LoggerMan, A Comprehensive Logging and Visualization Tool to Capture Computer Usage

Zaher Hinbarji, Rami Albatal, Noel O'Connor, and Cathal Gurrin

Insight Centre for Data Analytics,
Dublin City University
zaher.hinbarji@insight-centre.org
https://www.insight-centre.org

Abstract. As we become increasingly dependent on our computers and spending a major part of our day interacting with these machines, it is becoming important for lifeloggers and human-computer interaction (HCI) researchers to capture this aspect of our life. In this paper, we present LoggerMan, a comprehensive logging tool to capture many aspects of our computer usage. It also comes with reporting capabilities to give insights to the data owner about his/her computer usage. By this work, we aim to fill the current lack of logging software in this domain, which would help us and other researchers as well to build data sets for HCI experiments and also to better understand computer usage patterns. Our tool is published online [1] to be used freely by the community.

Keywords: logging tool, computer usage, lifelogging, human-computer interaction, user modeling

1 Introduction

Lifelogging is seen as the process of tracking personal data generated by our own behavioral activities. While most of the related work on lifelogging is focused on gathering and analysing visual and physical activity data, in this work we take this further by including HCI tracking and analysis. In our modern society computers have become a core part of our professional and personal environments, which makes capturing HCI activities a rich source of information for better insights into our life. Capturing HCI data is the first step toward several applications, ranging from simple automatic memory/diary generation, to the discovery of user's interests, skills, preferences and mood. Personalisation, profiling and security applications are common examples of how such lifelog data can be utilised. We believe that the passive continuous tracking of HCI can reveal important information about the user, which can be used to perform advanced analytics and to build various personalised services. In this work, we present a comprehensive logging tool to capture a wide range of our human-computer interactions.

[1] http://loggerman.org/

2 Background

Although one can easily find a commercial spyware/surveillance software providing the ability to monitor and record the computer activities, it is not easy to find dedicated and comprehensive tools for logging computer usage for analysis purposes. In [1], authors discussed using surveillance software for conducting HCI experiments. They described several issues related to time stamps, mouse clicks, web usage and some other missing functionality in such software. The current available software is designed to track interaction with one application. For example, *WOSIT*[2] can be used to observe user actions on one application's UI in UNIX systems. *OWL*[3] is another application-specific logging tool that tracks Microsoft Word usage.

Several works have utilised HCI logging for various applications. For example, the use of keystroke logging tools in the domain of cognitive writing research has created new possibilities by providing detailed information about the writing process that was not accessible previously. Particularly, key-loggers are utilised in studies on writing processes, description of writing strategies, children writing development, writing and spelling difficulties analysis [4] [5] [6]. Inputlog is an example of such logging tool [7]. Keyboard logging also can be integrated in educational fields for programming and typing skills [8]. In addition, detailed timing of keystrokes can be employed for security purposes [9], [10] and for stress [11] and emotion detection [12]. Screenshots are another example of possible logging technique that can be used to capture and analyse the content consumed or produced by the user. Screenshots were applied for search and task automation[13], and for building user friendly help manuals and demonstrations [14].

Furthermore, intrusion detection and authentication are common applications based on HCI logging. In [15], mouse curves are logged and used as a signature to authenticate users. Several mouse usage features are used to build user profile [16] [17]. Log files, resource and command usage are used to profile users behaviour [18] [19]. In [20], user identification was explored via logging various GUI interactions such as windows switching frequency, time between new windows and number of opened windows.

Motivated by the beneficial potential and applications of HCI logging, and by the lack of generic and comprehensive HCI tracking and visualisation tools, we present *LoggerMan* a generic, passive and application-independent logging solution for researchers and lifeloggers. LoggerMan provides a robust tool to assist data gathering, visualisation and analytics for research community.

3 LoggerMan Overview

LoggerMan helps researchers and lifeloggers to collect interaction data produced during normal computer usage. The main goal of LoggerMan is to work passively in the background, intercept usage events and store them for later analysis. It gathers wide range of keyboard, mouse and UI actions. LoggerMan is published online (LoggerMan.org) for interested researchers/lifeloggers. In the next sections, we will describe the system main components and functionality.

3.1 Privacy and Technical Specification

LoggerMan works under Mac OSX 10.7 or later. To avoid any privacy concerns among users, all captured data is stored locally on the computer. This provides the user with a full control of the data. Buffering techniques are used to store the stream of events efficiently to the hard disk. All modules are designed to be Unicode compatible. Thus, the tool can log texts of different languages properly. It adds itself to the computer's startup to run automatically after a system restart. To maintain user privacy, LoggerMan does not log data typed in secured fields (password fields).

3.2 Modules

LoggerMan has been designed to be flexible and modular. Our tool consists of multiple logging modules which can be switched on/off independently (Fig. 1). Each logging module captures a different type of data and every log-entry is timestamped to the current time.

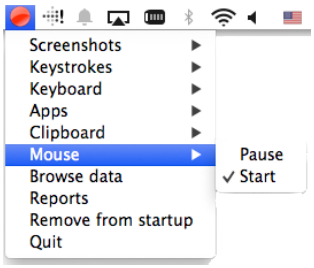

Fig. 1: LoggerMan Menu

Keyboard related Modules. These modules are responsible for capturing keystrokes events and storing them in local files. They provide two levels of detail: word based and key based. Word-based mode (visible as "Keyboard" menu item on Fig. 1) segments and concatenates the events stream into words. Whereas, key-based mode (visible as "Keystrokes" menu item on Fig. 1) tracks each key separately and stores it associated with its time stamp. This particularly important for the domain of keystrokes dynamics [9].

Mouse Module. All mouse actions are traced by this module: move, right/left up/down mouse buttons events, scroll and drag-and-drop actions. We previously utilised the data stored by this module to build an authentication system based on mouse curves [15].

Screenshots Module. This will capture a screenshot of the current active window. The user can select one of the shooting intervals (every 5, 10, 30 sec) or a smart-shooting option, which takes a screenshot: after every window transition (from app to app or even different windows in the same app) and at fixed one minute intervals also. This helps to ensure a reasonable trade-off between capture frequency and storage usage. The screenshot can then provide valuable information about the content that the user is consuming or non-texual content that the user is creating.

Apps Module. This module is designed to track apps transitions. It only logs the currently used app regardless of how it is being used, and as such it offers a simple overview of app usage, without any potential privacy issues of actually capturing the screens.

Clipboard Module. Clipboard module is responsible for tracking copy-paste operations. Any text the user copy to the clipboard is captured and logged.

3.3 Reporting and Insight

The current version of LoggerMan comes with a local reporting tool to visualise and give insights to the data owner about his/her computer usage. Specifically, the reporting tool presents three tag clouds: used apps, typed words and windows titles. These clouds are constructed by computing the normalised frequencies of items occurrence during the selected date range by the user (Fig. 2 and Fig. 4). Information about mouse clicks and typed keystrokes during a time frame is also presented as a graph (Fig. 3). In addition, LoggerMan organises the screenshots and the app usage data in a zoom-able timeline to provide a dyanmic and up-to-date view of computer usage (Fig. 5).

Fig. 2: Apps Cloud

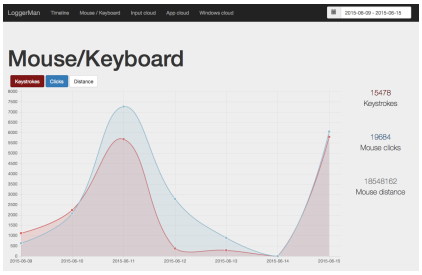

Fig. 3: Mouse/Keyboard Usage

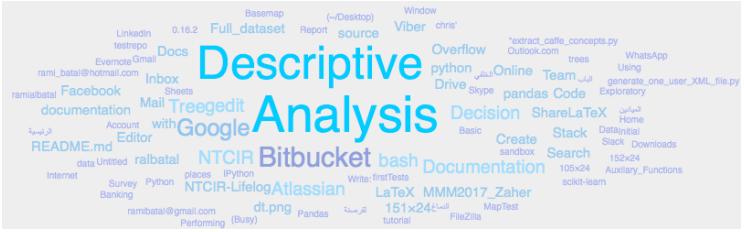

Fig. 4: Windows Titles Cloud

4 Demonstration and Evaluation

This tool is designed to be easy to install and run by any user. The user is all set by just double clicking the app icon for the first time. After that, the app will keep working automatically and passively in the background. Three subjects have been using LoggerMan to capture their computer usage for more than 2 months. Although the data size is very dependent of the running modules and the computer usage intensity, LoggerMan produced around 1 GB of data in a week for average computer usage by each subject. Most of this space (98%) is taken by screenshots. During MMM 2016, our demo will be conducted in a demo mode where conference participants will be able to log their interactions for a short period of time, then they can visualise their data by the reporting module of LoggerMan. Participants can get a copy of their interaction log files before

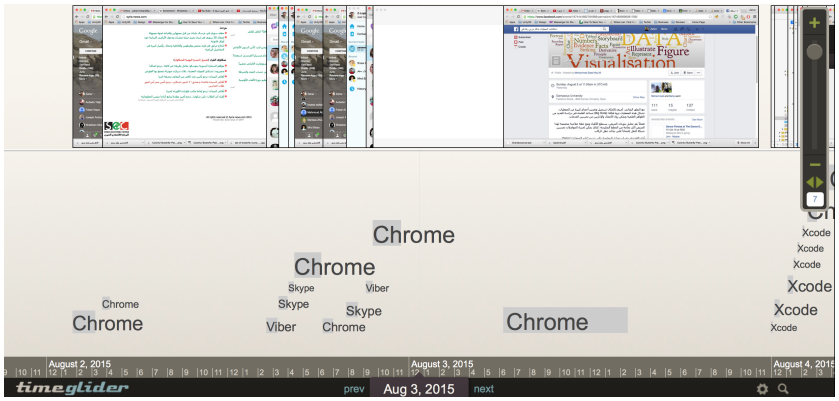

Fig. 5: Apps & Screenshots Timeline

the data being deleted at the end of the session. A personalized report will be generated for each user to show their uniqueness compared to other users.

5 Conclusions

There is a lack of a useful logging tools in the domain of human-computer interaction. In this paper, we presented LoggerMan, a logging and visualisation tool to capture computer activities and to give some insight based on this data. LoggerMan can easily support a wide range HCI related research such as user identification, personalisation, recommendation and cognition.

Acknowledgments This publication has emanated from research conducted with the financial support of Science Foundation Ireland (SFI) under grant number SFI/12/RC/2289.

References

1. Blaise W. Liffick and Laura K. Yohe. Using surveillance software as an hci tool. In *Information Systems Education Conference*, 2001.
2. The MITRE corporation. Widget observation simulation inspection tool. `http://www.openchannelfoundation.org/projects/WOSIT`. Accessed: 2015-08-07.
3. Frank Linton. Owl: A recommender system for it skills.
4. K.P.H. Sullivan and E. Lindgren. *Studies in Writing: Vol. 18. Computer Key-Stroke Logging and Writing: Methods and Applications*. 2006.
5. Marille Leijten and Luuk Van Waes. Keystroke logging in writing research: Using inputlog to analyze and visualize writing processes. *Written Communication*, 30(3):358–392, 2013.

6. Sven Stromqvist, Kenneth Holmqvist, Victoria Johansson, Henrik Karlsson, and Asa Wengelin. *What keystroke-logging can reveal about writing*, volume 18 of *Computer key-stroke logging and writing: methods and applications (Studies in Writing)*, pages 45–72. Elsevier, 2006.
7. Marille LEIJTEN and Luuk VAN WAES. Inputlog: A logging tool for the research of writing processes. Technical report.
8. Manuel Rodrigues, Srgio Gonalves, Davide Carneiro, Paulo Novais, and Florentino Fdez-Riverola. Keystrokes and clicks: Measuring stress on e-learning students. In Jorge Casillas, Francisco J. Martnez-Lpez, Rosa Vicari, and Fernando De la Prieta, editors, *Management Intelligent Systems*, volume 220 of *Advances in Intelligent Systems and Computing*, pages 119–126. Springer International Publishing, 2013.
9. Daniele Gunetti and Claudia Picardi. Keystroke analysis of free text. *ACM Trans. Inf. Syst. Secur.*, 8(3):312–347, August 2005.
10. T. Shimshon, R. Moskovitch, L. Rokach, and Y. Elovici. Clustering di-graphs for continuously verifying users according to their typing patterns. In *Electrical and Electronics Engineers in Israel (IEEEI), 2010 IEEE 26th Convention of*, pages 000445–000449, 2010.
11. Lisa M. Vizer. Detecting cognitive and physical stress through typing behavior. In *CHI '09 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '09, pages 3113–3116. ACM, 2009.
12. A. Kolakowska. A review of emotion recognition methods based on keystroke dynamics and mouse movements. In *Human System Interaction (HSI), 2013 The 6th International Conference on*, pages 548–555, 2013.
13. Tom Yeh, Tsung-Hsiang Chang, and Robert C. Miller. Sikuli: Using gui screenshots for search and automation. In *Proceedings of the 22Nd Annual ACM Symposium on User Interface Software and Technology*, UIST '09, pages 183–192. ACM, 2009.
14. Tom Yeh, Tsung-Hsiang Chang, Bo Xie, Greg Walsh, Ivan Watkins, Krist Wongsuphasawat, Man Huang, Larry S. Davis, and Benjamin B. Bederson. Creating contextual help for guis using screenshots. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, pages 145–154, 2011.
15. Zaher Hinbarji, Rami Albatal, and Cathal Gurrin. Dynamic user authentication based on mouse movements curves. In *21st International Conference on MultiMedia Modelling (MMM 2015)*, Sydney, Australia, 2015. Springer.
16. Ahmed Awad E. Ahmed and Issa Traore. A new biometric technology based on mouse dynamics. *IEEE Trans. Dependable Sec. Comput.*, 4(3):165–179, 2007.
17. Maja Pusara and Carla E. Brodley. User re-authentication via mouse movements. In *Proceedings of the 2004 ACM Workshop on Visualization and Data Mining for Computer Security*, pages 1–8. ACM, 2004.
18. James P. Anderson Co. *Computer Security Threat Monitoring and Surveillance*. 2002.
19. Dit yan Yeung and Yuxin Ding. Host-based intrusion detection using dynamic and static behavioral models. *Pattern Recognition*, 36:229–243, 2003.
20. T. Goldring. User profiling for intrusion detection in windows nt. *Computing Science and Statistics*, 35, 2003.