Logging All the Touch Operations on Android

Yuko Hirabe*, Yutaka Arakawa* and Keiichi Yasumoto*
*Graduate School of Information Science, Nara Institute of Science and Technology (NAIST)
8916-5 Takayama, Ikoma, Nara 630-0192, Japan
Email: {hirabe.yuko.ho2, ara, yasumoto}@is.naist.jp

***Abstract*—Touch operation has already become a common technique, and will be adopted to most electrical appliances in the future. Therefore, we suppose that a user's touch operation will be one of important contexts for realizing the future context-aware system. Through observation of the user's touch operation, a system could estimate the user's emotion, skill, etc. In this demonstration, we exhibit our system for logging all the touch operations on Android. In addition to the logging, our system can analyze and visualize several popular touch operations.**

I. Introduction

For a high quality awareness service, it is important to recognize a user's various context through various sensors. Recently, estimation techniques for not only a simple context like location and time but also rather complex context like human's activity receives increasing attention as one of important research fields. For example, HASC [1] aims to recognize various human activities such as walking, running, sitting, etc. by using an acceleration sensor in a smart phone.

Furthermore, a user's emotion is an important complicated context that should be estimated. There are many existing studies about an emotion estimation, and they can be categorized into four types as follows: text-based, voice-based, image-based and brain-based. A text-based approach analyzes textual information such as sentences, words, special characters and face marks. This approach can be widely adopted to various applications. However, it cannot estimate the user's emotion that cannot be expressed in a text, for example, impatience at busy situation, a stress in operating a smartphone. A voice-based approach measures user's tension from speech waveform, and estimates user's stress. However, it is not easy to keep recording all the user's voice unconsciously. Especially when talking with several persons, it is difficult to pick up only a certain user's voice. An image-based approach [2] has a similar problem in the view point of daily use. If a wearable system such as a google glass becomes popular, all-day monitoring of a user's face may be realized. However, even with such a leading-edge device, the battery amount restricts its camera to run less than 1 hour. Therefore, image-based lifelog system is not realistic. Brain-based approach is one of the most prospective approach that can precisely estimate user's emotion. Nevertheless, an external device for monitoring a user's brain wave is required and its cost is high.

As a novel approach, we focus on a user's touch operation behavior while using a smartphone. Like a keystroke dynamics [3], we expect that a user's emotion, personality and skill reflects the user's touch behaviors. A wipe-speed and its frequency might be different among a various situations (waiting response or not). To prove this assumption, we first develop a system for logging all the touch operations on a smartphone. Some application can obtain the user's touch operation for the application. Click Tile Touch [4] is a famous commercial software for embedding this function easily. However, the user's touch operation for other applications or OS itself cannot be observed from the application. Therefore, we need to develop a novel system for obtaining all the touch operations.

In this demonstration, we exhibit our developed system with some analytical results. Our system focuses on a device/event log output from an operating system. For example, if a user touches a display, some logs are output at */dev/input/event**. However, this directory is different between terminals, and the format is not easy to understand for a human. In addition, it includes neither timestamps nor application names. Therefore, we clarify the relationship between the log and real touch operations. We have implemented a system so that it can recognize several touch operations, and analyze a frequency and speed of each identified touch operation statistically.

II. Implementation

The current implementation of our system adopts Android Debug Bridge (as described below in adb) included in the platform-tools of Android SDK. It means that our system requires an external computer for analysis. The advantage of this architecture is that it does not require to add any special software to a target Android terminal. Also, rooted-terminal is not necessary for our system. (If we use a rooted-terminal, our system can run on arbitrary Android terminal without an external computer.)

For analysis and visualization, we use Python and matplotlib. Currently, we have already confirmed with real terminals that our system can be used for the following Android OS versions: 4.2.4, 4.2.1, 4.1.2, 4.0.4, 4.0.2, 2.3.4.

Our system is composed of 4 functions: (i) how to obtain a touch log with a timestamp and current application, (ii) how to recognize each touch operation, (iii) how to calculate a swipe speed, and (iv) how to visualize the result.

A. Getting touch operation log

Android OS outputs an event log to */dev/input/event** similarly to normal Linux OS. Therefore, our system keeps observing this log through adb tools. Since a timestamp in an event log is a relative value from the latest wakeup, and its format is slightly different among terminals, our system transforms it to an absolute unix time, as a common format. At the same time, our system recognizes a current application through a system dump. Figure 1 shows the actual touch log data that have been output.

978-1-4799-2231-4/14/$31.00 ©2014 IPSJ

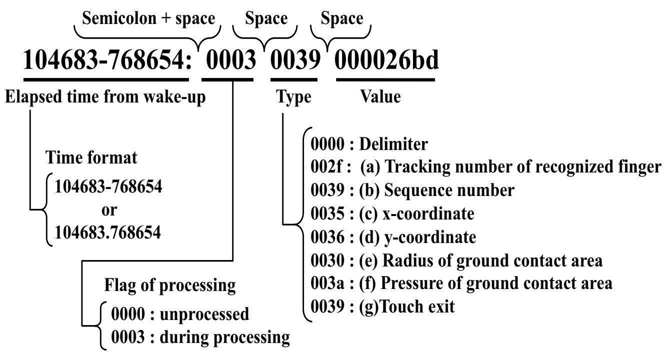

Fig. 1: Pure log one sentence

B. Analyzing touch operation log data

Since a raw log data is not easy to recognize for a human, we investigate the meaning of each line and definition of each touch operation. Touch operation behavior can be categorized into the following 7 behaviors: Single touch (Fig.2 Left), Multi touches (Fig.2 Right), Single swipe (Fig.3 Left), Multi swipes (Fig.3 Right), Pinch in (Fig.4 Left), Pinch out (Fig.4 Right) and Rotate (Fig.5). Our system supports recognition of all of these behaviors.

Fig. 2: single/multi touch

Fig. 3: single/multi swipe

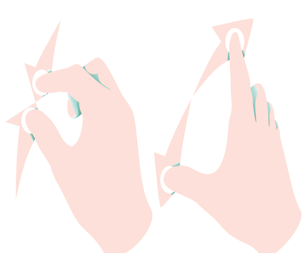

Fig. 4: pinch in/out

Fig. 5: rotate

C. Statistic analysis of touch operation

In order to estimate a user's context such as an emotion or a skill, we need to analyze features of each touch behavior. Perhaps, the features are different among persons, applications and situations. To define the feature, we not only count each behavior but also calculate an average speed.

D. Visualizing touch operations

Our system can depict a touch operation in real-time as shown in Fig. 6. We designed that each touch is drawn by a circle whose radius represents a touch pressure[1]. The color is automatically changed when a current application is changed.

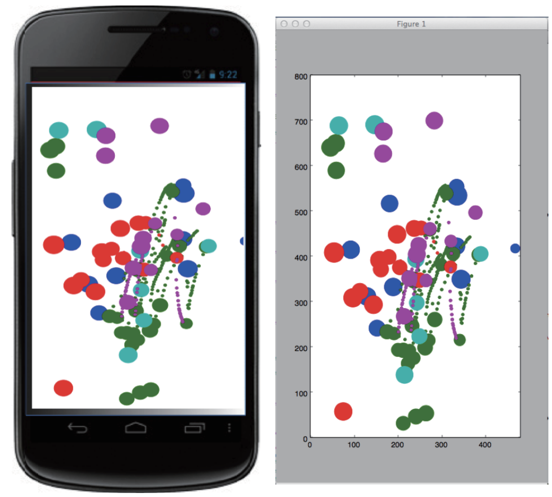

Fig. 6: Image of visualization

III. Demo

We demonstrate that our system can log all the touch behavior. Additionally, we show that a touch behavior is different depending on the person even when the same application is used. In our demo, we use GalaxyS II Android OS 4.0.2 as target devices.

IV. Conclusions and Future Work

The goal of our research is to obtain some of users' internal contexts through touch behavior. We implemented a novel system for logging all the touch operation in Android. Our system can recognize 7 popular touch operations and can analyze each operation's frequency and speed statistically. In the future, we will extend our system to estimate one's emotion and skills.

Acknowledgment

This work was partially supported by Grant-in-Aid for Exploratory Research (25540031).

References

[1] N. Kawaguchi, N. Ogawa, Y. Iwasaki, K. Kaji, T. Terada, K. Murao, S. Inoue, Y. Kawahara, Y. Sumi, and N. Nishio, "Hasc challenge: gathering large scale human activity corpus for the real-world activity understandings," in *Proceedings of the 2nd Augmented Human International Conference*. ACM, 2011, p. 27.

[2] K. Fukumoto, T. Terada, and M. Tsukamoto, "A smile/laughter recognition mechanism for smile-based life logging," in *Proceedings of the 4th Augmented Human International Conference*. ACM, 2013, pp. 213–220.

[3] F. Monrose and A. D. Rubin, "Keystroke dynamics as a biometric for authentication," *Future Generation Computer Systems*, vol. 16, no. 4, pp. 351–359, 2000.

[4] "Click tale touch," http://www.clicktale.com/products/clicktale-touch.

[1]However, a touch pressure is not always output in the log