Automated Rich Presentation of a Semantic Topic

Lie Lu and Zhi-Wei Li
Microsoft Research Asia
{llu, zli}@microsoft.com

ABSTRACT

To have a rich presentation of a topic, it is not only expected that many relevant multimodal information, including images, text, audio and video, could be extracted; it is also important to organize and summarize the related information, and provide users a concise and informative storyboard about the target topic. It facilitates users to quickly grasp and better understand the content of a topic. In this paper, we present a novel approach to automatically generating a rich presentation of a given semantic topic. In our proposed approach, the related multimodal information of a given topic is first extracted from available multimedia databases or websites. Since each topic usually contains multiple events, a text-based event clustering algorithm is then performed with a generative model. Other media information, such as the representative images, possibly available video clips and flashes (interactive animates), are associated with each related event. A storyboard of the target topic is thus generated by integrating each event and its corresponding multimodal information. Finally, to make the storyboard more expressive and attractive, an incidental music is chosen as background and is aligned with the storyboard. A user study indicates that the presented system works quite well on our testing examples.

Categories and Subject Descriptors

H.5.3 [**Information Interfaces and Presentation**]: Group and Organization Interfaces - *Organizational design;* H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing - a*bstracting methods*.

General Terms

Algorithms, Design, Management, Experimentation, Theory

Keywords

Rich presentation, multimodality, multimedia authoring, storyboard, events clustering, multimedia fusion

1. INTRODUCTION

In the multimedia field, a major objective of content analysis is to discover the high-level semantics and structures from the low-level features, and thus to facilitate indexing, browsing, searching, and managing the multimedia database. In recent years, a lot of technologies have been developed for various media types, including images, video, audio and etc. For example, various approaches and systems have been proposed in image content analysis, such as semantic classification [1], content-based image retrieval [2] and photo album management [3]. There are also a lot of research focuses on video analysis, such as video segmentation [4], highlight detection [5], video summarization [6][7], and video structure analysis [8], applied in various data including news video, movie and sports video. Since audio information is very helpful for video analysis, many research works on audio are also developed to enhance multimedia analysis, such as audio classification [9], and audio effect detection in different audio streams [10]. Most recently, there are more and more approaches and systems integrating multimodal information in order to improve analysis performance [11][12].

The main efforts of the above mentioned research have focused on understanding the semantics (including a topic, an event or the similarity) from the multimodal information. That is, after the multimedia data is given, we want to detect the semantics implied in these data. In this paper, we propose a new task, *Rich Presentation*, which is an inverse problem of the traditional multimedia content analysis. That is, if we have a semantic topic, how can we integrate its relevant multimodal information, including image, text, audio and video, to richly present the target topic and to provide users a concise and informative storyboard? In this paper, the so-called "semantic topic" is a generic concept. It could be any keyword representing an event or events, a person's name, or anything else. For example, "*World Cup 2002*" and "*US election*" could be topics, as well as "*Halloween*" and "*Harry Potter*". In this paper, our task is to find sufficient information on these topics, extract the key points, fuse the information from different modalities, and then generate an expressive storyboard.

Rich presentation can be very helpful to facilitate quickly grasping and better understanding the corresponding topic. People usually search information from (multimedia) database or the Internet. However, what they get is usually a bulk of unorganized information, with many duplicates and noise. It is tedious and costs a long time to get what they want by browsing the search results. If there is a tool to help summarize and integrate the multimodal information, and then produce a concise and informative storyboard, it will enable users to quickly figure out the overview contents of a topic that they want to understand. Rich presentation provides such a tool, and thus it could have many potential applications, such as education and learning, multimedia authoring, multimedia retrieval, documentary movie production, and information personalization.

In this paper, we will present the approach to rich presentation. In order to produce a concise and informative storyboard to richly

MM'05.

present a target topic, we need to answer the following questions. 1) How to extract the relevant information regarding the target topic? 2) How to extract the key points from the relevant information and build a concise and informative storyboard? 3) How to fuse all the information from different modality? and 4) how to design the corresponding rendering interface?

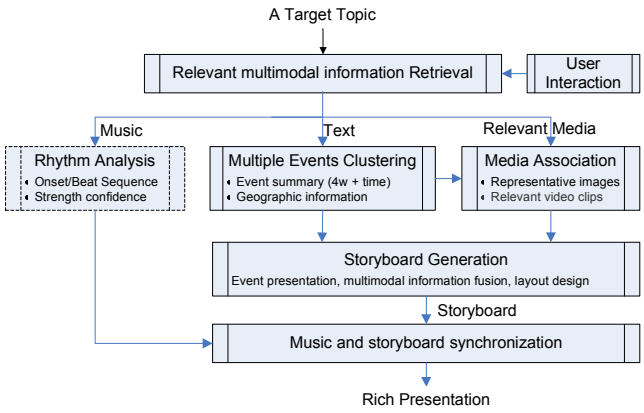

Fig. 1 The system framework of rich presentation of a target semantic topic. It is mainly composed of three steps, relevant multimodal information extraction, media analysis, and rich presentation generation.

In this paper, we propose a number of novel approaches to deal with the above issues and also present an example system. Fig. 1 illustrates the proposed system framework of rich presentation. It is mainly composed of three steps, relevant multimodal information extraction, media analysis including multiple events clustering, representative media detection and music rhythm analysis; and the final storyboard generation and music synchronization.

In the proposed system, given the semantic topic, the relevant information, including text, image, video and music, is first extracted from the available multimedia database or the web database. User interaction is also allowed to provide extra relevant material or give relevant feedback. Then, the information is summarized, with an event clustering algorithm, to give a concise representation of the topic and figure out the overview of the contents. Other multimedia materials, such as representative images (or image sequences) and geographic information, are subsequently associated with each event. In the next step, all the above information is integrated to generate a storyboard, in which each event is presented as one or multiple slides. An incidental music, which is also possibly relevant to the topic, is finally synchronized with the storyboard to improve its expressiveness and attractiveness. Thus, with these steps, a concise and informative rich presentation regarding the target topic is generated.

The rest of the paper is organized as follows. Section 2 discusses the relevant information extraction corresponding to the target topic. Section 3 presents our approach to the topic representation, including multiple events clustering, event description, and representative media selection. Section 4 describes the approach to rich presentation generation, including storyboard generation, incidental music analysis and synchronization. Experiments and evaluations are presented in the Section 5. Conclusions are given in the Section 6.

2. OBTAINING RELEVANT INFORMATION

To obtain the multimodal information which is relevant to the input topic (keyword), generally, we could search them from various databases which have been indexed with the "state-of-the-art" multimedia analysis techniques. However, in current stage, there is lack of such publicly available multimedia databases. The public search engine like MSN or Google indexes all the Internet web-pages and can return a lot of relevant information, but the search results usually contain much noise. We could also build a private database for this system to provide more relevant and clean results, but it will be too much expensive to collect and annotate sufficient multimedia data for various topics. In order to obtain relatively accurate and sufficient data for an arbitrary topic, in our system, we chose to collect the relevant multimodal information of the given topic from the news websites such as MSNBC, BBC and CNN, instead of building an available database from the scratch. These news websites are usually well organized and managed; and contain various kinds of high quality information including text, image and news video clips. Although the news websites are used as the information sources in our system, other various multimedia databases can be also easily incorporated into the system if they are available.

Instead of directly submitting the topic as a query and getting the returned results by using the search function provided by the websites, in our system, we crawled the news documents from these websites in advance and then build a full-text index. It enables us to quickly obtain the relevant documents, and also enable us to use some traditional information retrieval technologies, such as query expansion [13], to remove the query ambiguousness and get more relevant documents.

In our approach, user interaction is also allowed to provide more materials relevant to the topic, or give relevant feedback on the returned results. For example, from the above websites, we can seldom find a music clip relevant to the target topic. In this case, users could provide the system a preferred music, which will be further used as incidental music to accompany with the storyboard presentation. Users could also give some feedbacks on the obtained documents. For example, if he gives a thumb-up to a document, the relevant information of the document needs to be presented in the final storyboard. On the other side, users could also thumb-down a document to remove the related information.

3. TOPIC REPRESENTATION

A semantic topic is usually a quite broad concept and it usually contains multiple events. For example, in the topic "*Harry Potter*", the publication of each book and the release of each movie could be considered as an event; while in the topic "*World Cup 2002*", each match could also be taken as an event. For each event, there are usually many documents reporting it. Therefore, in order to generate an informative and expressive storyboard to present the topic, it would be better to decompose the obtained information and cluster the documents into different events.

However, event definition is usually subjective, different individuals may have different opinions. It is also confusing in which scale an event should be defined. Also take "*World Cup*" as an example, in a larger scale, "*World Cup 2002*" and "*World Cup 2006*" could also be considered as a big event. Therefore, due to the above vagueness, in this paper, we do not strictly define

each event of the target topic. Following our previous works on news event detection [14], an event is assumed as some similar information describing similar persons, similar keywords, similar places, and similar time duration. Therefore, in our system, an event is represented by four primary elements: who (persons), when (time), where (locations) and what (keywords); and event clustering is to group the documents reporting similar primary elements. As for the scale of event, in the paper, it could be adaptively determined by the time range of the obtained documents or the required event number.

In this section, we present a novel clustering approach based on a generative model proposed in [14], instead of using traditional clustering methods such as *K*-means. After event clusters are obtained, the corresponding event summary is then extracted and other representative media is associated with each event.

3.1 Multiple Event Clustering

To group the documents into different events, essentially, we need to calculate $p(e_j \mid x_i)$, which represents the probability that a document x_i belongs to an event e_j. Here, as mentioned above, an event e_j (and thus the document x_i describing the event) is represented by four primary elements: who (persons), when (time), where (locations) and what (keywords). That is,

$$Event / Docment = \{persons, locations, keywords, time\}$$

Assuming that a document is always caused by an event [14] and the four primary elements are independent, to calculate the probability $p(e_j \mid x_i)$, in our approach, we first determine the likelihood that the document x_i is generated from event e_j, $p(x_i \mid e_j)$ which could be further represented by the following generative model,

$$p(x_i \mid e_j) = p(name_i \mid e_j)\, p(loc_i \mid e_j)\, p(key_i \mid e_j)\, p(time_i \mid e_j) \quad (1)$$

where $name_i$, loc_i, key_i, and $time_i$ are the feature vectors representing persons, locations, keywords and time in the document x_i, respectively. In our approach, the above entities are extracted by the BBN NLP tools [15]. The tool can extract seven types of entities, including *persons*, *organizations*, *locations*, *date*, *time*, *money* and *percent*. In our approach, the obtained organization entity is also considered as a person entity; and all the words except of persons, locations, and other stop-words are taken as keywords.

In more detail, $name_i$ (similarly, loc_i and key_i) is a vector $<c_{i1}, c_{i2}, \ldots, c_{iNp}>$, where c_{in} is the occurrence frequency of the $person_n$ appears in the document x_i, and $person_n$ is the nth person in the *person vocabulary*, which is composed of all the persons appeared in all the obtained documents (similarly, we can define *keyword vocabulary* and *location vocabulary*). Assuming N_p is the size of person vocabulary, $p(name_i|e_j)$ could be further expressed by

$$p(name_i \mid e_j) = \prod_{n=1}^{N_p} p(person_n \mid e_j)^{c_{in}} \quad (2)$$

Since the person, location and keyword are discrete variables represented by words, and the probability of the location and keyword can be also defined similarly as that of the person in (2), in the flowing sections, we will not discriminate them and uniformly represent the probability $p(person_n \mid e_j)$ (correspondingly, the $p(location_n \mid e_j)$ and $p(keyword_n \mid e_j)$) as $p(w_n \mid e_j)$, which denotes the probability that the word w_n appears in the event e_j

On the other hand, the time of an event usually lasts a continuous duration. It is also observed, especially in the news domain, that the documents about an event usually increases at the beginning stage of the event and then decreases at the end. Therefore, in our approach, a Gaussian model $N(u_j, \sigma_j)$ is utilized to roughly represent the probability $p(time_i \mid e_j)$, where u_j and σ_j is the mean and standard deviation, respectively.

To this end, in order to estimate the probability $p(e_j \mid x_i)$, we need to estimate the parameters $\Theta = \{p(w_n \mid e_j), u_j, \sigma_j, 1 \le j \le K\}$, assuming K is the number of events (the selection of K is discussed in section 3.2). In our approach, the *Maximum Likelihood* is used to estimate the model parameters, as,

$$\begin{aligned} \theta^* &= \arg\max_\theta \log(p(X \mid \theta)) = \arg\max_\theta \log(\prod_{i=1}^{M} p(x_i \mid \theta)) \\ &= \arg\max_\theta \sum_{i=1}^{M} \log(\sum_{j=1}^{K} p(e_j)\, p(x_i \mid e_j, \theta)) \end{aligned} \quad (3)$$

where X represents the corpus of the obtained documents; M and K are number of documents and events, respectively.

Since it is difficult to derive a close formula to estimate the parameters, in our approach, an *Expectation Maximization* (*EM*) algorithm is applied to maximize the likelihood, by running *E-step* and *M-step* iteratively. A brief summary of these two steps is listed as follows, and more details can be found in [14].

- In *E-step*, the posterior probability $p(e_j \mid x_i)$ is estimated as:

$$p(e_j \mid x_i)^{(t+1)} = \frac{p(x_i \mid e_j)^{(t)}\, p(e_j)^{(t)}}{p(x_i)} \quad (4)$$

  where the upper script *(t)* indicate the *t*th iteration.

- In *M-step*, the model parameters are updated, as,

$$p(w_n \mid e_j)^{(t+1)} = \frac{1 + \sum_{i=1}^{M} p(e_j \mid x_i)^{(t+1)} \cdot tf(i,n)}{N + \sum_{i=1}^{M} (p(e_j \mid x_i)^{(t+1)} \cdot \sum_{s=1}^{N} tf(i,s))} \quad (5)$$

$$u_j^{(t+1)} = \frac{\sum_{i=1}^{M} p(e_j \mid x_i)^{(t+1)} \cdot time_i}{\sum_{i=1}^{M} p(e_j \mid x_i)^{(t+1)}} \quad (6)$$

$$\sigma_j^{2(t+1)} = \frac{\sum_{i=1}^{M} p(e_j \mid x_i)^{(t+1)} \cdot (time_i - u_j^{(t+1)})^2}{\sum_{i=1}^{M} p(e_j \mid x_i)^{(t+1)}} \quad (7)$$

  where $tf(i,n)$ is the term frequency of the word w_n in the document x_i and N is the corresponding vocabulary size. It is noted that, in (5), the *Laplace smoothing* [16] is applied to prevent zero probability for the infrequently occurring word.

  At last, the prior of each event is updated as:

$$p(e_j)^{(t+1)} = \frac{\sum_{i=1}^{M} p(e_j \mid x_i)^{(t+1)}}{M} \quad (8)$$

The algorithm can increase the *log-likelihood* consistently with the iterations; and then converge to a local maximum. Once the parameters are estimated, we can simply assign each document to an event, as following

$$y_i = \arg\max_j (p(e_j \mid x_i)) \quad (9)$$

where y_i is the event label of the document x_i.

The advantage of this generative approach is that it not only considers the temporal continuity of an event, it also can deal with the issue that some events overlap in some time durations. In this case, the Gaussian model of the event time can also be overlapped through this data-driven parameter estimation. From this view, the event clustering is also like a Gaussian mixture model (GMM) estimation in the timeline.

3.2 Determining the Number of Events

In the above approach to event clustering, the event number K is assumed known (as shown in (3)-(8)). However, the event number is usually very difficult to be determined *a priori*. In our approach, an intuitive way is adopted to roughly estimate the event number based on the document distribution along with the timeline.

As mentioned above, it is assumed that each document is caused by an event, and the document number of an event changes with the development of the event. According to this property, each peak (or the corresponding contour) of the document distribution curve might indicate one event [14], as the Fig. 2 shows. Thus, we can roughly estimate the event number by simply counting the peak number. However, the curve is quite noisy and there inevitably exist some noisy peaks in the curve. In order to avoid the noisy peaks, in our approach, only the salient peaks are assumed to be relevant to the event number.

To detect the salient peaks, we first smooth the document curve with a half-Hamming (raised-cosine) window, and then remove the very small peaks with a threshold. Fig.2 illustrates a smoothed document distribution with the corresponding threshold, collected on the topic "*US Election*" in four months. In experiments, the threshold is adaptively set as μ_d-σ_d/2, where μ_d and σ_d are the mean and standard deviation of the curve, respectively.

After the smoothing and tiny peaks removal, we further detect the valleys between every two contingent peaks. Thus, the range of an event (which is correlated to the corresponding peak) can be considered as the envelope in the two valleys. As shown in Fig2, the duration denoted by L_i+R_i is a rough range of the event correlated to the peak P_i. Assuming an important event usually has more documents and has effects in a longer duration, the *saliency* of each peak is defined as,

$$S_i = (\frac{P_i}{P_{avr}})(\frac{L_i + R_i}{D_{avr}}) \tag{10}$$

where P_i is the ith peak, L_i and R_i is the duration from the ith peak to the previous and next valley; P_{avr} is the average peak value and D_{avr} is average duration between two valleys in the curve. S_i is the saliency value of the peak P_i. It could also be considered as the normalized area under peak P_i, and thus, it roughly represents the document number of the corresponding event.

In our approach, the top K salient peaks are selected to determine the event number:

$$K = \arg\max_k \{\sum_{i=1}^{k} S_i^{'} / \sum_{i=1}^{N} S_i^{'} \le \eta\} \tag{11}$$

where $S_i^{'}$ is the sorted saliency value from large to small, N is total number of detected peaks and η is a threshold. In our experiments, η is set as 0.9, which roughly means that at least 90% documents will be kept in the further initialization of event clustering. This selection scheme is designed to guarantee there is no important information is missed in presentation. After the event number and initial clusters (the most salient peaks with their corresponding range) are selected, the event parameters could be initialized and then updated iteratively.

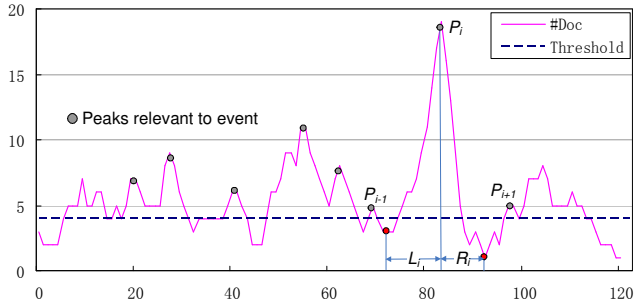

Fig.2 Peak saliency definition. It also illustrates the smoothed document distribution (document number per day) with the corresponding threshold for tiny peak removal. Each peak P_i is assumed to be correlated with each event.

It is noted that some technology such as Bayesian Information Criteria (BIC) or minimum description length (MDL) [17] could be used to estimate the optimal event number, by searching through a reasonable range of the event number to find the one which maximizes the likelihood in (3). However, these algorithms take long time, and it is usually not necessary to estimate the exact event number in our scenario of rich presentation. Actually, in our system, the most important point of event clustering is that the clustered documents 'really' represent the same event, rather than the event number, as observed in the experiments. Moreover, in the step of synchronization between the music and storyboard (in the section 4.2), the number of presented events may be further refined, based on the user's preference, in order to match the presentation duration with the music duration.

3.3 Event Description

After obtaining the events and the corresponding documents, we not only need a concise event summary, but also need to extract some representative media to describe each event.

3.3.1 Event Summary

A simple way to summarize an event is to choose some representative words on the persons, locations and keywords of the event. For example, for the event e_j, the 'leading actor' could be chosen as the person with the maximum $p(person_n \mid e_j)$, while the major location could be selected based on $p(location_n \mid e_j)$. However, such brief description might have a bad readability. Therefore, in order to increase the readability of the summary, in our system, we also provide an alterative way. That is, we choose a candidate document to represent an event. For example, the document with the highest $p(x_i|e_j)$ is a good candidate representative of the event e_j. However, a document might be too long to be shown on the storyboard. Therefore, in our system, only the "title-brow" (the text between the news title and news body) of the document, which usually exists and is usually a good overview (summary) of the document based on our observation (especially true in our case of news document), is selected to describe the event.

Fig. 3 The event template of the *Storyboard*, which illustrates (I) the representative media, (II)geographic information, (III) event summary, and (IV) a film strip giving an overview of the events in the temporal order.

3.3.2 Extracting Representative Media

In the obtained documents describing an event, there are usually many illustrational images, with possible flashes and video clips. These media information is also a good representative of the corresponding event. However, since the obtained documents are directly crawled from the news websites, they usually contain many noisy multimedia resources, such as the advertisements. Moreover, there also possible exist some duplicate images in different documents describing the same event. Therefore, to extract the representative media from the documents, we need to remove noisy media and possible duplicate images. Before this, we also performed a pre-filtering to remove all the images smaller than 50 pixels in height or width.

- *Noisy Media Detection*. In our approach, a simple but efficient rule is used to remove the noisy media resources. We find almost all advertisements are provided by other agencies rather than these news websites themselves. That is, the hosts of advertisement resources are from different websites. Thus, in our approach, we extract the host names from the URLs of all multimedia resources, and remove those resources with different host name.

- *Duplicate Detection*. A number of image signature schemes can be adopted here to accomplish duplicate detection. In our implementation, each image is converted into grayscale, and down-sampled to 8×8. That is, a 64-byte signature for each image is obtained. Then the Euclidean distance of the 64-byte signature are taken as the dissimilarity measure. Images have sufficiently small distance are considered as duplicates.

Once removing the noisy resources and duplicate images, we simply select the 1-4 large images from the top representative documents (with the top largest $p(x_i|e_j)$), and take them as representative media of the corresponding event. The exact number of the selected images is dependent on the document number (i.e., the importance) of the event and the total image number the event has. It is noted that, in our current system, we only associates images with each event. However, other media like video and flashes can be chosen in a similar way.

4. RICH PRESENTATION GENERATION

In the proposed system, the above obtained information, including event summary and representative media, are fused to generate a concise and informative storyboard, in order to richly present the target topic. In this section, we will first describe the storyboard generation for the target topic, by presenting each event with the multimodal information. Then, we present the approach to synchronizing the storyboard with an incidental music.

4.1 Storyboard Generation

In our approach, a storyboard of a target topic is generated by presenting each event of the topic slide by slide. To describe an event, we have obtained the corresponding information including the person, time, location, event summary and other relevant images. Therefore, to informatively present each event, we need first to design an event template (i.e., an interface) to integrate all the information.

Fig. 3 illustrates the event template used in our proposed system, with an example event in the topic '*US Election*". First, the template presents the representative images in the largest area (part I), since the pictures are more vivid than the words. As for each representative picture, the title and date of the document from which it is extracted is also illustrated. In the Fig.3, there are 4 pictures extracted from 3 documents. Then, the corresponding event summaries of these three documents are presented (part III), where each paragraph refers to the summary of one document. If a user is interested in one document, he can click on the corresponding title to read more details. Moreover, the geographic information of the event is shown with a map in the top-left corner (part II), to give users a view of the event location. The map is obtained from "MapPoint Location" service [18], which can

return a corresponding map based on user's location query. However, the mapping is usually difficult, especially when the event location is confusing so that the representative location is not accurately detected. For example, the event shown in the Fig 1 is mapped to Washington D.C. rather than New York where the republic convention is held, since Washington is the most frequently mentioned places in the documents. Finally, a film strip (part IV) is also presented, arranging each event in the temporal order, where each event is simply represented by a cluster of images, with the current event highlighted. It enables users to have a quick overview of the past and the future in the event sequence.

By connecting various events slide by slide, we could get an informative storyboard regarding the target topic. In order to catch the development process of a topic, the events are ordered by their timestamps in the generated storyboard.

4.2 Synchronizing with Music

To make the storyboard more expressive and attractive, and to provide a more relaxing way to read information, in the proposed system, we will accompany the storyboard with an incidental music and align the transitions between event slides with the music beats, following the idea in music video generation [19][20]. Sometimes, music could also provide extra information about the target topic. For example, when the target topic is a movie, the corresponding theme song could be chosen for the rich presentation. In this sub-section, we will present our approach to music analysis and synchronization with the storyboard.

4.2.1 Music Rhythm Analysis

In the proposed system, we detect the onset sequences instead of the exact beat series to represent music rhythm. This is because the beat information is sometimes not obvious, especially in light music which is usually selected as incidental music. The strongest onset in a time window could be assumed as a "beat". This is reasonable since there are some beat positions in a time window (for example, 5 seconds); thus, the most possible position of a beat is the position of the strongest onset.

The process of onset estimation is illustrated in Fig. 4. After FFT is performed on each frame of 16ms-length, an octave-scale filter-bank is used to divide the frequency domain into six sub-bands, including $[0, \omega_0/2^6)$, $[\omega_0/2^6, \omega_0/2^5)$, …, $[\omega_0/2^2, \omega_0/2]$, where ω_0 refers to the sampling rate.

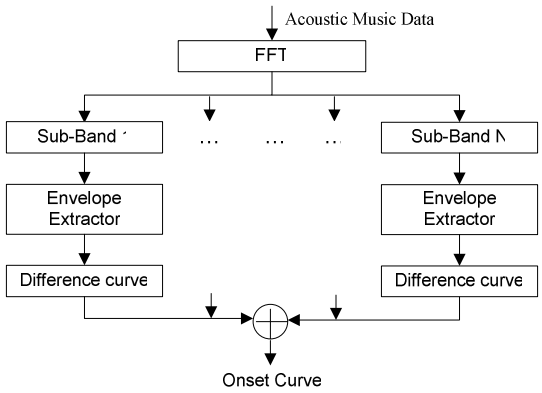

Fig. 4 The process of onset sequence estimation

After the amplitude envelope of each sub-band is extracted by using a half-Hamming window, a Canny operator is used for onset sequence detection by estimating its difference function,

$$D_i(n) = A_i(n) \otimes C(n) \tag{12}$$

where $D_i(n)$ is the difference function in the ith sub-band, $A_i(n)$ is the amplitude envelope of the ith sub-band, and $C(n)$ is the Canny operator with a Gaussian kernel,

$$C(n) = \frac{i}{\sigma^2} e^{-i^2/2\sigma^2} \qquad n \in [-L_c, L_c] \tag{13}$$

where L_c is the length of the Canny operator and σ is used to control the operator's shape, which are set as 12 and 4 in our implementation, respectively.

Finally, the sum of the difference curves of these six sub-bands is used to extract onset sequence. Each peak is considered as an onset, and the peak value is considered as the onset strength.

Based on the obtained onsets, an incidental music is further segmented into music sub-clips, where a strong onset is taken as the boundary of a music sub-clip. These music sub-clips are then used as the basic timeline for the synchronization in the next step. Thus, to satisfy the requirement that the event slide transitions of the storyboard should occur at the music beats, we just need to align the event slide boundaries and music sub-clip boundaries.

To give a more pleasant perception, the music sub-clip should not be too short or too long, also it had better not always keep the same length. In our implementation, the length of music sub-clips is randomly selected in a range of $[t_{min}, t_{max}]$ seconds. Thus, the music sub-clips can be extracted in the following way: given the previous boundary, the next boundary is selected as the strongest onset in the window which is $[t_{min}, t_{max}]$ seconds away from the previous boundary. In the proposed system, users can manually specify the range of the length of the music sub-clip. The default range in the system is set as [12, 18] seconds, in order to let users have enough time to read all the information on each event slide.

4.2.2 Alignment Scheme

To synchronize the transitions between different event slides and the beats of the incidental music, as mentioned above, we actually need to align the slide boundaries and music sub-clip boundaries. To satisfy this requirement, a straightforward way is to set the length of each event slide be equal to the corresponding length of the sub-music clip.

However, as Fig. 5 illustrates, the number of event slides is usually not equal to the number of music sub-clip. In this case, in our proposed system, we provide two schemes to solve this problem.

1) *Music Sub-clip Based.* In this scheme, only the top N important events of the target topic are adaptively chosen and used in the rich presentation, where N is supposed as the number of music sub-clip in the corresponding incidental music, as the Fig.5 shows. Although a formal definition of event importance is usually hard and subjective, in our approach, the importance score of an event is simply measured by the number of documents reporting it, assuming that the more important the event, the more the corresponding documents. The assumption is quite similar as that in the definition of (10).

2) *Specified Event Number Based*. In this scheme, users can specify the number of the event he wants to learn. For example, a user could choose to show the top 30 important events or all the events. Thus, to accommodate all the events in the music duration, we will repeat the incidental music if it is needed and then fade out the music at the end.

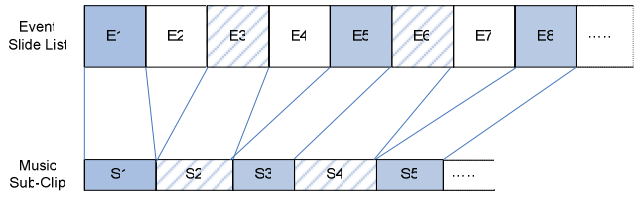

Fig. 5 Music and storyboard synchronization: a music sub-slip based scheme, that is, only the top important events are presented to match the number of music sub-clips.

4.2.3 Rendering

After the alignment between storyboard and incidental music, in our system, fifteen common transition effects, such as cross-fade, wipe and dissolve, are also randomly selected to connect the event slides, producing a better rich presentation in final rendering.

5. EVALUATIONS

In this section, we evaluate the performance of the proposed approach to rich presentation and its key component, event clustering. In the experiments, we randomly select 8 topics of different types, including *Earthquake*, *Halloween*, *Air Disaster*, *US Election*, *Nobel Prize*, *Britney Spears*, *David Beckham*, and *Harry Potter*, from some hot news topics in the end of 2004 and beginning of 2005. Once the topic is selected, the topic name is used as a query and the relevant documents are collected from CNN, MSNBC and BBC. More details about the selected topics and the corresponding documents are shown in the Table 1, which lists the topic name, the time range of the collected documents, and the number of documents and its corresponding events.

Table 1. A list of testing topics in the rich presentation evaluations

No.	Topic	Time	#doc	#event
1	Earthquake	1995-2004	976	17
2	Halloween	1995-2004	762	9
3	Air Disaster	1995-2004	210	13
4	US Election	1995-2004	2486	—
5	Britney Spears	2000-2004	1311	—
6	Nobel Prize	1995-2004	186	—
7	David Beckham	1995-2004	877	—
8	Harry Potter	2000-2004	841	—
Total	——	——	7649	—

It is noted that, in the table, only 3 topics have labeled events, while another 5 topics have not. This is because that, the labeling work of a topic is very subjective and usually hard for individuals to manually decide the event number of a given topic. Therefore, we only label the topics which are easily to be annotated based on the criterion in Topic Detection and Tracking (TDT) project [21]. For example, *Halloween* is a topic which is reported once a year, thus, each year's documents can be regarded as an event; as for *Earthquake* and *Air Disaster*, their events lists could be found from corresponding official websites. In the annotation, we remove the events which do not have or have few (less than 4) relevant documents, and also remove the documents not belonging to any events.

After parsing the obtained documents, for each topic, we usually can obtain 3.8 images per document in average. With further duplicate detection, only 1.6 images per document are remained. Moreover, from each document, we could also obtain about 3.0 unique location entities and 2.8 unique name entities. Other words except of these entities are taken as keywords. Fig.6 shows a real representation of an example document with extracted entities in the XML format, from which the event clustering is performed.

```
.....
<URL>http://news.bbc.co.uk/1/hi/world/americas/4071845.stm </URL>
<Abstract>The US battleground state of Ohio has certified the victory
of President George W Bush's in last month's poll. </Abstract>
<Date> 2004/12/6 </Date>
<NLPRESULT>
  <LOCATION>
    <entity> Ohio </entity>  <freq>4</freq>
    <entity> US </entity>  <freq> 2 </freq>
  </LOCATION>
  <PERSON>
    <entity> Bush </entity>  <freq> 3 </freq>
    <entity>David Cobb</entity>  <freq>1</freq>
    …
  </PERSON>
  …
  <DATE>
    <entity> 6 December, 200</entity>  <freq> 1 </freq>
    <entity> Friday </entity>  <freq> 2 </freq>
    …
  </DATE>
  <KEYWORDS>
    …
    <entity> recount </entity>  <freq>7</freq>
    <entity> elect </entity>  <freq>3</freq>
    <entity> America </entity>  <freq>3</freq>
    <entity> poll </entity>  <freq>3</freq>
    …
  </KEYWORDS>
</NLPRESULT>
```

Fig. 6. XML representation of a document on "*US Election*" with extracted entities

5.1 Event Clustering

As mentioned above, the evaluation of the approach to event clustering is evaluated on three topics, including *Earthquake*, *Halloween*, and *Air Disaster,* for which the corresponding event numbers are determined and the documents are labeled using a similar method in the TDT project. However, in the proposed appraoch, we actually do not estimate the optimal event number, but use a much larger one. Therefore, in order to better evaluate the performance of the event clustering algorithm and compare with its counterpart, we use the event number in the ground truth to initialize the cluster number in the proposed clustering algorithm.

In the experiments, *K-means*, which is another frequently used clustering algorithm (as well in TDT [22]), is adopted to compare with the proposed approach. The comparison results of two clustering approaches are illustrated in Table 2, with precision and recall for each topic.

Table 2. The performance comparison between our approach and K-means on the event clustering

	Precision		Recall	
	K-means	Ours	*K-means*	Ours
Earthquake	0.74	0.87	0.63	0.74
Halloween	0.88	0.93	0.72	0.81
Air Disaster	0.57	0.68	0.55	0.61
Average	***0.73***	***0.83***	***0.63***	***0.72***

From Table 2, it can be seen that the results of our approach are significantly better than those of *K-means,* both on precision and recall. On the three testing topics, the average precision of our approach is up to 0.83 and the average recall achieves 0.72, which is 10% and 9% higher than those of K-means, respectively. By tracing the process of *K-means*, we find that *K-means* usually assigns documents far away from each other on the timeline into the same cluster, since the time information affects little in *K-means*. It also indicates the advantages of our approach with time modeling.

The algorithms also show different performance on different kind topics. As for the "Air disaster", its performance is not as good as that of the other two, since the features (words and time) of its events are more complicated and intertwined in the feature space.

As for the topics (4-8 in Table I) which could not have an objective evaluation, the clustering performance on these topics could be indirectly reflected by the subjective evaluation of the rich presentation presented in section 5.2. This is because users will be more satisfied when the grouped documents shown in each event slide really belong to the same event; while users are not satisfied if the documents from different events are mixed in one event slide.

5.2 Rich Presentation

It is usually difficult to find a quantitative measure for rich presentation, since the assessment of the goodness of rich presentation is a strong subjective task. In this paper, we carry out a preliminary user study to evaluate the performance of the proposed rich presentation schemes.

To indicate the performance of rich presentation, we design two measures in the experiments, including '*informativeness'* and '*enjoyablity'*, following the criteria used in the work [7]. Here, the *informativeness* measures whether the subjects satisfy with the information obtained from the rich presentation; while *enjoyablity* indicates if users feel comfortable and enjoyable when they are reading the rich presentation. In evaluating the *informativeness*, we also provide the documents from which the rich presentation is generated. They are used as baseline, based on which the subjects can more easily evaluate if the important overview information contained in the documents is conveyed by the rich presentation. Moreover, in order to reveal the subjects' opinion on the design of the storyboard template, like the one shown in Fig 3, we also ask the subjects to evaluate the '*interface design'*.

In the user study, 10 volunteered subjects including 8 males and 2 females are invited. The subjects are around 20-35 years old, have much experience on computer manipulation, and usually read news on web in their leisure time. We ask them to give a subjective score between 1 and 5 for each measure of the rich presentation of each testing topic (an exception is '*interface design'*, which is the same for each rich presentation). Here, the score '1' to '5' stands for unsatisfied (1), somewhat unsatisfied (2), acceptable (3), satisfied (4) and very satisfied (5), respectively.

In experiments, we first check with the '*interface design'* measure. We find 7 out of 10 subjects satisfy with the event template design and the left three also think it is acceptable. The average score is up to 3.9. An interesting observation is that, some subjects like the template design very much at the first glance, but they feel a little boring after they finish all the user study since every slide in the rich presentation of each topic has the same appearance. It hints us that we had better design different templates for different topics to make the rich presentation more attractive.

As for the other two measures, we average the score across all the subjects to represent the performance for each topic, and list the detailed results in Table 3. It can be seen that the average score of both *enjoyablity* and *informativeness* achieves 3.7, which indicates that most subjects satisfy the provided overview information of the target topic, and they enjoy themselves when reading these rich presentations.

Table 3. The evaluation results of rich presentation on each topic

No.	Topic	Informative	Enjoyable
1	Earthquake	4.3	3.2
2	Halloween	3.6	4.0
3	Air Disaster	4.0	3.4
4	US Election	4.1	4.0
5	Britney Spears	3.6	4.1
6	Nobel Prize	3.3	3.4
7	David Beckham	3.4	4.0
8	Harry Potter	3.3	3.4
Average		***3.7***	***3.7***

In the experiments, we find *informativeness* is highly depended on the correlation between the presented documents and the target topic. If the presented information is consistent with the topic, subjects usually give a high score for *informativeness*, such as those on *Earthquake* and *US Election*; otherwise, they will give a low score, like those on *David Beckham* and *Nobel Prize*. It indicates that it is quite important to provide users clean information of the target topic with less noise. However, in current system, the documents are crawled from web and inevitably contain many noises. It affects much on the performance of *informativeness* in the current system. We need to consider how to prone the information of the target topic in the future works.

We also find that the *enjoyablity* score is usually related with *informativeness*. If the subjects do not get enough information from the rich presentation, they will be not enjoyable as well, such as the topics of *Nobel Prize* and *Harry Potter*. *Enjoyablity* is also topic-related, the subjects usually feel unconformable when they are facing with miserable topics, such as *Earthquake* and *Air*

Disaster, although their *informativeness* is quite high. On the contrary, users give a high score for *enjoyablity* on the interesting topics, such as *Britney Spears* and *David Beckham*, although their informative score is not high. This is because that there are usually many funny and interesting pictures in the presentation of these topics. Another finding is that users usually fell unenjoyable if the images and summaries in one event slide are not consistent with each other. From this view, the high *enjoyablity* score in our experiments also indicates that our event clustering algorithm works promisingly

6. CONCLUSIONS

To facilitate users to quickly grasp and go through the content of a semantic topic, in this paper, we have proposed a novel approach to rich presentation to generate a concise and informative storyboard for the target topic, with many relevant multimodal information including image, text, audio and video. In this approach, the related multimodal information of a given topic is first extracted from news databases. Then, the events are clustered, and the corresponding information, such as representative images, geographic information, and event summary, is obtained. The information is composed into an attractive storyboard which is finally synchronized with incidental music. A user study indicates that the presented system works well on our testing examples.

There is still some room for improving the proposed approach. First, the proposed approach could be extended to other multimedia databases or more general websites. For example, some standard multimedia database like NIST TRECVID could provide a nice platform for the implementation and evaluation of event detection and rich presentation. Second, to integrate more relevant multimedia information (such as video clips and flashes) and more accurate information regarding the target topic is highly expected by users. Thus, more advanced information retrieval/ extraction techniques and other multimedia analysis techniques are needed to be exploited and integrated, such as relevance ranking, mapping schemes, important or representative video clips detection and video clip summarization. We also need to design a much natural way to incorporate video clips in the event template. Third, we also consider designing various storyboard templates for different kind of topics. For example, each topic may be belonging to different clusters such as politics, sports and entertainments, each of which can have a representative template. Forth, appropriate user interaction will be added to further make the storyboard more interactive and easy to control. Finally, a thorough evaluation will be implemented to evaluate the effect of each component in the framework and storyboard template.

7. REFERENCES

[1] A. Vailaya, M.A.T. Figueiredo, A. K. Jain, and H.-J. Zhang. "Image classification for content-based indexing". *IEEE Transactions on Image Processing*, Vol.10, Iss.1, 2001

[2] F. J., M.-J. Li, H.-J. Zhang, and B. Zhang. "An effective region-based image retrieval framework". *Proc. ACM Multimedia'02*, pp. 456-465, 2002

[3] J. Platt "AutoAlbum: Clustering Digital Photographs using Probabilistic Model Merging" *Proc. IEEE Workshop on Content-Based Access of Image and Video Libraries*, pp. 96–100, 2000.

[4] A. Hanjalic, R. L. Lagendijk, J. Biemond, "Automated high-level movie segmentation for advanced video-retrieval systems", *IEEE Trans on Circuits and Systems For Video Technology*, Vol. 9, No. 4, pp. 580-588, 1999.

[5] J. Assfalg and et al, "Semantic annotation of soccer videos: automatic highlights identification," *CVIU'03*, vol. 92, pp. 285-305, 2003.

[6] A. Ekin, A. M. Tekalp, and R. Mehrotra, "Automatic soccer video analysis and summarization," *IEEE Trans. on Image Processing*, 12(7), pp. 796-807, 2003.

[7] Y. -F. Ma, L. Lu, H. -J. Zhang, and M.-J Li. "A User Attention Model for Video Summarization". *ACM Multimeida'02*, pp. 533-542, 2002.

[8] L. Xie, P. Xu, S.F. Chang, A. Divakaran, and H. Sun, "Structure analysis of soccer video with domain knowledge and hidden markov models," *Pattern Recognition Letters*, vol. 25(7), pp. 767-775, 2004.

[9] L. Lu, H. Jiang, H. J. Zhang, "A Robust Audio Classification and Segmentation Method," *Proc. ACM Multimedia'01*, pp. 203-211, 2001

[10] R. Cai, L. Lu, H.-J. Zhang, and L.-H. Cai, "Highlight Sound Effects Detection in Audio Stream," *Proc. ICME'03* Vol.3, pp.37-40, 2003.

[11] Y. Rui, A. Gupta, and A. Acero, "Automatically Extracting Highlights for TV Baseball Programs", *Proc. ACM Multi-media'00*, pp.105-115, 2000.

[12] C. Snoek, and M. Worring. "Multimodal Video Indexing: A Review of the State-of-the-art". *Multimedia Tools and Applications*, Vol. 25, No. 1 pp. 5 – 35, 2005

[13] E.M. Voorhees, "Query expansion using lexical-semantic relations" Proc. *ACM SIGIR Conference on Research and Development in Information Retrieval ,* pp 61 - 69, 1994

[14] Z.-W. Li, M.-J. Li, and W.-Y. Ma. "A Probabilistic Model for Retrospective News Event Detection", *Proc. SIGIR Conference on Research and Development in Information Retrieval*, 2005

[15] D. M. Bikel, R. L. Schwartz, and R. M. Weischedel. "An Algorithm That Learns What's in a Name". *Machine Learning*, 34(1-3), 1999

[16] K. Nigam, A. McCallum, S. Thrun, and T. Mitchell. "Text Classification from Labeled and Unlabeled Documents using EM". *Machine Learning*, 39(2-3), 2000

[17] T. Hastie, R. Tibshirani, and J. Friedman. "The Elements of Statistical Learning: Data Mining, Inference and Prediction". *Springer-Verlag*, 2001

[18] MapPoint Web Service http://www.microsoft.com/mappoint/ products/ webservice/default.mspx

[19] X.-S. Hua, L. Lu, H.-J. Zhang. "Automated Home Video Editing", Proc. *ACM Multimedia'03*, pp. 490-497, 2003

[20] J. Foote, M. Cooper, and A. Girgensohn. "Creating Music Videos Using Automatic Media Analysis". *ACM Multimedia'02*, pp.553-560, 2002.

[21] Topic Detection and Tracking (TDT) Project: http://www. nist.gov/speech/tests/tdt/

[22] J. Allan, R. Papka, and V. Lavrenko. "On-line New Event Detection and Tracking". Proc. SIGIR Conference on Research and Development in Information Retrieval 98, pp.37-45, 1998