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
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

\(^{1}\)The current version of our tool was publicly released in 2009.
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).

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 registra-
tion 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 pre-
sented to a team of five graphic designers that were not
usability experts. They wanted to redesign a corporate
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 be-
tween a user in the admin group and the super administra-
ator 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 lay-
out were causing confusion to most users; e.g., people often
hesitated over the main menu until deciding to click a navi-
gational item. Designers could also view that much of
the interaction with the site was concentrated around the
header section. Consequently, the team introduced modifi-
cations 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 tra-
cjectories, less clicks).

Overall, designers found the system very helpful. The
main advantages suggested were being able to reproduce ex-
actly 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 com-
mented on the value of the infographics used by smt2. Con-
cretely, the visualization layers that the team found most
useful were: mouse path, dwell times, clicks, direction/dist-
tances, and active areas. Designers also reported that the ‘path
centroids’ layer was too not 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 down-
loading 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 ad-
min 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 posi-
tively 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 inspect-
ing 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 it-
erate 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

every move – user activity tracking for website usability
S. Janet. UNIPEN project of on-line data exchange and
problem: Using cursor movements to understand and
movement activity on websites, a tool for user modeling. In

APPENDIX

As in other web tracking applications, this work raises privacy
concerns. First, we believe that logging keystrokes could be em-
ployed 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 recom-
mand 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 en-
courage 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