Event Driven Summarization for Web Videos

Richang Hong
School of Computing
National University of Singapore
hongrc@comp.nus.edu.sg

Jinhui Tang
School of Computing
National University of Singapore
tangjh@comp.nus.edu.sg

Hung-Khoon Tan
Department of Computer Science
City University of HongKong
hktan@cs.cityu.edu.hk

Shuicheng Yan
Department of ECE
National University of Singapore
eleyans@nus.edu.sg

Chong-Wah Ngo
Department of Computer Science
City University of HongKong
cwngo@cs.cityu.edu.hk

Tat-Seng Chua
School of Computing
National University of Singapore
chuats@comp.nus.edu.sg

ABSTRACT

The explosive growth of web videos brings out the challenge of how to efficiently browse hundreds or even thousands of videos at a glance. Given an event-driven query, social media web sites can easily return a ranked list of large but diverse and somewhat noisy videos. Users often need to painstakingly explore the retrieved list for an overview of the event. This paper presents a novel solution by mining and threading "key" shots, which can provide an overview of main contents of videos at a glance, by summarizing a large set of diverse videos. The proposed framework comprises three stages for multi-video summarization. Firstly, given an event query, a ranked list of web videos together with their associated tags are retrieved. Key shots are then established by near-duplicate keyframe detection, ranked according to informativeness and threaded in a chronological order. Finally, summarization is formulated as an optimization procedure which trades off between relevance of key shots and user-defined skimming ratio. The framework provides the summary with the way of dynamic video skimming. We conduct user studies on twelve event queries for over hundred hours of videos crawled from YouTube. The evaluation demonstrates the feasibility and effectiveness of the proposed solution.

Categories and Subject Descriptors

H.3.1 [**Content Analysis and Indexing**]: Abstracting methods; H.3.5 [**Online Information Services**]: [Web-based Services]

General Terms

Algorithm, Design, Experimentation.

Keywords

Event Evolution, Web Video Summarization.

WSM'09, October 23, 2009, Beijing, China.

1. INTRODUCTION

The modern Web 2.0 activities and contents have pervaded the internet. Their increasing popularity originates from their ease of operation and support for interactive services such as tagging, comments and ratings. One example is YouTube, which is one of the primary video sharing site. It offers users a new channel to deliver and share their videos. Studies have shown that YouTube serves 100 million distinct videos and 65,000 uploads daily [3], and traffic of this site accounts for over 20% of the web in total and 10% of the whole internet, covering 60% of the videos watched on-line [7].

The growing number of videos has motivated a real necessity to provide effective tools to support retrieval and browsing. However, given an event type query, the retrieved videos are diverse and somewhat noisy. Summarization may be a feasible way to help user efficiently browse the retrieved videos. However, most existing video summarization algorithms are designed only to handle a single video [1]. For our scenario, we have to face a large number of videos, most of which are derivatives from the key sub-events [4]. Thus the existing summarization methods cannot be applied directly.

Recent studies [3][11] on video sharing sites have shown that there exists a significant amount of over 25% of duplicate videos detected in the search results. We categorize the content redundancy on web videos into two classes: *near duplicate* and *overlap*. The former indicates that most of the frames from the two videos are duplicates and the latter indicates that the video pair shares some near duplicate frames. In this study, we focus on the case of overlap and look at it from a different perspective. We demonstrate that such content overlap in web video sharing system may be exploited for automatic video summarization [6][13].

As we know, an event is composed of a connected series of sub-events with a common focus or purpose that happens in specific places during a given temporal period [5]. For a given event, the few scenes that convey the main messages, such as the principal sub-events or key shots, will be presented more than once in news reports. We take "September 11 attacks" as an example and divide it into many sub-events. It contains several principal sub-events, such as "the airplane was hijacked", "airplane crashed into the world trade tower", "the world trade tower caught fire and collapsed" *etc.* We define the shots displaying these

sub-events in video as *key shots*, which are believed to unfold the dominant messages of the event. We observe that these types of key shots appear in many web videos retrieved by the event type query. Based on this observation, we can identify such shots by first extracting the keyframes and then performing near duplicate keyframe (NDK) detection [8]. This is especially so for those events happened during a limited time span and in a specific location.

Motivated by the above observation and analysis, we propose to utilize the key shots to summarize web videos. We first adopt a near duplicate keyframe (NDK) detection method to identify the key shots among the web videos. We then perform key shot ranking and thread them according to the chronological order based on the original videos. Finally the resulting summary is obtained by optimally selecting the key shots for video skimming. We conduct experiments on a real-world web video dataset, consisting of 102 hours of videos crawled from YouTube and with 12 queries. The user evaluation demonstrates the viability of the proposed system. The contributions of this research are twofold:

1. We propose an event driven web video summarization system to help users browse the results of web video search with an overview. To the best of our knowledge, this is the first attempt to summarize multiple web videos relevant to news events.
2. We identify overlap in content in web videos through the use of NDK detection technique and effectively leveraging the overlap to support video analysis and mining of the semantic relationships from the social sharing network.

Throughout this paper, we use the terms shot and keyframe interchangeably to mean the same thing. The rest of this paper is organized as follows. Section 2 illustrates the system framework and describes techniques for key shot processing, which includes linking, ranking and threading. Section 3 presents the strategy for summarization. In Section 4, we conduct experiments and user based evaluation to demonstrate the feasibility and effectiveness of the system. Section 5 conclude the paper.

2. VISUAL PROCESSING OF KEY SHOTS

The goal of the proposed summarization system is to give users a quick overview for an event type query by leveraging on the content overlap of the web video search results. Figure 1 illustrates the framework of the proposed system, which consists of three main stages, namely, web videos acquisition, visual processing of key shots and summary generation. A shot is defined as a sequence of contiguous frames that are captured by a single continuous camera action. It has been widely employed as the basic unit for video analysis. In general, a keyframe is extracted to represent the main content of a shot [10]. Thus in this study, the shot processing, if not specifically mentioned, is performed on its corresponding keyframe for simplicity.

2.1 Key Shot Ranking

In key shot ranking stage, we first extract the keyframes from the original videos sequentially and for each keyframe we extract its local point features, e.g., scale-invariant feature transform [9] features, for matching. To reduce the computation cost, local point features are mapped to a fixed

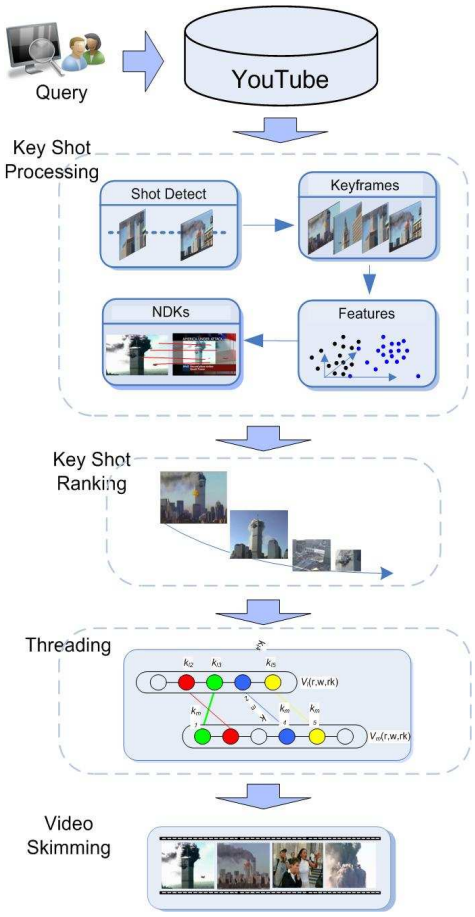

Figure 1: The flowchart of the proposed event driven web video summarization system. It comprises three main stages: video acquisition, visual processing of key shots and summary generation.

dimension and some keyframes are filtered by offline quantizing the keypoint descriptors. The keyframe pair with similarity value above a given threshold is retained as NDKs and their corresponding shots are defined as key shots. Given the key shot sequence, we would like to pick out those that are more informative to the event to form the resulting summary. Moreover, as NDK detection would inevitably bring false alarms, hence we should rank the key shots obtained by NDK detection [8]. Here, we denote the video corpus for a given query as $C = \{V_i,\ 1 \leq i \leq |C|\}$, where $|C|$ is the total number of videos in C. The set of shots for each V_i, which corresponds to NDKs, is $S_i = \{s_{im}, 1 \leq m \leq |S_i|\}$, where $|S_i|$ is the total number of shots in V_i. Note that if two shots are identified as near duplicate shots, both are defined as key shots. Thus the set of key shots can be segmented into several near-duplicate key shot groups $\{g_n, n = 1, 2, ...G\}$, where G is the total number of groups in total. We denote $|g_n|$ as the number of near-duplicate key shots in each group g_n.

According to the distribution analysis of users' ratings and interests (the options on YouTube site) against rank, we model the relevance score as a power law distribution: $rel(s_{im}) = ci^{-\gamma}$. Here, $rel(s_{im})$ denotes the relevance score of shot s_{im} in V_i. It should be emphasized that the distribu-

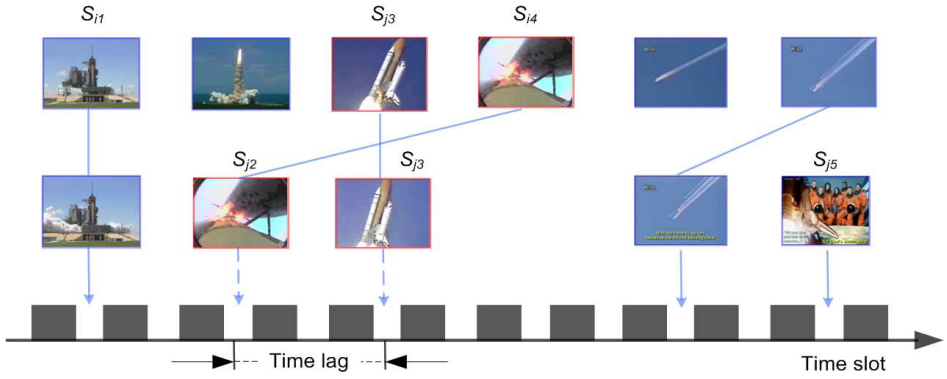

Figure 2: Key shot threading. The visual similarity links lie in the near-duplicate shots within different videos. Note that some links may wrap each other as the case of s_{i3} and s_{i4} linking to s_{j2} and s_{j3} respectively.

tion is somewhat related to the search query but not strictly agree with the power law distribution. In spite of that, the modeled distribution is capable of reflecting the tendency of videos' relevance scores [2]. We now define a metric to measure how informative the key shot is by linearly combining the importance and relevances of key shots. As discussed, all the key shots in g_n will be assigned to the same value of $|g_n|$. We then define the informative score of a key shot s_{im} as follows:

$$\begin{aligned} info(s_{im}) &= \log|g_n| + rel(s_{im}) \\ &= \log|g_n| + ci^{-\gamma} \end{aligned} \tag{1}$$

where $s_{im} \in g_n$, $1 \le n \le G$, and $\log|g_n|$ is normalized as:

$$\log|g_n| = \frac{\log|g_n| - \min(\log|g_n|)}{\max(\log|g_n|) - \min(\log|g_n|)} \tag{2}$$

Each key shot can be ranked according to the informative score in Eqn. (1). After that, we pick out the key shot with the highest informative score in each group g_n to form the list of unique key shot$\{s_l, 1 \le l \le G\}$. We can utilize the ranked s_l for static summarization directly. However, considering the evolution traits that lie in news event, it would be more desirable to embody the time constraint, even in static summarization of storyboard.

2.2 Key Shot Threading

Here, given the chronological order of key shots in the original videos, threading can be illustrated in Fig. 2, where the horizontal axis indicates the time slot. Note that the key shots located in the same video have no visual similarity links to each other. The ideal case is when all the identified key shots are ordered in a consistent way by the links. However, some links may wrap around each other as s_{i3} and s_{i4} are linking to s_{j2} and s_{j3} respectively in Fig. 2, where the superscript denote the shots lie in v_i and v_j.

In this study, we propose to mine the sequence of key shots by minimizing the time lag between the near duplicate key shot pairs in different videos. We first assign an initial value λ_{im} to each key shot s_{im}. λ_{im} is normalized so that the sum equals to 1 and $\lambda_{im} > \lambda_{in}$, $1 \le m, n \le |S_i|, m > n$. Considering that some key shots may appear at random location in many videos, we relax the second constraint to $\lambda_{im} > \lambda_{in}$, when $|m-n| < T$. Here, T is a threshold to control the time interval in which the established shots meet the chronological order requirements. Based on that, we can perform the key shots threading by solving the following minimization problem:

$$\begin{aligned} &\min \sum_{l=1}^{G} \sum_{g_l} \left\|\lambda_{im} - \lambda_{jn}\right\|^2 \quad \lambda_{im} \in g_l,\ \lambda_{jn} \in g_l \\ &s.t. \sum_i \sum_m \lambda_{im} = 1; \\ &\lambda_{im} > \lambda_{in},\ \lambda_{jm} > \lambda_{jn}\ if\ |m-n| < T \end{aligned} \tag{3}$$

Eqn. (3) is a standard quadratic programming problem and can be solved directly by general quadratic programming method. We denote the solution sequence as $\{\lambda_l, 1 \le l \le G\}$. The sequence of key shot groups, namely the sequence of non-near-duplicate shots, can be chronological ordered by the minimization process.

3. SUMMARIZATION

In Section 2.2, the key shots are chronologically ordered by minimizing the time lag between each key-shot pair s_{im} and s_{jm}. After the process of minimization, the resulting sequence is $\{\lambda_l, 1 \le l \le G\}$. For video skimming, the chronological order should be of primary importance followed by informative scores. It is the tradeoff between the informative score and the time order. In addition, the summary has to meet user's requirements for duration. We denote the duration as T_s. The selection strategy can be viewed as maximizing the tradeoff between the sum of relevance and time interval.

$$\begin{aligned} D = \arg\max_D (&\sum_{l \in D} info(s_l) + \beta \tfrac{1}{|D|} \sum_{l,m \in D} \|\lambda_l - \lambda_m\|^2) \\ s.t.\ &\sum_{l \in D} length(s_l) < T \end{aligned} \tag{4}$$

where D denotes the resulting key shots list and β controls the tradeoff $|D|$ indicates the size of the resulting list and $length(\cdot)$ reflect the duration of s_l. In general, we empirically set $\beta = 0.8$ for video skimming. λ_l and s_l corresponds to the same key shot since near-duplicate key shots are with close λ.

We can solve the equation through a greedy algorithm, in which every step selects the key shot with local maximal informative scores while keeping the time interval maximal. Based on the above analysis, we conclude the detailed steps for generating the summary as follows:

1. We link key shots by NDKs detection and rank each key shot according to its informative score.

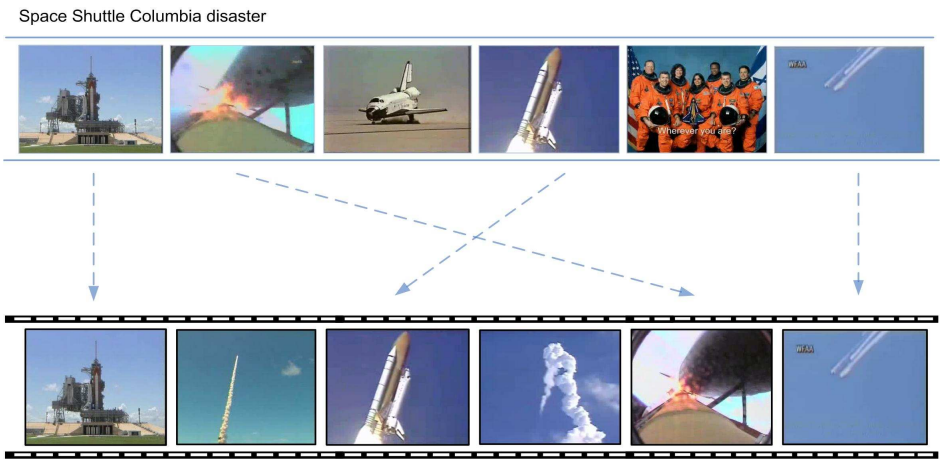

Figure 3: Video skimming for the event: "Space Shuttle Columbia Disaster". The top row corresponds to the key shots ranked with informative scores. We can see that the informativeness order of the key shots are rearranged after the tradeoff between informative scores and time constraints in optimization.

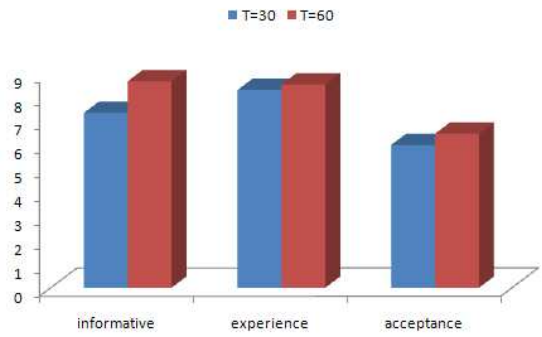

Figure 4: User based evaluation on the resulting video skimming.

2. We chronologically thread the key shots by optimizing the time lag between near duplicate key shot pair.

3. We optimally select the key shots for final summary according to Eqn. (4). Here β is set to control the tradeoff between informative scores and time constraints.

4. EVALUATION

We now present the results of evaluating our proposed system. We first describe the strategy for gathering the video collection from YouTube and elaborate on the characteristics of our dataset. We then analyze the performance of summarization on our dataset by user based evaluation.

4.1 Experimental Setup

In our scenario, we require access to the actual video content and its associated textual information, such as tags, titles and descriptions. Other contextual information such as comments and ratings are too noisy to be exploited in this study [12]. The requirement imposes a limit on the dataset size that can be stored in the corpus. Moreover, crawling is another activity that imposes high demand on the network. Here, we utilize 12 news event type queries as listed in Table 1 to search on YouTube and download the top videos in the retrieved list. The number of videos downloaded per query ranges from 49 to 120 because quite a number of videos are not accessible; where they may either be removed by the system or the owners themselves. The complete collection used for evaluation has a final size of 1024 videos with an average of 85 videos per query. The total duration is 102 hours with an average of 6.9 mins per video. The details of the data collection is provided in Table 1.

From Table 1, we can see that although YouTube has removed most of the exact duplicate videos, a few can still be found in the retrieved list. The third column of Table 1 shows the number of exact duplicate videos for each query. In this study, this type of videos are detected by NDK matching. If the keyframes in a video contain more than 90% of NDKs with another video, it is judged as a duplicate video and then removed. The number of keyframes ranges from 950 to 10428. It is not linearly correlated with the number or duration. Table 1 also reveals that there are large variation of ratio of key shots to the rest of shots across the range of news events. For some news events, which happened during a limited time span and in a specific location such as the "Handover 1997 HongKong" and "Obama inauguration", the ratio of key shots to other shots is as high as 10% since the shots displaying the dominant sub-events tend to be repeated frequently. While for general news events with large time span such as the "Russia Georgia war" , the ratio is about 3% since there are more sub-events and web videos may not display too many shots repeatedly. The key shot groups in Table 1 show the number of key shots with unique content.

4.2 Summarization Evaluation

In this section, we demonstrate the resulting summary for four event type queries and then evaluate the performance through user study. The parameters c and γ in key shot ranking are empirically set to 0.8 and 0.2 while λ_{im} in Eqn. (3) can be assigned based on the sequential number of keyframes directly. The time interval constraint T is set to 5. The video summaries in Fig. 3 are generated automatically on the query "Columbia Space Shuttle disaster". For

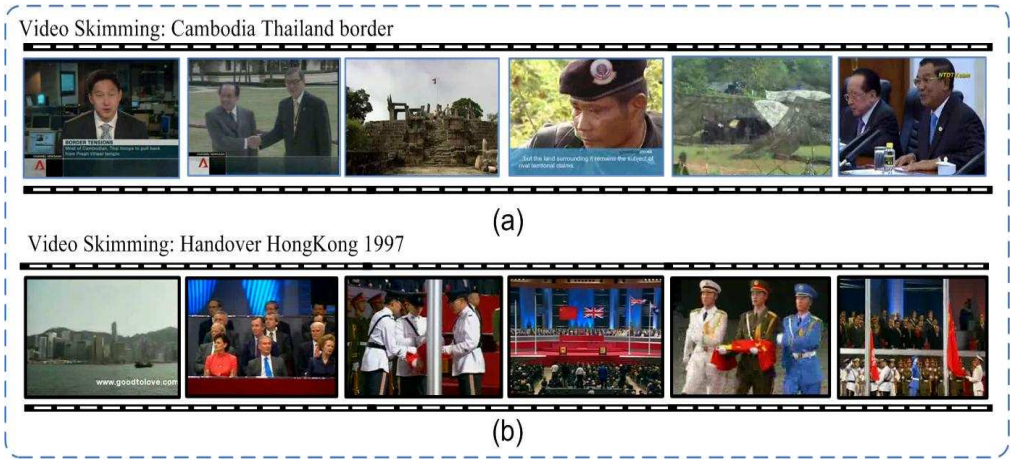

Figure 5: The dynamic summaries of video skimming. The key shots are ordered by optimizing the tradeoff between the informative scores and the time constraints. We can see the outlier (i.e., a shot for anchor person) exists in the "Cambodia Thailand border". Another example shows comparatively better performance for the "Handover 1997 HongKong".

comparison, the key shots with top six informative scores are presented above the video skimming. We can see that the key shots are re-arranged in chronological order based on the tradeoff between the informative scores and time constraints. The dash arrow line show the position of some key shots in the original informative ordered shot list. From Fig. 5, we can see that the sub-events are prepared to be launched in the first shot, and then lifted up, flew on the orbit and finally exploded when re-entered the atmosphere. The key shots with the third and the fifth maximal relevance scores are not used in the video skimming due to the time constraint (see Eqn. (4)). Here, for simplicity, we empirically set β in Eqn. (4) to 0.8 and show the six key shots in the resulting key shot list. In other words, the constraint condition is changed to $|D| = 6$.

Figure 5 shows another two examples for the summary of dynamic video skimming, i.e., Fig. 5(a) for "Cambodia Thailand border" and Fig. 5(b) for "Handover HongKong 1997" respectively. We can see that in Fig. 5(a), the first frame can be deemed as an outlier of an anchor person frame. It is probably introduced because the anchor person frame predominate the source videos of this query. However, it is ranked first due to the time constraint though it is only ranked fourth in informative scores. Another example is shown in Fig. 5(b), we can see that the key shots are organized in a chronological order and meanwhile each key shots represent the key sub-event which is often repeated in news videos. In general, this kind of summary is capable of giving an overview about the news event.

As it is difficult to objectively evaluate the proposed system, we evaluate the performance of summarization through user study. We ask eight evaluators to assess the performance of the dynamic summary. These evaluators are asked to give scores of between 1 to 10 based on their satisfaction, with higher score means better satisfaction. During evaluation, user can also set the duration of the summary. Shorter duration indicates that the summary would include only the most relevant shots that also meet the chronological order constraints. When user increases the duration, the summary will introduce more shots with decreasing relevance values. It should be emphasized that the video skimming produced by our system is not strictly constrained to the duration that the user set (see Eqn. (4)). We select the number of key shots to compose the summary. The sum of each key shot's time span optimally approaches the duration requirement.

We define three perspectives that evaluators should consider in their evaluation:

1. Informative: to what degree do you feel the summary retain the content coverage or capture the gist of the event?
2. Experience: do you think the summary is helpful for your understanding of the event?
3. Acceptance: if YouTube were to incorporate this function into their system, are you willing to use it for summary?

The average performance of eight evaluators' subjective tests are illustrated in Fig. 4. Here, the duration T_s is set to 30s. We can see that for most queries, users think that the summaries are informative and can help them to obtain an overview of the event. For further comparison of performance, we set the duration equals to 60s. We can see that the information conveyed by the summary increases rapidly while the other two metrics increase slowly. The evaluation conforms with our expectation that the metric of informativeness will approach the stable status shortly with the increase of duration. This is because further increase in duration will take in some shots with low relevance in summary.

5. CONCLUSION AND FUTURE WORK

In this paper, we proposed an event driven web video summarization system that can alleviate the need to painstakingly explore the retrieved web video list to obtain the gist of the event. We defined the concept of key shots related to the event and by observing the characteristics of social sharing system that many web videos are composed of the defined key shots. We utilized the content overlap in social sharing system to produce summary.

Table 1: Data Collection

No.	Qeury	Downloaded Videos			Key Shots		
		Number	Duration (hrs.)	Duplicate Videos	Key-frames	Key Shots /NDKs	Key Shot Groups
1	September 11 attacks	96	18.62	1	10428	535	189
2	Space Shuttle Columbia disaster	79	7.35	1	3647	535	168
3	2008 Mumbai attacks	71	5.38	2	3551	197	65
4	Tsunami Indonesia	90	6.61	2	5742	234	53
5	Handover 1997 Hong Kong	49	5.88	1	2010	209	32
6	the Berlin wall falls	96	9.50	2	5163	436	113
7	Columbine shootings	63	4.53	4	2372	201	43
8	Obama inauguration	120	13.60	9	4147	428	75
9	princess Diana funeral	99	12.54	1	6913	566	71
10	2008 Tibetan unrest	83	6.67	3	4815	307	75
11	Russia Georgia war	100	9.31	1	6944	174	56
12	FedEx Cargo Jet Crashing in Tokyo	78	2.66	1	950	105	35

Intuitively the scenes, which are presented as a set of key shots in this study, unfold the more informative messages of the event and many web videos are derived from that. Therefore we first identified key shots by NDK detection technique and performed key shot ranking and threading. Finally the dynamic summary is produced by selecting the corresponding key shots using a greed algorithm. The proposed framework can be extended to other applications such as multimedia QA [14]. Our work has some limitations: for example, it is not clear how to effectively utilize more information from social sharing system, such as the comments and ratings *etc.,* to improve the performance of summary. However, our work points to other applications that may benefit from leveraging on content redundancy. A direct way is to use such redundancy to propagate the tags between videos from social sharing system to improve the search. Furthermore, extension to other resources such as images in some datasets which originate from web resources [15] or large sharing sites, like Flickr and Picasaweb, *etc.,* deserves further exploration.

6. REFERENCES

[1] H. Benoit and M. Bernard. Automatic video summarization. *Chapter in "Interactive Video, Algorithms and Technologies", 27-41, 2006.*

[2] R. G. Capra, C. A. Lee, G. Marchionini, T. Russell, C. Shah, and F. Stutzman. Selection and context scoping for digital video collections: An investigation of youtube and blogs. In *Proceedings of the JCDL*, Pittsburgh, PA, USA, June 2008.

[3] M. Cha, H. Kwak, P. Rodriguez, YY. Ahn, and S. Moon. I tube, you tube, everybody tubes: analyzing the world's largest user generated content video system. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, San Diego, California, USA, September 2007.

[4] J. Chen, J. Yan, B. Zhang, Q. Yang, and Z. Chen. Diverse topic phrase extraction through latent semantic analysis. In *Proceedings of the 6th ICDM*, HongKong, China, December 2006.

[5] K. Y. Chen, L. Luesukprasert, and S. T. Chou. Hot topic extraction based on timeline analysis and multidimensional sentence modeling. *IEEE transactions on knowledge and data engineering*, 19:1016–1025, 2007.

[6] P. Duygulu, J.-Y. Pan, and D. A. Forsyth. Towards auto-documentary: Tracking the evolution of news stories. In *Proceedings of the 11th ACM Multimedia*, Berkeley, CA, USA, October 2003.

[7] P. Gill, M. Arlitt, Z. Li, and A. Mahanti. Youtube traffic characterization: a view from the edge. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, San Diego, California, USA, September 2007.

[8] H. K. Tan, C. W. Ngo, R. Hong and T. S. Chua. Scalable Detection of Partial Near-Duplicate Videos by Visual-Temporal Consistency. *proceedings of the ACM Multimedia 2009.*

[9] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer*, 60:91–110, 2004.

[10] Huamin Feng Tat-Seng Chua and Chandrashekhara A. An unified framework for shot boundary detection via active learning. In *ICASSP '2003*, 2003.

[11] X. Wu, A. G. Hauptmann, and C. W. Ngo. Practical elimination of near-duplicates from web video search. In *Proceedings of the 15th international ACM Multimedia*, Augsburg, Germany, October 2007.

[12] Meng Wang, Kuiyuan Yang, Xian-Sheng Hua and Hong-Jiang Zhang. "Visual Tag Dictionary: Interpreting Tags with Visual Words. *ACM Workshop on Web-Scale Multimedia Corpus, in association with ACM MM 2009*

[13] X. Wu, C. W. Ngo, , and Q. Li. Threading and autodocumenting news videos. *IEEE Signal Processing Magazine*, 23:59–68, 2006.

[14] T. S. Chua, R. Hong, G. Li and J. Tang. From Text Question-Answering to Multimedia QA. *ACM Workshop on Large-Scale Multimedia Retrieval and Mining, in association with ACm MM 2009.*

[15] T. S. Chua, J. Tang, R. Hong, H. Li, Z. Luo and Y. Zheng. NUS-WIDE: A Real-World Web Image Databased From National University of Singapore. *proceedings of the CIVR* . Greece. Jul. 8-10, 2009.