

# HCIVIEWER - A Tool for Human-Computer Interaction Practitioners

Anas Samara  
School of Computing  
Ulster University, Belfast  
BT37 0QB, UK  
samara-a@email.ulster.ac.uk

Leo Galway  
School of Computing  
Ulster University, Belfast  
BT37 0QB, UK  
l.galway@ulster.ac.uk

Raymond Bond  
School of Computing  
Ulster University, Belfast  
BT37 0QB, UK  
rb.bond@ulster.ac.uk

Hui Wang  
School of Computing  
Ulster University, Belfast  
BT37 0QB, UK  
h.wang@ulster.ac.uk

**This paper presents a tool that has been designed to provide user insight that might be beneficial for researchers working in Human-Computer Interaction disciplines, as well as user experience designers and usability testing practitioners. Using this tool, recordings of user facial information, eye-gaze data and the screen of the targeted user interface will be exploited such that these aspects may be viewed concurrently and in context. Accordingly, the tool permits the researcher to observe several important characteristics at the same time whilst users interact with different graphical user interfaces. We anticipate making this tool available for public use as soon as the design has been fully completed and implemented.**

*Human-Computer Interaction. Usability Testing. User Experience. Utilities.*

## 1. INTRODUCTION

Human-Computer Interaction (HCI) researchers and designers attempt to improve the quality of the interaction between humans and computer systems through a wide range of evaluation techniques and usability tests (Nielsen 1993). Furthermore, researchers often employ different approaches to attaining perceptions about the feelings of the user regarding the interaction, which often includes the use of subjective tools, physiological measurements, and expressive behaviour (Thüring & Mahlke 2007). Subsequently, a lot of usability testing software packages have been implemented; some of these are free to use while others are commercial, hence have an associated cost.

This paper presents a current work-in-progress, entitled *HCIVIEWER*, which is a tool that aims to give the HCI practitioner the ability to visualise different aspects of the user whilst interacting with graphical interfaces. The tool's functionality can potentially disclose usability issues about the interaction, together with emotional and cognitive information associated with the user at the same time. On playback post-session, the tool displays the recorded session through synchronised recordings acquired while the user interacts with the interface of a software application. However, the primary functionality of the *HCIVIEWER* is the analysis and display of the synchronised user data, rather than the recording task itself. Thus, it is the responsibility of the practitioner to prepare the necessary

peripherals and software for recording beforehand. Therefore, the practitioner should make use of two off-the-shelf peripheral devices; a webcam that records the upper part of the user (primarily the face) and an eye-tracker that records eye-gaze information. In addition, video frames of the screen upon which the software application's user interface is displayed are also recorded.

In the next section of this paper, an overview of the recorded measurements and their relationship with different aspects of human interaction is provided. This is followed by a description of the design and implementation of *HCIVIEWER*, along with an overview of the user interface of the tool. The paper concludes with a summary and discussion of forthcoming work.

## 2. BACKGROUND

Humans' performances are better when they are in positive emotional states (Mackie & Worth 1989). Subsequently, it is important for the HCI practitioner to have insights about users' emotional states. In this regard, facial expressions are deemed to be relevant features that can provide an indication about a user's emotional state, and it is known that they are instrumental in revealing affective states and clues to the user's feelings (Afzal et al. 2009; Akakin & Sankur 2010). On the other hand, eye tracking plays a crucial part in usability testing (Ehmke & Wilson 2007). Hence, by means of an eye tracker, the usability tester can track and see where

the user is looking at on the user interface at a particular time, along with the scan path pattern of the eye movement (Poole & Ball 2006). Additionally, measurements such as *fixation*, which is the moment when the gaze stopped at one point on the screen for a duration of time (Steichen et al. 2014), can relate the amount of processing being applied to that specific object in the interface. Furthermore, eye tracking behaviour through fixations indicates the area of interest where the eye movements and most fixations fall within, which implies that this part of the display is an interesting area for the user (Poole & Ball 2006). As well as the pattern of the eye movement, the entropy of the fixations around the interface is interpreted that the user is confused (Ehmke & Wilson 2007). Moreover, pupil dilation and size variation indicates the mental effort exerted and the cognitive load (Iqbal et al. 2004).

### 3. DESIGN

The HCIVIEWER tool aims to display the different recordings concurrently, in order to give the HCI practitioner the ability to infer different aspects about

the interface itself, whilst viewing the user's facial expressions conjointly with eye gaze behaviour when carrying out the interaction. Figure 1 illustrates the design of the primary user interface of the tool. As shown, a recording of the user's face is presented alongside the recording of the screen where the interaction with the user interface was being conducted. In addition, the fixations where the user's eyes were placed on the user interface are given synchronously with the video playback. Several associated measures are also displayed, together with instantaneous pupil size and the average of the pupil size during the whole session. Furthermore, a graph plot is drawn that gives the changes in pupil diameter throughout the session.

The tool gives the practitioner the ability to change a number of default options that entail the eye-gaze measurements: (1) the *window length* used during the aggregation of the measurements; (2) the *rate* that represents the denominator used to calculate the fixation rate; (3) the *area of interest*, which is the circle size that surrounds the fixation point, which is useful in tracking the fixation across different visualisation types and sizes.

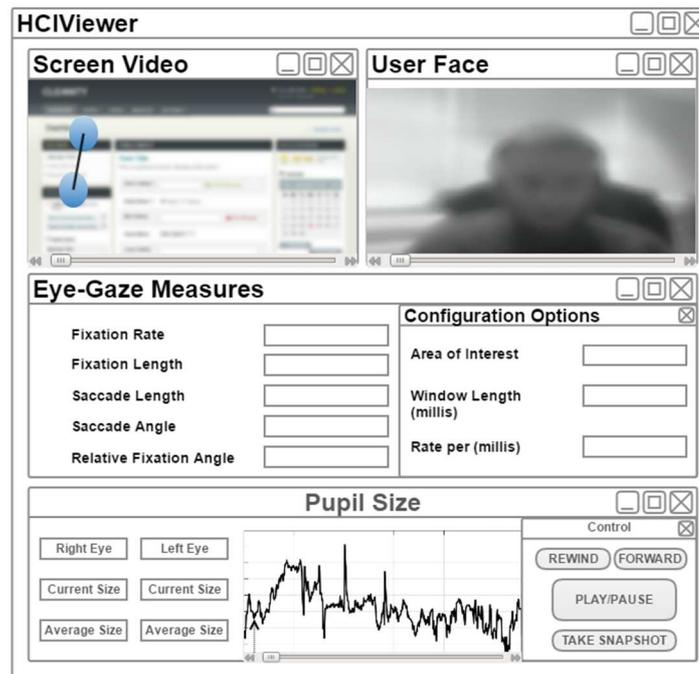


Figure 1: Mock-up design of the HCIVIEWER tool interface.

### 4. IMPLEMENTATION

This section will give technical details about the implementation of HCIVIEWER, the development kit, the exploited application packages, and the logic implemented within the tool.

#### 4.1 Frame and Layouts

The Java programming language has been used to implement HCIVIEWER, due to several advantages, including platform independence, fostering rapid development iterations and the abundance of supported libraries and packages. Subsequently, the *Swing* toolkit API has been used to construct the main graphical user interface components. A *JDesktopPane* container was created as the virtual base that contains the various sub-views, where each sub-view is an instance of a *JInternalFrame*.

Figure 2 presents a screenshot of the current iteration of the tool.

#### 4.2 Video Player

Internally, the VLCJ framework<sup>1</sup> has been employed for rendering the video frames of the recordings of both the user's face and the screen. VLCJ is the

Java bindings of the popular VLC multimedia player, through which one can programmatically access VLC native libraries using Java code (Caprica Software 2009). Therefore, it is assumed that VLC must be installed in order to use HCViewer. Subsequently, the buttons in the control box such as *Play/Pause* invoke the corresponding VLC player functions.

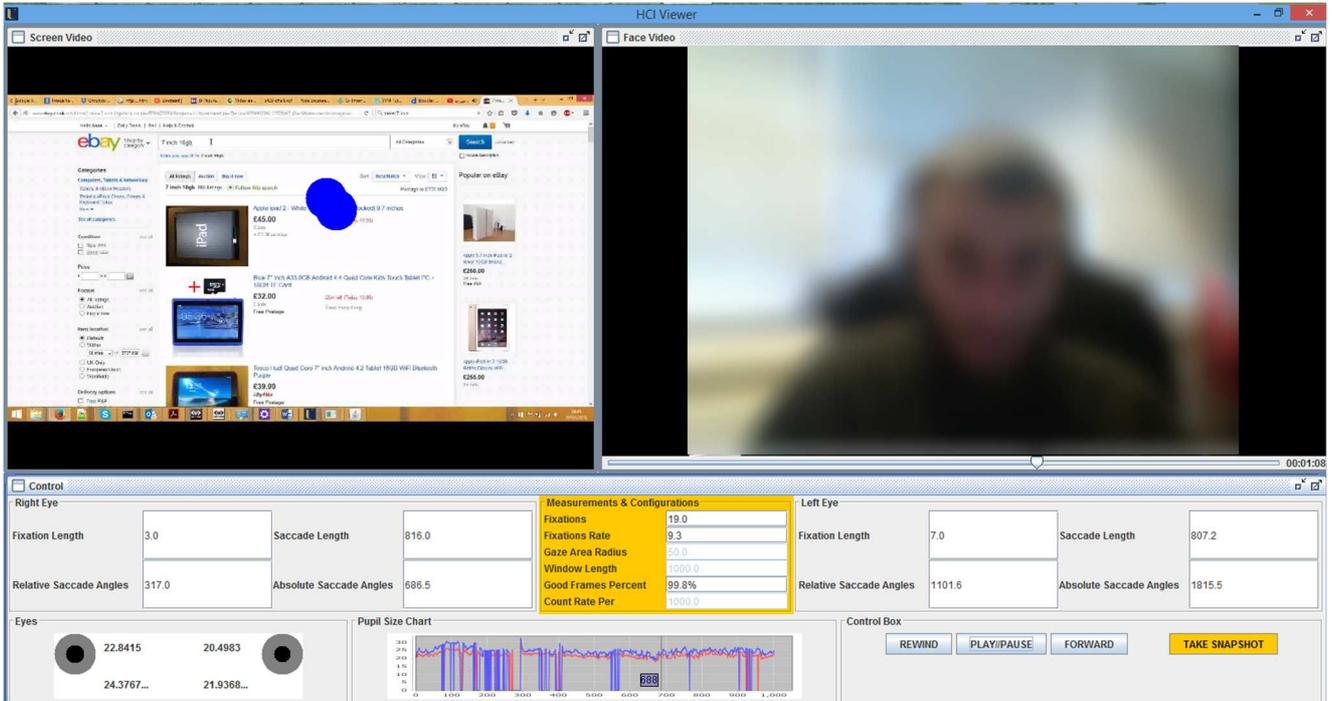


Figure 2: Screenshot of the current implemented iteration of the HCViewer tool

#### 4.3 Parsing of Eye-Gaze Tracking Data

Since the eye tracker usually represents data as a stream of serialised text, a pre-processing step is needed to make it usable. The common format of the data acquired by such devices are either JSON or XML, acquired at a specific rate. For instance, we used the Eye-Tribe infrared-light emitting eye tracker, which provides 30 samples per second as an array of JSON objects. Each JSON object represents a captured sample that contains measurements of both eyes including the pupil size, and the X, Y coordinates of the location of both gazes on the screen. Therefore, the pre-processing process transposes these samples into a map data structure, where the key represents the timestamp of a sample, and the value represents the sample object itself. Consequently, the eye-gaze data and associated measurements are updated and synchronised with the video player using the timestamps. Moreover, the window-based

measures are calculated using a sliding window of the samples.

#### 4.4 Fixations Overlay on the Screen Video

As previously discussed, the eye-gaze data is populated in a map data structure, thus, by accessing the corresponding eye-gaze sample object using the video player timestamp, the corresponding pupil size and coordinate information can be obtained. Subsequently, by embedding a *JWindow* container into the interface screen sub-view component, and using a *Graphics2D* object, a filled circle is painted at the specified coordinates, with the radius calculated based on the *area of interest* value provided in the configuration options sub-view. Furthermore, while the video is playing and updating the current timestamp, the corresponding eye-gaze samples are accessed from the map and the corresponding graphics painted.

<sup>1</sup> <http://capricasoftware.co.uk/#/projects/vlcj>

## 5. CONCLUSION AND FUTURE WORK

This paper presented the design and current implementation of a tool that can be used for practitioners working in the HCI domain. The tool potentially provides the researcher with insights into information that entails different aspects of the graphical interface of the software application being tested, conjointly with contextual information about the eye fixations, as well as the pupil size variation during a recorded user interaction session. At the same time, it can envision the user emotions by showing the video of the user's face whilst he/she is interacting with a software application. Currently, the first iteration of the tool has been used for data acquisition and analysis in a number of experiments by the author. It is the intention that further iterations of development will take place in order to fix some issues that currently reside in the tool. After this, a phase of testing will be conducted to assure the quality of the tool across different platforms before it is publicly released.

## 6. REFERENCES

- Afzal, S. et al., 2009. Perception of emotional expressions in different representations using facial feature points. In *Proceedings - 2009 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops, ACII 2009*.
- Akakin, H.Ç. & Sankur, B., 2010. Spatiotemporal-booster DCT Features for Head and Face Gesture Analysis. In *Proceedings of the First International Conference on Human Behavior Understanding*. HBU'10. Berlin, Heidelberg: Springer-Verlag, pp. 64–74. Available at: <http://dl.acm.org/citation.cfm?id=1881331.1881341>.
- Caprica Software, 2009. VLCJ. Available at: <http://capricasoftware.co.uk/#/projects/vlcj> [Accessed May 11, 2017].
- Ehmke, C. & Wilson, S., 2007. Identifying web usability problems from eye-tracking data. *Hci 2007*, pp.119–128. Available at: <http://portal.acm.org.proxy.lib.iastate.edu:2048/citation.cfm?id=1531294.1531311&coll=ACM&dl=ACM&CFID=74163200&CFTOKEN=35029952>.
- Iqbal, S.T., Zheng, X.S. & Bailey, B.P., 2004. Task-evoked pupillary response to mental workload in human-computer interaction. *Extended abstracts of the 2004 conference on Human factors and computing systems CHI 04*, p.1477. Available at: <http://portal.acm.org/citation.cfm?doid=985921.986094>.
- Mackie, D.M. & Worth, L.T., 1989. Processing deficits and the mediation of positive affect in persuasion. *Journal of Personality & Social*

*Psychology*, 57(1), pp.27–40.

- Nielsen, J., 1993. *Usability Engineering*, Available at: <http://www.useit.com/jakob/useengbook.html>.
- Poole, A. & Ball, L.J., 2006. Eye Tracking in HCI and Usability Research. *Encyclopedia of Human Computer Interaction*, pp.211–219. Available at: [http://go.galegroup.com.proxy1.ci.msu.edu/ps/retrieve.do?inPS=true&prodId=GvRL&userGroupName=msu\\_main&docId=GALE%7CCX3467000044&contentSegment=&searchId=R1&tabID=T003&resultListType=RESULT\\_LIST&currentPosition=1](http://go.galegroup.com.proxy1.ci.msu.edu/ps/retrieve.do?inPS=true&prodId=GvRL&userGroupName=msu_main&docId=GALE%7CCX3467000044&contentSegment=&searchId=R1&tabID=T003&resultListType=RESULT_LIST&currentPosition=1).
- Steichen, B., Conati, C. & Carenini, G., 2014. Inferring Visualization Task Properties, User Performance, and User Cognitive Abilities from Eye Gaze Data. *Tiis*, 4(2), p.11:1–11:29.
- Thüring, M. & Mahlke, S., 2007. Usability, aesthetics and emotions in human–technology interaction. *International Journal of Psychology*, 42(4), pp.253–264. Available at: <http://dx.doi.org/10.1080/00207590701396674>.