LV-Linker: Supporting Fine-grained User Interaction Analyses by Linking Smartphone Log and Recorded Video Data

Hansoo Lee, Sangwook Lee, Youngji Koh, Uichin Lee
School of Computing, KAIST, Republic of Korea
{hansoo,sangwooklee,youngji,uclee}@kaist.ac.kr

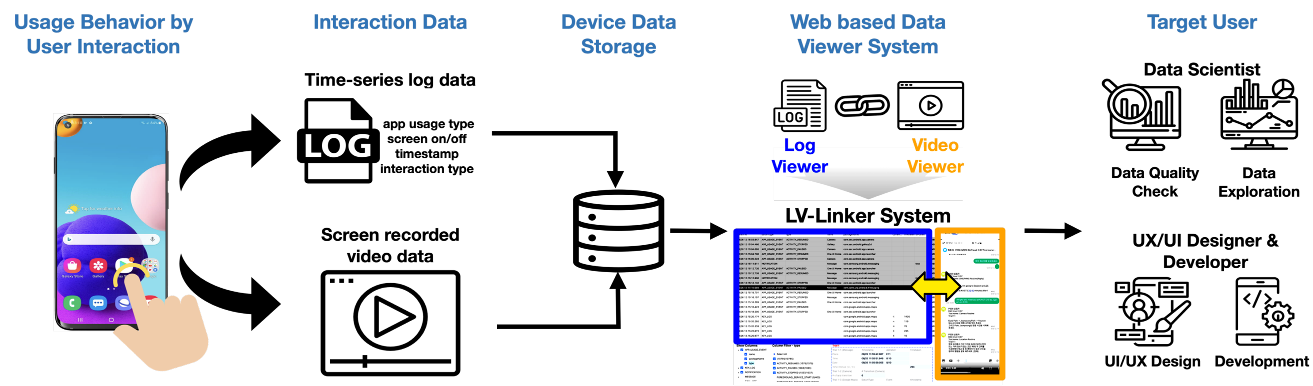

Figure 1: LV-Linker System Overview

ABSTRACT

Data-driven mobile design as an important UI/UX research technique often requires analyzing recorded screen video data and time-series usage log data, because it helps to obtain a deeper understanding of fine-grained usage behaviors. However, there is a lack of interactive tools that support simultaneously navigation of both mobile usage log and video data. In this paper, we propose LV-Linker (Log and Video Linker), a web-based data viewer system for synchronizing both smartphone usage log and video data to help researchers quickly to analyze and easily understand user behaviors. We conducted a preliminary user study and evaluated the benefits of linking both data by measuring task completion time, helpfulness, and subjective task workload. Our results showed that offering linked navigation significantly lowers the task completion time and task workload, and promotes data understanding and analysis fidelity.

CCS CONCEPTS

• **Human-centered computing** → **Ubiquitous and mobile computing**; **Systems and tools for interaction design**; **Visualization systems and tools**.

ACM ISBN 978-1-4503-9321-8/22/10.
https://doi.org/10.1145/3526114.3558714

KEYWORDS

Mobile Usage Behavior Analysis, Data-Driven Design, Mobile User Interface and Experience Design, Mobile Data Visualization Tool

ACM Reference Format:
Hansoo Lee, Sangwook Lee, Youngji Koh, Uichin Lee. 2022. LV-Linker: Supporting Fine-grained User Interaction Analyses by Linking Smartphone Log and Recorded Video Data. In *The Adjunct Publication of the 35th Annual ACM Symposium on User Interface Software and Technology (UIST '22 Adjunct), October 29-November 2, 2022, Bend, OR, USA.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3526114.3558714

1 INTRODUCTION

Objective data collected from smartphones facilitate mobile user interface and experience (UI/UX) design. Unlike self-reported user data, user interaction and app design data-driven design leverage fine-grained analyses of interaction behaviors such as app use routines, and reaction behaviors [3]. For example, UI/UX researchers can improve the accessibility of app interactions by examining frequent usage patterns. Furthermore, developers and data scientists can use objective usage data, along with passively collected sensor data, to enable novel data-driven computing services as well (e.g., digital phenotype, usability & performance testing, app development, privacy and security, data quality check) [12].

Objective usage data, which is mainly used in data-driven UI/UX research for smartphone usage behavior analyses, include screen recorded video data and time-series log data. Screen-recorded video data can be collected by using a smartphone system built-in or third-party screen recording app, and collecting the smartphone screen and surrounding context through a wearable camera [2, 9]. Screen-recorded video data shows detailed smartphone usage information, helping researchers to easily identify the user behavior patterns without learning difficulties. Time-series log data consists of diverse data sources and types: smartphone app usage history & status (e.g., foreground/background state & time, app name, screen state), touch interaction type (e.g., click, double click, scroll, typing), and touch interaction target (e.g., button, UI view hierarchy/elements, typing letter, notification) [12]. Those log data types are time-series numerical, categorical, and text data.

Log data has many advantages to analyze smartphone usage behavior compared to screen-recorded video data. Log data is easy to remove sensitive data through preprocessing and perform data analysis (e.g., statistical analysis and machine learning) and protect privacy issues when compared with screen recorded video data which captures a lot of personal information [10]. Furthermore, even if large-scale log data are collected for a long time, the required storage is relatively small compared to the video data. Log-data analysis allows researchers to quickly find and analyze a specific event of interest via simple text matching, whereas video data analysis requires complex computer vision techniques. Since log data can easily identify smartphone usage behavior, it is currently used in various research fields more than screen recorded video data to quantitatively analyze smartphone user behavior [1]. Furthermore, log data is widely used by developers or data scientists, but it is relatively less familiar to UI/UX researchers and designers or novice researchers. Log data analysis has a steep learning curve compared to video data analysis. Learning various smartphone usage log data types (e.g., user interaction log, system & device log, surrounding context log, physical activity log) and attributes (e.g., event, value) associated with smartphone usage behavior requires professional knowledge. Unfortunately, existing data collectors lack user-friendly data navigation features (e.g., AWARE [8]).

A simple solution is to synchronize the timestamp of log data and video data to visualize both data simultaneously. Most existing studies linked video and animation/text for audio description [11, 13, 15, 20]. Synchronized multi-modal usage data were also used by researchers. Morrison et al. [14] proposed a system replayer that can summarize statistical data or focus on specific factors of interest by synchronizing images taken while using mobile systems and logs generated from the system. Brown et al. [2] conducted a systematic study that allows users to check wearable cameras in which surrounding environments are recorded for user experience recording when using smartphones. However, there were no previous studies that link time-series smartphone usage behavior log data and video data to support fine-grained smartphone usage behavior analyses.

Therefore, we develop a Log and Video Linker (LV-Linker) system that connects smartphone screen recorded video data and time series log data. LV-Linker is a viewpoint movement interaction system that synchronizes the timestamp of screen recorded video and time-series log data and moves the viewpoint of one data point as well as the viewpoint of another data. Through this proposed system, researchers can easily understand the meaning of time-series log data by watching the corresponding screen recorded data in real time. Furthermore, researchers can analyze smartphone usage behavior patterns quickly and accurately by stitching multiple data sources. We evaluate the proposed system by comparing the performance of the proposed linked system and unlinked system

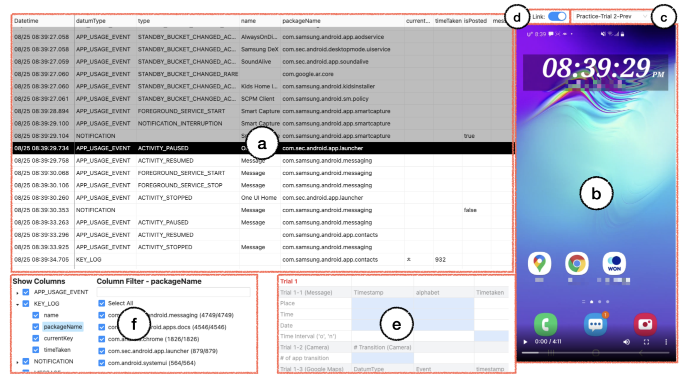

Figure 2: The user interface of LV-Linker: (a) log viewer, (b) video player, (c) dropdown for video selection, (d) link/unlink toggle switch, (e) task sheet, (f) filter function

in terms of user behavior analysis. In addition, we examine how the prior experience of data-driven UI/UX research affects the use of the proposed system.

2 LV-LINKER: LINKING USAGE LOG & VIDEO

2.1 System Description

We developed LV-Linker to help researchers evaluate how linking user's smartphone usage logs and videos affects data quality check (correctness), fast data analysis, and log data understanding. Log viewer shows the processed logs in the form of a spreadsheet as shown in Figure 2 **(a)**. The video player shows the smartphone usage video that records users' smartphone usage behavior pattern on the smartphone in Figure 2 **(b)**. It supports a basic seek bar to change into a specific frame. The users can select the video with the dropdown menu as depicted in Figure 2 **(c)**. LV-Linker integrates the log viewer that shows the processed logs in a spreadsheet into the video player that shows user behavior on the smartphone as depicted in Figure 2. We added a toggle switch to measure the effects of linking only to link or unlink the log and video in the same UI as shown in Figure 2 **(d)**.

We used a custom app usage logger by Android built-in APIs [4–7] to collect the usage log data such as the app usage status (e.g., start & end time, app & package name), notification (e.g., posted, app & package name), and typing event (typed letters & time, app & package name), touch interaction type (e.g., click), device event (screen on/off, battery status, power on/off), call & SMS logs. The usage log data consists of metadata such as timestamp, datum type, and JSON data with various attributes (e.g., package name, app name, type, is posted). Our system appends metadata to the parsed JSON data and presents the data on the sheet.

As shown in Figure 2 **(f)**, users can select the data they want to see using the filter function on the log viewer. The filter consists of two stages: (1) Show/Hide columns, and (2) Filter by the value of the selected column. The columns are grouped by datumType, which is the type of app usage log data (e.g., app usage event, notification, key log, and device event) because each datumType has different sets of columns. For example, in our experimental tasks, we used a filter feature to exclude non-relevant columns and datum types (e.g., device event, call & SMS, touch interaction type). In this case, the remaining logs include APP_USAGE_EVENT, KEY_LOG, and NOTIFICATION datumType. The remaining columns are timestamp, datumType, type (e.g., app usage interaction type), name, package name, currentKey, timeTaken, and isPosted.

The app usage log and video data are synchronized based on the timestamp. Thus, when users select a specific frame on the video player, the log viewer highlights a log recorded at the same frame. Conversely, when the users select a specific log on the log viewer, the video player seeks the corresponding time frame on the video player. Furthermore, the system embedded the task sheet for participants to immediately copy and paste the required analysis results as depicted in Figure 2 **(e)**. The system was implemented and deployed on the web frontend based on TypeScript and React.

3 PRELIMINARY EVALUATION

3.1 Methods

To showcase the benefits of LV-Linker, we consider the following data analysis tasks that could be used in the usability evaluation experiments. Twelve videos were recorded by a smartphone screen recording app at different times. Each video contains 6-minute app usage behaviors of the five app tasks (answering a call, sending a message, taking and deleting pictures, sharing a route, and transferring money) which are the most frequently used smartphone apps [19]. We ordered these videos randomly and placed them within the dropdown menu so that participants can move on to the following video for every task as depicted in Figure 2 **(c)**.

We recruited eight postgraduate students (5 female, 3 male) from HCI research laboratories. Before experiments, participants were asked to perform a preliminary survey. The preliminary survey asked questions about prior knowledge of smartphone usage log data. Through this survey, participants were classified into experienced groups (two people) and non-experienced groups (six people).

Then, we explained the LV-Linker system and the four tasks the participants should perform. Four tasks are selected based on the app usage data analyzed in previous studies [16–18]. The detailed description of each task is below.

- Task 1. App keyboard typing analysis: Finding text typed information in the SMS app and a time interval between two entered specific letters.
- Task 2. The number of app transition analysis: Finding the number of app transitions between camera & gallery app.
- Task 3. App usage start/end time analysis at a different time: There are two videos recorded at different times and log data which includes logs in two videos. In the first video, participants find the time when the notification was posted and the message was shared on Google Map. In the second video, participants find the time when the notification was unposted and the SMS app ended.
- Task 4. App usage start/end time analysis in continuous time: Finding when an app started and ended in the banking app.

In the main experiment, participants were divided into two groups (group A and B) according to the order of use of the linked and unlinked systems, and performed the above four tasks. Group A used the unlinked system after using the linked system, and group B is in the reverse order. The total experiment was 90 minutes long. We asked for the same four tasks for each session so that participants work on two iterations (2 practice and 2 main). In the practice session, participants learned how to use the system and basic knowledge of app usage data. For the system evaluation, we measured the task completion time and conducted a post-survey; i.e., a helpfulness survey and the NASA Task Load Index (NASA-TLX) survey. Task completion time was measured every time participants finished each task by a stopwatch. The post-survey asked questions about the helpfulness of the linked system for four tasks as 5-point Likert scales from three perspectives: 1) correct analysis, 2) quickness analysis, and 3) data understanding. We measured cognitive load using NASA Task Load Index (NASA-TLX) when analyzing smartphone usage data with linked and unlinked systems, respectively. Lastly, we asked for user feedback about the system.

3.2 Results

We examined the normality of Kolmogorov-Smirnov and Shapiro-Walk to determine statistical analysis methods. The two-way mixed ANOVA repeated measures (RM) and Tukey post hoc are used.

3.2.1 *Task Completion Time.*

In all participants and all tasks, the task completion time was reduced compared to the unlinked system. In terms of learnability, the main session of task completion time was much shorter compared to the practice session. In particular, group A, which used the linked system first, has much less task completion time than the practice session in the main session compared to group B. Accordingly, the linked system can help users to understand the log data, so the learning effect was high when the users used the linked system first. The difference in task completion time within linked/unlinked systems and between expert/novice groups was significant in Task 2, 3, not Task 1, 4 ($p<0.05$). Because Task 1, 4 is easier (see more video than log) than Task 2, 3. Therefore, the more difficult the task is, the more effective the linked system is than the unlinked system, and this tendency may vary more depending on the prior experience of using data.

3.2.2 *Helpfulness & Task Workload.*

In the helpfulness survey, 75% of respondents answered that the linked system was more helpful than the unlinked system in terms of data understanding, correctness, and quickness as shown in Figure 3. Especially they answered that the linked system was more helpful than the unlinked system in Tasks 3 and 4 in Figure 4. Therefore, the more difficult the usage behavior analysis task is, the more helpful the linked system is in terms of data understanding and accurate and fast analysis. Through this survey, we found that the linked system helps users quickly and correctly analyze smartphone usage behavior and easily understand each log. In the NASA TLX survey, the average load reduction of the linked system compared to the unlinked system of NASA TLX among all participants in the experiment was performance (42%), physical demand (35%), temporal demand (31%), effort (30%), mental demand (24%), and frustration (20%) as shown in Figure 5. We found that the linked system had a lower load compared to the unlinked system in both expert and novice groups.

3.2.3 *Subjective User Experience Reports.*

We received short answers on the advantages & disadvantages of linking log & video and system improvements from all participants. Participants responded that linking was very helpful for quickness and data understanding.

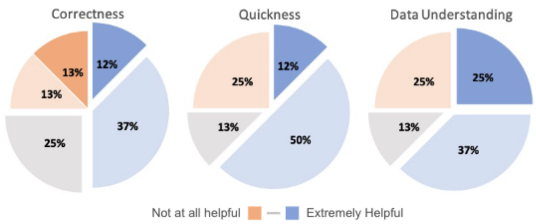

Figure 3: Helpfulness of Linked System

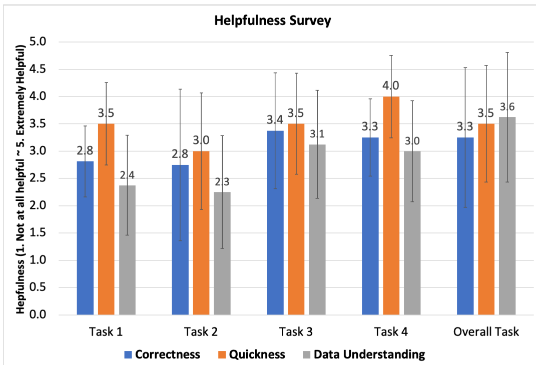

Figure 4: Helpfulness of Linked System for each Task Result

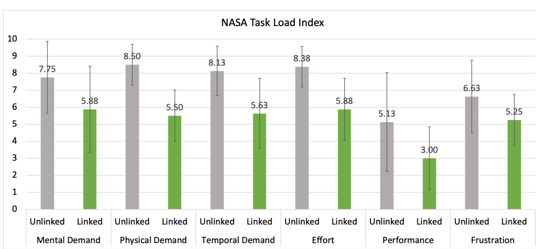

Figure 5: NASA Task Load Index Result

In particular, the linked system was helpful to reduce fatigue caused by tasks by minimizing scrolling when finding specific logs in the entire data. Non-experienced participants said that it was good to intuitively know which data is related to which activity. Experienced participants said that the location where the log starts can be immediately found through the video, so the action of scrolling to the corresponding part is omitted, which is likely to increase speed. As a system improvement, they responded that it would be good to add a function to automatically pause the video when certain log data is selected during video playback, and a function to label log event information on the video progress bar.

4 CONCLUSION AND FUTURE WORK

This study proposed an interactive visualization tool that supports a linked view of smartphone usage log and video data. A simple experiment was conducted to verify the effectiveness of smartphone user behavior analysis through quantitative and qualitative evaluation. The proposed LV-Linker system is expected to support diverse smartphone usage behaviors for data-driven UI/UX research. In future work, the LV-Linker system can be enhanced to examine data quality issues and identify the user behavior analysis of diverse touchscreen-based smart devices (e.g., smartwatches).

ACKNOWLEDGMENTS

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Korean government (MSIT) (2020R1A4A1018774, 2022R1A2C2011536).

REFERENCES

[1] Mehdi Boukhechba, Yu Huang, Philip Chow, Karl Fua, Bethany A Teachman, and Laura E Barnes. 2017. Monitoring social anxiety from mobility and communication patterns. In *Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers*. 749–753.

[2] Barry Brown, Moira McGregor, and Eric Laurier. 2013. iPhone in vivo: video analysis of mobile device use. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*. 1031–1040.

[3] Biplab Deka, Zifeng Huang, and Ranjitha Kumar. 2016. ERICA: Interaction mining mobile apps. In *Proceedings of the 29th annual symposium on user interface software and technology*. 767–776.

[4] Android Developer. 2022. AccessibilityService. Retrieved January 13, 2022 from https://developer.android.com/reference/android/accessibilityservice/AccessibilityService

[5] Android Developer. 2022. NotificationListenerService. Retrieved January 13, 2022 from https://developer.android.com/reference/android/location/LocationListener

[6] Android Developer. 2022. NotificationManager. Retrieved January 13, 2022 from https://developer.android.com/reference/android/app/NotificationManager

[7] Android Developer. 2022. UsageStatsManager. Retrieved January 13, 2022 from https://developer.android.com/reference/android/app/usage/UsageStatsManager#constants_1

[8] Denzil Ferreira, Vassilis Kostakos, and Anind K Dey. 2015. AWARE: mobile context instrumentation framework. *Frontiers in ICT* 2 (2015), 6.

[9] Maxi Heitmayer and Saadi Lahlou. 2021. Why are smartphones disruptive? An empirical study of smartphone use in real-life contexts. *Computers in Human Behavior* 116 (2021), 106637.

[10] Philipp Krieter and Andreas Breiter. 2018. Analyzing mobile application usage: generating log files from mobile screen recordings. In *Proceedings of the 20th international conference on human-computer interaction with mobile devices and services*. 1–10.

[11] Chiman Kwan, Jenson Yin, and Jin Zhou. 2018. The development of a video browsing and video summary review tool. In *Pattern Recognition and Tracking XXIX*, Vol. 10649. International Society for Optics and Photonics, 1064907.

[12] Hansoo Lee, Joonyoung Park, and Uichin Lee. 2021. A Systematic Survey on Android API Usage for Data-Driven Analytics with Smartphones. *ACM Computing Surveys (CSUR)* (2021).

[13] Martin Merkt, Sonja Weigand, Anke Heier, and Stephan Schwan. 2011. Learning with videos vs. learning with print: The role of interactive features. *Learning and Instruction* 21, 6 (2011), 687–704.

[14] Alistair Morrison, Paul Tennent, and Matthew Chalmers. 2006. Coordinated visualisation of video and system log data. In *Fourth International Conference on Coordinated & Multiple Views in Exploratory Visualization (CMV'06)*. IEEE, 91–102.

[15] Xiangming Mu. 2010. Towards effective video annotation: An approach to automatically link notes with video content. *Computers & Education* 55, 4 (2010), 1752–1763.

[16] Xiangang Qin, Chee-Wee Tan, Effie Lai-Chong Law, Mads Bødker, Torkil Clemmensen, Hequn Qu, and Diandi Chen. 2018. Deciphering the Role of Context in Shaping Mobile Phone Usage: Design Recommendations for Context-aware Mobile Services from a Cross-Cultural Perspective. In *Proceedings of the Sixth International Symposium of Chinese CHI*. 39–48.

[17] Choonsung Shin, Jin-Hyuk Hong, and Anind K Dey. 2012. Understanding and prediction of mobile application usage for smart phones. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*. 173–182.

[18] Aman Kr Singh, Ashish Kr Prajapati, Vikash Kumar, and Subhankar Mishra. 2017. Usage analysis of mobile devices. *Procedia computer science* 122 (2017), 657–662.

[19] Statista. 2020. Most frequently used smartphone apps in South Korea as of September 2020, by category. Retrieved January 13, 2022 from https://www.statista.com/statistics/897227/south-korea-frequently-used-smartphone-apps-by-category/

[20] Daisuke Yamamoto, Tomoki Masuda, Shigeki Ohira, and Katashi Nagao. 2008. Video scene annotation based on web social activities. *IEEE multimedia* 15, 3 (2008), 22–32.