A Video Timeline with Bookmarks and Prefetch State for Faster Video Browsing

Axel Carlier
University of Toulouse
axel.carlier@enseeiht.fr

Vincent Charvillat
University of Toulouse
charvi@enseeiht.fr

Wei Tsang Ooi
National Univ. of Singapore
weitsang@nus.edu.sg

ABSTRACT

Reducing seek latency by predicting what the users will access is important for user experience, particularly during video browsing, where users seek frequently to skim through a video. Much existing research strived to predict user access pattern more accurately to improve the prefetching hit rate. This paper proposed a different approach whereby the prefetch hit rate is improved by biasing the users to seek to prefetched content with higher probability, through changing the video player user interface. Through a user study, we demonstrated that our player interface can lead to up to 4× more seeks to bookmarked segments and reduce seek latency by 40%, compared to a video player interface commonly used today. The user study also showed that the user experience and the understanding of the video content when browsing is not compromised by the changes in seek behavior.

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

General Terms: Design, Human Factors, Performance

Keywords: Prefetching, Bookmarks, Video Browsing

1. INTRODUCTION

A tremendous amount of videos is uploaded and shared on the Internet. For instance, 5 hours worth of videos is uploaded to YouTube every second [2]; The amount of videos that appear in Facebook's news feed increases 3.6× year-over-year [1]. Presented with many videos to watch, and with limited time, users exhibit a behavior when watching videos on these social media web sites, known as *video browsing*, in which users quit watching a video frequently (if they find the video uninteresting) and seek frequently (to look for interesting content). Chen et al. conducted a study on video viewing behavior from 540 million sessions, and found that viewers do not complete watching the video in 80% of the sessions; Furthermore, users seek in 62.5% of the sessions, with an average of 9.36 seeks for movies and 4.75 seeks for music videos [5].

This new seeking behavior amplifies a challenge in ensuring the quality of experience (QoE) while viewing the video. A typical video player downloads and buffers the video segments immediately after the current playback point, as most of the time, a user watches the video linearly. Such *sequential* buffering mechanism ensures that, as long as the bandwidth does not drop unexpectedly, sufficient content is buffered for smooth playback without a freeze. Seeking, however, is a *non-linear* behavior that leads to a random access to a segment of video that may not be buffered, in which case, the playback freezes until (i) the target segment being seeked to is downloaded, and (ii) the player buffered enough subsequent segments to absorb the delay jitter. The time between the seek action and the playback of the target segment is called the *seek latency*.

To reduce the expected seek latency experienced, researchers have proposed to prefetch segments of videos beyond the current playback buffer in a non-contiguous manner, hoping that if and when the user seeks to the prefetched segment, zero seek latency is incurred. Such approach requires the system to predict the seek destinations. In this paper, we set out on a different path to reduce the seek latency: *instead of prefetching the segments that we predict the users would seek to, we bias the users to seek to the segments that we have prefetched.* Our idea is to show to the user, on the video timeline, the prefetched segments, i.e., segments that would lead to zero seek latency. We hypothesize that users would naturally prefer to seek to segments with zero seek latency, which would result in improving the overall video browsing experience.

We organize the rest of the paper into six sections. We start with an overview of the literature in Section 2. In Section 3, we explain the video timeline interface that we study in this paper. Section 4 explains how we setup the user studies. In Section 5, we present our results. Finally, we conclude in Section 6.

2. RELATED WORK

We now discuss related research in the literature and contrast our proposed video timeline with the existing work.

We found only few research efforts for which video user interfaces aim at optimizing system-centric criteria (such as network latency, cache hits etc.) as opposed to user-centric performance metrics. Dilip et al.[12] took advantage of the repeatability of user behaviors and proposed to reorder YouTube recommendations, such that the videos already in the cache are pushed to the top of the related list, to improve the performance of video caches. Brampton et al. [3] showed that, in the context of football matches, bookmarking critical events can lead to three to four times more views of the bookmarked segments compared to non-bookmarked segments. They then proposed to prefetch segments of videos ahead of the bookmark points, and show that up to 38% of the seeks would result in zero seek latency. In our work, we assume that in a video browsing scenario, users are more interested in a quick overview of the video content than on actually understanding the video content.

Displaying prefetching state becomes then a strong bias towards seeking to the prefetched temporal regions.

Some work have studied optimal prefetching policies for non-linear access. Huang and Hsu [9] proposed a data mining-approach in which the seeking pattern of other users is used to predict the seek destinations. Similar approaches have been proposed in the context of peer-to-peer video on-demand systems [7, 18, 13] and mobile video streaming [11], where the seek latency could be significantly larger. Simpler approaches to prefetching, such as prefetching evenly-spaced segments, segments that correspond to scene changes, or segments that summarize the video, were mentioned by Uchihara et al. [17]. In our work, we choose to prefetch bookmarked segments.

Guiding the users towards important portions of the video with bookmarks, thumbnails, or recommendations can ease browsing tasks [15]. Many papers address the improvement of video players to facilitate searching and seeking. We categorize these research into content-based, user-based, and hybrid recommendations. For instance, the content-aware timeline [14] extracts keyframes with content analysis and plays a video snippet around these points when the user scrubs the timeline.

In the user-aware category, Syeda-Mahmood and Ponceleon interpret the video playback interactions in the temporal domain (play, fast-forward, pause) and infer the most interesting temporal segments from the video [16]. This pioneer work inspired many subsequent papers including the work by Gkonela et al. [8] that rethinks the underlying user models. Kim et. al [10] introduced a 2D video timeline with an innovative exploitation of collective navigation traces.

The current trend is to mix content-aware and user-aware recommendations. In [4], Carlier et al. proposed to compliment content analysis with implicit feedback of a community of users to recommend interesting video regions. A similar idea is used in the SmartPlayer [6], which adapts the playback speed from both the visual richness of the scene (estimated from content analysis) and the user preferences (learnt from their video interactions). In this work, we assume that such recommendations are available and can be strategically included in our video timeline as bookmarks.

3. VIDEO PLAYER UI

In this section, we present our approach to improve the predictability of user access patterns through a subtle change in the user interface of a video player. In particular, we propose to augment the video timeline with information about the prefetch buffer along with bookmarks.

Regular Video Timeline Interface. Figure 1(a) shows the video timeline of a typical video streaming player that can be found on many Web-based video hosting sites, such as YouTube or Vimeo. The timeline is divided into three sections, depicted in different colors in Figure 1(a). The left most section (in red) stretches from the beginning of the video to the current playback point. The middle section (light gray) shows the segments that has been buffered (sequentially prefetched ahead of the playback point). The right most section (in dark gray) shows the segments that have not been buffered. We denote this interface as the Regular interface.

Consider a user who wants to skip forward beyond the current playback point. If the user clicks on the timeline section that has been buffered, the playback can resume almost immediately with negligible seek delay, as the segment that the user wants to seek to has already been downloaded and buffered. On the other hand, if the user clicks on the timeline section that has not been buffered, the video player would pause the playback, until the segment that is seeked to has been downloaded and sufficient subsequent segments have been downloaded to absorb network jitter.

Video Timeline with Prefetching States. It is also common for a video timeline to display bookmarks. These bookmarks serve as annotations to important or interesting points in the video, such as commercial breaks, beginning of chapters, and critical events in the narration of the video. Figure 1(b) shows an example. These bookmarks can be determined manually by content provider, or automatically through content analysis or user behavior analysis [3, 19], to identify popular or semantically important points in the video content. Bookmarks can be annotated with text describing the content or images that depict the content of the videos (as shown in Figure 1(b)).

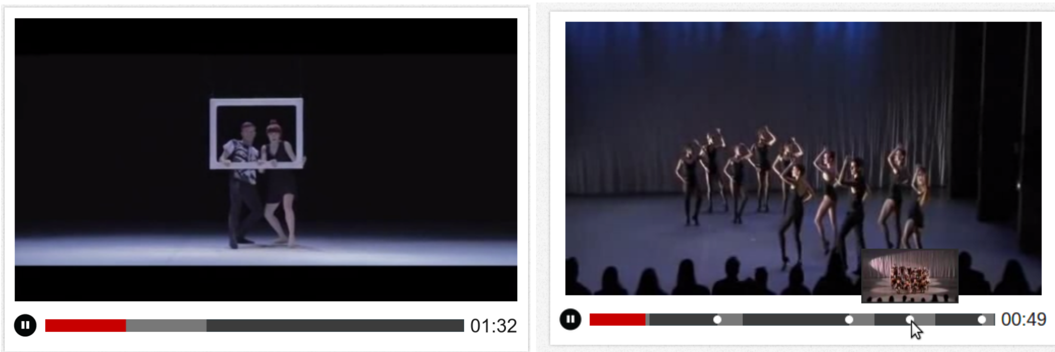
The UI we investigate in this paper not only shows (i) the playback point, (ii) bookmarks, (iii) thumbnails of the bookmarks which will become visible with mouse hover, and (iv) buffered segments ahead of the playback point, it also shows (v) the prefetched segments ahead of the bookmarks. Figure 1(b) shows a screenshot of this video timeline. Since users would experience zero seek latency when seeking to any of the prefetched segment, we hypothesize that, by depicting (v) in the video timeline, users would seek more frequently to prefetched segments than to non-prefetched segments. Combining features (i)-(v), this video timeline interface would therefore lead to more prefetching hits and lower seek delay, compared to a regular video timeline that depicts only (i) and (iv). Due to the additional information depicted in this video timeline, we denote this interface as Extra.

One question that arises is whether users would appreciate less the content of the videos if we nudge users to click on the bookmarked and prefetched segments more, especially for the case where the bookmarks are automatically generated and may not accurately cover all the key events or interesting segments in the videos. This is true in general, especially for video with informational content (documentary, news, lectures) or narrative entertainment (drama, sports). Research on automatically generating appropriate bookmarks is outside the scope of our work in this paper. We assume that bookmarks are manually generated. Furthermore, this work is done in the context of video browsing behavior on social video sharing sites, where videos tend to be short (less than 10 minutes) and placement of bookmarks does not affect the understanding of the video as much, as long as they are spread throughout the video so that the users get to sample a broad range of segments as they skim through the video.

4. USER STUDY

To evaluate the effectiveness of our proposed video timeline design, we conducted a user study, where each user viewed six videos using two different video timeline interfaces. The user seek actions while viewing the videos were then logged for analysis.

Videos Used: We chose six fairly similar videos depicting dance performances with the length of $5' 45''$, $5' 59''$, $7' 53''$, $7' 59''$, $8'$ and $8' 6''$ respectively for the user study. All videos are VBR encoded and segmented into 1-second chunks. For video timeline that requires bookmarks, we manually selected four bookmarks on each of these six videos that are reasonably spaced out, with different lighting and camera angle. It is our intention to choose videos where no particular chunk is more important or interesting than other chunks throughout the video, so that our bookmarking decisions do not affect the viewing experience. We also chose dance performances without a strong narrative, so that seeking does not interfere with the understanding of videos. Snapshots of two of the videos are shown in Figure 1.



(a) (b)

Figure 1: Video Timeline Interfaces: (a) A typical video player showing only playback point and sequentially buffered region; (b) A video player with timeline showing bookmarks and prefetched video segments.

Task: Since our experiment aims to learn how users behavior would change when presented with a different video timeline design, we designed a task that is doable without relying on specific user interaction, except play, pause, and seek. We showed six videos to each participant and asked if he/she likes each video, and whether he/she would recommend it to a friend. The videos were shown to the participant on a Web page, one after another, in randomized order, using one of two video timeline interfaces. The first three videos were shown with either the Regular interface or Extra interface, chosen randomly with uniform probability. The next three videos were shown with the other interface. We did not give specific instructions to the participants on how to use the video timeline (to avoid biasing them to seek), nor did we ask them to watch the video completely (to see if seeking behavior would arise naturally). The task mimics existing natural behavior when users watch videos online.

Setup: We developed an HTML5-based Web video player in which we can configure different video timeline design. The video player logs all user interactions with the video and relays it back to a backend server for logging and later analysis. Even though our user study targets a video streaming scenario, where chunks are downloaded while the video is being watched, our video player pre-loads the complete video before it allows the user to play. As the video plays, the player *simulates* buffering and prefetching of chunks, as well as the seek latency when the player seeks to a chunk that has not been downloaded according to the simulation. The player pauses for 2 seconds, before the playback continues.

There are two reasons why the videos are pre-loaded and the buffering, prefetching, and seek latency are simulated: (i) Different participants may experience different network conditions and thus the viewing experience and user behavior may be affected in an unpredictable way if our player actually downloads and plays the video simultaneously; (ii) By simulating the downloading process, our setup is predictable, reproducible, and controllable (e.g., we can tune the bandwidth allocated to prefetching).

The Regular video player has a simple prefetching policy that sequentially prefetch ahead of the current playback time. The video player with Extra interface (Figure 1(b)) prefetches the chunks according to the following algorithm: the chunks ahead of the current playback point have the highest priority of being prefetched. When the player has at least 5 seconds of data prefetched ahead of the current playback point, the algorithm starts to prefetch chunks after the bookmarks, in two phases. First, the algorithm fetches 5 seconds of chunks ahead of each bookmark, one bookmark after another, starting from the bookmarks closest to (and after) the playback point. Second, once at least 5 seconds of chunks are buffered after each bookmark, the algorithm prefetches one more chunk ahead of each bookmark, in a round-robin manner.

Our prefetch algorithm above arbitrarily assumes that a 5-second buffer is sufficient to absorb variations in network bandwidth. Further, the algorithm is not optimized (e.g., according to the probability of seek to a particular bookmark). Our goal is to study how user behavior differs under different video timeline interface, not the effects of prefetching algorithm. Varying the prefetching algorithms would introduce too many variations into our user study.

To reduce pre-loading time for low-bandwidth participants, we resample the videos to resolution 480×270. The video player has the size of 480×270 pixels as well and displays the videos without resizing. The video timeline has the size of 400×12.

5. RESULTS

A total of 21 participants participated in the user study. The result, summarized in Table 1, is consistent (or even better) with our hypothesis – 44% of seeks landed on bookmarks, and the average seek latency is reduced by 40%.

	Regular	Extra
% Seek to Bookmarks	12 %	44 %
Average Seek Latency	1.09s	0.65s

Table 1: Video Browsing Seeks and Hits

One could wonder why 12 % of the seeks landed on bookmarks even when the bookmarks are not displayed in the Regular UI. This number is actually very close to the expected percentage if the seeks landed randomly and uniformly along the timeline.

At the end of participation, we ask each participant to answer to a survey. The questions and results of the survey are summarized in Table 2. Note that we gave the participants three choices for each question: Extra, Regular, or the same[1].

Table 2 shows that, less than a quarter of the participants prefer Regular over Extra for browsing the video. No participant finds that Regular is more helpful than Extra to understand the content

[1]In the actual study, we named the interfaces A and B to avoid biasing the participants

of the video (about half feels that both UIs are equally helpful, the rest finds Extra more helpful). Finally, 14 out of 21 participants answered that Extra helps them scan through the videos faster.

In the quantitative comments, a participant who prefers Regular over Extra responds that Regular does not suggest to them where to click so he/she is free to click anywhere. Another participant with the same preference mentioned that the bookmarks remind him/her of advertisements.

Question	Regular	Extra
Which interface do you prefer to use to browse through the videos?	5	13
Which interface helps you more in understanding the content of the video?	0	11
Which interface helps you in scanning the content of the video faster?	2	14

Table 2: Video Browsing Survey Results

In summary, this user study shows that depicting the prefetch states of bookmarked segments can bias users to seek to the prefetched segments, reducing the average seek latency experience, and (ii) for a majority of participants, showing bookmarks and prefetch state is preferred, and helps to understand the video content as well as allow faster skimming through the video.

6. CONCLUSION AND FUTURE WORK

Researchers have been trying to reduce seek latency by predicting where a user would seek to when watching videos through content analysis and user behavior mining. In this paper, we show that, we can simply reduce seek latency by tweaking the user interface, therefore altering the user behavior. Instead of prefetching where we think the user would seek to, we show that it is possible to influence the user to seek to what we have prefetched. As a result, users' behaviors are more predictable, bringing lower seek latency. Our technique can complement existing prediction techniques to improve the user experience while browsing videos online. This question will be further studied in our future work.

7. REFERENCES

[1] Facebook video: New universal language. `http://insights.fb.com/2015/01/07/new-universal-language/`. posted January 7, 2015.

[2] YouTube statistics. `https://www.youtube.com/yt/press/statistics.html`. retrieved March 31, 2015.

[3] A. Brampton, A. MacQuire, M. Fry, I. A. Rai, N. J. Race, and L. Mathy. Characterising and exploiting workloads of highly interactive video-on-demand. *Multimedia Systems*, 15(1):3–17, 2009.

[4] A. Carlier, G. Ravindra, V. Charvillat, and W. T. Ooi. Combining content-based analysis and crowdsourcing to improve user interaction with zoomable video. In *Proceedings of the 19th ACM International Conference on Multimedia*, MM '11, pages 43–52, Scottsdale, AZ, USA, Nov. 2011.

[5] L. Chen, Y. Zhou, and D. M. Chiu. A study of user behavior in online VoD services. *Computer Communications*, 46:66–75, 2014.

[6] K.-Y. Cheng, S.-J. Luo, B.-Y. Chen, and H.-H. Chu. SmartPlayer: User-centric video fast-forwarding. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI'09, Boston, MA, USA, 2009.

[7] Y. Feng, B. Li, and B. Li. Peer-assisted VoD prefetching in double auction markets. In *Prceedings of the 18th IEEE International Conference on Network Protocols*, ICNP'10, pages 275–284, Kyoto, Japan, Oct. 2010.

[8] C. Gkonela and K. Chorianopoulos. VideoSkip: event detection in social web videos with an implicit user heuristic. *Multimedia Tools and Applications*, 69(2):383–396, 2014.

[9] C.-M. Huang and T.-H. Hsu. A user-aware prefetching mechanism for video streaming. *World Wide Web*, 6(4):353–374, 2003.

[10] J. Kim, P. J. Guo, C. J. Cai, S.-W. D. Li, K. Z. Gajos, and R. C. Miller. Data-driven interaction techniques for improving navigation of educational videos. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST'14, pages 563–572, Honolulu, HI, USA, 2014.

[11] C. Koch and D. Hausheer. Optimizing mobile prefetching by leveraging usage patterns and social information. In *Prceedings of the 22nd IEEE International Conference on Network Protocols*, ICNP'14, pages 293–295, Raleigh, NC, USA, Oct. 2014.

[12] D. K. Krishnappa, M. Zink, C. Griwodz, and P. Halvorsen. Cache-centric video recommendation: an approach to improve the efficiency of youtube caches. In *Proceedings of the 4th ACM Multimedia Systems Conference*, MMSys'13, pages 261–270, Oslo, Norway, February 2013.

[13] C. Lee, E. Hwang, and D. Pyeon. A popularity-aware prefetching scheme to support interactive P2P streaming. *IEEE Transactions on Consumer Electronics*, 58(2):382–388, 2012.

[14] S. Pongnumkul, J. Wang, G. Ramos, and M. Cohen. Content-aware dynamic timeline for video browsing. In *Proceedings of the 23nd Annual ACM Symposium on User Interface Software and Technology*, UIST'10, pages 139–142, New York, NY, USA, 2010.

[15] K. Schoeffmann and L. Boeszoermenyi. Video browsing using interactive navigation summaries. In *Proceedings of the 7th International Workshop on Content-Based Multimedia Indexing*, CBMI'09, pages 243–248, Chania, Crete, June 2009.

[16] T. Syeda-Mahmood and D. Ponceleon. Learning video browsing behavior and its application in the generation of video previews. In *Proceedings of the 9th ACM International Conference on Multimedia*, MM'01, pages 119–128, Ottawa, Canada, 2001.

[17] N. Uchihara, H. Kasai, Y. Suzuki, and Y. Nishigori. Asynchronous prefetching streaming for quick-scene access in mobile video delivery. *IEEE Transactions on Consumer Electronics*, 56(2):633–641, 2010.

[18] Z. Wang, L. Sun, S. Yang, and W. Zhu. Prefetching strategy in peer-assisted social video streaming. In *Proceedings of the 19th ACM International Conference on Multimedia*, MM'11, pages 1233–1236, Scottsdale, AZ, USA, Nov. 2011. ACM.

[19] Y. Zhao, Y. Tian, and Y. Liu. Extracting viewer interests for automated bookmarking in video-on-demand services. *Frontiers of Computer Science*, pages 1–16, 2014.