LikeLines: Collecting Timecode-level Feedback for Web Videos through User Interactions

Raynor Vliegendhart
R.Vliegendhart@tudelft.nl

Martha Larson
M.A.Larson@tudelft.nl

Alan Hanjalic
A.Hanjalic@tudelft.nl

Multimedia Information Retrieval Lab, Delft University of Technology, Delft, The Netherlands

ABSTRACT

Conventional online video players do not make the inner structure of the video apparent, making it hard to jump straight to the interesting parts. Our LikeLines system provides its users with a navigable heat map of interesting regions for the videos they are watching. Its novelty lies in its combination of content analysis and both explicit and implicit user interactions. The system can be readily used and deployed to collect large amounts of interaction data needed for in-depth research on timecode-level feedback.

Categories and Subject Descriptors:
H.5.1 [Multimedia Information Systems]: Video

General Terms: Human Factors, Design

Keywords: viewer interactions, implicit feedback, explicit feedback, Web video player

1. INTRODUCTION

Most online videos are opaque since their inner structure is not apparent to the outside world. Video players often only offer a time slider, making it hard to navigate through videos. Using a video player that provides no hints of a video's inner structure is similar to flipping through pages of a book that lacks an index or a table of contents. Users would benefit from a visualization of clickable regions of potential interest. In this paper, we present LikeLines, a system that provides such a visualization by combining multimedia content analysis with user feedback. The feedback is provided either implicitly or explicitly by users during viewing and interaction.

Our system was designed and implemented within the larger framework of an investigation of user interactions during the consumption of Web videos. Large quantities of these user interactions are needed for further understanding of their usefulness in various video domains as a source for implicit user feedback. Our system provides the means for collecting this information. Ultimately, this information will be useful for not only visualizing interesting points in a video, but also for, e.g., reranking entire videos, fragment-level video retrieval and video summarization.

Our contribution is an open and fully implemented system.[1] It is deployable out-of-the-box and able to collect user interaction data for large amounts of existing Web video content accessible through HTML5 and the YouTube API. The system combines multimedia content analysis and user interactions and allows these two interact with each other.

[1] https://github.com/delftmir/likelines-player

MM'12, October 29–November 2, 2012, Nara, Japan.
ACM 978-1-4503-1089-5/12/10.

2. USER FEEDBACK FOR MULTIMEDIA

This work combines recent trends in video retrieval systems: interactions at the timecode level and implicit feedback. The importance of addressing viewer interest at the timecode level is evidenced by users willing and wanting to refer to particular points in multimedia files. On sites like Nico Nico Video [4] and Indaba Music [2], comments are inherently tied to a particular time-point in a video or song. YouTube gives its users the option to deep-link to parts of a video in the comments [5] and also encourages interactions at the timecode level by offering a preview when using the time slider [6].

Implicit feedback such as view counts is used in video retrieval systems as a supplement to other forms of relevance, since automatic content analysis still suffers from the semantic gap [10], i.e., the problem of deriving sufficiently high level descriptions from multimedia features in order to match these to the query. Our system captures implicit interactions, which users perform naturally when watching a video in order to derive feedback at the timecode level. These interactions include playing and pausing a video and skipping and rewinding to a particular point in the video. The captured user interactions augment automatic content analysis performed on the video content. Automatic analysis is needed during cold start, when user interactions are not yet available. The analysis could give the first viewers initial hints of potentially interesting regions in the video. User interactions with the output of automatic analysis could reinforce or de-emphasize the importance of a specific time-point, leading to the discovery of new interesting regions.

LikeLines brings together features that exist only individually in previously proposed systems. The VideoSkip system captures similar implicit user interactions as our system for the purpose of generating representative thumbnails [9] and detecting events within a video [7]. Songle [8] tries to make music less opaque by visualizing user-contributed songs in various ways based on automatic content analysis and offering easy browsing within the songs themselves (e.g., jumping to the next verse or chorus). The Songle system accepts explicit corrections provided by the users in order to improve the initial analysis done by the system. A winning HTML5 video demo from a Mozilla contest allows users to explicitly cast votes for particular time-points during the video playback [3]. The demo displays a clickable histogram enabling users to jump to, e.g., a point in the video with the most votes. Similarly, the proprietary and closed-source video player by the content distribution platform Hulu [1] features a navigable heat map that informs the viewer which parts of the video have been watched most often. We only know of two systems that combine both automatic analysis and explicit user feedback in their approaches [8, 11], but none of them considers the use of implicit feedback.

3. SYSTEM OVERVIEW

3.1 User Interface

The user interface of the LikeLines system is similar to that of any other Web video player as depicted in Figure 1. Users can play and pause the video using the conventional buttons and they can seek or skip parts of the video by clicking on or dragging the time slider. In contrast to conventional players, the player in LikeLines has been augmented with a heat map and a time-sensitive "like" button, both of which are shown below the main user interface controls.

The heat map is an aggregation of all interactions performed by previous viewers and shows which points in the video can be possibly interesting. Parts of the video that are potentially more interesting are represented by a more intense color on the heat map. The viewers can interact with the heat map by clicking on it. Doing so will take them directly to the corresponding point in the video. When no previous interactions have been recorded yet, the output of multimedia content analysis serves as a source for a preliminary heat map.

The time-sensitive "like" button offers users the option to explicitly like particular points in the video they are watching. Clicking this button will result in a visible mark of the current playback position on the timeline, informing users their "like" has been registered. This marker can also serve the purpose of a bookmark, allowing the user easy access to earlier "liked" points in the video.

3.2 Technical Details

The LikeLines system consists of two main components: a Web video player component that resides in a browser on the user's system and a server component (Figure 1). The user only directly interacts with the player component. This component is implemented in JavaScript and uses HTML5 or Flash for video playback. User interactions such as playing and pausing the video are captured by the LikeLines player and are sent to the server component. The server component is responsible for storing and aggregating all these user interactions. The player component communicates with the server using the HTTP protocol and can make the following requests: *a*) Create a new interaction session for a video; *b*) Add new interactions to an existing session; and *c*) Aggregate content analysis and all sessions for a particular video to compute a heat map. The server's reply messages are encoded in the JSON or JSONP format.

The heat map is computed by representing an interaction session for an n seconds video as n bins. Each bin is initially set to 0 and each interaction can contribute, possibly negatively, to a bin's value. Content analysis of a video is modeled as an interaction session as well. The heat map is then obtained by aggregating all sessions and mapping each bin's accumulated value to a color.

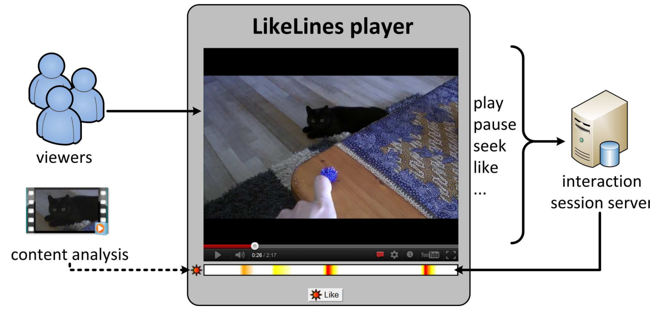

Figure 1: Users' interactions (play, pause, etc.) with the LikeLines player are stored at a server and are aggregated into a heat map. Viewers can use this heat map to jump to a particular point in the video. Content analysis of the video can seed the heat map.

4. CONCLUSION AND OUTLOOK

Our video system, which provides users with a navigable heat map, can be used to collect the large quantities of user interaction data that are needed to start studying both implicit and explicit timecode-level user feedback. We want to understand for what kinds of video this feedback is useful, how it should be interpreted and how it can be fused with content analysis without encouraging snowball effects that could lead to inappropiate reinforcement of arbitrary regions. Further, we want to know if timecode-level data can be linked to certain queries in order to recommend relevant jump points. Finally, we are interested in addressing the problem of collecting a critical mass of timecode-level data by incentivizing users to interact with the system.

5. ACKNOWLEDGMENTS

The research leading to these results has received funding from the European Commission's 7th Framework Programme under grant agreement № 287704 (CUbRIK).

6. REFERENCES

[1] Hulu – Labs: Captions Search. http://www.hulu.com/labs/captions-search.
[2] Indaba Music. http://www.indabamusic.com.
[3] Mozilla DemoStudio – HTML5 Video Voting. https://developer.mozilla.org/en-US/demos/detail/html5-video-voting.
[4] Nico Nico Douga. http://nicovideo.jp.
[5] YouTube – Deep Links. http://support.google.com/youtube/bin/answer.py?hl=en&answer=116618.
[6] YouTube – Looking ahead in the YouTube player. http://youtube-global.blogspot.com/2012/03/looking-ahead-in-youtube-player.html.
[7] C. Gkonela and K. Chorianopoulos. VideoSkip: event detection in social web videos with an implicit user heuristic. *Multimed Tools Appl*, pages 1–14, 2012.
[8] M. Goto et al. PodCastle and Songle: Crowdsourcing-Based Web Services for Retrieval and Browsing of Speech and Music Content. In *CrowdSearch (WWW)*, April 2012.
[9] I. Leftheriotis et al. Efficient video indexing on the web: A system that leverages user interactions with a video player. In *UCMEDIA*, 2010.
[10] A. Smeulders et al. Content-based image retrieval at the end of the early years. *IEEE T Pattern Anal*, 22(12):1349–1380, 2000.
[11] E. Smits and A. Hanjalic. A system concept for socially enriched access to soccer video collections. *IEEE Multimedia*, 17:26–35, 2010.