Interactive Hypervideo Visualization for Browsing Behavior Analysis

Luis A. Leiva and Roberto Vivó
Departament de Sistemes Informàtics i Computació
Universitat Politècnica de València (Spain)
{luileito,rvivo}@dsic.upv.es

ABSTRACT

Processing web interaction data is known to be cumbersome and time-consuming. State-of-the-art web tracking systems usually allow replaying user interactions in the form of mouse tracks, a video-like visualization scheme, to engage practitioners in the analysis process. However, traditional online video inspection has not explored the full capabilities of hypermedia and interactive techniques. In this paper, we introduce a web-based tracking tool that generates interactive visualizations from users' activity. The system unobtrusively collects browser events derived from normal usage, offering a unified framework to inspect interaction data in several ways. We compare our approach to related work in the research community as well as in commercial systems, and describe how ours fits in a real-world scenario. This research shows that there is a wide range of applications where the proposed tool can assist the WWW community.

Categories and Subject Descriptors

H.5.1 [**Multimedia Information Systems**]: Video, Evaluation/methodology; H.5.4 [**Hypertext/Hypermedia**]: Navigation

Keywords

User Interaction, Information Visualization, Remote Tracking, Video Synthesis, Interactive Analysis

General Terms

Experimentation, Human Factors, Design

1. INTRODUCTION

Over the years, video analysis has been considered a key evaluation in user interface (UI) design. However, video data is time-consuming to process. Analyzing video has traditionally involved a human-intensive procedure of recruiting users and observing their activity in a controlled lab environment. Such an approach is known to be costly (equipment, personnel, etc.) and rapid prototyping sometimes requires just preliminary studies. Problems like these have led to the development of remote activity tracking for web UI evaluation and user behavior analysis.

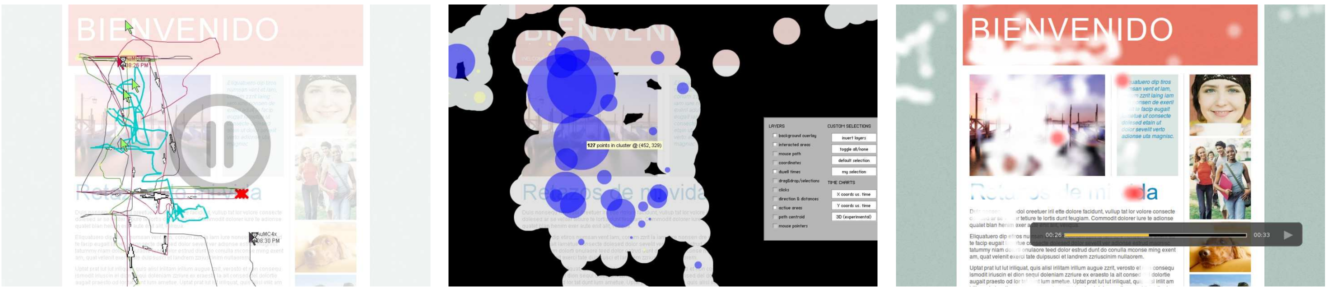
State-of-the-art user tracking systems employ client-side logging tools, which include mouse and keyboard tracking, since these input devices are ubiquitous and therefore neither specific hardware nor special settings are required to collect interaction data remotely. Modern mouse tracking systems usually support replaying the user interactions in the form of mouse tracks, a video-like visualization scheme, to allow researchers to easily inspect what is going on behind such interactions; e.g., *In which order did the user fill in the form fields? Do users ever scroll the web page? If so, how far exactly?* However, traditional online video inspection has not benefited from the full capabilities of hypermedia and interactive techniques. We claim that mixing both channels can better assist the usability practitioner. Therefore, we envision hypervideo to be a useful inspection tool for web tracking. Our system lets the viewer combine multiple user logs in a non-linear structure. The generated movies contain embedded interactive elements, allowing the viewer to manipulate different information layers that modify the video content. This tool is released as Open Source software, and can be downloaded and inspected at `http://smt2.googlecode.com`.

2. RELATED WORK

Mueller and Lockerd [6] set a precedent in client-side tracking, presenting preliminary research on mouse behavior trends and user modeling. Arroyo et al. [1] introduced the concept of collaborative filtering and the idea of using a web-based proxy to track external websites. Finally, Atterer et al. [2] developed an advanced HTTP proxy that tracked the user's every move, being able to map mouse coordinates to DOM elements. Beyond the usefulness of these systems, only [2] could track complex AJAX websites, and visualization was solely the primary focus of [1], although it was limited to an image overlaid on top the HTML pages. Nonetheless, we argue that incorporating the temporal information may enhance the intentionality of mouse movements and hence it may ease human interaction understanding. For instance, hesitations on a text paragraph may indicate interest about that content; or moving the mouse straight to the link of interest would show familiarity with the page. This is where video capabilities come into play, which, to some extent, were implemented lately in industry systems.

Amongst the popular commercial systems at present, ClickTale and UserFly are deeply oriented to web analytics, with limited support for (non-interactive) visualizations. On the other hand, Mpathy and Clixpy are more visualization centered, but they use Flash sockets to transmit data, and so they only would work for users having the Flash plugin installed. Finally, other approaches for visualizing user's

WWW 2012 Companion, April 16–20, 2012, Lyon, France.
ACM 978-1-4503-1230-1/12/04.



(a) (b) (c)

Figure 1: Visualization possibilities. [1a] Replaying users' trails simultaneously, highlighting the average mouse track, and overlaying direction arrows. [1b] Clusters of mouse movements, displaying also masked areas of activity. [1c] Dynamic heatmaps of mouse coordinates and clicks.

activity are DOM based (Tag tracker), or heatmap based (CrazyEgg). Our tool, besides incorporating most of the state-of-the-art features[1], differs significantly from previous work, as stated below.

3. RESEARCH CONTRIBUTIONS

First, a notable strength of our tool, entitled *smt2*, is the ability to composite multiple interaction logs in a single video visualization, which can be rendered at runtime by mixing a variety of infographic layers. This feature has been proven to be useful in assessing the usability of websites, and also to discover common usage patterns by simply inspecting the hypervideos (see Section 5). Secondly, our tracking approach performs a discretization in time of user interactions, following the *polling* technique, i.e., sampling the status of the mouse at regular intervals. This way, our system tracks the user actions as they were exactly performed, allowing also to modify the speed at which movies can be replayed. This is helpful to normalize trajectories that were acquired at different sampling rates when compositing a multi-track hypervideo — specifically, in that case we set a common frame rate of 24 fps; otherwise we use the original user-defined frame rate. Thirdly, another contribution of our approach is the generation of user and page models based on the automatic analysis of collected logs. In this regard, we did not find any related tracking tool that would perform implicit feature extraction from users' interaction data, i.e., interaction metrics inherently encoded in mouse trajectories. We believe that this is a promising line of research, and currently has gained the attention of other researchers (e.g., [4]). In Section 6 we describe some applications of this contribution.

4. SYSTEM DESCRIPTION

Our tool is built on web technologies and hence does not need to install additional software on the client side. The only requirement is a web browser with JavaScript support, so it applies to any modern device capable of accessing the Internet, including smartphones and tablets.

4.1 Architecture

This system uses the WWW infrastructure to log the user activity in a MySQL database. A JavaScript program tracks in the background mouse and browser-related events at a (configurable) registration frequency, measured in fps, and sends the data at fixed-time intervals. Such a program does not interfere with user's browsing experience or other page scripts. Hypervideos are then synthesized on the web server by taking into account the gathered data, the visited pages, and the viewport size of clients' browsers.

4.1.1 Logging Users' Interactions

The system can be invoked manually, but it also can fetch external websites by using a PHP proxy that automatically inserts the required tracking code. We also take into account the user agent string to cache an exact copy of the page as it was originally requested, if desired. Additionally, it is possible to store interaction data from different domains in a single database.

Finally, every lower-level action can be recognized automatically, since the tracking script relies on the DOM event propagation model. We use the UNIPEN format [3] —a popular scheme for handwriting data exchange and recognizer benchmarks— to store the mouse coordinates.

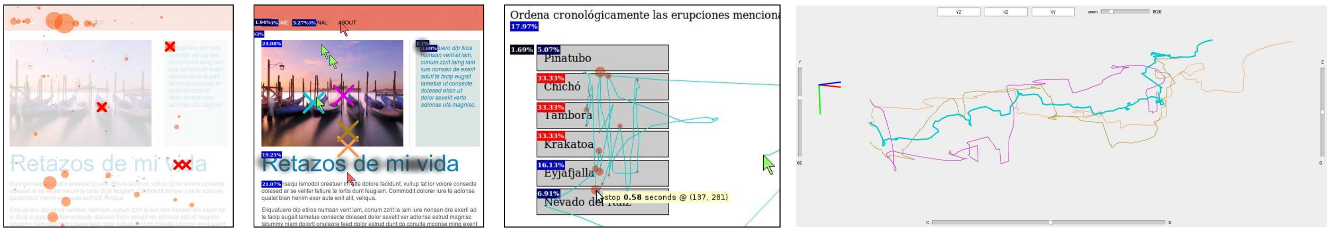
4.1.2 Accessing and Synthesizing Hypervideos

As previously introduced, on the server side a multi-user admin interface is used to manage and deliver hypervideos. First, the system queries the database with the information that the viewer provides. For example, she might request to filter logs by, say, operating system and page ID. In this case the data are merged into a single hypervideo when replaying (Figure 1). On the contrary, though, the viewer might want to visualize a single browsing session to study individual interactions in detail (e.g., Figure 2c). In that case the system will retrieve the subsequent logs to compose a video that will enclose all tracks sequentially. Different mouse trajectories will be normalized according to the original viewport of the user's browser and the actual viewport of the viewer's browser, to bypass discrepancies between screen sizes. Then, a cached copy of the browsed page and the above-mentioned interaction data are both bundled in an hypermedia player.

4.1.3 Analyzing and Interacting with the Data

The viewer can toggle different information layers interactively while she visualizes the videos by means of a control panel. Automatic analysis of interaction features is also feasible for mining patterns within the admin interface, since

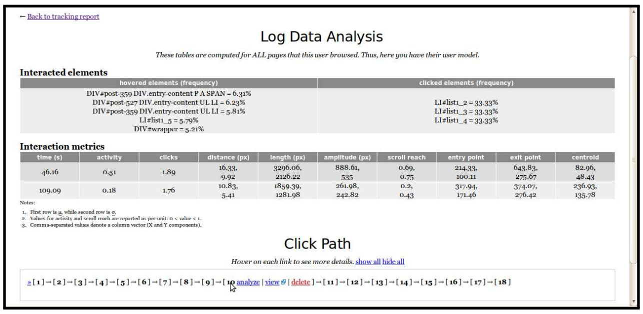
[1]The current version of our tool was publicly released in 2009.



(a) (b) (c) (d) Interactive 3D visualization

Figure 2: [2a] Displaying hesitations (circles) and clicks (small crosses). [2b] Displaying entry/Exit coordinates (cursor bitmaps), motion centroids (big crosses), drag&drop activity (shaded fog), and interacted DOM elements (numbered rectangles). [2c] Analyzing a decision process; the user rearranged items in a list. Small circles represent dwell times. Hovered DOM elements are labelled based on frequency (percentage of browsing time), including a blue color gradient (100% blue: most hovered items). The same scheme is used to analyze clicked items, but using the red palette to colorize labels instead. [2d] Time charts visualization. Bold line is the averaged mouse track, taking into account the selected users. The viewer can rotate the axes with 3 sliders (one for each direction), zoom, and project the lines in the usual YZ, XZ, and XY planes.

collected data are readily available in the database (Figure 3). In this way, besides explicit metadata that is assigned to content, implicit knowledge can help to get a better picture on the nature of such content (see Section 6). Furthermore, movies can be generated for individual users or by taking into account different kinds of segmentations; e.g., time or date intervals, city locations, first-time users, and so on. For instance, the viewer can segment the tracking logs by user ID, and determine which elements were most interacted, or notice the percentage of scroll to infer lostness (e.g., if all browsed pages for a certain user have a minimal scroll reach, that user might be searching for a specific content with no success).



← Back to tracking report

Log Data Analysis

These tables are computed for ALL pages that this user browsed. Thus, here you have their user model.

Interacted elements

hovered elements (frequency)	clicked elements (frequency)
DIV#post-359 DIV.entry-content P A SPAN = 6.31% DIV#post-527 DIV.entry-content UL LI = 6.23% DIV#post-359 DIV.entry-content UL LI = 5.81% LI#list1_5 = 5.79% DIV#wrapper = 5.21%	LI#list1_2 = 33.33% LI#list1_3 = 33.33% LI#list1_4 = 33.33%

Interaction metrics

time (s)	activity	clicks	distance (px)	length (px)	amplitude (px)	scroll reach	entry point	exit point	centroid
46.16	0.51	1.89	16.33, 9.92	3296.06, 2126.22	888.61, 535	0.69, 0.75	214.33, 100.11	643.83, 275.67	82.96, 48.43
109.09	0.18	1.76	10.83, 5.41	1859.39, 1281.98	261.98, 242.82	0.2, 0.43	317.94, 171.46	374.07, 276.42	236.93, 135.78

Notes:

1. First row is μ, while second row is σ.
2. Values for activity and scroll reach are reported as per-unit: 0 < value < 1.
3. Comma-separated values denote a column vector (X and Y components).

Click Path

Hover on each link to see more details. show all hide all

» [1] → [2] → [3] → [4] → [5] → [6] → [7] → [8] → [9] → [10 analyze | view | delete] → [11] → [12] → [13] → [14] → [15] → [16] → [17] → [18]

Figure 3: A log analysis example, reporting all visited pages for a certain user. The first table summarizes the most interacted DOM elements (hovered and clicked, respectively). The second table computes interaction metrics based on mouse activity; e.g., distances, path length, etc. In this table, the first row are mean values, and the second are standard deviations. The click path shows the sequence of browsed pages, allowing three actions over its corresponding log file: visualize, analyze, or delete.

5. APPLICATIONS

The following list briefly illustrates the pragmatic utility of the *smt2* system.

Which areas of the page concentrate most of the interaction? To answer this question, a K-means clustering is performed each time a mouse track ends replaying. So, concentrating on the clustered areas allows to visually notice where users are focusing their actions. Each cluster is represented by a circle with a radius proportional to the cluster population (Figure 1b). This visualization layer is notably appropriate when tracking data are rendered as a static image.

Where do users hesitate? How much? We followed the notion of *dwell time*, i.e., the time span that people remain nearly motionless during pointing at objects, often associated with ambiguous states of mind. In *smt2* dwell times are displayed as circles with a radius proportional to the time in which the mouse does not move (Figure 2a). This visualization helps to measure the time needed to perform certain operations.

Do users perform drag&drop operations? How? A web application can support rearranging widgets to customize their layout. At a lower level, users perform drag and drop to select HTML content. Since we are using the UNIPEN format to encode each pair of mouse coordinates, the status of the click button can be easily represented, so *smt2* provides a specific visualization type for these cases (Figure 2b).

What elements is the user actually interacting with? Whenever an mouse event is dispatched, the tracking script traverses the DOM hierarchy to find if there is an element that relates to that event. Each tracking log holds a list of interacted elements, sorted by time frequency (Figure 2c), so such list can be inspected either quantitatively (by looking at the numbers) or qualitatively (by looking at the colors). This visualization can be helpful to answer low-level questions such as if the users go straight to the content or whether the mouse hovered over a link without clicking.

What is the persistence of the page through time? In this case, a 3D visualization might be useful (Figure 2d). The 3D chart renders each pair of coordinates x, y along the z axis, and provides simple interactive controls to ease further inspection. This way, for a given page, the viewer can observe at a glance the duration of each visit and relate to the rest of them.

Do different mouse tracks correlate? The viewer can project in 2D the z axis of the time chart, and thus observe the evolution of the x, y components of mouse tracks against time. The coordinates are normalized in width and height according to the available chart size, to avoid possible visual biases. Each tick in the x-axis corresponds to the registration frequency used while tracking (e.g., for 24 fps each tick is 1/24 s).

6. EVALUATION: A CASE STUDY

To test *smt2* in a real-world scenario, the system was presented to a team of five graphic designers that were not usability experts. They wanted to redesign a corporative website, and they all used the tool for one month. One of them assumed the super administrator role, and everyone could access to all admin sections. (The only difference between a user in the admin group and the super administrator is that admin users cannot delete the gathered tracking logs.)

6.1 Qualitative Results

By running an informal usability test, potential problems could be identified when visually inspecting the hypervideos. Designers noticed that some areas of the main page layout were causing confusion to most users; e.g., people often hesitated over the main menu until deciding to click a navigational item. Designers could also view that much of the interaction with the site was concentrated around the header section. Consequently, the team introduced modifications to the website and they could compare the generated interactions to previous data. Such updates had notable repercussions specially for first-time visitors (e.g., faster trajectories, less clicks).

Overall, designers found the system very helpful. The main advantages suggested were being able to reproduce exactly what users did in a web page, and the speed with which a redesign could be verified. They also commented that visualizing simultaneous logs was particularly time-saving. It was also reported that *"This tool provides a remarkably deep insight on the user's browsing context"*. They also commented on the value of the infographics used by *smt2*. Concretely, the visualization layers that the team found most useful were: mouse path, dwell times, clicks, direction/distances, and active areas. Designers also reported that the 'path centroids' layer was not too relevant. They liked the option of being able to switch to a static representation, specially when working with a large number of aggregated tracking logs.

6.2 Quantitative Results

Additionally, we asked permission to the team for downloading their gathered tracking logs for an offline study. They provided us with near 5000 XML files. We processed them to build regression models of user activity and to create interaction profiles, by clustering pages according to users' behavior, using the interaction metrics provided by the admin interface. We were able to predict with close to 70% of accuracy the expected time on a page based on the amount of mouse motion. This information was received very positively by the design team, as they could consider rephrasing some paragraphs to enhance the visibility of the content of the home page.

We also found that a 95% over all browsed pages could be explained by looking at just 3 clusters [5]. By inspecting those groupings, we could identify which pages were clubbing active users (e.g., rapid mouse movements, slight scroll reach, few clicks, etc.) or which ones caused people to hesitate most (e.g., repeated patterns of 'move-stop-move'). Designers could then review the pages belonging to each cluster, focusing on the identified behaviors, and could iterate over the design-test process. We concluded that *smt2* can be easily integrated with third-party tools to analyze usage data.

7. DEMONSTRATION SCENARIOS

The value of this tracking tool has been showcased to the WWW audience in two scenarios.

In the first scenario, we described how to configure the tool from scratch — since the package is downloaded until it is configured and uploaded to a web server. We prepared a custom server which could be publicly accessed throughout the conference. By using this sample server, we described to the audience how the system works. We also showed how to track external websites and how to prepare experiments; e.g., how to set random sampling, change the registration frequency, or display a warn dialog to ask for user's tracking consent.

In the second scenario, we described the admin interface. Attendants were able to filter, refine, and combine user logs, as illustrated in Section 4.1.2. The audience were also able to analyze and interact with the data.

Acknowledgments

We thank Jeff Huang for his valuable feedback while reviewing the first draft of this manuscript. This work is partially supported by the Spanish programme Consolider Ingenio 2010: MIPRCV (CSD2007-00018) and the project TIN2009-14103-C03-03.

8. REFERENCES

[1] E. Arroyo, T. Selker, and W. Wei. Usability tool for analysis of web designs using mouse tracks. In *Proc. EA CHI*, 2006.
[2] R. Atterer, M. Wnuk, and A. Schmidt. Knowing the user's every move – user activity tracking for website usability evaluation and implicit interaction. In *Proc. WWW*, 2006.
[3] I. Guyon, L. Schomaker, R. Plamondon, M. Liberman, and S. Janet. UNIPEN project of on-line data exchange and recognizer benchmarks. In *Proc. ICPR*, pp. 29–33, 1994.
[4] J. Huang, R. W. White, and S. Dumais. No clicks, no problem: Using cursor movements to understand and improve search. In *Proc. CHI*, pp. 1225–1234, 2011.
[5] L. A. Leiva and E. Vidal. Assessing user's interactions for clustering web documents: a pragmatic approach. In *Proc. HT*, pp. 277–278, 2010.
[6] F. Mueller and A. Lockerd. Cheese: Tracking mouse movement activity on websites, a tool for user modeling. In *Proc. EA CHI*, pp. 279–280, 2001.

APPENDIX

As in other web tracking applications, this work raises privacy concerns. First, we believe that logging keystrokes could be employed for unfair purposes. For that reason, we rejected to log raw keystroke data and track only keyboard events instead, without registering the associated character codes. Second, we recommend to ask always the user's consent before tracking takes place. This is a webmaster's responsibility, but not doing so could be considered unethical in some countries. And third, we believe that logged data should be stored in a server the webmaster owns, and not in a remote domain that he/she cannot control. We encourage commercial tracking systems to do so, since chances are there and current web technologies do support it.

Video: `http://vimeo.com/luileito/smt2-www`
Code & Demos: `http://smt2.googlecode.com`