A Multi-Agent Architecture for Tracking User Interactions in Browser-based Games

A.A. Bagherzadeh, S. Rezvankhah, S. Farahi, K. Khalvati, P. Mousavi, A. Dehghan, B. Ghaderi, L. Kashani, H. Moradi

Electrical and Computer Engineering Department

University of Tehran

Tehran, Iran

e-mail: moradih@ut.ac.ir

Abstract—This paper presents a tracking system developed for browser-based games to track user interactions remotely. To eliminate the restriction of accessing local file system, the client side of the system runs on a standalone web server. The tracking agent writes the user interactions with a game on the local file system, and the log-transfer agent sends the results to the server side of the system as soon as an internet connection is available. This system is used in a research to investigate the effects of playing certain games on visual working memory of children under the age 7. This tracking system has provided the games with better user-friendliness by removing browser's permission prompts in JavaScript and Flash-based games. Furthermore, allowing remote data collection from all the gamers provides vast data for future research.

I. INTRODUCTION

Today, playing with computer games is a very popular activity. Many children and even adults spend large amount of their time playing with them [1]. That is why the effects of computer games on people, especially young players, have been the focus of many different studies [2][3]. There are conflicting results showing both positive and negative effects of computer games on users. Although, there are concerns about the effects of violent video games on young players, research results show that some games improve learning problem solving, fine motor skills and hand-eye coordination in young children, it has been investigated and shown in special education as well (if you want to use this sentence, let me know so I can provide you with refs.).

Along the same goal of evaluating the effects of computer games, we started a study to investigate the effects of a set of computer games on the visual working memory of young children under the age 7. Furthermore, we were interested to explore the possibility of using such games as assessment tools.

The evaluation of the impacts of games on users requires a tracking system to log their interactions with the games, without violating the gameplay and overall user-friendliness of the games. Such a tracking system may collect all the mouse movements, keyboard key-presses, head and eye movements, voices and state of the games. In order to process the logged data remotely, the data should be sent to a system for further processing.

In this paper, we present the architecture that is designed to meet all the mentioned concepts such as collecting the mouse movements and clicks, key-press and head-eye movements during a game and sending them to a remote server for the purpose of evaluating the effects of certain games on children. Two versions of the system are available for onsite and offsite usage. The onsite version which is designed to be used in places like research centers is used to collect full motion tracking details and head-eye motion tracking, while the offsite version only tracks coarse mouse motion, mainly at object level, and key-presses to reduce network communication overhead.

The tracking system stores the data locally on the user's computer (Fig. 1), and sends it to the central server for processing. This information will then be used by the psychology team to evaluate the effects of games. One of the features of our tracking system is its user friendliness such that a user is not disturbed by the constant permission prompts and alerts generated by browsers for security issues. It should be noted that the user still has the capability to turn off the tracking system totally, if she or he wishes so. Consequently, the privacy of the users is not at risk. Furthermore, no private information would be collected and only the interactions with the game are tracked.

II. RELATED WORK

Most of the tracking systems are based on one of thefollowing two methods: 1) onsite tracking system and 2) offsite or remote tracking systems. The onsite tracking systems use sophisticated devices such as gaze tracking and Brain Computer Interface (BCI) to follow the users' activities and interaction with games. Consequently, these systems are fairly expensive and require delicate setup. On the other hand, the offsite tracking systems, specifically those which run on web browsers, mainly use cookies to capture the general motion of the user. Although the web-based systems are fairly inexpensive and remotely available, however they cannot collect detailed information. Furthermore, such tracking systems need real-time network connection for collecting and saving user interactions.

To overcome the above limits, we have designed an architecture in which high end equipments are not needed and the information logging can be done remotely even without internet connection during play time. Furthermore, the system is not limited to coarse details, and can collect mouse movements and key-presses for further analysis. Moreover, the system can even capture the user's face on front of computer for further processing. Finally, the proposed tracking system bypasses the permission prompts and alerts while the games run in a web browser to ease the user experience with the game. Based on our best knowledge, there is no similar research which has a tracking system as scalable as the tracking system presented here.

IEEE computer society

III. CLIENT SIDE DESIGN

The games are written such that the user interaction is logged and written to a file to be sent to the server later. Accessing the clients' disk directly by games running under a web browser requires permissions which will be asked every time a game runs, thus reducing the user-friendliness and destroying the flow of the game being played. To take care of this problem, the game runs through a stand-alone web server. The games send the tracking data to the web-server through the browser, and the web-server stores the information locally. This information consists of mouse movements, key-presses and game states, plus the time for each event during a game.

Beside the web browser's security checks, the Flash technology used in this system also has its own Global Security Settings [4] that would not allow scripts and programs access local file system to read or write. To overcome this limit, we used Flash-JavaScript interaction which has no extra security prompts when being run in a web server.
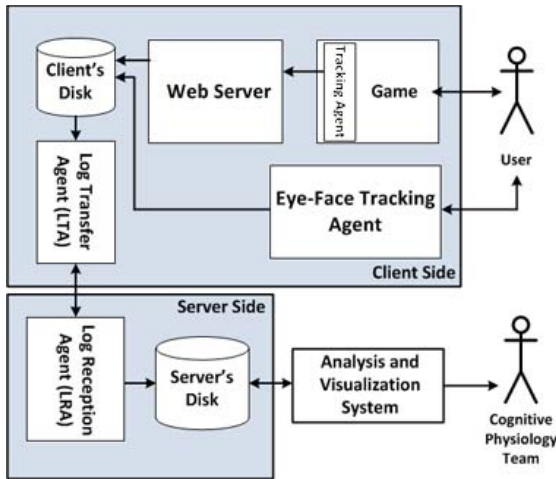

Figure 1. A simplified architecture of the tracking system

To better analyze the collected data, it is important to know whether a user is attended to a game or not. One of the methods to accomplish this task is to evaluate the head pose compared to the computer screen. If the head is fairly tilted away from the screen, it is assumed that the user is distracted and disengaged from the game.

Toward achieving the above goal, our system is capable of using the built-in webcams to track a user's eyes and face to determine if the user is actually looking at the monitor and plays a game or he/she is distracted and is not attended to the game (Fig. 3). In other words, the eye-face tracking system is developed for the purpose of noise reduction. This system detects the time slots that a user is not looking at the monitor, and removes related gathered data.

It is important to mention that due to the privacy issues and the network bandwidth constraints, this feature is not deployed for the remote locations (offsite) and only used in the lab.

However, it is possible to make it available for the remote users if they desire to do so. For instance, such feature is suitable for daycares and kindergartens which have public environments and the privacy issues are less important.

The tracking agent is implemented in two modes: full motion tracking and object-based tracking. The first one tracks all the mouse movements on the screen recorded in (x,y) position of the cursor. This set of data can be used to simulate the exact game conditions and evaluate how the user was playing the game. In the second mode, i.e. object based tracking, the game is partitioned into objects and the tracking system collects the motion of the cursor based on the time the cursor moves on the object, leaves the object or clicks the object.
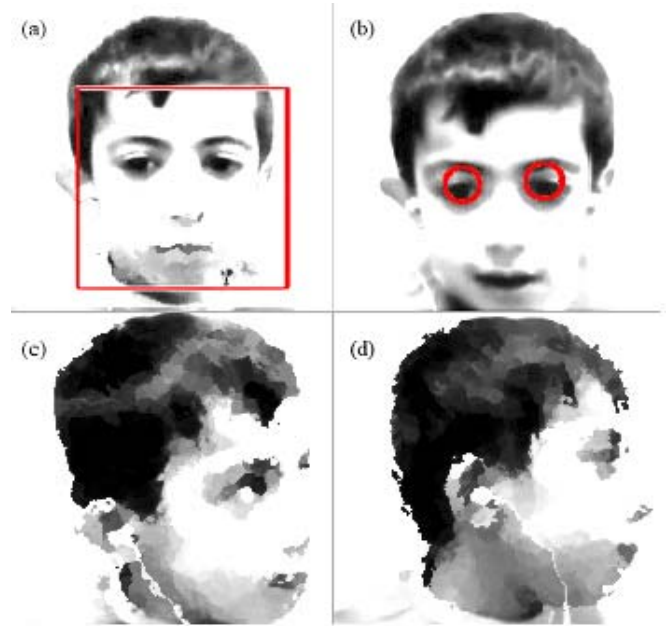

Figure 2. A sample output of the head-eye tracking system. The system is designed to detect users' head if the user is looking at the monitor (a). However, if the system fails to detect the head (due to special conditions such as partially outside the screen), the eye tracking system is used (b). The head-tracking system requires less processing than the eye-tracking system. This comes to the reason of using eye-tracking if the head-tracking system fails. Figures (c) and (d) show two cases which both head and eye-tracking systems fail. As it can be seen the user is not attended to the monitor. The image is partially distorted to preserve the confidentiality of the child.

Finally, after the data is stored on the client side by the web server, the Log Transfer Agent (LTA) transfers the logged data to the server side. LTA resides on the client side and looks for an active connection. As soon as one is available, the logged data is transferred from the client side to the server side and it gets removed from the client side.

IV. SERVER SIDE DESIGN

The server contains the Log Reception Agent (LRA), which receives the data from every client and stores this data on the server. In general, the clients' information is private and no personal information is collected. However, if a user wishes to

cooperate further with the research team, he/she can fill out an online form.

A visualization system is under development to help the psychology team in their evaluation and assessment. The visualization system acts like a replay of a user's interaction with the game, and shows the users' mouse movements step by step.

V. IMPLEMENTATION

The games are based on Flash and JavaScript. We used Server2Go as the stand-alone Apache server in which the games run. The client side tracking agent uses Ajax technology to send the logged data to a logging module written in PHP running on the Apache server. The Log Transfer Agent is written in C# which checks for internet connection and sends this data to the Log Reception Agent that is also written in C#. The LRA and LTA are both designed as Windows Services.

VI. RESULTS AND ANALYSIS

The system has been tested in a preschool, a kindergarten and in our laboratory. As expected the users did not feel any difference in their game play experience and were not distracted by any of the browsers' notifications. The system has proven to be very useful in providing the research with a wide set of data.

The data has been used to determine the correlation between the game output and CANTAB psychological test[1] results. Fig. 3 shows an example of correlation between CANTAB DMS test and the game, which is a high correlation between the game data and CANTAB results. This indicates the possibility of using the games along side or as a replacement of typical cognitive tests as an assessment tool [5].
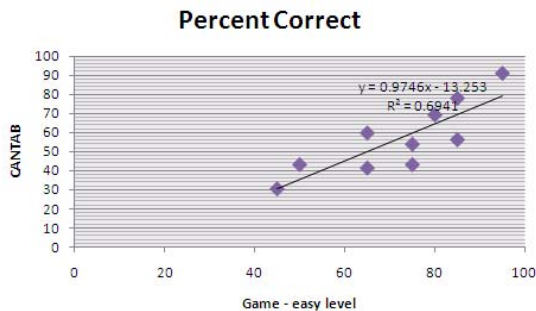

Figure 3. The linear correlation between correct answers in a game and percent correct in CANTAB DMS. This diagram shows that game outputs are significantly correlated to related CANTAB outputs.

Recently, the system is released in limited numbers to the public to further evaluate the capabilities of the tracking and logging agents. It is obvious that this multi-agent tracking system would allow researchers to have access to a large number of test cases. This tracking system not only provided us with the raw data for the sake of research, but also maintains the privacy of the users.

VII. CONCLUSIONS AND FUTURE WORK

A user friendly tracking system based on client-server architecture using multi-agent approach is presented. The client side contains a web-server in which the games run, and an agent which sends the data to the server. The server side contains an agent, which receives the data from clients. The data would be visualized by the visualization system and analyzed by the researchers later.

This multi-agent tracking architecture allowed us to enhance the user-friendliness of the games by removing permission prompts asked every time by the browsers while using highly secured web-server preserves users' privacy.

The future work would be to implement intelligent agents for noise reduction in data and decision making to use the games as assessment tools automatically. Also a compression algorithm will be used to reduce the size of log files and network traffic. In the future, Google Analytics is also going to be tested and modified to be used in the tracking system. One issue in using Google Analytics would be the possibility of sending voice and video information which needs to be explored or implemented. Also we will consider using Microsoft KINECT to determine the engagement of the user with the game. However, it would limit our usage to systems/games that has/uses KINECT.

REFERENCES

[1] http://www.aacap.org/cs/root/facts_for_families/children_and_video_games_playing_with_violence

[2] W.R. Boot, A.F. Kramer, D.J. Simons, M. Fabiani, G. Gratton, "The Effects of Video Game Playing on Attention, Memory, and Executive Control", Journal of Acta Psychologica, 129(3), 378-398, 2008.

[3] G. C. Delacruz, G.K.W.K. Chung, E.L. Baker, "Validity Evidence for Games as Assessment Environments", presented at 9th International Conference of the Learning Sciences, Chicago, 2010.

[4] http://www.macromedia.com/support/documentation/en/flashplayer/help/settings_manager04.html

[5] B. Ghaderi, S. Ekhtiari, H. Moradi, M. Asadpour, "Using a Computer Game for Assessing Visual Memory of Children", Int. Technology Enhanced Education Conf., ICTEE, India, 3-5 January, 2012.

[1] http://www.cambridgecognition.com/