# A Practical Method to Eye-tracking on the Phone: Toolkit, Accuracy and Precision

Thanh Le
LMU Munich
Munich, Germany

Felix Dietz
University of the Bundeswehr Munich
Munich, Germany

Ken Pfeuffer
Aarhus University
Aarhus, Denmark

Florian Alt
University of the Bundeswehr Munich
Munich, Germany

## ABSTRACT

While eye-tracking has become a core asset for many computing environments, mobile phones, as a prime computing device, are lacking a practical platform to conduct eye-tracking studies and develop gaze applications easily. In this work, we aim to tackle this issue by investigating a system concept that allows for the deployment of remote eye-trackers for mobile devices. We describe a toolkit that supports eye-tracking in mobile apps based on a simple phone, PC, and remote eye tracker setup. We evaluate our approach through a technical evaluation of accuracy and precision in various user contexts important for mobility (sitting, standing, walking, lying). Our results show that eye-trackers can be easily used with high accuracy, and how it is impacted through body posture and motions of the user. Our work paves the way for enabling easy-to-use eye-tracking studies on mobile devices.

## CCS CONCEPTS

• **Human-centered computing → User studies**; **Ubiquitous and mobile computing systems and tools**.

## KEYWORDS

eye tracking, gaze detection, toolkit, accuracy, precision

**ACM Reference Format:**
Thanh Le, Felix Dietz, Ken Pfeuffer, and Florian Alt. 2022. A Practical Method to Eye-tracking on the Phone: Toolkit, Accuracy and Precision. In *21th International Conference on Mobile and Ubiquitous Multimedia (MUM 2022), November 27–30, 2022, Lisbon, Portugal*. ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3568444.3568463

## 1 INTRODUCTION

Eye-tracking has become a popular and relatively inexpensive way to improve computer interaction. Eye-tracking systems that are commercially accessible are reliable and offer important features to customers in industry and research, such as marketing studies, website analytics, and research investigations. They are unobtrusive and can be attached to laptops, mounted on a display, or integrated into head-mounted devices. As a result, there is growing interest in employing eye gazing data not only as a research tool but also to improve our daily interactions with computers [10].

Because of the great user experience given by direct touch and their high mobility, mobile phones have become one of the most popular computing devices today. The grasp and reach issue with mobile devices was investigated by researchers [1, 13, 18]. Users of smart phones, for example, frequently adopt a variety of grip styles, whether one-handed or two-handed. The thumb is technically free for input during grasp, but its reach is rather limited [1]. This can be improved by combining different input modalities. Explicit gaze input, such as pointing or gaze motions [15, 16], to attentive programs that use gaze to infer the user's intentions and improve the input with other modalities [12, 13] are examples of gaze-enabled applications and interaction strategies. However, only a few efforts conducted eye-tracking research on an actual mobile phone. Those usually involved highly complex technical setups, e.g., external motion tracking [8] and deep learning-based methods [17].

Surprisingly, less effort has been put into understanding the needs of such apps as they become more integrated into our daily smartphone usage. There is no standardized foundation on which designers and developers may build easy-to-use and reliable gaze-enabled applications. When designing applications for input modalities, such as touch, they have to face basic questions such as: Which region of the screen is easiest to interact with? What level of precision can we expect from the user's input? How does eye-tracking accuracy change with varying mobile phone usage / context?

Furthermore, prior research has discovered that tracking quality differs from data obtained in manufacturers' facilities, and that tracking accuracy (the offset from the true gaze point) and precision (the dispersion of the gaze points) vary greatly between tracking situations and users [3, 9]. However, few formal findings that could inform the design of gaze-enabled applications or interaction strategies have been drawn. Typically, research investigations strive to minimize variance to a minimum by carefully controlling tracking circumstances, user placement, recalibrating the eye tracker on a regular basis, and removing participants who do not track well [6]. While this is feasible in theory, it is not always viable in mobile contexts. To close this research gap, our contributions are as follows:

- We develop a toolkit specifically tailored to a mobile phone + PC + remote eye tracker system, that allows us to collect gaze information and integrate it into mobile apps on phones.


*MUM 2022, November 27–30, 2022, Lisbon, Portugal*

ACM ISBN 978-1-4503-9820-6/22/11…$15.00
https://doi.org/10.1145/3568444.3568463

- We present results of a user study that assesses the quality of eye-tracking in several mobile contexts such as sitting, standing, walking, and lying down while demonstrating the utility of our approach wrt. good accuracy, and how screen location and mobile context affect the accuracy.

## 2 RELATED WORK

Even though eye-tracking applications and interfaces have been extensively explored for a long time, only little work explored its feasibility on mobile phones. The mobile phone is a ubiquitous device with many potential users, and thus provides rich opportunities to employ eye-tracking. "Mobile" eye-tracking has often referred to the use of a head-worn device that enables the sensing of eye information [2], which implies different challenges. A head-worn eye-tracker can be used to track gaze on a mobile device, which involves high precision 3D tracking of the device in space in order to enable the transformation of the coordinate system of eye-tracker and mobile phone [8]. It is also possible without 3D tracking, e.g., by using computer vision to detect the phone within a scene camera of the eye-tracker [14], but the computational resources needed are often difficult to support live running applications (e.g., at 30 Hz). We, therefore, investigate a relatively cheap setup and toolkit to enable mobile eye-tracking that could be as easy as on the desktop. Recently, the Apple IPhone X and later series have included gaze-tracking capability in the ARKit. It has been noted that this gaze tracker's accuracy is severely lacking yet [11]. As such, deep learning approaches aim to address this through a more data-intensive approach [17], indicating the potential that eye-gaze may find itself as a core asset in future mobile devices. But as of yet, there is no work that explored a simple approach of remote eye-trackers combined with a mobile phone.

Our work is related to other research-oriented toolkits and software solutions, and to literature on measuring the gaze estimation error. Individual circumstances, such as movement or rest, affect the output of eye-trackers [9]. Accuracy is defined as the absolute difference between the focused target and the mean of the estimated gaze points. Precision is calculated using the gaze points' standard deviation (SD) [3, 9]. Feit et al. assessed eye tracking quality for a desk setup with a laptop/tablet system. They found that there is a lot of variety in accuracy and precision. As they found consistently bigger standard deviations in the y-direction, gaze-enabled areas should be built somewhat larger in height than in breadth. It's best to keep gaze-enabled objects away from the bottom and right edges of the screen, where accuracy and precision have been shown to be much lower [3]. Kapp et al. provided an open-source toolkit for eye-tracking research in augmented reality utilizing the Microsoft HoloLens 2 device. The spatial accuracy and spatial precision of gaze data from their toolkit were investigated in user research (n = 21). The gaze signal was captured while the individuals were sitting or walking. The findings indicate that when the distance between *fixation* targets rises, spatial accuracy improves. Furthermore, they discovered evidence that participants' spatial accuracy and precision decrease when walking versus standing motionless [9]. In this work, we aim to provide a toolkit for eye-tracking on a mobile phone and assess accuracy and precision requirements across novel mobile settings. We particularly investigate how accuracy changes in different areas on the screen when being used in different settings.

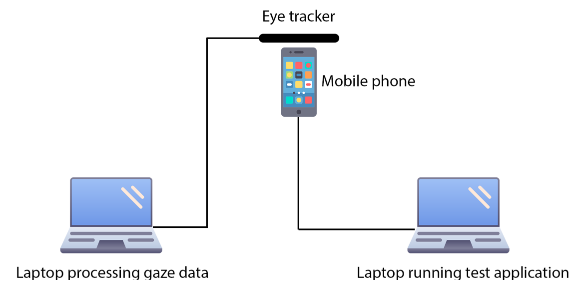


**Figure 1: The study setup. The Tobii Pro Nano Eye Tracker was connected to a PC which transferred the gaze data via UDP to another PC controlling the content on the smart phone.**

## 3 MOBILE PHONE EYE TRACKING TOOLKIT

We develop an eye-tracking toolkit for mobile phone applications using the cross-platform framework Flutter. Our goal is to simplify access to eye-tracking data from the Tobii Eye Pro device for research purposes or advanced interaction techniques. Work by Kapp et al. [9] is most relevant to our aims as they have been designing a similar user study to measure accuracy and precision using a VR device.

We aim at providing raw gaze data robustly at a fixed data rate, without delay, and with the highest possible spatial accuracy and precision. For this, we implement an easy-to-use interface to control recordings. The toolkit, detailed documentation, and an example project are available on GitHub [1] under the MIT open-source license. Our recording tool consists of two major components. On the one side, we have our main application running on our test mobile device, which is built using Google's Flutter cross-platform framework, in the programming language Dart. A Python server running on a Windows PC is receiving gaze data from the eye tracker and sends it through UDP to the mobile device. An overview of our system's architecture and the interplay of individual components is shown in Figure 1. The manufacturer offers a SDK to easily find and get data from our eye tracker[2].

The Tobii Pro SDK makes it simple to set up the eye-tracker. It has built-in methods for listing all available eye-trackers, connecting to them, and subscribing to their data. The data will be available in the form of a Python class called *GazeData*, which we'll use in a callback function. This function is called whenever a new gaze data sample is available. In our scenario, we want it to be sent to our actual application through UDP. This function is called 60 times per second - exhausting the Tobii Pro Nano's refresh rate.

[1] Eye Tracking Toolkit on GitHub, https://github.com/overdoz/gaze_and_touch accessed 10/10/2022
[2] Tobii Pro SDK, https://developer.tobiipro.com/python/python-getting-started.html, accessed 10/10/2022

# 4 EVALUATION

Our goal is to show that there is large variability in accuracy and precision when eye-tracking on mobile phones is used in more practical conditions. We collected gaze data across a broad range of people in four different everyday scenarios. We maintained key variables constant throughout the investigation, such as the distance between the user's eyes and the eye-tracker, but made the setup as natural as possible to reflect how much eye-tracking accuracy and precision may change in real-world tracking situations.

## 4.1 Procedure

After welcoming the participants, they received a brief verbal overview of the study setup and procedure. After providing informed consent for participation in the study, they filled out a demographics questionnaire. Before the actual study started, we calibrated the eye tracker with a 5-point calibration method on the tablet computer provided by the manufacturer. The eye tracker was placed and angled towards the user until the calibration software revealed a good tracking position. The participants were advised to attempt to keep as motionless as possible, but that they could naturally move their heads. We then proceeded with the actual study. After finishing all conditions, the participants were thanked for their time and debriefed.

## 4.2 Independent Variables

During our study, we manipulate or alter the independent variable, which is expected to have a direct influence on our dependent variables. Therefore we want to analyze the following two variables:

*Task.* The participant's task was to look at five targets presented on the screen in a random order for three seconds each. After a five seconds countdown, the participant should focus on the inner circle of each appearing target, equally distributed over the spatial area. Four targets are positioned in the corners and one is located in the center of the screen. The participants should maintain a consistent distance between their eyes and the screen (however we did not enforce this) and were permitted to avert their attention and rest their eyes when the fifth dot disappeared from the screen. We will consider the following five targets for our investigation: TOP LEFT, TOP RIGHT, CENTER, BOTTOM LEFT, BOTTOM RIGHT.

*Conditions.* The task was repeated four times in different conditions. After each task, the user was instructed to pause for at least 90 seconds to avoid eye fatigue effects. In total, the study lasted approximately 20 minutes per participant.

- **A - Sitting** The participant was instructed to sit at a desk with their backs straight. Because we want this position to feel as natural as possible, we won't give the participants any specific instructions other than to lay their main forearm on the table. Previous testing revealed that the distance between the eyes and the mobile phone is between 50 and 70 cm. Therefore we determined 60 cm as a fixed distance throughout the entire study.
- **B - Standing** Our participants were instructed to stand up straight and use one hand to grasp the device. They were once again asked to finish the task motionless. We also strive to maintain a natural arm posture here. The forearm-to-upper-arm angle should be between 80 and 100 degrees. Their heads are somewhat tilted downward.
- **C - Walking** During this task, our participants walked many laps until the last target vanished off the screen. The participants were able to entirely focus their cognitive capabilities on the visual task since there were no obstacles on the preset route. We also supplied a three-meter USB cable to provide greater walking freedom.
- **D - Lying** The last task requires you to interact with your smartphone while lying down. The participant is lying on his back with the gadget held vertically above his head. The neck is supported by a cushion, which tilts the posture slightly. We attempted to keep a distance of 60 cm between the eyes and the device here as well.


**Figure 2: The 3D-printed phone case with the attached Tobii Pro Nano eye-tracker.**

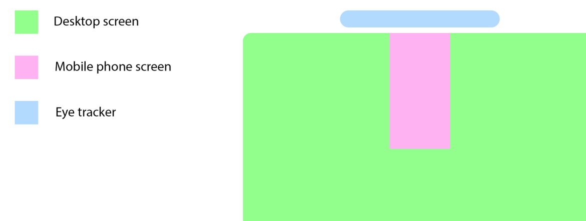


**Figure 3: Positioning of the eye-tracker above the smart phone. It also shows the relationship between the Smart Phones screen and the PCs screen on which the calibration software is executed.**

## 4.3 Dependent Variables

Spatial precision and accuracy are two typical measures for evaluating the gaze estimation error, and they also act as dependent variables in our study. Spatial precision is calculated as the root

**(a) Sitting** **(b) Standing** **(c) Walking** **(d) Lying**

**Figure 4: Example of setting A, B, C and D in our study with the participant holding the test device. The eye-tracker is mounted on top of the mobile phone. All tasks were completed in one single room throughout the entire study.**

mean square error or standard deviation of individual gaze samples from their centroid, and spatial accuracy is calculated as the mean angular deviation of fixations to the real position [10]. Both pixel and degree of visual angle measurements are reported. The euclidean distance between the predicted gaze point and the target based on their locations is used to determine the distance in pixels. The angle between the reported 3D gaze ray from the gaze origin to the gaze point and the 3D ray from the gaze origin to the target position is used to determine the visual angle.

## 4.4 Participants

Ten participants took part in the experiment, of which 5 were female. Their age ranged between 24 to 30 (mean = 26.7, SD = 2.06) years. Height was in range 1.57 to 1.88 m (mean = 1.72, SD = 0.11). Two of the 10 participants already had experience with eye-tracking and three of the participants wore glasses.

## 4.5 Data Collection and Preprocessing

Without filtering, all gaze points were recorded as a (x, y) position in screen coordinates, as provided by the tracker's API, as well as translated values in pixels. The first and last 1s were discarded during analysis to compensate for the time it took participants' eyes to travel to the target and anticipatory movements. We didn't employ a *fixation* detection technique since it can't tell the difference between *saccades* and noise artifacts consistently. The API from the manufacturer also includes the pupil diameter for each eye. In addition, data from an inertial measurement unit (IMU) was gathered at the same time. This includes data from the accelerometers and gyroscopes, which provides us with more information about the phone's orientation. For each frame the eye tracker sent to the device, IMU data was captured.

## 4.6 Results

The analyzed dataset consists of 56,342 gaze samples from 10 users looking at 5 targets. 45,885 samples were registered during fixation. By "target fixation" we simply denote the data recorded while a user was looking at a target. No algorithmic method was involved to extract fixation events. In the following analysis, we first compute aggregate measures for each target. Where not denoted differently, we then average across the 5 targets for each user to account for the fact that fixations by the same participant are not independent observations. Data loss occurs when the eye tracker cannot estimate the gaze position. On average, each target consists of 20 gaze points recorded over 1 s (ca. 60 frames). We believe this data loss to have occurred due to the handheld thus unsteady usage. The given data was analyzed using a Friedman and a Wilcoxon signed-rank test to show statistically significant differences across our independent variables. We compute accuracy and precision individually in the x- and y-direction for all gaze positions during a target fixation. The absolute offset between the focused target and the mean of the estimated gaze points is referred to as accuracy. The standard deviation (SD) of the gaze points is used to calculate the precision.

## 4.7 Accuracy across Context

Figure 5 presents the gaze estimation error for each setting denoted as degrees of visual angle. The horizontal angle is represented by the light blue bars, while the vertical angle is represented by the dark blue bars. According to the chart, settings with phone rotation or movement result in higher horizontal than vertical inaccuracy. Furthermore, setting A, B and D seem to yield quite similar results compared to setting C. For setting A, we report the metrics for all targets with a horizontal mean angular accuracy of 2.27 degrees (SD = 1.56) and vertical accuracy of 2.55 degrees (SD = 1.47). Setting B shows a horizontal mean spatial accuracy of 1.54 (SD = 1.20) and 2.16 degrees (SD = 1.79) on the vertical axis. The recordings for setting C yield a horizontal mean accuracy of 7.71 degrees with a precision of 3,00 degrees and a vertical accuracy of 5.12 degrees (SD = 1.62). The mean angular accuracy for setting D is 3.06 degrees with a precision of 4.09 degrees on the X-axis and 3.14 degrees (SD = 2.43) on the y-axis.

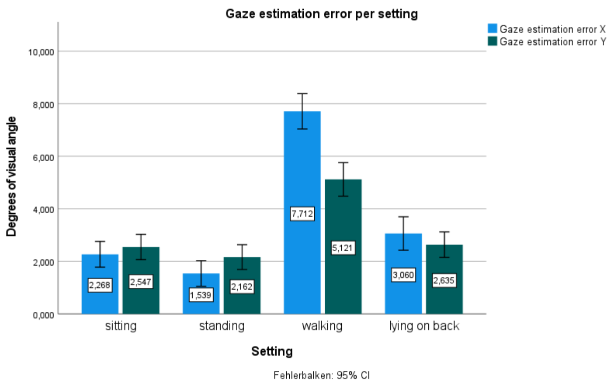


**Figure 5: Gaze estimation error per setting measured in horizontal and vertical degrees of visual angle.**

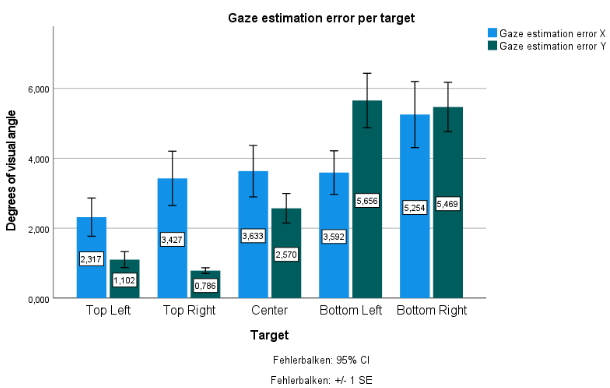


**Figure 6: Gaze estimation error per target measured in horizontal and vertical degrees of visual angle.**

## 4.8 Accuracy across Spatial Area

Figure 6 shows us the gaze estimation error for each target. Again the y-axis represents the inaccuracy in degrees of visual angle. While there are no noteworthy discrepancies between the horizontal values (2.3 - 5.3 degrees), the vertical inaccuracy differs greatly between the upper and lower targets (1.1 - 5.7 degrees). Also, the lower targets show a higher vertical than horizontal inaccuracy compared to the remaining targets. For the top left target, we report a horizontal mean angular accuracy of 2,32 degrees (SD = 1.32) and vertical accuracy of 1.10 degrees (SD = 0.63). The top right target shows a horizontal mean spatial accuracy of 3.43 (SD = 2.20) and 0.79 degrees (SD = 0.40) on the vertical axis. The recordings for the center target yield a horizontal mean accuracy of 3.63 degrees with a precision of 2.71 degrees and a vertical accuracy of 2.57 degrees (SD = 1.43). The mean angular accuracy for the bottom left target is 3.59 degrees with a precision of 2.45 degrees on the x-axis and 5.66 degrees (SD = 3.14) on the y-axis. The target at the bottom right corner shows a horizontal accuracy of 5.25 (SD = 3.80) and vertical accuracy of 5.47 (SD = 2.80) degrees.

**Table 1: Results of the Wilcoxon signed-rank tests for the comparison of the horizontal and vertical accuracy between settings. *the Bonferroni corrected significane level is p < 0.008.**

| Comparison | Z (x) | p (x) | Z (y) | p (y) |
|---|---|---|---|---|
| Standing - Sitting | -1.580 | 0.114 | -1.070 | 0.285 |
| Walking - Sitting | -2.497 | 0.013 | -2.497 | 0.013 |
| Lying - Sitting | -0.153 | 0.878 | -0.764 | 0.445 |
| Walking - Standing | -2.701 | 0.007* | -2.599 | 0.009 |
| Lying - Standing | -1.478 | 0.139 | -1.478 | 0.139 |
| Lying - Walking | -2.701 | 0.007* | -2.547 | 0.011 |

## 4.9 Statistical Analysis

The Friedman test reported an overall statistically significant difference between the mean ranks of the related groups. We can see that there was a statistically significant difference in perceived effort depending on which posture was taken whilst testing accuracy, $x^2(2)$ = 11.400 for x / 10.636 for y, p = 0.010 for x / 0.014 for y. To evaluate the difference in spatial accuracy over all targets we also conducted the same test. It shows a significant difference in accuracy between the different spatial areas, $x^2(3)$ = 6.800 for x / 27.440 for y, p < 0.147 for x / <0.001 for y.

Table 1 shows the output of the Wilcoxon signed-rank test comparing each setting regarding horizontal and vertical accuracy. The table provides the significance level for each direction (p) and the corresponding Z score. It is important to note that the significance values have not been adjusted in our statistics program to compensate for multiple comparisons. We must manually compare the significance values produced by the program to the Bonferroni-adjusted significance level. We can see that at the p < 0.005 significance level, only perceived effort between setting B (standing) and C (walking) was statistically significantly different (p = 0.007 on the x-axis). There was a statistically significant difference between all targets in the vertical accuracy depending on which type target was fixated, $x^2(2)$ = 27.440, p = < 0.001, whereas the Wilcoxon signed-rank test doesn't point out any significant differences.

# 5 DISCUSSION

The eye tracking toolkit renders it simple for researchers to employ eye tracking on a mobile phone and gives users a first glance at a possible use case to enhance the user interface. It should make it simple for eye trackers to operate with Flutter-based apps and allow for the recording of a wide range of eye-tracking signals.

We tested a commercially available eye-tracker in a variety of scenarios. Because tracking relies on video images of the eye, an unobstructed view of the eye is required [7]. Changing light conditions, reflections from eyeglasses, and droopy eyelids are all factors that can impair the quality of tracking [4, 10]. Under different climatic circumstances and/or with different equipment, the particular accuracy and precision metrics we reported may vary. However, our results confirm, that variances in movement and resting have a significantly great impact on tracking quality, according to our research.

We used a consumer remote eye-tracker to evaluate the accuracy on a mobile phone. The requirement is that both eye data and screen coordinate systems are mapped with the right method. Such a method could refine the result and erase errors. Nevertheless, systematic errors can be caused by inaccurate calibrations, head motions, astigmatism, eyelid closure, and other factors that are highly dependent on the individual participant's characteristics [5]. Further, we could have applied filters to ensure robust and smooth interaction. There is no way to post-process data in interactive gaze-enabled programs, therefore any filtering must be done in real-time. On the other hand, this reduces the capacity to detect outliers and artifacts, which cause a multiple-frame latency [3]. Nonetheless, because the p-value of the Wilcoxon signed-rank test is barely over 0.005, we may have taken extra care during our investigation to exclude outliers and data loss in order to prove a significant decrease in accuracy toward to bottom of the screen. In addition, a larger number of participants could have supported our expected results.

We found that accuracy becomes slightly worse toward the bottom of the screen (see Figure 6), similar to the findings of Feit et al. [3]. The tested eye-tracker can be used with screens up to 24", which is larger than the 5.5" phone screen we used. Testing with larger phone screens could result in more accurate gaze estimation due to more precision during the mapping process, where the spectrum of interest is not trimmed drastically. We expect tracking to also get worse toward the top corner as targets are placed closer to the spatial limits of the tracker; however, a similar study on a larger screen is necessary to confirm the expectation. During the accuracy evaluation, we only tested 5 targets with a certain distance ( > 0 pixel) to the borders of the screen. It would be interesting to conduct in-depth investigations of edge cases, for example, the already mentioned top corners where most of the notification elements are located in modern mobile operating systems. Another interesting way to extend the accuracy evaluation would be to create a model which also contains different lighting environments. We could extend our existing scenarios by including different times of the day. This includes natural daylight and artificial lighting at night.

In order for eye-tracking to be integrated into mobile phone applications, it may require some sort of stabilizer for current eye-tracking devices to work outside of controlled labs. We conclude, that the interface design must adapt to the accuracy of such eye-trackers. The study showed that UI elements must satisfy a certain size to allow some degree of precision. Here, UI designers should also think about possible different tracking accuracy in the horizontal and vertical dimensions based on the UI element placement on the screen. In addition, calibration can improve by omitting the second layer of coordinate mapping. Standalone calibration software for mobile devices, that also covers smaller resolutions, could achieve better accuracy results.

## 6 CONCLUSION

In this work, we presented an open-source toolkit that enables video-based eye-tracking research on mobile devices. We addressed the gap of missing research tools by implementing a Flutter toolkit for reliable gaze data acquisition. We conducted a user study (n = 10) to investigate the spatial accuracy and precision of gaze data from our toolkit. The results suggest that the spatial accuracy and precision decrease slightly toward the bottom of the screen. Further, we found evidence that spatial accuracy drops when participants are moving compared to resting motionless. Eye-tracking has the potential to revolutionize the way we interact with mobile phones. Attentive applications seamlessly integrate gaze input into existing interaction paradigms and explicit gaze control opens new design spaces for Human-Computer interaction. We were able to bridge a gap in understanding the requirements of such apps when they are delivered to customers since researchers have created several interaction approaches and applications that employ gaze information. This showed that these techniques can improve the usability and utility of interacting with mobile devices, which led to the conclusion that the technology has great potential and can be used easily when eye tracking is established more.

## REFERENCES


[1] Joanna Bergstrom-Lehtovirta and Antti Oulasvirta. 2014. Modeling the functional area of the thumb on mobile touchscreen surfaces. *Conference on Human Factors in Computing Systems - Proceedings* (2014), 1991–2000. https://doi.org/10.1145/2556288.2557354

[2] Andreas Bulling and Hans Gellersen. 2010. Toward Mobile Eye-Based Human-Computer Interaction. *IEEE Pervasive Computing* 9, 4 (2010), 8–12. https://doi.org/10.1109/MPRV.2010.86

[3] Anna Maria Feit, Shane Williams, Arturo Toledo, Ann Paradiso, Harish Kulkarni, Shaun Kane, and Meredith Ringel Morris. 2017. Toward Everyday Gaze Input: Accuracy and Precision of Eye Tracking and Implications for Design. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 1118–1130. https://doi.org/10.1145/3025453.3025599

[4] Kenneth Holmqvist, Marcus Nyström, and Fiona Mulvey. 2012. Eye tracker data quality: What it is and how to measure it. *Eye Tracking Research and Applications Symposium (ETRA)* February (2012), 45–52. https://doi.org/10.1145/2168556.2168563

[5] Anthony J. Hornof and Tim Halverson. 2002. Cleaning up systematic error in eye-tracking data by using required fixation locations. *Behavior Research Methods, Instruments, and Computers* 34, 4 (2002), 592–604. https://doi.org/10.3758/BF03195487

[6] Mike Horsley, Matt Eliot, Bruce Allen Knight, and Ronan Reilly. 2014. Current trends in eye tracking research. *Current Trends in Eye Tracking Research* (2014), 1–345. https://doi.org/10.1007/978-3-319-02868-2

[7] Jukka Hyönä, Ralph Radach, and Heiner Deubel. 2003. *The Mind's Eye: Cognitive and Applied Aspects of Eye Movement Research.* 1–741 pages. https://doi.org/10.1016/B978-0-444-51020-4.X5000-2

[8] Xinhui Jiang, Yang Li, Jussi P.P. Jokinen, Viet Ba Hirvola, Antti Oulasvirta, and Xiangshi Ren. 2020. How We Type: Eye and Finger Movement Strategies in Mobile Typing. *Conference on Human Factors in Computing Systems - Proceedings* (2020), 1–14. https://doi.org/10.1145/3313831.3376711

[9] Sebastian Kapp, Michael Barz, Sergey Mukhametov, Daniel Sonntag, and Jochen Kuhn. 2021. Arett: Augmented reality eye tracking toolkit for head mounted displays. *Sensors* 21, 6 (2021), 1–18. https://doi.org/10.3390/s21062234

[10] Päivi Majaranta and Andreas Bulling. 2014. Eye Tracking and Eye-Based Human–Computer Interaction. (2014), 39–65. https://doi.org/10.1007/978-1-4471-6392-3_3

[11] Sven Mayer, Gierad Laput, and Chris Harrison. 2020. *Enhancing Mobile Voice Assistants with WorldGaze.* Association for Computing Machinery, New York, NY, USA, 1–10. https://doi.org/10.1145/3313831.3376479

[12] Ken Pfeuffer, Jason Alexander, Ming Ki Chong, and Hans Gellersen. 2014. Gaze-Touch: Combining Gaze with Multi-Touch for Interaction on the Same Surface. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology* (Honolulu, Hawaii, USA) *(UIST '14)*. Association for Computing Machinery, New York, NY, USA, 509–518. https://doi.org/10.1145/2642918.2647397

[13] Ken Pfeuffer and Hans Gellersen. 2016. Gaze and touch interaction on tablets. *UIST 2016 - Proceedings of the 29th Annual Symposium on User Interface Software and Technology* (2016), 301–311. https://doi.org/10.1145/2984511.2984514

[14] Ken Pfeuffer and Yang Li. 2018. Analysis and modeling of grid performance on touchscreen mobile devices. *Conference on Human Factors in Computing Systems - Proceedings* 2018-April (2018). https://doi.org/10.1145/3173574.3173862

[15] Radiah Rivu, Yasmeen Abdrabou, Ken Pfeuffer, Mariam Hassib, and Florian Alt. 2020. Gaze'N'Touch: Enhancing Text Selection on Mobile Devices Using Gaze. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI EA '20)*. Association for Computing Machinery, New York, NY, USA, 1–8. https://doi.org/10.1145/3334480.3382802

[16] Sheikh Rivu, Yasmeen Abdrabou, Thomas Mayer, Ken Pfeuffer, and Florian Alt. 2019. GazeButton: Enhancing Buttons with Eye Gaze Interactions. In *Proceedings of the 11th ACM Symposium on Eye Tracking Research & Applications* (Denver, Colorado) *(ETRA '19)*. Association for Computing Machinery, New York, NY, USA, Article 73, 7 pages. https://doi.org/10.1145/3317956.3318154

[17] Nachiappan Valliappan, Na Dai, Ethan Steinberg, Junfeng He, Kantwon Rogers, Venky Ramachandran, Pingmei Xu, Mina Shojaeizadeh, Li Guo, Kai Kohlhoff, et al. 2020. Accelerating eye movement research via accurate and affordable smartphone eye tracking. *Nature communications* 11, 1 (2020), 1–12.

[18] Pierre Weill-Tessier, Jayson Turner, and Hans Gellersen. 2016. How Do You Look at What You Touch? A Study of Touch Interaction and Gaze Correlation on Tablets. In *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications* (Charleston, South Carolina) *(ETRA '16)*. Association for Computing Machinery, New York, NY, USA, 329–330. https://doi.org/10.1145/2857491.2888592