Front.Comput.Sci.
DOI

RESEARCH ARTICLE

Extracting Viewer Interests for Automated Bookmarking in Video-on-Demand Services

Yang ZHAO [1], **Ye TIAN** (✉)[1], **Yong LIU** [2]

1 University of Science and Technology of China, Hefei, Anhui 230027, China
2 Polytechnic Institute of New York University, Brooklyn, NY 11201, USA

Abstract Video-on-Demand (VoD) services have become popular on the Internet in recent years. In VoD, it is challenging to support the VCR functionality, especially the jumps, while maintaining a smooth streaming quality. Previous studies propose to solve this problem by predicting the jump target locations and prefetching the contents. However, through our analysis on traces from a real-world VoD service, we find that it would be fundamentally difficult to improve a viewer's VCR experience by simply predicting his future jumps, while ignoring the intentions behind these jumps.

Instead of the prediction-based approach, in this paper, we seek to support the VCR functionality by bookmarking the videos. There are two key techniques in our proposed methodology. First, we infer and differentiate viewers' intentions in VCR jumps by decomposing the inter-seek times, using an Expectation-Maximization (EM) algorithm, and combine the decomposed inter-seek times with the VCR jumps to compute a numerical interest score for each video segment. Second, based on the interest scores, we propose an automated video bookmarking algorithm. The algorithm employs the time-series change detection techniques of CUSUM and MB-GT, and bookmarks videos by detecting the abrupt changes on their interest score sequences. We evaluate our proposed techniques using real-world VoD traces from dozens of videos. Experimental results suggest that with our methods, viewers' interests within a video can be precisely extracted, and we can position bookmarks on the video's highlight events accurately. Our proposed video bookmarking methodology does not require any knowledge on video type, contents, and semantics, and can be applied on various types of videos.

Keywords Video-on-Demand (VoD), highlight bookmarking, time-series change detection

1 Introduction

In recent years, Video-on-Demand (VoD) services have become very popular on the Internet. In such a service, on-demand video streams are delivered to viewers either through a dedicated content delivery network (CDN) (e.g., YouTube and Netflix) or using a peer-to-peer (P2P) network (e.g., PPTV). With the prevalence of the services, users are expecting better interactive experiences, which are usually referred to as the *VCR functionality*. In a VCR-enabled VoD service, users are free to pause/resume the playback, skip forward and backward, and jump to any arbitrary positions. Unfortunately, for both CDN and P2P-assisted VoD systems, it is not easy to support the VCR functionality in a large scale, as in these systems, the video seeking delay, which is the time lag from a viewer issuing a VCR jump request to the rendering of the requested content is usually several orders of magnitude longer than the delay that a viewer would tolerate without suffering a noticeable streaming quality degradation.

A number of previous studies propose to support the VCR functionality by predicting a viewer's VCR jumps, and prefetching the contents that the viewer is predicted to jump to, under the context of a P2P network. Usually, viewers' "VCR patterns" [1], or an "interest maps" among different

Received month dd, yyyy; accepted month dd, yyyy

E-mail: yetian@ustc.edu.cn

video parts [2] [3], are constructed by mining their historical VCR behaviors. One implicit assumption behind the prediction-based approaches is that a viewer would jump from one position where he has a low viewing interest, to another position where he has high interest, thus by accurately predicting the jumps, the viewer's quality of VCR experience will be improved.

Another approach for supporting the VCR functionality is to *bookmark* the video, that is, to position a number of bookmarks that label interesting events, or highlights, on the video. In a bookmark-assisted VoD service, a viewer could choose to directly jump to a bookmark, and the client could prefetch all the bookmarked contents in advance for reducing the seeking delay. A recent study [4] reported that when accurately positioned bookmarks are available, viewers' VCR experiences can be improved considerably.

Many highlight event detection and summarization techniques have been proposed for video bookmarking in recent years (e.g., [5–9]). However, these methods are generally based on analyzing semantic features of the video contents, such as the key frames, motions, replays, textures, etc.. As a result, the developed techniques are not generic, but work with only certain specific video types, like sport videos. On the other hand, in large-scaled VoD services providing various types of videos, such as Youku [1)], the largest Internet VoD site in China, bookmarks have to be positioned manually. In this paper, we refer to the VCR jumps that follow bookmarks as the *guided jumps*, and refer to the ones without the assistance of any bookmarks as the *unguided jumps*.

In this work, we seek to support the VCR functionality in VoD services by bookmarking the videos. We first show that it is fundamentally difficult to improve a viewer's VCR experience by simply predicting his unguided jumps, while ignoring the intentions behind these jumps. Instead, we infer and differentiate viewers' intentions behind their video jumps based on the *inter-seek time*, i.e., the duration between two consecutive jumps. We then extract viewers' interest levels for video segments. We further propose an algorithm that detects the abrupt changes on the interest levels within a video for automatically positioning bookmarks. The detailed contributions of this paper are listed as follows.

- We investigate viewers' unguided VCR jumps from traces of a real-world VoD service, and find that they have different intentions when making VCR jumps. We observe that 60 ∼ 80% of the unguided jumps do not lead to attractive contents, and it is very common for a viewer to make consecutive jumps with very short inter-seek times. Our observation suggests that it would be fundamentally difficult to improve a viewer's VCR experience by simply predicting his future unguided jumps without understanding the intentions behind.
- We propose to differentiate viewers' intentions behind their VCR jumps by decomposing the inter-seek times, using an Expectation-Maximization algorithm [10]. We then use viewers' intentions of the unguided VCR jumps to compute a score for measuring the interest level held by the viewers on each video segment. We show that with the scores, viewers' interests within a video could be accurately extracted.
- We design an algorithm for automatically positioning bookmarks on a video by detecting the abrupt changes on the video's segment interest score sequence. The algorithm employs the time-series change detection techniques of CUSUM [11] and MB-GT [12].
- We apply our proposed method on various types of videos, and find that it is able to position the bookmarks on a video's highlight events with high accuracies, thus effectively improves the viewers' VCR experiences.

Compared with the previous studies, our proposed video bookmarking methodology exploits viewers' VCR behaviors that are already recorded in most existing VoD systems. The method is independent of the video type, and works well without any knowledge on the video's contents and semantics. The paper is organized as the following: In Section 2, we introduce the VoD traces we use in this work; In Section 3, we characterize the unguided VCR jumps; Section 4 presents our proposed technique for extracting viewers' interests, and we describe the video bookmarking algorithm in Section 5; We evaluate the proposed video bookmarking methodology in Section 6; Section 7 discusses the related work and we conclude this paper in Section 8.

2 VoD Traces under Study

In this section, we introduce the VoD traces that we use in this work. Our study is based on the log files collected from a campus-wide VoD platform that provides streaming services of over 17,000 videos in our university[2)]. The VoD platform is built upon the Microsoft's Window Media

[1)] http://www.youku.com

[2)] The service is at `http://video.ustc.edu.cn`, however, it can only be accessed from within the campus network.

Table 1 Videos under study

Name	Title	Duration (sec.)	Sessions	VCR jumps	Highlight events
Gala09	2009 CCTV Chinese New Year Gala	17,272	968	19,351	Beginnings of popular programs
Gala10	2010 CCTV Chinese New Year Gala	18,270	1,862	58,081	Beginnings of popular programs
Soc09	2009 FIFA Confederations Cup Final, USA vs. Brazil	9,636	172	4,247	Goals
Soc10	2010 FIFA World Cup Semi-Final, Uruguay vs. Netherlands	7,379	253	2,857	Goals
Doc	Dream and Glory of USTC	4,664	11,410	61,384	Ends of commercials

Services (WMS) system, and in WMS, when a viewer requests a video, or when he initiates a VCR jump, WMS writes an entry in its log file, recording the information such as the video's URL, client ID, etc.. By analyzing these records, we can track the viewer's behaviors. In particular, for each VCR jump, we can collect the information on its starting and destination positions, as well as the *inter-seek time*, which is the duration of the video content rendered by the viewer's player before his next jump. The VoD traces we study in this work were collected from Mar. 11, 2009 to Apr. 26, 2012. The traces contain $1,702,511$ VoD sessions and a total number of $22,581,465$ VCR jumps.

One good feature of the campus VoD traces is that as the system is built upon WMS that doesn't support bookmarks, all the VCR jumps are unguided. Moreover, as the service is restricted within a high-speed campus network, we can assume that a viewer's VCR jump is mostly driven by his view interest, rather than by the playback performances (e.g., buffering delay, jitter, etc.) that is influenced by network conditions during the VoD sessions.

Instead of focusing on one video or one video type, in this work we study a wide range of videos from various types. In particular, we select five videos representing different types from the campus VoD traces, and carry out most of our analysis and evaluations on them. The selected videos are listed in Table 1. Among the five videos, two are the CCTV Chinese New Year galas in 2009 and 2010 (referred to as *Gala09* and *Gala10*), two are soccer games (referred to as *Soc09* and *Soc10*), and one is a documentary TV program with five commercials inserted (referred to as *Doc*).

The reason that we select the five videos in our study is that the videos have attracted many viewers thus have rich sets of VCR records; more importantly, all the five videos have the *ground-truth highlight events* (that is, their highlight events can be identified with explicit criteria), which enables us to perform an objective evaluation on different bookmarking methodologies. For example, for the gala type videos of Gala09 and Gala10, which are composed of a series of programs, their highlight events[3] are the programs that are most enjoyed by the viewers, and bookmarks should be positioned at beginnings of these programs; for the soccer games of Soc09 and Soc10, goals can be considered as their highlight events; finally, for the documentary video of Doc, although no obvious highlights exist, however, as we can see in Section 4, viewers apparently lack interests in the five commercials that are inserted in the program, therefore we may consider all the contents between two commercials as a highlight event, and a bookmark should be positioned at the end of each commercial. From Table 1, we can see that the five videos under study have totally different contents and semantics in their highlight events, and we seek to develop techniques for detecting all of them. Besides the five videos, we also employ the videos that have no ground-truth highlights (e.g., movies and TV dramas) to evaluate our proposed bookmarking methodology in Section 6.

3 Characterizing Unguided VCR Jumps

In this section, we study viewers' unguided VCR jumps, and show that they have different intentions in making these jumps. For simplicity, we assume that videos are sequentially divided into segments, each of which has playback time of one second.

Recent studies ([1–3, 13]) seek to improve a viewer's quality of VCR experience by predicting the destinations of his VCR jumps and prefetching the contents at the predicted positions, based on the assumption that a viewer would jump to the positions that are more interesting. Thus, we first study the VCR jump destinations. We select two videos, Gala09 and Doc, and present the frequencies of the segments chosen as the seeking destinations in Fig. 1. From the figure one can see that the entire distribution is noise-like with some peaks. For the video of Gala09 in the top figure, one obvious peak is around the $12,500^{th}$ second, which corresponds to the most popular program in the gala, suggesting that viewers are more likely to jump to the part

[3] Strictly, a highlight event has its start and end boundaries, however for bookmarking, only the start position is useful. In this work, we also refer to the start position of a highlight event as a highlight for simplicity.

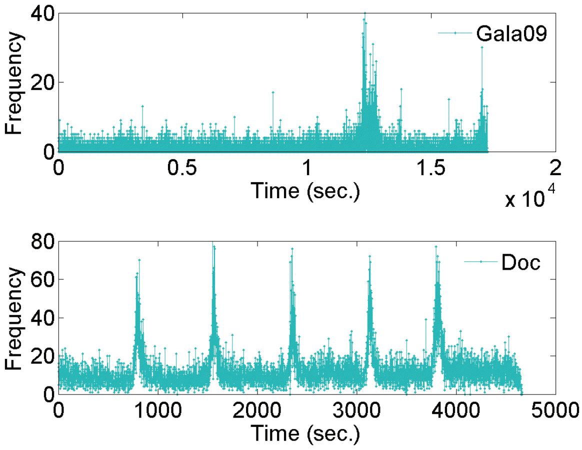


Fig. 1 Frequencies of the segments selected as the seeking destinations in Gala09 and Doc.

that is attractive. However, the other popular programs in the gala are less obvious, due to the noisy nature of the unguided VCR jumps. For the video of Doc in the bottom figure, five obvious peaks can be observed. However, we find that these peaks are indeed corresponding to the five commercials inserted into the video, and viewers make many VCR jumps around these peaks precisely for skipping the commercials that they are not interested in. From the two examples, we can see that *viewers have different intentions in making VCR jumps: a viewer could make a VCR jump either to look for an interesting content, or for skipping a boring part.*

Table 2 Satisfactory and unsatisfactory jumps in the five videos

	τ(in sec.)	Satisfactory jump	Unsatisfactory jump
Gala09	14	5,085 (26.3%)	14,266 (73.7%)
Gala10	13	12,313 (21.2%)	45,768 (78.8%)
Soc09	9	1,715 (40.0%)	2,857 (60.0%)
Soc10	11	1,496 (35.2%)	2,751 (64.8%)
Doc	14	23,237 (38.7%)	37,649 (61.3%)

As a viewer would make a VCR jump either for interest or for lack of interest, we then try to differentiate the two types of jumps. For each VCR jump, we compare the inter-seek time till the next jump with a threshold τ: if the inter-seek time is longer than τ, we refer to the corresponding jump as a satisfactory jump, as the viewer is satisfied with the content after the jump and does not make further jumps within τ; otherwise, we refer to the jump as an *unsatisfactory jump*. Table 2 lists the numbers and percentages of the two types of jumps, as well as the τ values for the five videos. From the table we can see that there are much more unsatisfactory jumps than satisfactory ones. Moreover, we find that it is very common for viewers to make consecutive unsatisfactory jumps; in fact, the mean consecutive unsatisfactory jumps in the five videos of Gala09, Gala10, Soc09, Soc10, and Doc are 5.01, 6.12, 4.31, 3.86, and 4.06 respectively. Note that here we only employ a naive threshold for differentiating the jumps, in Section 4, we will explain how we choose the τ values for different videos, using a probabilistic approach based on Expectation Maximization.

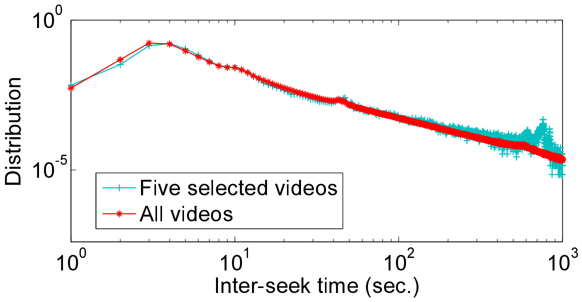


Fig. 2 Aggregated inter-seek time distributions from traces of the five videos in Table 1 and from all videos.

In summary, our observations on the seeking frequencies and inter-seek times of the unguided VCR jumps suggest that *the statistical mapping between the starting and destination positions of the jumps doesn't necessarily correspond to viewers' "interest map", and it would be fundamentally difficult to improve a viewer's quality of VCR experience without understanding his intentions when making these jumps.*

4 Extracting Viewer Interests

In this section, we seek to extract viewers' interests within a video from their VCR jumping behaviors. We first differentiate viewers' intentions in their VCR jumps by decomposing the inter-seek times, using an Expectation-Maximization (EM) algorithm [10]. Then we combine the decomposed inter-seek times with the VCR jumps to compute their interest levels on the video segments.

4.1 Decomposing Inter-Seek Times

4.1.1 Inter-seek time analysis

To study the inter-seek times in VoD services, in Fig. 2 we present the empirical distribution of the aggregated inter-seek times from VoD sessions of all videos in the campus VoD traces, as well as the distribution from the five videos in Table 1 under the log-log scale. From the figure one can see that the distributions are heavily skewed. For instance, the mean and median inter-seek times from all videos are 110.8 and 8.0 seconds respectively, suggesting that most of the inter-seek times are very short. Furthermore, by carefully studying the figure, we find that the distribution curve after 5 seconds, which cover from median to long inter-seek times, actually decays slower than power-law distribution. The observation suggests that it would be impropriate to model the inter-seek times using a single power-law distribution, like lognormal [4] or Weibull [14] as suggested in previous studies.

4.1.2 Decomposing inter-seek time with EM

Motivated by the observation, we seek to model the empirical inter-seek times with a mixture of two lognormal distributions, using an Expectation-Maximization (EM) algorithm [10].

Specifically, for an observed inter-seek time x, we consider that it is drawn from one of the two lognormal distributions, where the j^{th} ($j = 1, 2$) distribution has a probability density function (pdf) as

$$Lg_j(x) = \frac{1}{x\sqrt{2\pi\sigma_j^2}} e^{-\frac{(\ln x - \mu_j)^2}{2\sigma_j^2}},$$

and the probability that the inter-seek time x is drawn from Lg_j is θ_j, with $\theta_1 + \theta_2 = 1$.

To fit the empirical data, our objective is to find the parameters $\Theta = (\theta_1, \theta_2)$, $\mathrm{M} = (\mu_1, \mu_2)$, and $\Sigma = (\sigma_1, \sigma_2)$, so that for all the recorded inter-seek times, $\{x^{(1)}, \cdots, x^{(N)}\}$, the log-likelihood, defined as

$$l(\Theta, \mathrm{M}, \Sigma) = \sum_{i=1}^{N} \log \left(\sum_{j=1}^{2} Lg_j\left(x^{(i)}\right) \times \theta_j \right),$$

is maximized.

Instead of directly solving the problem, we consider the problem of finding the optimal mixture of two Gaussian distributions with parameters of Θ, M, and Σ, so as to best fit the samples $\{y^{(1)}, \cdots, y^{(N)}\}$, with $y^{(k)} = \log x^{(k)}$. This problem can be solved with an Expectation-Maximization (EM) algorithm as suggested in [10]. More specifically, the algorithm works in rounds: for each round, the log-likelihood is computed using the current values of Θ, M, and Σ, and the algorithm updates these parameters as:

Table 3 Inter-seek time decomposing results

		Gala09	Gala10	Soc09	Soc10	Doc
Lg_1	θ_1	0.316	0.252	0.396	0.467	0.446
	μ_1	3.891	4.070	3.755	3.682	4.398
	σ_1^2	2.842	3.133	2.320	2.615	2.974
Lg_2	θ_2	0.684	0.748	0.604	0.533	0.554
	μ_2	1.610	1.546	1.520	1.448	1.615
	σ_2^2	0.216	0.215	0.150	0.148	0.249

$$\theta_j = \frac{1}{N}\sum_{i=1}^{N} \omega_j^{(i)}$$

$$\mu_j = \frac{\sum_{i=1}^{N} \omega_j^{(i)} y^{(i)}}{\sum_{i=1}^{N} \omega_j^{(i)}}$$

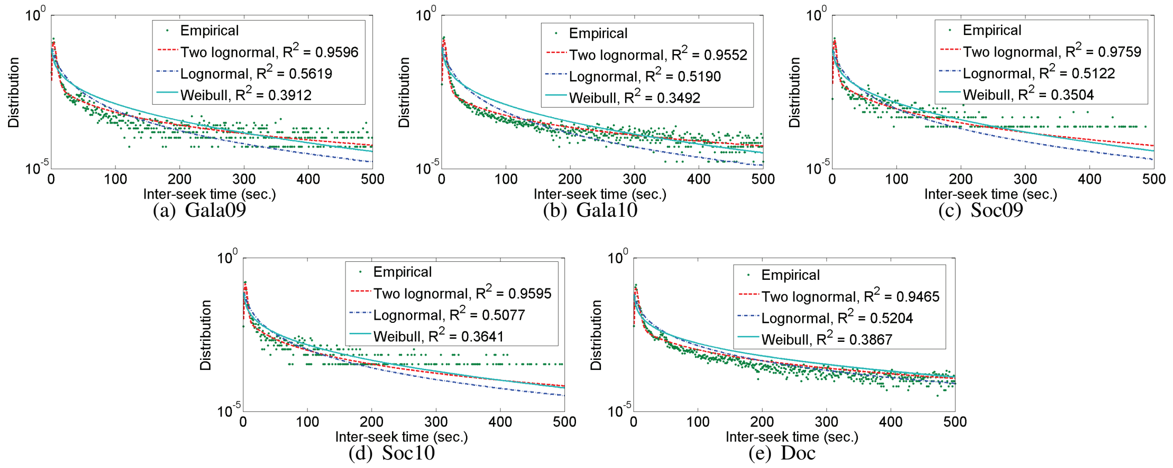
$$\sigma_j^2 = \frac{\sum_{i=1}^{N} \omega_j^{(i)} (y^{(i)} - \mu_j)^2}{\sum_{i=1}^{N} \omega_j^{(i)}},$$

where $\omega_j^{(i)} = \frac{\theta_j \frac{1}{\sqrt{2\pi\sigma_j^2}} e^{-(y^{(i)} - \mu_j)^2 / 2\sigma_j^2}}{\sum_{k=1}^{2} \theta_k \frac{1}{\sqrt{2\pi\sigma_k^2}} e^{-(y^{(i)} - \mu_k)^2 / 2\sigma_k^2}}$, using the values of Θ, M, and Σ in the previous round. After the algorithm stops with convergence, the optimal values for Θ, M, and Σ can be obtained.

4.1.3 Evaluation and interpretation

We apply the EM algorithm above described to the five selected videos. For each video, two lognormal distributions, as well as the parameters Θ, M, and Σ, are obtained. Table 3 lists the decomposition results. For each video, we can see that the inter-seek times from one distribution are much longer than the other. Since longer inter-seek intervals suggest that viewers have higher interest levels in viewing the content [15], we therefore refer to the corresponding distribution Lg_1 as the *interested* inter-seek time distribution, and refer to the other lognormal Lg_2, producing much shorter inter-seek times, as the *non-interested* distribution.

We plot the empirical inter-seek time distributions from the five videos and the corresponding mixtures of the two lognormals obtained from the EM-based algorithm in Fig. 3. For comparison, we also apply the models of one lognormal distribution as suggested in [4] and one Weibull distribution in [14] to fit the empirical data. For each fitting, we compute the determination coefficient R^2, which indicates how good the model fits the empirical data, and display the values in the figures. By comparison, we can see that for each video, the two-lognormal mixture matches the empirical data very well, with the R^2 values very close to one; on the other hand,



(a) Gala09 (b) Gala10 (c) Soc09

(d) Soc10 (e) Doc

Fig. 3 Fitting inter-seek times for the videos of (a) Gala09, (b) Gala10, (c) Soc09, (d) Soc10, and (e) Doc, using the models of two-lognormal mixture, one lognormal distribution, and one Weibull distribution respectively.

the models of single lognormal or Weibull distribution do not fit the data well, with much smaller R^2 values.

Note that here we don't claim that the mixture of two lognormals can best fit the inter-seek times in VoD services. In fact, better fitting can be obtained by mixing more lognormal distribution components. However, when more than two lognormals are used, most of the inter-seek times are still falling into the first two distributions (i.e., Lg_1 and Lg_2) with large probabilities of θ_1 and θ_2, and the contributions from the other distributions are trivial.

Finally, we explain how we set the value of the threshold τ in Section 3. For the threshold based classification, the log-likelihood $l(\tau)$ can be expressed as

$$l(\tau) = \sum_{i=1}^{N} \log\left(I\left(x^{(i)} > \tau\right) Lg_1\left(x^{(i)}\right) + I\left(x^{(i)} \leqslant \tau\right) Lg_2\left(x^{(i)}\right)\right),$$

where τ is the threshold and $I(\cdot)$ is an indicator function. For a given video, we find x that satisfies $Lg_1(x) = Lg_2(x)$, and let $\tau = x$ as the threshold, so as to maximize the log-likelihood $l(\tau)$.

4.2 Scoring Video Segments with Viewer Interests

Based on the EM-based inter-seek time decomposition, in this section, we seek to extract viewers' interests in different parts within a video.

4.2.1 Interest scoring scheme

We consider a video divided into segments. For each segment, say, the k^{th} segment, a numerical score is assigned to indicate the level of the interest that viewers hold for it. We refer to such a score as the segment's *interest score*, denote as $s(k)$. Higher score suggests higher interest level.

Suppose that the video is sequentially divided into M segments, and there are N VCR jump records. For the i^{th} jump, denote its destination segment as $p_s^{(i)}$, and the inter-seek time between it and the next jump as $t^{(i)}$ seconds, then after the jump, a range of the segments from $p_s^{(i)}$ to $p_e^{(i)}$ are rendered by the viewer's VoD client, with $p_e^{(i)} = p_s^{(i)} + t^{(i)}$. Recall that with the EM-based decomposing algorithm, we can assign each inter-seek time with an interested probability and a non-interested probability. Specifically, for the inter-seek time $t^{(i)}$, the chances that it is drawn from the interested and from the non-interested inter-seek time distributions (i.e., Lg_1 and Lg_2) can be expressed as

$$P_j\left(t^{(i)}\right) = \frac{\theta_j Lg_j\left(t^{(i)}\right)}{\theta_1 Lg_1\left(t^{(i)}\right) + \theta_2 Lg_2\left(t^{(i)}\right)},$$

with $j = 1$ or 2 corresponding to the "interested" and "non-interested" probabilities respectively.

We compute the interest score of the k^{th} segment as

$$s(k) = \frac{\sum_{i=1}^{N} I\left(k \in \left[p_s^{(i)}, p_e^{(i)}\right)\right)\left(w_1 P_1(t^{(i)}) + w_2 P_2(t^{(i)})\right)}{N}, \qquad k = 1, 2, \cdots, M \quad (1)$$

where $I(\cdot)$ is an indicator function. In other words, for the k^{th} segment, we consider all the VCR jumps after which the segment is rendered: if $k \in \left[p_s^{(i)}, p_e^{(i)}\right)$, then the contribution from the i^{th} jump to segment k's interest score is $w_1 P_1(t^{(i)}) + w_2 P_2(t^{(i)})$, where w_1 and w_2 are two parameters deciding how an interested and a non-interested jump contribute to the score respectively. By summing up all the contributions and normalizing with the total number of the VCR jumps, we can compute the interest score for the segment.

One remaining issue is how to set the values for the two parameters, w_1 and w_2. Recall that in our analysis, an inter-seek time drawn from Lg_1 suggests that the viewer has interest in the content, while from Lg_2 suggests that he doesn't, therefore our principle is: w_1 should be set as positive, while w_2 be negative. For simplicity, in the following part of this paper, we always let $w_1 = 1$ and $w_2 = -1$.

4.2.2 Evaluation

We apply the scoring scheme on the five selected videos, and present their segment interest score sequences in Fig. 4. By comparing the seeking frequencies in Fig. 1, we can see that the videos' highlight events can be clearly identified from the interest scores. For example, for the gala videos of Gala09 and Gala10, which are composed of a series of programs, peaks and valleys that are corresponding to the popular and unpopular programs can be easily observed on the score sequences in Fig. 4 (a) and (b). To better support our claim, for Gala09 and Gala10, we compare the programs' interest scores with the voting results on the Internet[4], and compute the Pearson's correlation coefficient between the two: the correlations for Gala09 and Gala10 are 0.928 and 0.873 respectively, suggesting that the interest scores can precisely extract the viewers' interests within the galas. Besides the gala videos, for Soc09 and Soc10, we can also easily identify the peaks that are corresponding to the goal events on the score sequences in Fig. 4 (c) and (d). And for the video of Doc, five steep valleys that are corresponding to the five inserted commercials can be found in Fig. 4 (e).

For each video, we then manually bookmark its highlight events. As shown in Fig. 4 (a) and (b), for the videos of Gala09 and Gala10, we consider a program with its peak interest score higher than the median score of the entire video as a popular program, and place a bookmark at its start position; however, when there are consecutive popular programs, only one bookmark is placed at the first program. With this method, we have placed 13 and 8 bookmarks on the two gala videos. For Soc09 and Soc10, each video has 5 bookmarks placed at the five goal events, as shown in Fig. 4 (c) and (d). Finally, for the documentary video of Doc in Fig. 4 (e), we place 5 bookmarks at the ends of the five inserted commercials. Note that all the highlight bookmarks in Fig. 4 are manually places with our knowledge of the video contents. In the next section, we will propose a methodology that does not require any knowledge and semantic analysis on the video contents, and bookmarks the videos in an automated way based only on the videos' segment interest score sequences.

5 Bookmarking Algorithm

In this section, we present an algorithm that automatically bookmarks highlight events for various types of videos based on the videos' segment interest score sequences. The algorithm is motivated by the observation that viewers' interests before and after the highlights are changing rapidly (see Fig. 4), thus by detecting the abrupt positive changes on the interest scores, we can accurately locate positions of the highlight events and bookmark the videos.

Although the idea is straightforward, however, it is challenging to develop a methodology that is accurate enough with both low false positive and false negative ratios. Note that it is conflicting to achieve the two goals simultaneously. Moreover, the solution should be independent of the video type, and should work without any knowledge and analysis on the video contents.

Since the problem of placing bookmarks based on a video's segment interest score sequence is essentially a problem of detecting changes on data series, in our solution, we incorporate the time-series change detection algorithm of *CUSUM* developed by Page [11] and the *Memory-Based Graph Theoretic (MB-GT)* algorithm proposed in [12] for this purpose.

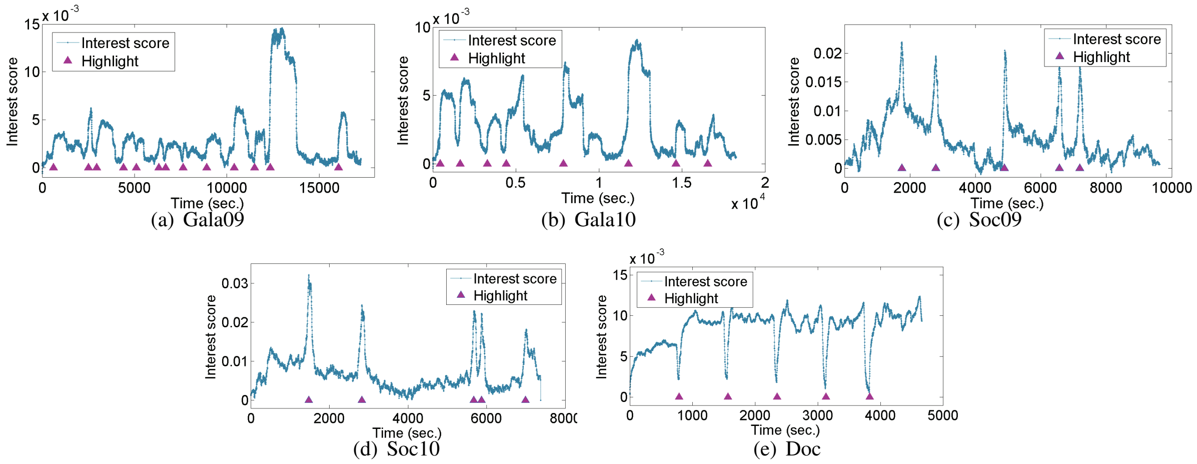
5.1 Preliminary

Before presenting the algorithm, we first review CUSUM and MB-GT briefly.

5.1.1 CUSUM

In CUSUM, data samples come in a sequence of $\{s_i\}$. The algorithm employs two control variables, U_i and L_i, to indicate the changes in the positive and negative directions (i.e., increasing and decreasing trends of the data).

[4] After the galas were first broadcasted on the New Year's Eves, viewers from all over the world voted for their favorite programs on the Internet. The voting for Gala09 is at `http://bbs.people.com.cn/voteResult.do?voteId=246` and the voting for Gala10 is at `http://page.vote.qq.com/?id=68246\&result=yes`.



(a) Gala09 (b) Gala10 (c) Soc09 (d) Soc10 (e) Doc

Fig. 4 Segment interest scores and manually identified highlights for the videos of (a) Gala09, (b) Gala10, (c) Soc09, (d) Soc10, and (e) Doc.

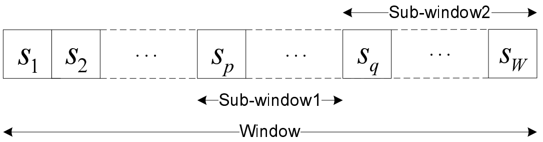


Fig. 5 Demonstration of the MB-GT algorithm.

The CUSUM algorithm starts with $U_0 = L_0 = 0$, and for each new sample s_i, U_i and L_i are updated using the following formula

$$U_i = \max\{0, (U_{i-1} + s_i - k_1)\}$$
$$L_i = \min\{0, (L_{i-1} + s_i - k_2)\},$$

where k_1 and k_2 are reference parameters that are usually set as above and below the mean of the samples respectively.

After processing each data sample, say s_i, the newly updated U_i and L_i are compared with thresholds. More specifically, given a threshold parameter ζ, if $U_i \geqslant \zeta$, a positive change is detected, and for $L_i \leqslant -\zeta$, a negative one is announced. After sequentially processing all the samples, a number of positive and negative changes will be detected on the sample sequence. Note that CUSUM is very efficient with a time complexity of $O(M)$, with M being the size of the sample sequence.

5.1.2 MB-GT

The MB-GT algorithm is developed to detect if there exists an abrupt change within a given window on the sample sequence. As demonstrated in Fig. 5, for a sample window as $\{s_1, \cdots, s_W\}$, where W is the constant window size, we can use two parameters p $(1 < p < W - 1)$ and q $(p + 1 < q < W)$ to define two sub-windows as $\{s_p, \cdots, s_{q-1}\}$ and $\{s_q, \cdots, s_W\}$, and the Euclidean distance between the two sub-windows can be computed as

$$d_{p,q} = \frac{\sum_{k=p}^{q-1} \sum_{l=q}^{W} \Delta_{k,l}}{(q-p)(W-q+1)},$$

where $\Delta_{k,l} = (s_l - s_k)$ or $(s_k - s_l)$, depending on whether the to-be-detected change is positive or negative.

The algorithm exhaustively searches all the feasible (p, q) pairs and find the maximum distance as $d_{\max} = \arg\max_{p,q}\{d_{p,q}\}$, and compare $d_{\max}$ with a threshold ξ: if $d_{\max} \geqslant \xi$, MB-GT claims that there exists an abrupt change, and use the corresponding parameter of q as the change position; otherwise, the algorithm announces that no abrupt change is detected within the window.

For exhaustively searching through all the (p, q) pairs, the algorithm's time complexity is $O(W^4)$. However, by employing the dynamic programming technique as suggested in [12], the overhead can be reduced to $O(W^2)$.

5.2 Algorithm Description

With a complexity of $O(M)$, CUSUM is very efficient, and it is proved to be optimal, in term of the detection delay (i.e., the distance between the announced change position and the position that the change actually happens), when the sample distribution parameters before and after the change are already known [16]. However, we find that it is impropriate to run CUSUM directly on a video's segment interest score sequence, because of two reasons: First, as the interest scores fluctuate with the viewers' interests significantly, by directly applying CUSUM on an entire video's score sequence, the algorithm will be influenced by the global

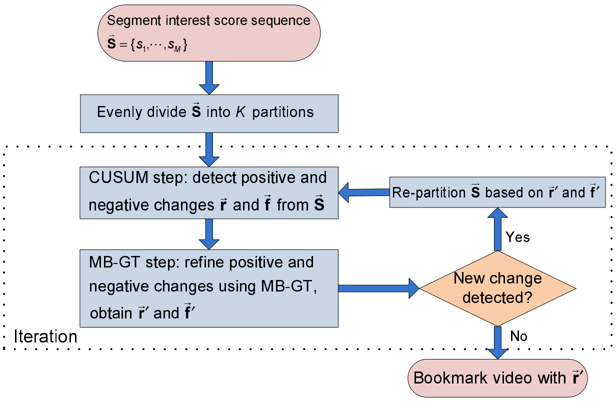

Segment interest score sequence $\vec{\mathbf{S}} = \{s_1, \cdots, s_M\}$
Evenly divide $\vec{\mathbf{S}}$ into K partitions
CUSUM step: detect positive and negative changes $\vec{\mathbf{r}}$ and $\vec{\mathbf{f}}$ from $\vec{\mathbf{S}}$
Re-partition $\vec{\mathbf{S}}$ based on $\vec{\mathbf{r}}'$ and $\vec{\mathbf{f}}'$
Yes
MB-GT step: refine positive and negative changes using MB-GT, obtain $\vec{\mathbf{r}}'$ and $\vec{\mathbf{f}}'$
New change detected?
Iteration
No
Bookmark video with $\vec{\mathbf{r}}'$

Fig. 6 High-level description of the bookmarking algorithm.

characteristics of the interest scores, and may fail to capture the local abrupt changes. Taking the gala type videos as examples, as we can see in Fig. 4 (a) and (b), different programs in a gala have different interest levels. Since there are dozens of programs, the average and variance of the scores from the entire video do not accurately reflect the viewers' normal interest levels and interest dynamics in one particular program, therefore can not be used to detect the significant deviations of the viewers' interests from their normal interest level in this program. Second, although CUSUM is proved to be optimal when the score distributions before and after the changes are pre-specified, however, in our application, it is impossible to know such distributions in advance, and sometimes the changes detected by CUSUM have considerably large delays. To overcome the first problem, we propose to divide the video into partitions, apply CUSUM on each partition independently, and adjust the partitioning in an iterative way. Ideally, we would like to divide the video in a way such that each partition contains at most one highlight event. This is because by removing the interferences from the other highlights, the algorithm will be more sensitive to the abrupt changes on the score sequence caused by the local highlight events. For handling the second problem, we incorporate the MB-GT algorithm to repeatedly evaluate and refine the changes detected by CUSUM.

Fig. 6 presents a high-level description of our proposed bookmarking algorithm. As shown in the graph, given a video's segment interest score sequence $\vec{\mathbf{S}} = \{s_1, \cdots, s_M\}$, we first evenly divide $\vec{\mathbf{S}}$ into K partitions, then the algorithm works iteratively to discover the highlight events on the video. During each iteration, there are three steps: the CUSUM step, the MB-GT step, and the checking and re-partitioning step. In the CUSUM step, the algorithm independently applies CUSUM on each video partition to detect abrupt positive and negative changes; we use $\vec{\mathbf{r}}$ to denote the detected positive changes concatenated from all the partitions, and use $\vec{\mathbf{f}}$ to denote all the negative ones. Each change detected by CUSUM is evaluated and refined in the MB-GT step, and we denote the refined positive and negative changes as $\vec{\mathbf{r}}'$ and $\vec{\mathbf{f}}'$ respectively. After the CUSUM and MB-GT steps, if at least one abrupt change is newly detected, the algorithm re-partitions $\vec{\mathbf{S}}$ based on $\vec{\mathbf{r}}'$ and $\vec{\mathbf{f}}'$, and enters into the next iteration; on the other hand, if no new abrupt change is found, the algorithm returns the currently detected positive changes $\vec{\mathbf{r}}'$ as the positions for the bookmarks. In the following, we describe the CUSUM step, the MB-GT step and the step of checking and re-partitioning in details.

5.2.1 The CUSUM step

In this step, we apply CUSUM on each video partition, and concatenate the detected abrupt positive and negative changes as $\vec{\mathbf{r}}$ and $\vec{\mathbf{f}}$ respectively.

For the parameters of k_1, k_2, and ζ in CUSUM, we let $k_1 = \mu + \frac{\delta}{2}$ and $k_2 = \mu - \frac{\delta}{2}$, where μ and δ are the mean and standard deviation of the interest scores of the segments in the video partition. We let $\zeta = 3 \times \delta$ if not otherwise specified.

5.2.2 The MB-GT step

In this step, we apply the MB-GT algorithm to evaluate and refine the changes $\vec{\mathbf{r}}$ and $\vec{\mathbf{f}}$ detected by CUSUM. More specifically, for each $r \in \vec{\mathbf{r}}$, we construct a window of size W on the sequence $\vec{\mathbf{S}}$, with r as the center of the window, then we apply the MB-GT algorithm to find the segment that introduces the maximum distance $d_{\max}$ within the window. If $d_{\max}$ is larger than a threshold ξ, we denote the corresponding position q as a new positive change r', and use it to replace r in $\vec{\mathbf{r}}$; otherwise, we consider that the change within the window is not significant enough, and remove r from $\vec{\mathbf{r}}$. After applying MB-GT upon all the changes in $\vec{\mathbf{r}}$, we can obtain a list of the refined positive changes on $\vec{\mathbf{S}}$, which we denote as $\vec{\mathbf{r}}'$. We evaluate the negative changes and obtain a refined change series $\vec{\mathbf{f}}'$ through a similar approach.

For running the MB-GT algorithm, we set the window size W as 200 segments. For the threshold ξ, it is indeed the key parameter that determines the accuracy of the bookmarking results: If ξ is too large, the algorithm is too conservative, and may cause false negative errors; on the other hand, when ξ is too small, the algorithm is too aggressive, and will introduce considerable false positive errors. In Section 6, we will explore the influence of this parameter with experiments.

5.2.3 Checking and re-partitioning

As previously discussed, in our bookmarking algorithm we prefer to divide a video's interest score sequence into partitions, so that each partition encompasses no more than one highlight event. Here we describe our methodology for partitioning the video in the checking and re-partitioning step.

After the CUSUM and MB-GT steps, we compare the detected positive and negative changes $\vec{\mathbf{r}'}$ and $\vec{\mathbf{f}'}$ with the ones that are detected in all the previous iterations, denote as $\vec{\mathbf{r}'}_{prev}$ and $\vec{\mathbf{f}'}_{prev}$, to see if any new changes are detected. More specifically, for any $r' \in \vec{\mathbf{r}'}$, if there exists a $r'_{prev} \in \vec{\mathbf{r}'}_{prev}$, such that $|r' - r'_{prev}| \leqslant 50$; or for any $f' \in \vec{\mathbf{f}'}$, there exists a $f'_{prev} \in \vec{\mathbf{f}'}_{prev}$, such that $|f' - f'_{prev}| \leqslant 50$, we say that no new change is detected, and the algorithm returns $\vec{\mathbf{r}'}$ as the final positions for bookmarking the video.

On the other hand, if at least one change is newly detected, we re-partition the video, based on the currently detected change series $\vec{\mathbf{r}'}$ and $\vec{\mathbf{f}'}$, and enter into a new iteration. The video is re-partitioned using the following rules:

1. For the first positive change r'_0 in $\vec{\mathbf{r}'}$, let $f'_p \in \vec{\mathbf{f}'}$ be its immediate next negative change, i.e., $f'_p = \min\{f'_j | f'_j > r'_0, f'_j \in \vec{\mathbf{f}'}\}$.
2. Repeat
   (a) Let r'_q be f'_p's immediate next positive change, that is, $r'_q = \min\{r'_i | r'_i > f'_p, r'_i \in \vec{\mathbf{r}'}\}$. If r'_q could be found, then we partition the video at $\frac{(r'_q + f'_p)}{2}$, and go to Step 2b; otherwise, we have reached the end of the video, and exit the re-partitioning.
   (b) Let f'_p be the immediate next negative change of the positive change r'_q found in Step 2a, i.e., $f'_p = \min\{f'_j | f'_j > r'_q, f'_j \in \vec{\mathbf{f}'}\}$. If f'_p can be found, go to Step 2a; otherwise, we have reached the end of the video, and exit the re-partitioning.

From the description of the re-partitioning rules, we can see that in each algorithm iteration, we partition the video at the mid-position of each pair of consecutive negative and positive changes that is currently detected. By partitioning the video at each "valley" on the interest score sequence, we can avoid including multiple highlight events into one partition (otherwise, the partition will have consecutive negative and positive change pairs, and should be further divided).

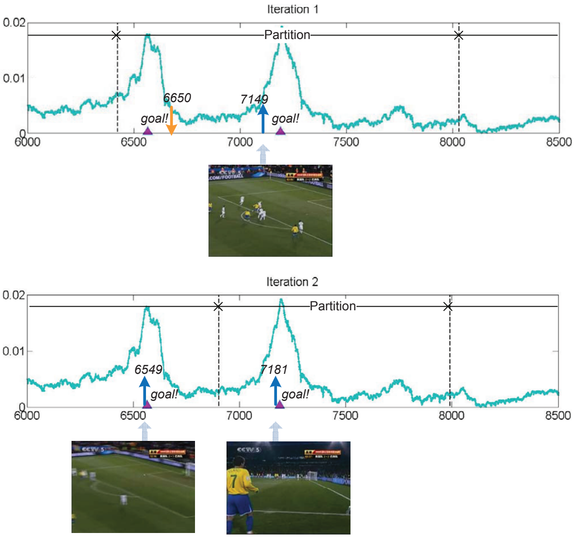


Fig. 7 An example of re-partitioning.

We use a part of the interest score sequence on Soc09 to demonstrate the re-partitioning idea. As shown in Fig. 7, in the initial iteration in which the video is evenly partitioned, as shown in the top figure, the two goals at the $6,566^{th}$ and $7,190^{th}$ seconds are included in one partition, and the CUSUM and MB-GT steps only detect one positive change, which is indeed far away from the second goal. After re-partitioning the video at the "valley" at $6,899^{th}$ second, which is the mid-position of the consecutive negative and positive changes at the $6,650^{th}$ and $7,149^{th}$ seconds, we put the two goals into two partitions, and both of them can be detected with much better accuracies, as we can see from the bottom figure in Fig. 7.

Finally, we claim that our algorithm runs efficiently. Note that in MB-GT, we use the parameter of ξ to filter out the changes that are not significant enough. Since the number of the changes that incur a distance larger than ξ is limited, the algorithm will eventually return with the bookmark positions. In fact in our evaluation, the algorithm runs for at most five iterations.

6 Evaluation

In this section, we carry out a comprehensive and in-depth evaluation on the methodologies we have proposed for video bookmarking. In our evaluation, we employ both the videos with ground-truth highlight events as listed in Table 1, and the videos without ground-truths (e.g., movies and TV dramas). We also investigate the influences of the historical VCR data and the key algorithmic parameters on the bookmarking accuracies.

6.1 Evaluation with Ground Truths

We first use the five selected videos of Gala09, Gala10, Soc09, Soc10, and Doc in our evaluation. We choose the five videos for the reason that all of them have ground-truth highlight events, which enables an objective evaluation.

6.1.1 Methodology and metrics

In our experiments, we evaluate and compare two bookmarking methodologies, both are based on the algorithm we have described in Section 5:

- *EM+CUSUM+MB-GT*: In this method, we compute the interest scores for the segments in a video, based on differentiating viewers' intentions, using the EM-based algorithm as described in Section 4; We then apply the algorithm that incorporates CUSUM and MB-GT as described in Section 5 on the segment interest score sequence, and detect abrupt changes for bookmarking the video.
- *CUSUM+MB-GT*: In this approach, we do not differentiate the intentions in viewers' VCR jumps, but use the times that a segment is rendered by the viewers as its interest score (or, equivalently, by letting $w_1 = w_2 = 1$ in Equation (1)); then we apply the CUSUM and MB-GT based bookmarking algorithm as described in Section 5 upon the score sequence for bookmarking.

Besides the two methods, for the soccer game videos of Soc09 and Soc10, we also apply the *cinematic template* [5] and the *logo based* [9] techniques, and compare them with our bookmarking methods. The cinematic technique exploits the high-level semantics in soccer game videos to detect the goal events, where the detection is based on four semantic rules including 30 $\sim$ 120 seconds of break duration, occurrence of close-up/out of field shot, existence of slow-motion replay, and the relative position of the replay shot. Note that although proposed ten years ago, this technique is still considered as state-of-the-art for highlighting the soccer game videos [17] [18].

The logo based technique highlights the most exciting events based on detecting the frames that showing the soccer game's logo (for example, the logo of FIFA) before and after the replay of the event. The logo based approach can be viewed as an extension of the cinematic method.

We compare the bookmarks that are positioned by the automated methods (i.e., "EM+CUSUM+MB-GT", "CUSUM+MB-GT", the cinematic template and the logo based techniques) with the ground-truth highlights in Fig. 4. For each detected bookmark, if the distance from its nearest ground-truth highlight is shorter than a threshold D, we consider it as a *properly positioned bookmark*; otherwise, it is referred to as a *mis-positioned bookmark.* In our evaluation, we let $D = 90$ seconds. For each automated bookmarking method applied on a video, we use the following metrics to evaluate its accuracy.

- **Precision**: Precision is the ratio between the properly positioned bookmarks divided by all the bookmarks detected by algorithm, i.e.,

$$\text{precision} = \frac{\text{\# of properly positioned bookmarks}}{\text{\# of all the bookmarks detected by algorithm}};$$

- **Recall**: Recall is the ratio between the properly positioned highlights divided by all the ground-truth bookmarks on the video, i.e.,

$$\text{recall} = \frac{\text{\# of properly positioned bookmarks}}{\text{\# of ground-truth highlights}};$$

- **F-measure**: F-measure is defined as

$$\text{F} - \text{measure} = \frac{2 \times \text{precision} \times \text{recall}}{(\text{precision} + \text{recall})}.$$

Obviously, among the three metrics, precision and recall are indeed related to the false positive and false negative errors respectively, and F-measure provides an overall evaluation on the bookmarking accuracy by combining the two.

In addition to the three metrics, we also consider the mean absolute error (**MAE**) of the bookmarks. That is, we compute an averaged absolute distance between a detected bookmark and its associated ground-truth highlight event, however, for the case that a bookmark is mis-positioned, and for a highlight event that is missed by the bookmarking algorithm, we consider its erroneous distance as $D = 90$ seconds in our evaluation.

6.1.2 Bookmarking accuracies

In the first experiment, we apply the bookmarking methods of "EM+CUSUM+MB-GT" and "CUSUM+MB-GT" on the five selected videos, and for the soccer game videos of Soc09 and Soc10, we also apply the cinematic and the logo

Table 4 Bookmarking results for the five videos with ground-truth highlights

		Precision	Recall	F-measure	MAE (sec.)
Gala09	EM+CUSUM+MB-GT	10/11=0.91	10/13=0.77	0.83	42.71
	CUSUM+MB-GT	9/10=0.90	9/13=0.69	0.78	46.86
Gala10	EM+CUSUM+MB-GT	7/7=1.00	7/8=0.88	0.93	32.50
	CUSUM+MB-GT	5/8=0.63	5/8=0.63	0.63	55.09
Soc09	EM+CUSUM+MB-GT	4/6=0.67	4/5=0.80	0.73	51.71
	CUSUM+MB-GT	4/7=0.57	4/5=0.80	0.67	61.25
	Cinematic template	3/9=0.33	3/5=0.60	0.43	78.64
	Logo based	5/5=1.0	5/5=1.0	1.0	41.20
Soc10	EM+CUSUM+MB-GT	4/6=0.67	4/5=0.80	0.73	51.57
	CUSUM+MB-GT	4/4=1.00	4/5=0.80	0.89	45.50
	Cinematic template	4/7=0.57	4/5=0.80	0.67	50.63
	Logo based	2/3=0.67	2/5=0.40	0.50	62.20
Doc	EM+CUSUM+MB-GT	4/5=0.80	4/5=0.80	0.80	33.00
	CUSUM+MB-GT	3/3=1.00	3/5=0.60	0.75	41.20

based methods. Table 2 presents the precisions, recalls, F-measures, and MAEs of the bookmarks positioned by different methodologies. Note that for the bookmarking approaches of "EM+CUSUM+MB-GT" and "CUSUM+MB-GT", we always let the initial partition number $K = 6$, and tune the parameter of ξ for the best performances (i.e., with the highest F-measure values). We also tune the cinematic and the logo based methods and present their best bookmarking results in Table 2.

By examining the results in the table, we can make the following observations: First, the bookmarking techniques of "EM+CUSUM+MB-GT" and "CUSUM+MB-GT" are quite accurate. For instance, for the soccer games of Soc09 and Soc10, our methods actually detected same (for Soc09) or more (for Soc10) goals than the cinematic method, although the latter is designed specifically for soccer game videos. For the logo based technique, although the method has very accurate goal detections for Soc09, it has the poorest performance on Soc10. The reason is that in this video, the logo image is overlapped upon the frames instead of replacing them, which makes the logo very difficult to be detected. Besides, all the properly positioned bookmarks of our proposed methodologies are ahead of the goal events, which is necessary for the goal detection in soccer game videos.

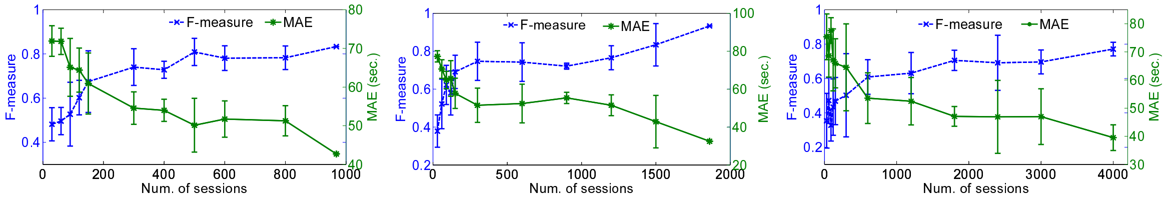
Second, we find that the approach of "EM+CUSUM+MB-GT" has better accuracies in term of F-measure values than "CUSUM+MB-GT", with the only exception on Soc10. We explain the better performance of "EM+CUSUM+MB-GT" with the fact that in this method, we apply the EM-based algorithm to differentiate viewers' intentions in their VCR jumps when computing the segment interest scores, thus by detecting the abrupt changes upon the score sequences, better bookmark accuracies can be obtained. For the videos of Soc09 and Soc10, after carefully examining the bookmarks positioned by "EM+CUSUM+MB-GT", we find that the mis-positioned bookmarks are actually the kick-offs and award ceremony, which can also be considered as the highlight events for a soccer game video (although we only use the goals as the ground-truth highlights). In fact, if we include kick-offs and ceremony as the ground-truth highlight events, then the scheme of "EM+CUSUM+MB-GT" will have better accuracies than "CUSUM+MB-GT", regarding both the F-measure values and MAEs.

In conclusion, through evaluation using videos with ground-truth highlights, we can see that *our proposed bookmarking technique of "EM+CUSUM+MB-GT" can detect the highlights and position the bookmarks with considerable accuracies, and it has superior performances than the "CUSUM+MB-GT" method that doesn't differentiate viewers' VCR jump intentions; moreover, unlike the semantic methods that can only work for certain specific kind of videos (e.g., soccer videos), our propose technique has a good performance for all the video types.*

6.1.3 How long a history is enough?

Unlike the approaches that exploit videos' semantics, our proposed bookmarking technique relies on viewers' historical VCR jump records. In this experiment, we seek to answer the question: how many historical data is enough for accurately positioning the bookmarks?

To answer this question, we apply the "EM+CUSUM+MB-GT" method on the videos of Gala09, Gala10, and Doc. We do not select the soccer game videos as they have relatively fewer VCR jump records. In each round of bookmarking, we only use a subset of the sessions from the video's campus VoD trace, and apply the algorithm.



(a) Gala09 (b) Gala10 (c) Doc

Fig. 8 Bookmarking accuracies when only partial historical VCR jump data is employed for the videos of (a) Gala09, (b) Gala10, and (c) Doc.

The experiment results are presented in Fig. 8. In the figures, the *x*-axle is the number of the sessions randomly selected from the VoD trace used for bookmarking, and the two *y*-axles are the F-measure value and MAE respectively.

From Fig. 8, we can see that in general, when more historical VCR jump records are employed, bookmarks with better accuracies can be expected, in terms of both higher F-measure values and smaller MAEs. Furthermore, by studying the curves in the figures, we can see that the bookmark qualities are improved in two phases: Initially, when a moderate amount of the historical data is employed, the bookmarking accuracies are improved dramatically. For example, for the videos of Gala09 and Gala10, the F-measure values increase by 40.1% and 82.4% respectively, when as few as 150 VoD sessions are employed for bookmarking; for Doc, the F-measure value increases by 41.9% when 300 sessions are used. Afterwards, when more and more historical data is employed, the bookmarking accuracies are further improved, but not as aggressively as in the initial phase. Note that when a rich set of the historical VCR jump records is available, bookmarks with very high qualities can be expected. For example, for the video of Gala10, when jumps in all its 1,862 VoD sessions are used for bookmarking, we can precisely detect seven of the eight highlights with an F-measure value as high as 0.93, as shown in Table 2.

The observation from this experiment suggests that our bookmarking methodology can achieve considerably high accuracies with only moderate amount of VCR jump records; and the qualities of the bookmarks can be further improved when more VCR data is available. With this "fast start" feature, the method is very suitable for the large-scaled Internet VoD services with huge viewer populations.

6.1.4 Influence of threshold parameter

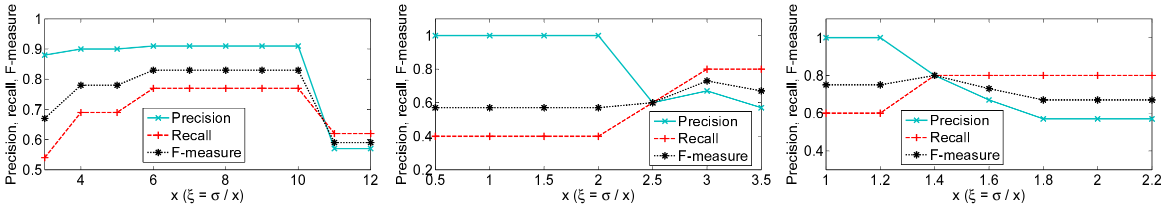
In our bookmarking algorithm described in Section 5, we use the parameter ξ to evaluate and refine the candidate bookmarks that are detected by CUSUM. In this experiment, we explore the tradeoff between the false positive (in terms of recall) and false negative (in terms of precision) of the bookmarking algorithm introduced by varying the parameter value.

We select the videos of Gala09, Soc09, and Doc, each representing one video type, and apply the "EM+CUSUM+MB-GT" method on their segment interest score sequences, but each time we use different ξ values. In particular, we let $\xi = \sigma/x$, where σ is the standard deviation of the interest scores for the segments in the window around the candidate bookmark, and x is a variable. Fig. 9 presents the precision, recall, and F-measure of the bookmarking results under various x values. From the figure one can see that when ξ is too large (with small x), the algorithm is too selective with higher precisions, but makes considerable false negative errors and results in lower recalls; on the other hand, when ξ is set too small, the algorithm makes false positive errors, and has low precision values.

Unfortunately, there is no universal optimal setting for the parameter ξ in our bookmarking algorithm. In fact, ξ depends on how viewers' interests on the highlight events deviate from their interest levels on the other parts of the video. For example, for the soccer game and documentary videos, in which viewers' interests on the goals and the commercials are significantly higher and lower than the interests on the ordinary video contents, we have relatively larger ξ values as the optimal. Meanwhile, for the gala video, where the interest scores for highlights and regular programs are not dramatically different, the optimal value for ξ is much smaller. We remark that for highlight event detection approaches based on video semantics analysis, it is also generally infeasible to have an universal parameter setting that is optimal for all the videos.

6.2 Evaluation without Ground Truths

In addition to the five videos with ground-truth highlight events as listed in Table 1, we also apply our proposed automated bookmarking techniques on the videos without



(a) Gala09 (b) Soc09 (c) Doc

Fig. 9 Influences of the threshold parameter ξ on the bookmarking accuracies for (a) Gala09, (b) Soc09, and (c) Doc.

Table 5 Bookmarks on videos without ground-truth highlight events

	Movie	TV drama	Cartoon
Videos	5	4	1
Bookmarks	37	26	9

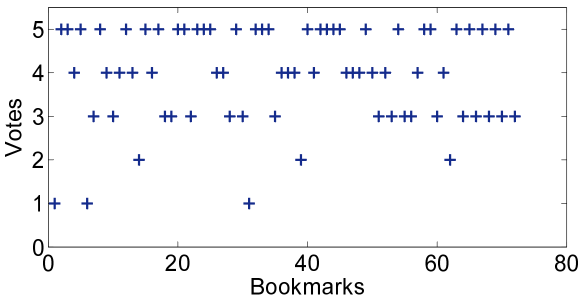


Fig. 10 Voting results on the bookmarks by volunteers.

ground truths, and evaluate the bookmark accuracies.

We randomly select 10 videos from the campus VoD traces, and apply the "EM+CUSUM+MB-GT" method on them. The videos include five movies, four TV dramas, and one cartoon. A total number of 72 bookmarks were generated. By manually examining these bookmarks, we find that they cover a wide range of highlight events, including big scenery switches, debuts of the main characters, starts of fighting and funny parts, ends of commercials, etc.. Table 5 lists the numbers and distributions of the bookmarks.

As there are no ground-truth highlights in these videos, we evaluate the bookmarks through subjective user tests. For each bookmark, five volunteers were asked to decide whether the bookmark suggests a highlight or not; and for each detected bookmark, it would have 0 to 5 votes, indicating how many volunteers considered it as properly positioned.

Fig. 10 shows the votes on the 72 bookmarks. From the figure, we can see that the bookmarks are quite accurate: among the 72 bookmarks, 30 of them have 5 votes, which means everyone consider these bookmarks as properly positioned; in addition, none of the bookmarks has zero vote; if we employ a majority rule by considering a bookmark with over 3 votes as properly positioned, then 66 bookmarks, constituting 91.7% of all the bookmarks generated by our method, are accurately positioned on the videos.

7 Related Work

Many ideas were proposed to support the VCR functionality under the context of P2P-assisted VoD services in recent years. Zheng et al. [1] considered viewer's VCR pattern, and developed a distributed prefetching scheme for minimizing the expected seeking delay. He et al. [2] mined the associations inside videos, and proposed to predict and prefetch the requested contents in future VCR jumps based on peers' VCR information collected from a gossip protocol. Xu et al. [3] mined viewers' access patterns in VoD services, and developed a personalized prefetching scheme based on collaborative filtering. Although these methods are proved to be able to predict viewers' future jumps to some extent, however, as we have seen in this paper, since viewers have different intentions in making VCR jumps, and most of the jumps do not necessarily lead to attractive contents, it is fundamentally difficult to improve a viewer's overall VCR service experience by passively predicting his future jumps.

Automated event detection technique based on semantic analysis has been a hot topic in the area of video technology for years. Ekin et al. [5] proposed a cinematic template technique for detecting highlights in soccer game videos. Eldib et al. [9] summarizes the soccer game highlights by detecting the logo image appearing before and after the events. Tong et al. [7] exploited replay scenes to identify and locate highlights in sport videos. Zhu et al. [19] analyzed the human behaviors for ranking the highlight segments extracted from racket game videos. Xu et al. [6] exploited the webcast texts and combined with the video structure analysis to detect the highlight event moments and moment boundaries in sport game videos. Although these techniques are effective in detecting the highlight events in certain types of videos, like sport videos, they can not be applied to videos of other types. On the other hand, our proposed technique is applicable for a wide-range of video types.

Besides the semantic analysis based approaches, some recent studies propose to capture viewers' physiological signals for highlight detection. In [20], five signals (electrodermal activity, heart rate, blood volume pulse, respiration rate, and respiration amplitude) are measured from the viewers to generate the video highlight summaries. Joho et al. [21] used web cameras to capture the facial activities of the viewers for detecting the video highlight events. Our approach, however, does not require any physiological signal capturing sensors, but exploits the viewers' VCR behavior data that is already recorded in most existing VoD systems to infer their intentions in the VCR jumps, therefore is more applicable for large scale Internet VoD services.

Some recent studies are focusing on analyzing and understanding key characteristics of the viewers' behaviors in IPTV and on-demand streaming media systems. Chang et al. [22] reported that viewers' inter-arrival times could be modeled with a mixture of two exponential distributions. While in [14], Garcia et al. proposed to use two exponential distributions to estimate a viewer's total watching length in a video, and how the two distributions are mixed depends on the viewer's interest level; they also reported that viewers' inter-seek times can be modeled with a Weibull distribution. Brampton et al. [4] suggested that the inter-seek times follow a lognormal distribution, and through a controlled experiment, they showed that with manually positioned bookmarks, viewer will have a much better service experience. Our work differs from these studies in that we propose to use mixtures of two lognormal distributions to fit the empirical inter-seek times, moreover, we present a bookmarking algorithm that accurately positions bookmarks on various types of videos with high accuracies.

8 Conclusion

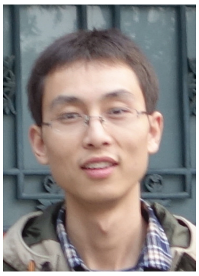
In this paper, we proposed an automated bookmarking methodology that extracts viewers' interests from their historical VCR behaviors, and positions bookmarks on videos for supporting the VCR functionality in VoD services. By analyzing traces from a real-world VoD service, we first showed that it is fundamentally difficult to improve a viewer's VCR experience without understanding his intentions in making the VCR jumps. We differentiated viewers' intentions in VCR jumps by decomposing the inter-seek times, using an Expectation-Maximization (EM) algorithm, and extracted viewers' interests within a video by computing an interest score for each video segment. Based on the interest scores, we proposed a bookmarking algorithm that incorporates the time-series change detection techniques of CUSUM and MB-GT, and bookmark the videos by detecting the abrupt changes on the videos' segment interest score sequences. Finally, we evaluated our proposed techniques with VCR traces of various types of videos. Experimental evaluations suggest that our proposed automated bookmarking methodology can be used to improve the viewers' VCR service experiences effectively.

Acknowledgements This work was supported by National Natural Science Foundation of China under Grant No. 61202405, the sub task of Strategic Priority Research Program of the Chinese Academy of Sciences under Grant No. XDA06010301, and Anhui Provincial Natural Science Foundation under Grand No.1208085QF106; Yong Liu was partially supported by National Science Foundation of USA, under contracts CNS-0953682 and CNS-1018032.

References

1. Zheng C, Shen G, Li S. Distributed prefetching scheme for random seek support in peer-to-peer streaming applications. In: Proc. of ACM Workshop on Advances in Peer-to-Peer Multimedia Streaming (P2PMMS'05). Nov. 2005
2. He Y, Liu Y. VOVO: VCR-oriented video-on-demand in large-scale peer-to-peer networks. IEEE Trans. Parallel Distrib. Syst., 2009, 20(4): 528 – 539
3. Xu T, Ye B, Wang Q, Li W, Lu S, Fu X. APEX: A personalization framework to improve quality of experience for dvd-like functions in P2P VoD applications. In: Proc. of IEEE International Workshop on Quality of Service (IWQoS'10). Jun. 2010
4. Brampton A, MacQuire A, Fry M, Rai I A, Race N J P, Mathy L. Characterising and exploiting workloads of highly interactive video-on-demand. Multimed. Sys., 2009, 15(1): 3 – 17
5. Ekin A, Tekalp A M, Mehrotra R. Automatic soccer video analysis and summarization. IEEE Trans. Image Process., 2003, 12(7): 796 – 807
6. Xu C, Zhang Y F, Zhu G, Rui Y, Lu H, Huang Q. Using webcast text for semantic event detection in broadcast sports video. IEEE Trans. Multimedia, 2008, 10(7): 1342 – 1355
7. Tong X, Liu Q, Zhang Y, Lu H. Highlight ranking for sports video browsing. In: Proc. of ACM Multimedia'05. Nov. 2005
8. Qian X, Wang H, Liu G, Hou X. HMM based soccer video event detection using enhanced mid-level semantic. Multimed. Tools Appl., 2012, 60(1): 233 – 255

9. Eldib M Y, Zaid B S A, Zawbaa H M, El-Zahar M, El-Saban M. Soccer video summarization using enhanced logo detection. In: Proc. of ICIP'09. Nov. 2009
10. Hastie T, Tibshirani R, Friedman J. The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd edition. Springer, 2009
11. Page E S. Cumulative sum charts. Technometrics, 1961, 3(1): 1 – 9
12. Nikovski D, Jain A. Fast adaptive algorithms for abrupt change detection. Machine Learning, 2010, 79(3): 283 – 306
13. He Y, Shen G, Xiong Y, Guan L. Optimal prefetching scheme in P2P VoD applications with guided seeks. IEEE Trans. Multimedia, 2009, 11(1): 138 – 151
14. Garcia R, G.Paneda X, Garcia V, Melendi D, Vilas M. Statistical characterization of a real video on demand service: User behaviour and streaming-media workload analysis. Simulation Modelling Practice and Theory, 2007, 15(6): 672 – 689
15. Claypool M, Le P, Waseda M, Brown D. Implicit interest indicators. In: Proc. of International Conference on Intelligent User Interfaces (IUI'01). Jan. 2001
16. Basseville M, Nikiforov I V. Detection of Abrupt Changes: Theory and Application. Prentice-Hall, 1993
17. Tjondronegoro D, Chen Y P. Knowledge-discounted event detection in sports video. IEEE Trans. Syst. Man Cybern A, 2010, 40(5): 1009 – 1024
18. Chênes C, Chanel G, Soleymani M, Pun T. Highlight detection in movie scenes through inter-users, physiological linkage. Social Media Retrieval, 2010, 217 – 237
19. Zhu G, Huang Q, Xu C, Xing L, Gao W, Yao H. Human behavior analysis for highlight ranking in broadcast racket sports video. IEEE Trans. Multimedia, 2007, 9(6): 1167 – 1182
20. Money A G, Agius H. ELVIS: Entertainment-led video summaries. ACM Trans. Multimedia Computing, Communications and Applications, 2010, 6(3): 1 – 17
21. Joho H, Staiano J, Sebe N, Jo J M. Looking at the viewer: analysing facial activity to detect personal highlights of multimedia contents. Multimed. Tools Appl., 2011, 51(2): 505 – 523
22. Chang B, Dai L, Cui Y, Xue Y. On feasibility of P2P on-demand streaming via empirical VoD user behavior analysis. In: Proc. of IEEE ICDCS'08. Jun. 2008



Yang ZHAO received the B.S. degree in Computer Science from University of Science and Technology of China (USTC) in 2009. He is currently a Ph.D. candidate in the Department of Computer Science and Technology in USTC. His research interests include multimedia networks and vehicular ad hoc networks.

Ye TIAN is an associate professor at the School of Computer Science and Technology, University of Science and Technology of China (USTC). He joined USTC in August 2008. He received his Ph.D. degree from the Department of Computer Science and Engineering at The Chinese University of Hong Kong (CUHK) in December 2007. His research interests include Internet and network measurement, information-centric networks, online social networks, and multimedia networks. He is a member of IEEE, and a senior member of China Computer Federation (CCF). He is currently serving as an associate editor for Frontiers of Computer Science.

Yong LIU is an associate professor at the Electrical and Computer Engineering department of the Polytechnic Institute of New York University (NYU-Poly). He received his Ph.D. degree from Electrical and Computer Engineering department at the University of Massachusetts, Amherst, in May 2002. His general research interests lie in modeling, design and analysis of communication networks. His current research directions include Peer-to-Peer systems, overlay networks, network measurement, online social networks, and recommender systems. He is the winner of the IMC Best Paper Award in 2012, INFOCOM Best Paper Award in 2009, and the IEEE Communications Society Best Paper Award in Multimedia Communications in 2008. He is a senior member of IEEE and member of ACM. He is currently serving as an associate editor for IEEE/ACM Transactions on Networking, and Elsevier Computer Networks Journal.