Psychophysiological Observing and Analysis Tool for User Experience

Ahmed Husseini Orabi
EECS, University of Ottawa
800 King Edward Ave, Ottawa
Ontario, Canada K1N 6N5
ahuss045@uottawa.ca

Mahmoud Husseini Orabi
EECS, University of Ottawa
800 King Edward Ave, Ottawa
Ontario, Canada K1N 6N5
mhuss092@uottawa.ca

Timothy Lethbridge
EECS, University of Ottawa
800 King Edward Ave, Ottawa
Ontario, Canada K1N 6N5
tcl@eecs.uottawa.ca

ABSTRACT
Studies related to the analysis of psychophysiological responses have become increasingly relevant in HCI. However, the available tools lack systematic and extensive analyses that cover the complexity and asymmetry of psychophysiological sensors, and appropriately synchronize signals with user interactions. In this paper, we present a tool that enables evaluating user experience by linking the psychophysiological responses of users with their interactions. The tool helps collecting and observing the psychophysiological signals, navigation traces, and attention focuses of users while they are interacting with applications. We provide a pipeline-based approach that systemizes automatic analysis, data annotation, and data visualization, as well as affective state detection. We pay attention to the necessity to support extensibility as a way to handle the analysis complexity.

Keywords
Navigation traces; psychophysiological traces; pipeline; extensibility; affective computing; User Emotion Experience

1. INTRODUCTION
Our tool is designed to provide solutions to evaluate user experience, specifically for web applications. The tool provides an easy way to record user interactions and observe their psychophysiological signals. Data is observed and collected remotely.

We support lazy rendering of the massive amount of captured data from multiple sensors.

Real-time synchronization leads to reliable analysis and interpretation of the data observed. Sensor signals must be captured in a real-time manner, such that data recorded becomes associated based on time of occurrences. For example, a trace entry needs to be linked to user interactions including their *gaze* data. This helps an evaluator to know what UI elements the user was interacting with at the time each entry was recorded, as well as the other parts of the screen they looked at.

SEmotion'16, May 17 2016, Austin, TX, USA

ACM 978-1-4503-4169-1/16/05…$15.00
DOI: http://dx.doi.org/10.1145/2897000.2897004

We adopt a *pipeline*-based approach to provide a high level of abstraction, which can also be represented graphically. This approach promotes component-based development, in which a system is modularized as a number of components interacting and transmitting data via connectors and ports. A component is used analyze and interpret the data observed.

Component-based development allows for extensibility, such that a component is developed in a target language of interest; it can be either simple or bridged to external libraries. For instance, computer-vision-related classifiers can use OpenCV to interpret data observed.

A user of our tool can use components developed by other experts, and may only need to arrange the components and connect them together.

Our main contributions can be summarized as follows:

- Collect observational data of users' from different data sources, in a way that
  - Incorporates real-time trackers that allow remote observation of the psychophysiological signals and interactions of users.
  - Enables additional trackers to be integrated.
  - Links between users' affective states and their interaction traces with an application.
- Implement a pipeline-based approach that eases the process of creating user defined analyses and interpretation modules.
- Implement an annotation board, which is used to synchronize the data observed from different sensors based on the user-defined modules developed as pipelines.
- Implement a number of nonintrusive real-time trackers to detect affective states, facial expressions, and eye gazes.
- Implement novel visualization techniques, 3D heatmaps and scanpaths, that associate user experience with observed data, by representing the affective states of users in the third dimension.

2. ARCHITECTURE
The architecture followed in our tool is client/server. Session recording takes place at the client side, and the storing and analysis of the observed data is done on the server.

Our tool is divided into four main independent components, session management service, management dashboard, pipeline workflow, and visualization (top of Figure 1).

A client needs to include a JavaScript file in their main application, and provide contextual details, such as user data. A JavaScript client provides REST services to connect to the