. Bergstrom-Lehtovirta and Oulasvirta [3] conducted a controlled lab study to determine what the authors call the *comfortable area* of the thumb [3]. They provide a model that describes the thumbs' movement range. Le et al. [20] confirm this work and extend it by describing the movement range of all five fingers. Le et al. [21] further extend their work, confirming their findings also for walking scenarios using a treadmill. Bergstrom-Lehtovirta and Oulasvirta [3], and Le et al. [20, 21], base their findings on highly controlled studies, using a small set of device sizes, while participants were in a calm environment.

Fundamental work by Bergstrom-Lehtovirta and Oulasvirta [3], and Le et al. [20, 21], can inform the placement of interaction controls when designing for optimal conditions. However, mobile interaction is characterized by very diverse contexts. Especially smartphones are used while walking, standing, and running and they are often used while performing other tasks. Smartphones' form factors are more diverse than what can be considered in a lab study. Thus, it is unclear how the proposed models perform when the interaction context is less restricted. In summary, lab studies yield a high internal validity but less external validity. Thus, in-the-wild studies are used to counteract this drawback [14, 17]. Examples to foster external validity are works by Henze et al. [15, 16] which showed how the precision of target section can be increased regardless of device size and context. Therefore, we study finger placement on mobile devices in two in-the-wild studies. We investigate how the thumbs can interact with a smartphone's touchscreen in a context outside of lab environments.

We conducted two in-the-wild studies by publishing games in mobile application stores. We collected 45,899,268 touch events from 607 devices. Both millions of touch events in potentially different setting as well as a wide range of devices contribute to the external validity. We extend prior work which showed that with larger mobile devices the average input is shifted more towards the left side of the device. However, for tablets, we observed the effect of the device getting too big for one-handed interaction, here our players started to also interact with the left hand. Further extending previous work, we highlight the *sweet spot*, the position that is touched if the interaction is not bound to specific areas.

RELATED WORK

Using games to understand how people interact with technology has a long history e.g. [11, 15, 26]. Thus we present related work in the domain of in-the-wild studies with a focus touch interaction. The second part focuses on smartphone input modeling as this is the core contribution of the paper.

Touch interaction in-the-wild

In-the-wild studies are commonly used for observation and ethnographic studies; however, with the availability of app stores such as the Google Play Store and the Apple App Store distributing apps to a large number of participants became a new form of in-the-wild studies especially in the human-computer interaction (HCI) domain.

The interaction with touchscreens has in particular been studied using in-the-wild deployments. Alt et al. [1], for example, conducted an in the wild study to better understand graphical passwords and Schneegass et al. [33] extended this work to counteract the smudge traces left on the screen by the finger through an in-the-wild study. Previous work also used in-the-wild deployments to improve the interaction with touchscreens. Poppinga et al. [31] used a mobile application to understand touchscreen gestures. Henze et al. [15] used an in-the-wild study to collect more then 100 million touch events to better understand how users interact with the screen and how to improve users' accuracy. The authors also used a mobile game to understand systematic errors when using on-screen keyboards and how to correct them [11]. Goguey et al. [10] used an in-the-wild study to model whole touch interfaces rather than just a simple touch.

In-the-wild studies conducted by publicly deploying apps and games in mobile application stores have increased our knowledge about specific types of applications and the interaction with smartphones in general. While the previous work focused on different aspects, the approach has successfully been used to learn and improve the interaction with touchscreens.

Modeling Smartphone Input

Studying and modeling smartphone input has a long tradition for various reasons. On the one hand, modeling the input can increase usability, e.g by increasing input accuracy [15], or reducing input lag [12, 13]. On the other hand, modeling the input can foster a better understanding and thus increase usability in the long run. Such an understanding can include challenges such as the fat-finger [4, 34] and the occlusion problem [37].

In this paper, we are mainly interested in modeling the finger and hand movements as the range of the fingers inform about graphical user interface (GUI) design. One important aspect when designing new GUIs is to consider possible ergonomic constraints. Le et al. [22] investigated smartphone grips and derived ergonomic implications for Back-of-Device (BoD) interaction. Lee et al. [23] modeled the index fingers touch areas also for BoD interaction. Eardly et al. [6, 7] studied changes in the grip during smartphone interaction. They revealed the effect of device size and target distance effects and proposed use cases for adaptive user interfaces. However, finger movements can also directly inspire new GUI layouts. Therefore, understanding the main input fingers, the index finger and the thumb, is important [27]. Park et al. [29] showed that the size of buttons influences the thumb's touch accuracy which is in line with challenges predicted by the fat-finger problem [34]. Additionally, Boring et al. [4] exploited the limitation of the thumb and proposed using it as a metric to simulate pressure. Moreover, Xiong et al. [38] found that interacting with small targets will cause fatigue problems.

Bergstrom-Lehtovirta and Oulasvirta [3] modeled the thumb's range on smartphones in one-handed interaction scenarios. The developed model predicts the thumb's range mainly based on the user's hand size and the position of the index finger. Beyond that, Trudeau et al. [35] modeled the motor performance in different flexion states. Finally, Le et al. [20, 21] studied fingers' range and comfortable area for all fingers in a one-handed interaction scenario both while sitting and walking on a treadmill. They used four different smartphone sizes ranging from 4.0″ to 5.96″ inches. The authors derived design guidelines to arrange buttons for BoD interaction based on their findings. However, as there was no real task performed on the front touchscreen, their design implications focus only on the placement of additional input controls for BoD interaction.

Summary

A large corpus of work used in-the-wild deployments and studied the interaction with touchscreens. Previous work showed that in-the-wild deployment can be used to study diverse questions. A common approach is to use playful tasks instead of the highly controlled and often synthetic tasks used in lab studies. Compared to lab studies, information from a diverse set of users and a much larger set of devices can be collected. Previous work focusing on modeling touchscreen input relied on small sets of users and potentially even more important, had to limit the number of devices. While previous work provided models that describe fingers' range and comfortable area [3, 20, 21], the insights are based on studies in calm environments without participants performing meaningful actions beyond the studies' tasks themselves.

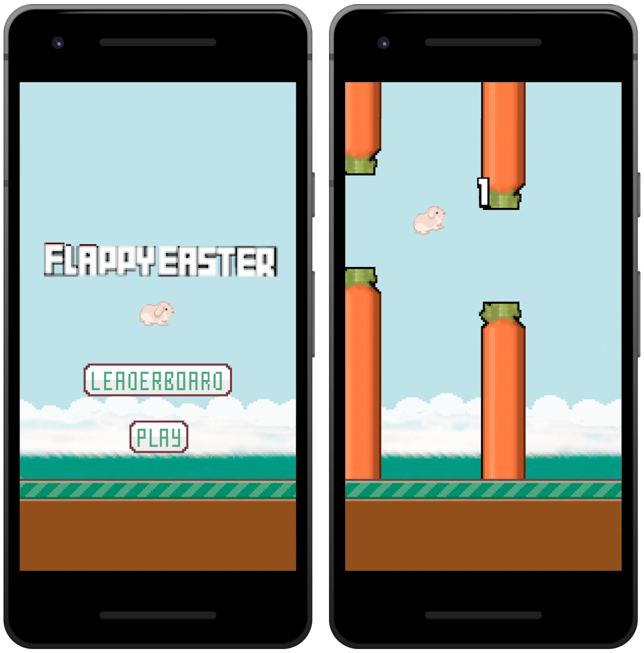
In this paper, we investigate scenarios where users touch the screen when they are not forced by the user interface (UI) to perform the input in a specific area of the screen. To consider diverse types of devices, we use in-the-wild deployments of two mobile games. Using the first game, we collect simple touch input and using the second game we collect swipe gestures. Thereby, we aim to understand where users interact with the touchscreens when the GUI does not require the user to touch a certain area.

GAME: FLAPPY BIRD

To understand how single tap input influences the area where users comfortable can tap the screen, we implemented a clone of Flappy Bird, see Figure 1. We used Flappy Bird as a study apparatus, as the game has the unique feature that the tap can be performed on the whole GUI. This allows the user to tap on the screen wherever it is most comfortable.

Game Design

The game consists of two screens, a start screen, and a play screen. The start screen contains a "Play" and a "Leaderboard"



(a) (b)

Figure 1. Two screenshots of the Flappy Bird game implementation, showing the game at the start (a) and during the game after the first pipe (b).

(a) (b)

Figure 2. Two screenshots of the 2048 game implementation, showing the game at the start (a) and during the game after a number of swipe gestures (b).

button, see Figure 1a. The "Leaderboard" button was implemented in a traditional way. Only the button area triggered the action; however "Play" can be triggered by touching not only the button itself but also the rest of the screen excluding the "Leaderboard" button. This was done to avoid participants center their finger towards the "Play" button which otherwise would have led to results with a bias towards the button. The "Leaderboard" was implemented using the Play Games Services[1] provided by Google. In the game phase, see Figure 1b, the bird needs to be steered tough the vertical pipes. The bird continuously moves to the right, while gravity pulls the bird down. To elevate the bird, players have to tap the screen, whenever the bird touches a pipe or the ground the player loses. The player scores one point per mastered pipe pair.

We released our version of Flappy Bird[2] using the Google Play Store to the Android eco-system. We asked for users consent when they started the app the first time to collect users touch events. This method to ask for consent is in line with previous work, e.g., Henze et al. [11], and Weber at al. [36].

Data Set

In total, the game was installed and started by 665 players who provided consent. As the game can be played without Internet connection, we only collected data from 459 players from which we observed at least one game. We manually decoded all devices with at least 300 games per player. Additionally, we removed all players with less than 100 touch inputs. In the following, we analyze the remaining 6,834,309 touch events produced by 2,111,501 tap gestures in a total of 186,676 games played by 386 players. A tap gesture had on average 2.99 touch events ($SD = 4.16$) with an average movement length of $0.86mm$ ($SD = 8.52$, $95\%CI = [0.85, 0.87]$). We used all touch events for our analysis to respect all movements performed while playing. Players played on various devices with a minimum screen size of $3.6''$ up to a maximum screen size of $10.5''$, see Figure 3. The collected locales and time zones show that there is a strong bias towards western countries among the players. In particular, the two most common locales are English USA (47.0%) and GB (23.6%). This is followed by Germany (7.9%), Brazil (4.9%), Canada (3.2%), France (3.0%), and others (9.9%).

GAME: 2048

In contrast to the first study where we focused on discrete touch interaction, the second study focuses on gesture input by analyzing directional swipe gesture.

Game Design

The game consists of a single screen. It always shows the play area which resembles a 4×4 grid, the game starts with two randomly selected grid cells (tiles) showing either a 2 or a 4, see Figure 2a. By swiping up, down, left, or right, the tiles slide in the direction of the swipe gesture. A tile stops sliding when it either collides with the end of the grid or with another tile with a number. If the different numbers collide, nothing happens. However, when two colliding numbers are

[1]https://developers.google.com/games/services/

[2]Our Flappy Bird clone called *Flappy Easter* is available in the Google Play Store at https://play.google.com/store/apps/details?id=de.makufunk.easterbunny

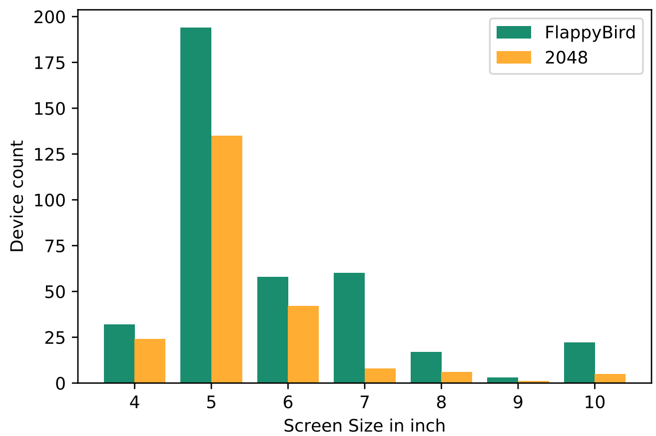


Figure 3. The screen size distribution of the 2048 and Flappy Bird players on which we base our analysis on.

the same, the values are added and displayed on one tile. After each swipe gesture (each turn of the user) a new tile appears and will randomly get assigned a 2 or 4. The player loses when there is no tile left without a number after the users' turn. The overall goal of the game is to collapse tiles until one tile displays the number 2048. However, the player can continue playing and therefore archive even larger numbers.

We implemented a clone of 2048, see Figure 2. We released 2048[3] using the Google Play Store. Again, we asked for users consent to collect users touch events in line with previous work (e.g., Henze et al. [11], and Weber at al. [36]).

Data Set

In total the game was installed, started, and played by 532 players who provided consent. We manually decoded all devices sizes with at least 100 games per player. Additionally, we removed all players that performed less than 100 gestures. In the following, we analyze the remaining 39,064,959 touch events produced by 4,921,290 swipe gestures from 221 players. The average length of a swipe gesture was $18.02mm$ ($SD = 10.60$, $95\%CI = [18.1, 18.03]$). Players played on various devices with a minimum screen size of $3.5''$ up to a maximum screen size of $10.1''$, see Figure 3. The collected locales and time zones show that there is a strong bias towards western countries among the players. In particular, the two most common locales are English USA (22.6%) and GB (10.4%). This is followed by Germany (16.3%), Russia (8.1%), France (4.1%), Vietnam (3.2%), Iran (3.2%), and others (32.1%).

RESULTS

From the two games, Flappy Bird and 2048, we collected a total of 45,899,268 touch events from 607 devices. Moreover, due to the design of the games, all touch events were performed while the phone was in portrait mode. Our goal is to extend the findings by Le et al. [20, 21] for the front screen using an in-the-wild setting. They studied how far users' fingers can reach and which finger positions are comfortable. While the study provides fundamental insights, the generalizability is limited by the use of four discrete device sizes, the lack of an interactive task participants had to perform, and the very specific environment in which the study has been conducted. In contrast, our data set contains data from diverse devices and users interacted with their devices in-the-wild with a task focusing on front screen input. While users performed a task, they were free in where they perform the touch input which gives a unique setting to understand the preferred grip and resting position of the finger used for input.

[3]2048 is available the Google Play Store: https://play.google.com/store/apps/details?id=org.hcilab.projects.game2048

Screen		***Tap***		***Swipe***	
Size in inch	**Area**	**Area**	**%**	**Area**	**%**
$3 < s \leq 4$	41.2	16.2	39.3	23.2	56.3
$4 < s \leq 5$	60.7	19.5	32.1	29.3	48.2
$5 < s \leq 6$	83.6	23.9	28.5	33.6	40.2
$6 < s \leq 7$	122.7	23.9	19.5	22.4	18.2
$7 < s \leq 8$	164.1	23.6	14.4	36.9	22.5
$8 < s \leq 9$	217.	7.2	3.3	31.9	14.7
$9 < s \leq 10$	258.1	27.2	10.5	32.6	12.6
$10 < s \leq 11$	307.6	31.9	10.4	23.4	7.6

Table 1. The touched areas for the different screen size both for tap and swipe input. Area measurements are reported in cm^2 per $1''$ step.

Screen Size Annotation

While Android provides app-developers with the model and manufacturer name, as well as the width and height of the screen in pixels, the Android OS does not provide the developer with the physical screen size. However, to compare different smartphone models it is essential to have the physical dimensions of the screen. Two researchers coded all devices with the physical dimensions by looking up the model on the website of the manufactures.

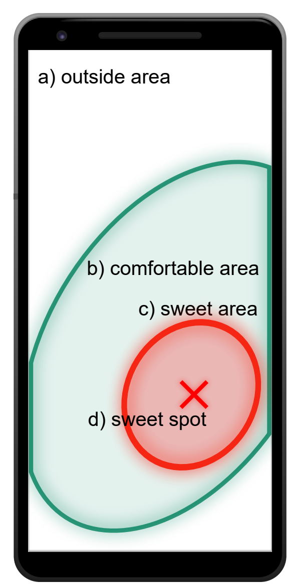


Figure 4. Sketch of the three areas we divide the screen into, the a) area outside of the comfortable area, the b) comfortable area, c) the *sweet area*, and d) the *sweet spot*.

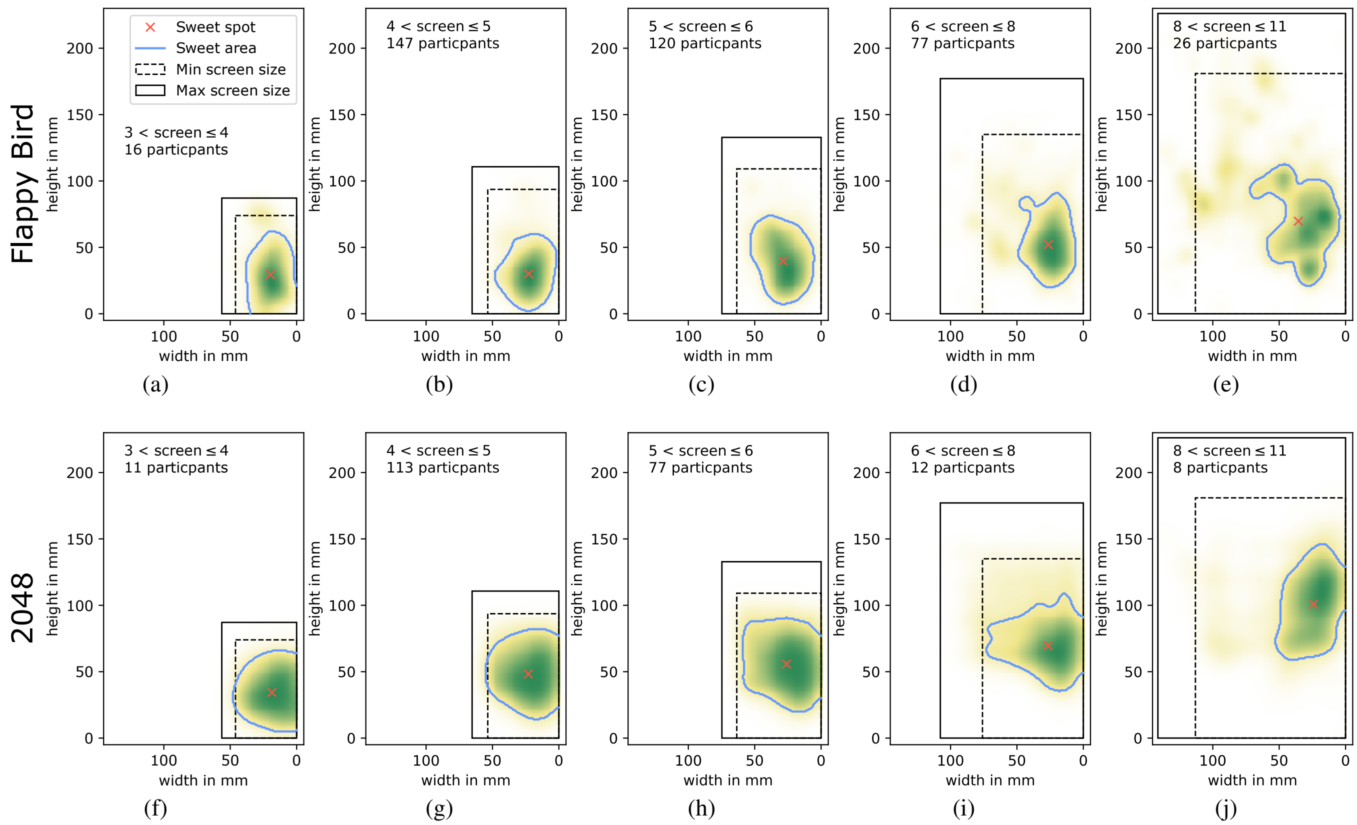


Figure 5. In the top row are the Flappy Bird heat map results and in the bottom row are the results of the 2048 touch events for five different screen size brackets ranging all the way from $3.5''$ up to $10.1''$. The blue line around the green hotspot indicates the *sweet area*. The red $\times$ represents the centroid of the blue outline, thus the *sweet spot*. Device sizes are indicated by black dashed lines for the minimum screen size and solid black lines for the maximum screen size.

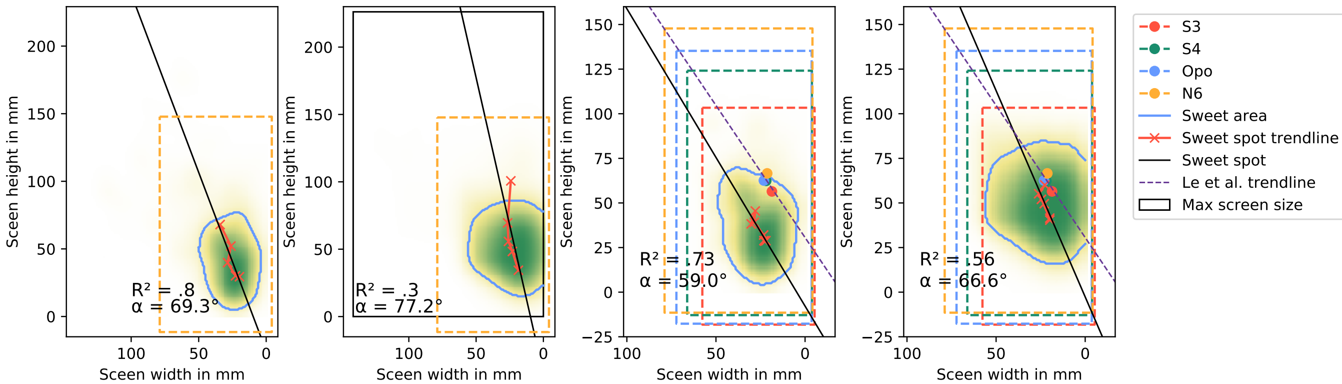
Preprocessing

To make the results comparable to previous work, we adopted the preprocessing pipeline form Le et al. [20, 21]. However, as they handled motion tracking data and our data is touch data, our filter steps are different. Original filtering was substituted with an average and a Gaussian filter. We first transformed the touch data retrieved from the games from pixels into metric units (*mm*). From there we generated heat maps for each participant with a raster size of $1 \times 1mm$ which therefore represents the distribution where participants performed the touch input. We normalized the heat maps for each participant to give all participants the same weight independent of the amount of data they contributed. We aligned the different devices to their lower right corner and group them by screen size. Afterward, we summed the normalized heat maps up to derive the average distribution, see Figure 5. To calculate the *sweet area* and *sweet spot* (see Figure 4), we first applied a mean filter ($< M + 3SD$) to reduce outliers and then a Gaussian filter ($sigma = 2$ and $order = 0$) to reduce noise. This was followed by the find contours algorithm [25] to determine the *sweet area* as done in prior work to determine the comfortable areas. From the *sweet area* we determined the center of mass which results in the *sweet spot*, see Figures 5 and 6.

Sweet Area

For the *sweet area* obtained from the heat maps using the find contours algorithm we calculated the area size of the sweet spots per screen size, see Figure 5 and Table 1. We then performed a paired t-test to compare the size of the *sweet area* in the *Tap* and *Swipe* conditions. There was a significant difference in the size of the *sweet area* for the two conditions *Tap* ($M = 23.6$, $SD = 6.4$) and *Swipe* ($M = 30.1$, $SD = 3.7$); $t(5) = -2.779$, $p < .05$.

Le et al. [20, 21] showed for four smartphones ranging from $4.0''$ to $5.96''$ inches the area which humans can comfortably reach on front screens of the phones while sitting and walking. We show that for similar phones the actual area is always smaller when participants are not forced to tap or swipe on a specific UI position, see Figure 5. Moreover, we found that the *sweet areas* in our observations are always within the comfortable areas. We performed a paired t-test to understand if the *sweet areas* are significantly smaller than the comfortable areas for the screen sizes from $3.5''$ to $6''$. The t-test revealed the *sweet areas* for both conditions *Tap* and *Swipe* are significantly smaller than the corresponding comfortable area, $t(4) = 5.893$, $p < .001$ and $t(4) = 2.871$, $p < .028$, respectively. Therefore we argue that this is the *sweet area*.



(a) Flappy Bird (zoomed-out) (b) 2048 (zoomed-out) (c) Flappy Bird (zoomed-in) (d) 2048 (zoomed-in)

Figure 6. The *sweet area* and the *sweet spot* in the two zoomed-out visualization are showing all phone sizes ranging from 3.5″ to 10.1″, while the zoomed-in versions are only showing devices between 3.5″ and 6.0″ enabling comparability to findings by Le et al. [20, 21].

Sweet Spot Position

Similar to Le et al. [20, 21] we calculated the centroid of the *sweet area*, see Figure 5, to estimate the most preferred input position, the *sweet spot*. Finally, to understand the relation between device size and *sweet spot* we perform a line regression. The regression revealed a possible trend that the *sweet spot* shifts towards the upper left corner with increasing device size, see Figure 6. For the full device range from 3.5″ to 10.1″ the *sweet spot* movement can be modeled with a line with an average R^2 of .55. Here the *sweet spot* shifted with 73.3°. Moreover, for smaller devices (from 3.5″ to 6.0″) the average R^2 is .65 and the angle is 64.5°. These results are in line with findings by Le et al. [20] who reported a R^2 of .67 and an angle of 61.1° and Le et al. [21] were they reported a R^2 of .77 with 59.3° for sitting and R^2 of .88 and 61.1° for walking.

DISCUSSION

In contrast to previous work, we conducted in-the-wild studies to understand how smartphone users interact with their devices. While this allows us to understand various situations which users might be in, we can not ensure that they covered all possible situations. In our analysis, we combine the touch event from 607 different users which result in a wide range of devices. While typical user interfaces force the user to interact with specific on-screen elements on the GUI, in our implementation users are free to tap and swipe at any position. This enables to understand where users want to interact with their device when no UI element required touching a specific point. From the raw touch events, we generated heat maps from which we extract the commonly used areas for interaction for each device size. Le et al. [20, 21] presented insight on the comfortable area, which is defined as the area which can be reached without a grip change. By combining the user specific heat maps we get a similar representation of the touched area; however, our results show that the area is significantly smaller than the comfortable areas. Thus, we argue that while Le et al. [20, 21] explored the full comfortable area, our results represent the area where users prefer to interact in. Thus, we call this region the *sweet area*.

From the *sweet area*, we used the centroid to calculate the *sweet spots* for each screen size, in line with Le et al. [20, 21]. While our regression R^2 values are low, they are in line with related work for the shift of the comfortable area. Le et al. [21] reported mean R^2 of 60.2 across their sitting and walking conditions. In Figure 6, we highlight how our observations relate to the studies in controlled environments presented by Le et al. [21]. The results of the regression indicate a slight trend of the *sweet spot* possibly moving towards the devices' upper left corner. This, however needs to be confirmed in a separate study. Moreover, we found that the position where the offset vector for touch input is zero, as shown by Henze et al. [15], is close to the *sweet spots*. However, as Holz and Baudisch [18] showed, the touch input itself is multi-dimensional; thus, we argue that the offset zero position is not the same as the *sweet spot* but located in the same area.

While techniques like ThumbSpace [19] or RayForce [5], offer a way to overcome the reachability problem, they do not solve the underlying problem of interactive elements placed outside of the reach. However, we provide designers and developers with the fundamentals for the thumb *sweet spots* for a wide range of devices. Thus, we provide a better understanding to design more user-friendly GUIs in the first place.

In our setting, we controlled that players only used the games in portrait mode as this is the dominant phone orientation mode [32]. Moreover, this covers the most common grips [24]. We were not able to control with which hand or finger the users interacted with the devices during playtime. However, we argue that around 80% of the world population is right-handed [28, 30] and our results are in line with previous work which only had right-handed participants interacting with their right thumb [20, 21]. This leads us to believe that either most users use their phone one-handed and interact with the thumb or that the interaction areas are located at the same position for the second most common grip where the user holds the phone with the left hand and interacts with the index finger of the right hand.

Limitations

When choosing the games, we picked games where the number of buttons which could potentially influence the grip is low. Moreover, during gameplay no buttons need to be pressed, lowering the impact of the GUI to a very minimum. However, we see the potential that the players may not be aware of this fact and possibly change the grip anyhow. Moreover, the visual representations of the game itself could have potentially influenced the grip. Here, we argue that the results for our smaller devices are in line with previous work [20, 21]. Moreover, the results for our games are similar considering that the interaction (tap vs swipe gestures) is different. Thus, the effect of the GUI has to be minimal.

In this work, we only investigate portrait mode as the grip is fundamentally different from the grip in landscape mode. We choose to study only portrait mode as Sahami Shirazi et al. [32] showed that this is the more common mode. They further show that landscape is most used for content consumption such as video and images. And this is true even though that 84.5% of activities can be oriented according to the users needs [8]. Thus, landscape mode is less interesting to study as a first step. However, to build a full understanding of possible grips and the *sweet spot* this will be a next step to research.

We informed participants when they started the game that we record the touch events. However, participants did not have the immediate chance to ask questions. However, as an implication of our large-scale study, inviting every participant into the lab is not feasible. To still comply with the GDPR rules, no personal informations can be recorded. Thus, the data recorded is fully anonymized. However, this poses the limitation that we can not draw conclusions from the demographic sample such as age and gender. Both games have been uploaded to the Google Play Store to address a wide range of potential users. Moreover, the games have been advertised on Facebook which potentially skewed the user-base towards younger people. Moreover, the potentially young user base may also have effected the context in which they interacted with the games. Additionally, the fact that we used games in both studies might have eliminated certain contexts such as office environments.

Today's mobile devices have various body and screen sizes as well as different aspect ratios. As a second implication of the GDPR rules, we also can not record the handedness of the participants. Thus, we cannot ensure that all users used their right hand. We argue that today's devices mostly rely on right-handed GUI and moreover, around 80% of the world population is right-handed [28, 30]. Thus, the presented results are limited to right-handed usage. We hypothesize that the mirrored results are true for left-handed interaction, e.g. the trendline in Figure 6 is from lower left to upper right. This, however, needs to be studied in future work in more detail.

CONCLUSION

Using observations from 45.8 million touch events from 607 users we showed that smartphone input is device size dependent. This needs to be taken into account when designing smartphone interfaces. In detail, we showed that for both tap and swipe input, the sweet spot shifts gradually towards the upper right corner with increasing device size. Guided by our observations, we present a model which divides the screen into three different areas and fundamentally extends previously presented models and heuristics for UI element positioning.

This paper presents the first step towards understanding touch input of many different device sizes with a huge number of environmental influences. However, while in-the-wild studies can support a more extensive variety, they often lack the possibility of controlled environmental influences which would be possible in a lab study. Thus, to minimize the effect of environmental influences, these should be taken into consideration in future work. In particular, taking the activity and grip into account using the smartphones IMU sensor data with a grip classifier [9] or an activity recognition algorithm [2] are the next steps. Moreover, as this will allow calculating the *sweet spot* per-user and directly inform the GUI. This, therefore, is the next step towards user-based adaptation for more user-friendly designs.

ACKNOWLEDGMENTS

This work is financially supported by the German Research Foundation (DFG) within Cluster of Excellence in Simulation Technology (EXC 310/2) at the University of Stuttgart.

REFERENCES

[1] Florian Alt, Stefan Schneegass, Alireza Sahami Shirazi, Mariam Hassib, and Andreas Bulling. 2015. Graphical Passwords in the Wild: Understanding How Users Choose Pictures and Passwords in Image-based Authentication Schemes. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '15)*. ACM, New York, NY, USA, 316–322. DOI:http://dx.doi.org/10.1145/2785830.2785882

[2] Alvina Anjum and Muhammad U. Ilyas. 2013. Activity recognition using smartphone sensors. In *IEEE 10th Consumer Communications and Networking Conference (CCNC '13)*. 914–919. DOI: http://dx.doi.org/10.1109/CCNC.2013.6488584

[3] 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

[4] Sebastian Boring, David Ledo, Xiang 'Anthony' Chen, Nicolai Marquardt, Anthony Tang, and Saul Greenberg. 2012. The Fat Thumb: Using the Thumb's Contact Size for Single-handed Mobile Interaction. In *Proceedings of the 14th International Conference on Human-computer Interaction with Mobile Devices and Services (MobileHCI '12)*. ACM, New York, NY, USA, 39–48. DOI:http://dx.doi.org/10.1145/2371574.2371582

[5] Christian Corsten, Marcel Lahaye, Jan Borchers, and Simon Voelker. 2019. ForceRay: Extending Thumb Reach via Force Input Stabilizes Device Grip for Mobile

Touch Input. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. ACM, New York, NY, USA, Article 212, 12 pages. DOI:http://dx.doi.org/10.1145/3290605.3300442

[6] Rachel Eardley, Steve Gill, Anne Roudaut, Stephen Thompson, and Joanna Hare. 2016. Investigating How the Hand Interacts with Different Mobile Phones. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct (MobileHCI '16)*. ACM, New York, NY, USA, 698–705. DOI: http://dx.doi.org/10.1145/2957265.2961840

[7] Rachel Eardley, Anne Roudaut, Steve Gill, and Stephen J. Thompson. 2017. Understanding Grip Shifts: How Form Factors Impact Hand Movements on Mobile Phones. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 4680–4691. DOI: http://dx.doi.org/10.1145/3025453.3025835

[8] Umar Farooq and Zhijia Zhao. 2018. RuntimeDroid: Restarting-Free Runtime Change Handling for Android Apps. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services (MobiSys '18)*. ACM, New York, NY, USA, 110–122. DOI: http://dx.doi.org/10.1145/3210240.3210327

[9] 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. DOI: http://dx.doi.org/10.1145/2380116.2380184

[10] Alix Goguey, Géry Casiez, Andy Cockburn, and Carl Gutwin. 2018. Storyboard-Based Empirical Modeling of Touch Interface Performance. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 445, 12 pages. DOI: http://dx.doi.org/10.1145/3173574.3174019

[11] Niels Henze. 2012. Hit It!: An Apparatus for Upscaling Mobile HCI Studies. In *CHI '12 Extended Abstracts on Human Factors in Computing Systems (CHI EA '12)*. ACM, New York, NY, USA, 1333–1338. DOI: http://dx.doi.org/10.1145/2212776.2212450

[12] Niels Henze, Markus Funk, and Alireza Sahami Shirazi. 2016. Software-reduced Touchscreen Latency. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '16)*. ACM, New York, NY, USA, 434–441. DOI: http://dx.doi.org/10.1145/2935334.2935381

[13] Niels Henze, Sven Mayer, Huy Viet Le, and Valentin Schwind. 2017. Improving Software-reduced Touchscreen Latency. In *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '17)*. ACM, New York, NY, USA, Article 107, 8 pages. DOI: http://dx.doi.org/10.1145/3098279.3122150

[14] Niels Henze and Martin Pielot. 2013. App Stores: External Validity for Mobile HCI. *interactions* 20, 2 (March 2013), 33–38. DOI: http://dx.doi.org/10.1145/2427076.2427084

[15] Niels Henze, Enrico Rukzio, and Susanne Boll. 2011. 100,000,000 Taps: Analysis and Improvement of Touch Performance in the Large. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services (MobileHCI '11)*. ACM, New York, NY, USA, 133–142. DOI:http://dx.doi.org/10.1145/2037373.2037395

[16] Niels Henze, Enrico Rukzio, and Susanne Boll. 2012. Observational and Experimental Investigation of Typing Behaviour Using Virtual Keyboards for Mobile Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 2659–2668. DOI: http://dx.doi.org/10.1145/2207676.2208658

[17] Niels Henze, Alireza Sahami Shrazi, Albrecht Schmidt, Martin Pielot, and Florian Michahelles. 2013. Empirical Research through Ubiquitous Data Collection. *Computer* 46 (06 2013), 74–76. DOI: http://dx.doi.org/10.1109/MC.2013.202

[18] Christian Holz and Patrick Baudisch. 2011. Understanding Touch. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, New York, NY, USA, 2501–2510. DOI:http://dx.doi.org/10.1145/1978942.1979308

[19] Amy K. Karlson and Benjamin B. Bederson. 2007. ThumbSpace: Generalized One-Handed Input for Touchscreen-Based Mobile Devices. In *Human-Computer Interaction – INTERACT 2007*. Springer Berlin Heidelberg, Berlin, Heidelberg, 324–338.

[20] Huy Viet Le, Sven Mayer, Patrick Bader, and Niels Henze. 2018. Fingers' Range and Comfortable Area for One-Handed Smartphone Interaction Beyond the Touchscreen. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 31, 12 pages. DOI:http://dx.doi.org/10.1145/3173574.3173605

[21] Huy Viet Le, Sven Mayer, Benedict Steuerlein, and Niels Henze. 2019. Investigating Unintended Inputs for One-Handed Touch Interaction Beyond the Touchscreen. In *Proceedings of the 21st International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '19)*. ACM, New York, NY, USA, 14. DOI:http://dx.doi.org/10.1145/3338286.3340145

[22] Huy Viet Le, Sven Mayer, Katrin Wolf, and Niels Henze. 2016. Finger Placement and Hand Grasp During Smartphone Interaction. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '16)*. ACM, New York, NY, USA, 2576–2584. DOI: http://dx.doi.org/10.1145/2851581.2892462

[23] Songil Lee, Gyouhyung Kyung, Jungyong Lee, Seung Ki Moon, and Kyoung Jong Park. 2016. Grasp and index finger reach zone during one-handed smartphone rear interaction: effects of task type, phone width and hand length. *Ergonomics* 59, 11 (2016), 1462–1472. DOI: http://dx.doi.org/10.1080/00140139.2016.1146346

[24] Florian Lehmann and Michael Kipp. 2018. How to Hold Your Phone When Tapping: A Comparative Study of Performance, Precision, and Errors. In *Proceedings of the 2018 ACM International Conference on Interactive Surfaces and Spaces (ISS '18)*. ACM, New York, NY, USA, 115–127. DOI: http://dx.doi.org/10.1145/3279778.3279791

[25] William E. Lorensen and Harvey E. Cline. 1987. Marching Cubes: A High Resolution 3D Surface Construction Algorithm. In *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '87)*. ACM, New York, NY, USA, 163–169. DOI: http://dx.doi.org/10.1145/37401.37422

[26] Sven Mayer, Lars Lischke, Jens Emil Grønbæk, Zhanna Sarsenbayeva, Jonas Vogelsang, Paweł W. Woźniak, Niels Henze, and Giulio Jacucci. 2018. Pac-Many: Movement Behavior when Playing Collaborative and Competitive Games on Large Displays. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 539, 10 pages. DOI: http://dx.doi.org/10.1145/3173574.3174113

[27] Emiliano Miluzzo, Alexander Varshavsky, Suhrid Balakrishnan, and Romit Roy Choudhury. 2012. Tapprints: Your Finger Taps Have Fingerprints. In *Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services (MobiSys '12)*. ACM, New York, NY, USA, 323–336. DOI: http://dx.doi.org/10.1145/2307636.2307666

[28] R.C. Oldfield. 1971. The assessment and analysis of handedness: The Edinburgh inventory. *Neuropsychologia* 9, 1 (1971), 97 – 113. DOI: http://dx.doi.org/https://doi.org/10.1016/0028-3932(71)90067-4

[29] Yong S. Park and Sung H. Han. 2010. One-handed thumb interaction of mobile devices from the input accuracy perspective. *International Journal of Industrial Ergonomics* 40, 6 (2010), 746 – 756. DOI: http://dx.doi.org/10.1016/j.ergon.2010.08.001

[30] Ira B. Perelle and Lee Ehrman. 1994. An international study of human handedness: The data. *Behavior Genetics* 24, 3 (01 May 1994), 217–227. DOI: http://dx.doi.org/10.1007/BF01067189

[31] Benjamin Poppinga, Alireza Sahami Shirazi, Niels Henze, Wilko Heuten, and Susanne Boll. 2014. Understanding Shortcut Gestures on Mobile Touch Devices. In *Proceedings of the 16th International Conference on Human-computer Interaction with Mobile Devices & Services (MobileHCI '14)*. ACM, New York, NY, USA, 173–182. DOI: http://dx.doi.org/10.1145/2628363.2628378

[32] Alireza Sahami Shirazi, Niels Henze, Tilman Dingler, Kai Kunze, and Albrecht Schmidt. 2013. Upright or Sideways?: Analysis of Smartphone Postures in the Wild. In *Proceedings of the 15th International Conference on Human-computer Interaction with Mobile Devices and Services (MobileHCI '13)*. ACM, New York, NY, USA, 362–371. DOI: http://dx.doi.org/10.1145/2493190.2493230

[33] Stefan Schneegass, Frank Steimle, Andreas Bulling, Florian Alt, and Albrecht Schmidt. 2014. SmudgeSafe: Geometric Image Transformations for Smudge-resistant User Authentication. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '14)*. ACM, New York, NY, USA, 775–786. DOI: http://dx.doi.org/10.1145/2632048.2636090

[34] Katie A Siek, Yvonne Rogers, and Kay H Connelly. 2005. Fat finger worries: how older and younger users physically interact with PDAs. In *Human-Computer Interaction-INTERACT 2005*. Lecture Notes in Computer Science, Vol. 3585. Springer Berlin Heidelberg, Berlin, Heidelberg, 267–280. DOI: http://dx.doi.org/10.1007/11555261_24

[35] Matthieu B. Trudeau, Justin G. Young, Devin L. Jindrich, and Jack T. Dennerlein. 2012. Thumb motor performance varies with thumb and wrist posture during single-handed mobile phone use. *Journal of Biomechanics* 45, 14 (2012), 2349 – 2354. DOI: http://dx.doi.org/10.1016/j.jbiomech.2012.07.012

[36] Dominik Weber, Alexandra Voit, Jonas Auda, Stefan Schneegass, and Niels Henze. 2018. Snooze!: Investigating the User-defined Deferral of Mobile Notifications. In *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '18)*. ACM, New York, NY, USA, Article 2, 13 pages. DOI: http://dx.doi.org/10.1145/3229434.3229436

[37] 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

[38] Jinghong Xiong and Satoshi Muraki. 2014. An ergonomics study of thumb movements on smartphone touch screen. *Ergonomics* 57, 6 (2014), 943–955. DOI: http://dx.doi.org/10.1080/00140139.2014.904007