International Journal of Advanced Smart Convergence Vol.10 No.4 30-37 (2021)
http://dx.doi.org/10.7236/IJASC.2021.10.4.30

IJASC 21-4-4

Analyzing Input Patterns of Smartphone Applications in Touch Interfaces

Hyokyung Bahn and Jisun Kim

Professor, Department of Computer Engineering, Ewha University, Korea
bahn@ewha.ac.kr

Abstract

Touch sensor interface has become the most useful input device in a smartphone. Unlike keypad/keyboard interfaces used in electronic dictionaries and feature phones, smartphone's touch interfaces allow for the recognition of various gestures that represent distinct features of each application's input. In this paper, we analyze application-specific input patterns that appear in smartphone's touch interfaces. Specifically, we capture touch input patterns from various Android applications, and analyze them. Based on this analysis, we observe a certain unique characteristics of application's touch input patterns. This can be utilized in various useful areas like user authentications, prevention of executing application by illegal users, or digital forensic based on logged touch patterns.

Keywords: *Input pattern, touch gesture, Smartphone touch interface, Smartphone Application.*

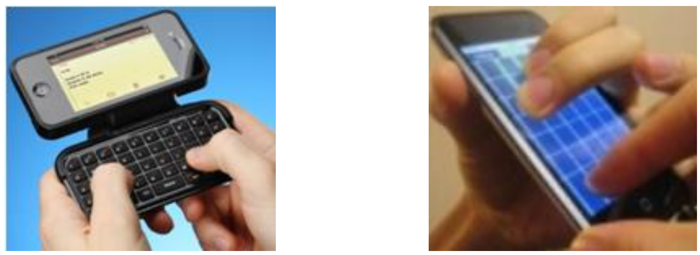
1. Introduction

With the recent advances in mobile hardware and software technologies, smartphones have become an indispensable device in our lives [1]. People are increasingly accessing social media, personal live-streaming, and location-based services through smart devices [2, 3, 4]. Unlike traditional computer systems that use keyboard and mouse as input devices, smartphones adopt touch interfaces for receiving user input [5]. Smartphone's touch interface provides convenience to users and it has a powerful function of recognizing complicated gesture. Accordingly, touch interfaces are widely used in various applications, which require user identification and sophisticated input data [6, 7].

In the last decades, studies on smartphone systems have focused on the performance improvement of mobile storage as it was the performance bottleneck of mobile systems [8, 9]. Recently, the performance of a smartphone has become sufficient to perform multitasking like desktop systems, and the focus of smartphone studies are shifting to I/O interfaces like sensors and touch devices. Specifically, touch interfaces support various input patterns and control through touch positions, directions, and pressure of touch than traditional keyboard interfaces as shown in Figure 1. This paper analyzes user gestures appear in each application, which is displayed in touch interface of a smartphone. Specially, we extract touch data from various application categories such as game, web browser, image viewer, video player, camera, and map applications in Android smartphones and analyzed them. Through our analysis, we observed a certain unique characteristics for each

Manuscript Received: September. 24, 2021 / Revised: September. 29, 2021 / Accepted: October. 4, 2021
Corresponding Author: bahn@ewha.ac.kr
Tel: +82-2-3277-2368, Fax: +82-2-3277-2306
Professor, Department of Computer Engineering, Ewha University, Korea

application, which will be meaningful to some security-related functions. We anticipate that the result of this paper will be helpful in the design of new security functions like fingerprinting of applications/users and digital forensic.



(a) Keyboard interface (b) Touch interface

Figure 1. Comparison of Keyboard and Touch Interfaces

2. Input Patterns of Smartphone Applications in Touch Interfaces

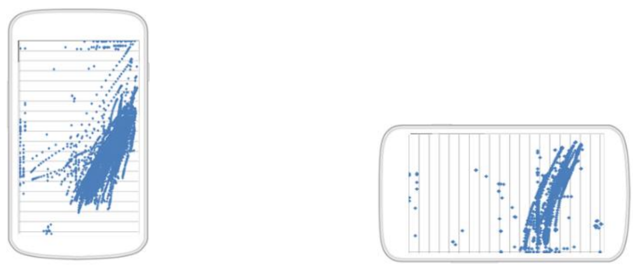
In this section, we collect input patterns of various applications that appear in touch interfaces and analyze them. There are various types of applications executed in modern smartphones, and we selected six categories based on the diversity of touch input patterns. We excluded application types depending mainly on keypad character inputs as we focus on the touch input characteristics of applications. The six categories we selected are game, web browser, image viewer, video player, camera, and map applications. We extracted touch input patterns that appear in these categories of applications. In our experiment, three graduate students in our research group participated in the process of collecting touch input data. For each application, we performed the experiments for 10 to 20 minutes, and the experiments were tried three times. Of these, we depicted the captured result of one sample for each application category as the patterns of the same applications exhibited similar results.

2.1 Game

We select Angrybird and Pokopang as game applications. Figure 2 shows the distributions of touch input patterns while playing each of the games. Angrybird is played by placing the smartphone in the transverse direction. In Angrybird, the user performs a number of zoom-in and zoom-out gestures as the game needs to determine the location of the target. Thus, as shown in Figure 2(a), a number of touches are logged at the bottom screen line. Also, many touch data are logged at the left side of the screen as the launch button of Angrybird is located at the left side of the screen. Pokopang is a game that draws lines to join blocks of the same color. Thus, as shown in Figure 2(b), the touch points of Pokopang are evenly distributed throughout the screen. Another interesting result is that Angrybird has a lot of linear touch data, whereas Pokopang has a curved type of touch data.

(a) Angry Bird (b) Pokopang

Figure 2. Touch input patterns in games.



(a) Portal Site (b) YouTube

Figure 3. Touch input patterns in Web browser

2.2 Web browser

We visited popular portal site and YouTube to investigate the touch data in Web browser. Figure 3(a) shows the touch points when a portal site was visited. As shown in the figure, the touch points are displayed linearly in the right side of the screen because the user frequently drags the vertical scroll bar. In most cases, the scrolling is caused by the thumb, and thus the touch data are diagonally shaped. Figure 3(b) shows the touch data when the user watch videos on YouTube. YouTube has a small number of touch logs as touch data are not generated while watching a video for a long time. However, there are some distinct patterns in YouTube as various activities like uploading, watching, sharing, or searching videos may happen while the user uses YouTube. Like the portal site case, scrolling appears on the right side in YouTube since there are contents consisting of several series files. The figure also shows that watching the content on YouTube is available by the horizontal direction.

2.3 E-book and Gallery Viewer

In this section, we analyze the touch input patterns while a user views text and image files by E-book and Gallery applications, respectively. In the case of E-book, as shown in Figure 4(a), a lot of touch gestures are generated in both the left and the right side of the screen. This is because the E-book application has a function of turning pages when touching a portion of each side. With this simple touch data on both sides, there are also many touch gestures in the horizontal form. This is because the user takes gesture as if they are turning the pages due to the sliding effect provided by E-book. Figure 4(b) shows touch points extracted from the Gallery application. Gallery is an application to view multiple photos. Thus, the touch data are mostly composed of side to side gestures. The figure also shows diagonal touch data that occur when the users zoom-in or zoom-out photos.

(a) E-book **(b) Gallery**

Figure 4. Touch input patterns in E-book and gallery

2.4 Video player

Figure 5 shows touch points extracted from video player. Video player has a few touch data. This is because video data are sequentially accessed [10] and users spend most of their time in watching videos when they use the video player. A few touch data are logged at the center of the screen because users check current timeline in playing video by touching the screen. Note that the video player has a feature that makes the timeline disappear unless the user enters motion. Touch data at the bottom of the screen occur when the user moves backwards or skips parts of the video.

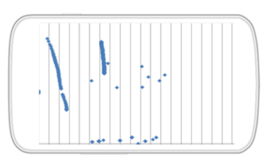


Figure 5. Touch input patterns in video player

2.5 Camera applications

Figure 6 shows the touch data collected while the camera application is executed. As shown in the figure, touch data are clustered around the center of the screen. This is because touch data in the camera application occurs while trying to zoom-in and zoom-out with both hands. Many touch data also appear in a certain position of the screen where the shutter button is located.

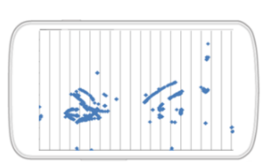


Figure 6. Touch input patterns in camera application

2.6 Map applications

Unlike the touch data of applications in the previous section, a map application exhibits some irregular touch access patterns. This is because a map application makes use of all available gestures like touch, horizontal scrolling, vertical scrolling, zoom-in, and zoom-out. Figure 7(a) shows touch data while searching a certain location. In this figure, we can observe a lot of zoom-in and zoom-out gestures near the central horizontal line to check the current and target locations. Figure 7(b) shows touch data while checking the route to the desired location. When compared with Figure 7(a), lines are longer and close to a straight line.

(a) Map search (b) Navigation

Figure 7. Touch input patterns in Map Application

3. Analysis and Implications

In this section, we briefly summarize the analysis results of each application's input patterns obtained through touch interfaces and briefly discuss how they can be utilized.

3.1 Analysis of the Captured Patterns

The analysis results for each application's touch input in the previous section can be summarized as follows: 1) Touch patterns of games are determined based on a certain unique behaviors for each game. For example, Angrybird generates zoom-in/zoom-out and firing input patterns, whereas Pokopang generates a curved input pattern across the screen by one stroke. 2) Web browser exhibits touch inputs for scrolling diagonally, and Youtube also has many scroll inputs for video search although it does not have frequent touch inputs. 3) In the case of image and e-book viewers, a number of left and right touch inputs to move to the next photo or page appears. 4) Video player has very few touch input for moving video positions and skipping to the next video. 5) For camera applications, touch input for pressing zoom and shoot buttons appears. 6) In map applications, various touch inputs such as vertical scrolling, left and right scrolling, and zoom-in/zoom-out of the screen appear.

3.2 Further Implications

Similar to fingerprints, unique touch input patterns exist for each smartphone application. Thus, it can be possible to identify which application is being performed through the touch pattern analysis results performed in this paper. Specifically, by making use of unique input patterns of applications, monitoring of smartphone user behaviors can be possible without direct and strong invasion of privacy. For example, instead of monitoring the full context of smartphone usage (e.g., the contents searched or the videos watched), the touch logs can be alternatively used to estimate the type and duration of applications executed with limited information. This can be applied to monitor whether a child or a student uses some particular applications without exposing sensitive information.

In addition, since the touch input pattern appears differently depending on users [11-17], it will be also possible to identify who executes applications by combining our application-specific study with previous works that authenticate user-specific touch patterns. However, this needs more detailed analysis of touch inputs as previous studies showed that not only the touch shape, but also other inputs like time between the start and end of an event, velocity, touch pressure, etc., are necessary to distinguish users [11, 12].

For user authentication of a smartphone, passwords, pattern locks, or biometrics are currently being used to protect personal information of a smartphone. However, it is not easy to check whether a malicious user finds out and secretly executes an application or steals personal data. As various financial information (e.g., credit cards and bank accounts) and private data (e.g., personal photos and videos) are stored in smartphones, the problem of privacy invasion between malicious users and even family members is becoming increasingly serious. The analysis result in this paper can be applied for identifying whether someone other than the smartphone's owner executed a certain smartphone application by logging touch input patterns. It may also be possible to detect whether an input pattern is different from the original user in an online manner and block the device instantly. The logged patterns may also be used for digital forensic when a crime occurs.

4. Related Works

As a touch input interface provides complicated information that represents distinct features of applications and users compared to text input interfaces such as keyboard or keypad, a lot of studies have been conducted to utilize touch gestures for user authentication.

Shen et al. [11] observed that touch-interaction in smartphones have user-specific behaviors with respect to rhythm, strength, and angle preferences. They extracted features from various users' touch operations based on some classification techniques. Through this analysis, they presented a new authentication technique for smartphones.

Luca et al. [12] claimed that pattern-based authentication for smartphone unlocking is vulnerable in security, and introduced an implicit approach that enhances pattern-based authentication with an additional security layer. Specifically, their scheme does not authenticate users only by the input shape, but also by the way they perform the input like pressure, coordinates, size, speed, time etc.

Fierrez et al. [13] analyzed the one finger swipe gestures in touch input data for personal authentication. Their analysis results showed some new insights into the distinctiveness of swipe interaction, i.e., 1) data from landscape orientation is more stable, and 2) horizontal gestures are more discriminative than vertical ones.

Saebae et al. [14] argued that multi-touch gestures can enhance the authentication security significantly. Specifically, they defined five-finger touch gestures based on the movement characteristics of the center of the palm and fingertips. By using pattern recognition techniques, they built a classifier to recognize unique biometric gesture characteristics of an individual with their defined gesture set.

Frank et al. [15] investigated the possibility of using touch behavior for continuous authentication on smartphones. They used a background application to capture simple touch movements in smartphones, and extracted a set of 30 features from each stroke to characterize user profiles. Their results showed that continuous authentication based on touch behavior can reach a practically useful level, but the applicability in real-world scenarios need to be further addressed for putting it into practice.

Murmuria et al. [16] proposed strangeness-based outlier detection to monitor user behaviors based on power consumption, touch gestures, and physical movement. Their algorithms exhibited competitive performance when sufficient data are available to model each user.

Perera and Patel [17] proposed a quick intrusion detection method for continuously verifying the identity of a mobile device user. They presented Bayesian and MiniMax versions of the quickest change detection algorithms and suggested how to utilize them without heavy resource overhead by periodically activating the sensing module. Their results showed that it is possible to detect a high percentage of intrusions with a relatively small number of gestures.

5. Conclusion

In this paper, we extracted and explored application-specific user gestures that appear in smartphone touch interfaces. Unlike keypad or keyboard interfaces, smartphone's touch interfaces allow for the recognition of various gestures that represent distinct features of each application's input. Specifically, we analyzed the touch input patterns of various Android applications and observed a certain unique characteristics of application's touch input patterns. We anticipate that the result of this paper will be helpful in the design of new security functions like application fingerprinting and digital forensic.

Acknowledgement

This work was supported in by the ICT R&D program of MSIP/IITP (2020-0-00121, development of data improvement and dataset correction technology based on data quality assessment) and (2019-0-00074, developing system software technologies for emerging new memory that adaptively learn workload characteristics).

References

[1] T. Li, M. Zhang, H. Cao, Y. Li, S. Tarkoma, and P. Hui, "What apps did you use? understanding the long-term evolution of mobile app usage," Proceedings of the World Wide Web Conference, pp 66–76, 2020.
DOI: https://doi.org/10.1145/3366423.3380095

[2] D. Kim, S. Lee, and H. Bahn, "An adaptive location detection scheme for energy-efficiency of smartphones," Pervasive and Mobile Computing, vol. 31, pp. 67-78, 2016.
DOI: https://doi.org/10.1016/j.pmcj.2016.04.012

[3] E. Lee and H. Bahn, "Electricity usage scheduling in smart building environments using smart devices," The Scientific World Journal, vol. 2013, Article 468097, pp. 1-11, 2013.
DOI: https://doi.org/10.1155/2013/468097

[4] T. Kim, H. Bahn, and K. Koh, "Popularity-aware interval caching for multimedia streaming servers," Electronics Letters, vol. 39, no. 21, pp. 1555-1557, 2003.
DOI: https://doi.org/10.1049/el:20030965

[5] R. Ikeda, K. Hakka, and B. Shizuki, "Hover-based reachability technique for executing single-touch gesture on smartphone," in Proc. Asian CHI Symposium, pp. 9-15, 2021.
DOI: https://doi.org/10.1145/3429360.3468171

[6] A. Acien, A. Morales, and R. Vera-Rodriguez, “Smartphone sensors for modeling human-computer interaction: General outlook and research datasets for user authentication,” in Proc. 44th IEEE COMPSAC Conf., 2020.
DOI: https://doi.org/10.1109/COMPSAC48688.2020.00-81

[7] A. Suharsono and D. Liang, “Hand stability based features for touch behavior smartphone authentication,” in Proc. 3rd IEEE Conf. Knowledge Innovation and Invention, 2020.
DOI: https://doi.org/10.1109/ICKII50300.2020.9318982

[8] D. Kim, E. Lee, S. Ahn, H. Bahn, “Improving the storage performance of smartphones through journaling in non-volatile memory,” IEEE Trans. Consumer Electronics, vol. 59, no. 3, pp. 556-561, 2013.
DOI: https://doi.org/10.1109/TCE.2013.6626238

[9] J. Park, H. Lee, S. Hyun, K. Koh, H. Bahn, “A cost-aware page replacement algorithm for NAND flash based mobile embedded systems,” in Proc. the seventh ACM international conference on Embedded software (EMSOFT), 2009.
DOI: https://doi.org/10.1145/1629335.1629377

[10] O. Kwon, H. Bahn, and K. Koh, “Popularity and prefix aware interval caching for multimedia streaming servers,” in Proc. 8th IEEE International Conference on Computer and Information Technology, pp. 555-560, 2008.
DOI: https://doi.org/10.1109/CIT.2008.4594735

[11] C. Shen, Y. Zhang, X. Guan and R. A. Maxion, “Performance analysis of touch-interaction behavior for active smartphone authentication,” IEEE Trans. Information Forensics and Security, vol. 11, no. 3, pp. 498-513, 2016.
DOI: http://doi.org/10.1109/TIFS.2015.2503258

[12] A. Luca, A. Hang, F. Brudy, C. Lindner, and H. Hussmann, “Touch me once and I know it’s you!: implicit authentication based on touch screen patterns,” in Proc. ACM SIGCHI Conf. on Human Factors in Computing Systems, pp. 987-996, 2012.
DOI: http://doi.org/10.1145/2207676.2208544

[13] J. Fierrez, A. Pozo, M. Martinez-Diaz, J. Galbally and A. Morales, “Benchmarking touchscreen biometrics for mobile authentication,” IEEE Trans. Information Forensics and Security, vol. 13, no. 11, pp. 2720-2733, 2018.
DOI: http://doi.org/10.1109/TIFS.2018.2833042.

[14] N. Sae-Bae, K. Ahmed, K. Isbister and N. Memon, “Biometric-rich gestures: A novel approach to authentication on multi-touch devices,” in Proc. ACM SIGCHI Conf. on Human Factors in Computing Systems, pp. 977-986, 2012.
DOI: http://doi.org/10.1145/2207676.2208543

[15] M. Frank, R. Biedert, E. Ma, I. Martinovic and D. Song, “Touchalytics: On the applicability of touchscreen input as a behavioral biometric for continuous authentication,” IEEE Trans. Information Forensics Security, vol. 8, no. 1, pp. 136-148, 2013.
DOI: http://doi.org/10.1109/TIFS.2012.2225048

[16] R. Murmuria, A. Stavrou, D. Barbará and D. Fleck, “Continuous authentication on mobile devices using power consumption touch gestures and physical movement of users,” in Proc. Int. Workshop Recent Advances in Intrusion Detection, pp. 405-424, 2015.
DOI: http://doi.org/10.1007/978-3-319-26362-5_19

[17] P. Perera and V. M. Patel, “Efficient and low latency detection of intruders in mobile active authentication,” IEEE Trans. Information Forensics Security, vol. 13, no. 6, pp. 1392-1405, 2018.
DOI: http://doi.org/10.1109/TIFS.2017.2787995