Beyond Search: Event Driven Summarization for Web Videos

RICHANG HONG, JINHUI TANG
National University of Singapore
HUNG-KHOON TAN, CHONG-WAH NGO
City University of HongKong
and
SHUICHENG YAN, TAT-SENG CHUA
National University of Singapore

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 usually return a large number of videos that are diverse and noisy in a ranking list. Exploring such results will be time-consuming and thus degrades user experience. This paper presents a novel scheme that is able to summarize the content of video search results by mining and threading "key" shots, such that users can get an overview of main content of these videos at a glance. The proposed framework mainly comprises four stages. First, given an event query, a set of web videos is collected associated with their ranking order and tags. Second, key-shots are established and ranked based on near-duplicate keyframe detection and they are threaded in a chronological order. Third, we analyze the tags associated with key-shots. Irrelevant tags are filtered out via a *representativeness* and *descriptiveness* analysis, whereas the remaining tags are propagated among key-shots by random walk. Finally, summarization is formulated as an optimization framework that compromises relevance of key-shots and user-defined skimming ratio. We provide two types of summarization: video skimming and visual-textual storyboard. We conduct user studies on twenty 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—*documentation*; H.3.5 [**Online Information Services**]: Web-based Services

General Terms: Algorithm, Design, Experimentation

Additional Key Words and Phrases: Event Evolution, Key-shot Threading, Key-shot Tagging, Web Video Summarization

Author's address: R. Hong, J. Tang (corresponding author), School of Computing, National University of Singapore. COM1, 13 Computing Drive, Singapore, 117417; email: {hongrc,tangjh}@comp.nus.edu.sg; H.-K. Tan, C. -W. Ngo, Department of Computer Science, City University of HongKong, 83 Tat Chee Avenue Kowloon Tong, Hong Kong; email: {hktan,cwngo}@cs.cityu.edu.hk; S. Yan, Department of Electric and Computer Engineering, National University of Singapore. 4 Engineering Drive 3, Singapore, 117576; email: eleyans@nus.edu.sg; T. -S. Chua, School of Computing, National University of Singapore, COM1, 13 Computing Drive, Singapore, 117417; email: chuats@comp.nus.edu.sg.

1. INTRODUCTION

The modern Web 2.0 activities and contents have pervaded the internet. Their increasing popularity originates from the 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 website. 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 [Cha et al. 2007] and covers 60% of the videos watched on-line. Its traffic accounts for over 20% of the web and 10% of the whole internet [Shen et al. 2009][Cheng et al. 2007].

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 website may return thousands or even more videos that are diverse and noisy. For example, YouTube returns more than 1 million videos for the query "September 11 attacks". Figure 3 illustrates the thumbnails of the top eight videos returned for a range of event oriented queries including "September 11 attacks". We can see that the evolution of the entire event is not directly observable by simply watching these videos one after another. Even worse, some videos are indeed weakly or not relevant to the query, which are highlighted by yellow box in Fig. 3. This distracts users from the gist of the event and forces them to painstakingly explore the returned videos for an overview of the event. To this end, a system that can provide users with a quick overview for each event is highly desired.

Intuitively, the system should be able to generate a condensed and succinct summary of the event related web videos. In this scenario, summarization technique appears as a typical approach to helping user more efficiently browse the retrieved videos. However, most existing video summarization algorithms are designed only to handle an individual video. The existing methods can be mainly divided into two categories [Benoit and Bernard 2006]: static summarization and dynamic summarization. The former usually generates a mosaic of keyframes, a slide-show or a storyboard, while the later produces a video previewer or skim which captures the essential content of a video. For our problem scenario, nevertheless we need to handle a large number of videos and most of them are only related to the key sub-events. Thus the existing summarization methods cannot be directly applied.

For genre-specific videos, such as news and documentary videos, techniques which perform multi-video summarization by utilizing multi-modality clues has been proposed. In [Duygulu et al. 2003], the authors tracked the evolution of news story and clustered video semantics across sources based on co-clustering. [Wu et al. 2006] presented a strategy for news video auto-documentary by extending co-clustering to exploit the duality between the stories and the textual-visual concepts. In [Chen et al. 2003], the authors proposed to mine the story structure, which includes four kinds of entities: who, what, where and when, to construct the summary for documentary videos. These methods demonstrate that summarization on multiple videos is feasible by leveraging the visual content and textual information available such as the transcripts from news videos. However, in the scenario of web videos, such information is not always available and thus these approaches cannot be directly employed. For examples, speech transcripts are absent in web videos, and instead user tags which could be noisy and redundant are available. Furthermore,

web videos are not as well organized as news and documentary videos. Some successive shots which are consistent with the evolution of an event may be separated or disorganized.

Recent studies [Cha et al. 2007][Wu et al. 2007] on video sharing sites have shown that there exists a significant amount of over 25% of duplicate videos detected in the search results. Although the exact or near duplicates have greatly reduced in recent version of YouTube, content redundancy still exist in various forms. We categorize the content redundancy on web videos into two classes: *near duplicate* and *overlap*. The former case is that most of the frames from the two videos are duplicates and the latter case is that the video pair shares some near duplicate frames. In this study, we leverage the case of overlap and investigate it from a different perspective. We demonstrate that web video summarization can benefit from the content overlap between web videos.

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 [Shen et al. 2008]. Typically, the scenes that convey the main information of an event, 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. It contains several principal sub-events, such as "the airplane was hijacked", "airplane crashed into the world trade tower" and "the world trade tower caught fire and collapsed". We define the shots displaying these sub-events in video as *key-shots*, which are believed to convey the dominant information of the event. We can observe that these types of key-shots appear frequently in the retrieved videos for a specific query. Therefore, we can identify such shots by first extracting the keyframes and then performing near duplicate keyframe (NDK) detection on the retrieved videos.

Motivated by the above observation and analysis, we propose to utilize the key-shots and their associated tags to summarize web videos. We first employ a near duplicate keyframe (NDK) detection method to identify the key-shots among the web videos. We then rank these key-shots and thread them according to the chronological order based on the original videos [Hong et al. 2009]. In addition, we exploit the tags associated with web videos to spread the visual semantics between the established shots through random walk on a visual similarity based graph. Finally the resulting summary is obtained by optimally selecting either the key-shots for video skimming or keyframes for a visual-textual storyboard. The contributions of this research are threefold:

1. We propose an event driven web video summarization system to help users browse the video search results with an overview. To the best of our knowledge, this is the first attempt to perform query-oriented summarization of multiple web videos relevant to news events.
2. We identify content overlap in web videos through a NDK detection technique and take advantage of the overlap information to support the analysis and mining of semantic relationships in videos.
3. We propose a hybrid approach that analyzes both the video content and its associated tags to produce the summary. The summary is capable of presenting storyboard and video skimming in one system.

Throughout this paper, we use the terms shot and keyframe interchangeably

for simplicity. The rest of this paper is organized as follows. In Section 2, we introduce the related work on NDK detection and video summarization. Section 3 gives an overview of the proposed system. Section 4 briefly reviews the techniques for key-shot processing including linking, ranking and threading. We discuss tags filtering and the mining of semantic relationship between tags in Section 5. Section 6 presents the strategy for summarization. In Section 7, we introduce our experiments and user study to demonstrate the feasibility and effectiveness of the system.

2. RELATED WORK

The main focus of this study is to utilize the content overlap in web video scenario and synthesize the visual and its associated meta-data to support event driven web video summarization. Our work is related to video summarization and NDK detection techniques.

2.1 Video Summarization

Depending on the presentation manner, video summarization can be categorized into static and dynamic summaries. Static summary presents the content on a static storyboard [Peng and Ngo 2006] with an emphasis on its importance or relevance, whereas dynamic summary (also known as video skimming [Ngo et al. 2005]) combines video and audio information to generate a shorter video clip [Truong and Venkatesh 2007]. It should be highlighted that in some cases, these two categories can be transposed into each other.

To date, most existing video summarization techniques aim to abstract a single video to facilitate content based access to the desired content in the video. With the advance of Web 2.0, more external information is available on the internet [Money and Agius 2007]. In this scenario, [Yang et al. 2003] extended the techniques from text processing to video question answering by modeling the web linguistic knowledge. [Neo et al. 2007] enhanced news video searching by leveraging extractable video semantics coupled with relevant external information resources to support event-based analysis. In [Zhu et al. 2003], a hierarchical video content description and summarization strategy supported by a joint textual and visual similarity is presented. The approach adopts video content description ontology and utilizes video processing to construct semi-automatic video annotation for a multi-layer video summary with different granularities. [Wu et al. 2006] proposed a method of threading and auto-documenting news stories according to topic themes. Story clustering is performed by exploiting the duality between stories and textual-visual concepts through a co-clustering algorithm. Then the dependency among stories of a topic is tracked by exploring the textual-visual novelty and redundancy of stories. Finally, a novel topic structure that chains the dependencies of stories is presented to facilitate fast navigation of the news topic.

However, as previously mentioned, the existing methods focus on either individual video or only multiple news and documentary videos. When facing the web videos, those methods can not be directly applied and we have to try off the beaten path.

2.2 Near Duplicate Keyframe Detection

Near duplicate keyframes are defined as keyframes that are similar to each other in spite of the variations of viewpoint, motion, lighting and acquisition time. Recently

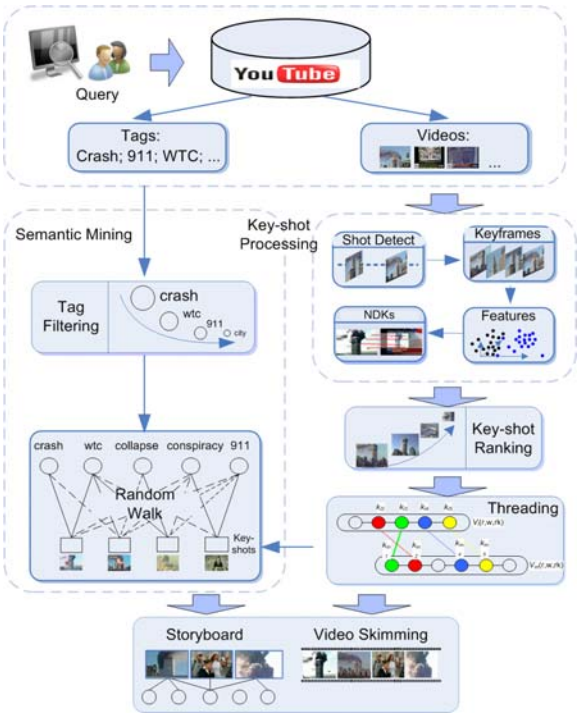


Fig. 1. The flowchart of the proposed event driven web video summarization system. It comprises four main stages: video and meta-data acquisition, key-shot processing, key-shot semantics mining and summary generation.

near duplicate keyframes detection has received a lot of attentions by the multimedia research community, especially after the emergence of TRECVID benchmark dataset[1]. [Duygulu et al. 2003] proposed to detect NDKs heuristically by investigating the derivation of keyframe similarity. [Zhang and Chang 2004] proposed a stochastic attributed relational graph matching, which is fully connected with the detected smallest uni-value segment assimilating nucleus corners as vertices for NDK identification. To accelerate the matching speed, [Ke et al. 2004] proposed an index structure based on locality sensitive hashing to fasten the process of PCA-SIFT descriptors matching.

In addition, a multidimensional index structure for local interest points to accelerate filtering and matching is proposed in [Zhao et al. 2007]. It was later extended to measure the spatial regularity of the matching patterns with high efficiency in the scenario of large scale video corpus [Zhao and Ngo 2008]. Another similar work deserving mention is the network-aware identification of video footages [Pedro and Dominguez 2007], which relies on robust hash functions. In their proposed video copy detection system, all videos involved in the detection are first converted into hash values before searching in the hash space. In this work, we employ the NDK detection algorithm in [Zhao and Ngo 2008] for determining the set of key-shots.

[1]http://www-nlpir.nist.gov/projects/trecvid/

Since we summarize web videos based on key-shots, the performance and efficiency of NDK detection will make a difference in our proposed method.

3. SYSTEM OVERVIEW

The goal of the proposed summarization system is to give users a quick overview for an event-type query by leveraging the content overlap in the web video search results. Figure 1 illustrates the framework of the proposed system, which consists of four main stages, namely, web videos and meta-data collection, key-shot visual processing including linking, ranking and threading, key-shot semantic analysis (tag filtering and key-shot tagging) and summary generation. Given an event query, a ranking list of videos and the associated meta-data, including tags for analysis, descriptions and titles for user based evaluation, can be obtained from YouTube.

In key-shot processing stage, we first extract the keyframes from the original videos sequentially and for each keyframe we extract its local point features for matching. Here we adopt scale-invariant feature transform (SIFT) features [Lowe 2004] because of that SIFT is a widely-used feature in high-level concept detection and near-duplicate detection for its detection effectiveness. To reduce computational cost, SIFT features are mapped to a fixed dimension and some keyframes are filtered by offline quantizing the keypoint descriptors. After that, the keyframe pairs with similarity measurement above a threshold are retained as NDKs and their corresponding shots are defined as key-shots. The NDK identification is based on the prior work [Zhao and Ngo 2008] which is able to provide high accuracy of detection even on noisy keyframes, and the source program is also publicly available. We then rank the key shots according to their *informative score*, which is defined as the linear combination of relevance and normalized significance. After that, the shots are chronologically threaded. Here, we utilize the chronological order lies in the original videos and formulate the threading as the minimization of the time lag between the key-shot pairs in different original videos.

The third stage propagates the tags based on visual similarity-based graph. We first define the *representativeness* and *descriptiveness* of tags and less informative tags with respect to the query are removed. A random walk [Hsu et al. 2007] is then performed based on the visual similarity-based graph to propagate the tags to other key-shots connected by NDK links. The final stage performs two kinds of summarization: video skimming summarization from the selected key-shots with a greedy algorithm and a visual-textual storyboard that contains the keyframes extracted from the key-shots as well as their tag descriptions. Both the duration of the skimming and the size of the storyboard are flexible according to users' requirements.

4. VISUAL PROCESSING OF KEY-SHOTS

We briefly review the visual processing of key-shots including key-shot linking, key-shot ranking and key-shot threading in [Hong et al. 2009]. In key-shot linking, we first perform keyframe extraction and NDK detection with the method described in [Zhao and Ngo 2008]. Denote the video corpus for a given query as $C = \{V_i,\ 1 \leq i \leq |C|\}$, where $|C|$ is the number of videos in C, and $S_i = \{s_{im}, 1 \leq m \leq |S_i|\}$ as the set of shots for V_i that corresponds to NDKs, where $|S_i|$ is the number of shots

in V_i. If two shots are identified as near-duplicate, both of them are defined as key-shots. Thus the set of key-shots can be segmented into certain near-duplicate key-shot groups $\{g_n, n = 1, 2, ...G\}$, where G is the number of groups in total. We denote $|g_n|$ as the number of near-duplicate key-shots in each group g_n. The number of key-shots in the whole corpus is equivalent to the total number of key-shots in all the near-duplicate groups: i.e., $\sum_n |g_n| = \sum_i |S_i|$. We then construct the visual similarity graph by inter-linking the key-shots. The graph is connected where each vertex is linked to all other vertices except for those within the same video. In the graph, the vertices are the set of all key-shots s_{im} and the weights of the edges are set to the confidence scores of near-duplicates [Zhao and Ngo 2008]. Note that after we remove the edges that are below the detection threshold used in [Zhao and Ngo 2008], the graphs would be segmented into multiple key-shot group islands.

In key-shot ranking, we first 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 distribution is somewhat related to the search query but not strictly agree with the distribution. In spite of that, the modeled distribution is capable of reflecting the tendency of videos' relevance scores [Capra et al. 2008]. Moreover, since that all the key-shots in g_n will be assigned to the same value of $|g_n|$, we define *informativeness* score of a key-shot s_{im} as follows:

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

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 *informativeness* score in Eqn. (1). After that, we select the key-shot with the highest *informativeness* 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 to directly generate storyboard. However, considering the attributes of evolution lie in news event, it would be more desirable to also embody the time constraint.

We propose to mine the sequence of key-shots by minimizing the time lag between the near duplicate key-shot pair in different videos in key-shot threading. 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:

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

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

5. MINING OF KEY-SHOT SEMANTICS

Mining the links on web has received growing research interests. The most popular example is Google PageRank, where the graph of web document nodes is interconnected by the hyperlinks. In the rich multimedia scenario, [Wu et al. 2007] presented a content-based copy retrieval algorithm to promote diversity on search results by removing redundant entries. [Jing and Baluja 2008] proposed to improve ranking for photo search based on visual links between images.

In contrast, we aim to utilize the visual similarity based graph to propagate the tags by tagging key-shots for producing visual-textual storyboard. Since the tags from the web videos with content overlap can convey different users' comprehension to the video [Siersdorfer et al. 2009], they would be valuable for exploiting the inherent semantics in video content. However, in our scenario, the tags associated may not convey the visual semantics of the key-shots since users assign tags only according to their comprehension of the whole video but not just the key-shots. Thus in next two Sections, we focus on tag filtering and tagging of key-shots based on visual similarity graph.

5.1 Tag Filtering

The tags from social sharing website are somewhat noisy and biased by different users. Intuitively, we can filter out the noisy tags with a general dictionary such as WordNet[2]. However, there are some tags that are absent in general dictionary but make sense for the news event, such as the tag "WTC (world trade center)" and "FDNY (the New York City Fire Department)" in "September 11 attacks". The filtering scheme should retain them. Some observations are detailed as follows:

1. For an event query, some tags appear in a large number of videos. Take "Columbia Space Shuttle disaster" as an example, "space", "tragedy" and "NASA" belong to this type. They are either part of the query phrase or terms strongly related to the query.
2. On the other hand, some tags may appear in few videos, such as "onvi" and "gaza". In these cases, they may be typos or only convey the comprehension from an *ad hoc* user.

Motivated by the observations, we define two metrics named *descriptiveness* and *representativeness* for filtering noisy tags. The former measures to what extent the tags describe the event, while the latter measures to what extent the tags can represent the event.

We employ the Google distance [Cilibrasi and Vitanyi 2007] between the query and the tags as the metric for *representativeness* since Google distance is a measurement of semantic interrelatedness derived from the number of hits returned

[2]http://wordnet.princeton.edu

by the popular Google search engine and such distance metric may explore the semantic distance between different query-tag pairs. Given the unique tags list $T = \{t_i,\ i = 1, 2...N\}$, i.e., no duplicate tags in T, we define the *representativeness* between the query q and each item in T as follows:

$$D(q, t_i) = \frac{\log f(q) - \log f(q, t_i)}{\log M - \log f(t_i)}, \tag{4}$$

where M is the number of the retrieved web pages. $f(q)$ and $f(t_i)$ are the number of hits for event query and tag respectively. $f(q, t_i)$ is the number of web pages on which both query q and tag t_i appear.

On the other hand, we define the *descriptiveness* as the normalized logarithm tag frequency. Given the frequency of tag t_i in T, the *descriptiveness* of tag t_i is defined as:

$$R(T, t_i) = \frac{\log t_i}{\max(\log t_i | t_i \in T)}. \tag{5}$$

From Eqn. (5), we can see that the lower the frequency, the lower the *descriptiveness* of tag t_i for the news event. In this way, the tags such as "onvi", which is probably a typo can be removed. Finally we linearly combine the two metrics to obtain the relevance score of tag t_i:

$$M(t_i) = (1 - \zeta) \cdot R(T, t_i) + \zeta \cdot (1 - D(q, t_i)). \tag{6}$$

Here, ζ is a modulation parameter ranges from 0 to 1 and is used to control the tradeoff between the two metrics. Since the *descriptiveness* is supposed to contribute more to the relevance of the query-tag pair, thus the metric of *representativeness* $R(T, t_i)$ in Eqn. (6) is restrained by the setting of $\zeta > 0.5$. If the tag appears only once, its value of *descriptiveness* will be downgraded to zero. We can utilize Eqn. (6) to obtain a rank list of tags with precise relevance to the query. In addition, more meaningful tags can be picked out by setting an appropriate threshold. The last column in Table 1 shows the number of tags whose relevance scores are greater than 1.25.

5.2 Key-shot Tagging

Up to now, we have established the key-shot groups, constructed the visual similarity based graph and filtered out tags relevant to the event. Intuitively key-shots in the same group should contain tags with high semantic similarity. However, this is not the case in social sharing network. Figure 2 shows two key-shots (shot 1 and 2) in one group which illustrates the explosion of "Space Shuttle Columbia". From the tags (left black font), we can see that they do share some identical tags but also some with lower semantic similarity. The results are that users assign tags at the video level but not at the shot level, and their contributed tags tend to reflect their own comprehension for the event which may be incomplete. Thus we propose a key-shot tagging scheme to propagate the tags within the key-shot groups for producing textual-visual storyboard.

Our assumption is that given a number of web videos, the tags that accurately describe the visual content of the key-shot do exist but only lie in some of the videos. Thus it is necessary to propagate such tags based on the visual similarity graph.

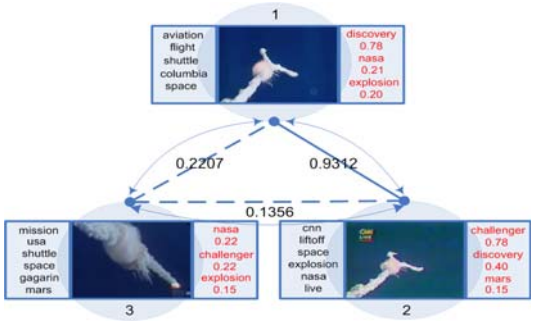


Fig. 2. Random walk is performed on three key-shots, in which key-shots 1 and 2 are near-duplicate shot pair. The correct tag "explosion" is propagated to the other two key-shots.

Figure 2 illustrates an example of tag propagation between key-shots, where initial tags for each key-shot are shown in the left (in black) and the tags propagated are shown in the right (in red). We can see that the tag "explosion" appears in one key-shot only due to users' bias and the different visual content in which the key-shot lies. After propagation, the tag "explosion" emerges in another two key-shots for providing a more relevant and meaningful annotation with community knowledge. Note that only the key-shots 1 and 2 are within the same key-shot group in Figure 2 and we cannot directly combine the tags associated with the two key-shots. Although the key-shot pairs 1-3 and 2-3 are not within the same key-shot group, the semantic relationship may be established by other key-shots from the videos that key-shots 1 and 2 lie in. Simple combination of all the tags may be a potential textual description for the event but not the shot. Considering that, in this study, we employ random walk for propagating the tags at shot level based on the visual similarity graph.

In random walk, the transition matrix $\boldsymbol{P} = [p_{ij}]_{n\times n}$, i.e., the similarity between key-shot, is used to control the transition of random walk process, where p_{ij} is the transition probability from state i to state j. We denote $f_{(k)} = [p_{(k)}(i)]_{n\times 1}$, which is a column vector of the probabilities residing in the nth vertex at iteration k, as the state probability at that iteration. The stationary probability $f_\pi = \lim_{k\to\infty} f_{(k)}$ is the state probability of the random walk process as the number of iterations approaches infinity if the convergence conditions are satisfied.

In this framework, the state probability $f_{(k)}(j)$ of vertex j at iteration k is:

$$f_k(j) = \alpha \sum_i f_{k-1}(i) p_{ij} + (1-\alpha) v_j \tag{7}$$

where v_j is the initial relevance score of tag t_j, and α is a weight parameter ranges from 0 to 1. The above process will propagate the tags associated with the key-shots to others with high visual similarity.

Intuitively $f_k(j)$ is parameterized by all the identified key-shots at iteration $k-1$ and its own initial relevance score v_j. They are then fused linearly with the weight α and $1-\alpha$ respectively. In our scenario, the visual similarity between the key-shots from different key-shot groups is comparatively lower and would not influence the propagation. The relationship in Eqn. (7) is updated recursively until all vertices in the graph converge. Here, we prove the convergence of the iteration in Eqn. (7),

which is re-written in matrix form as follows:

$$\mathbf{f}_\pi = \lim_{n\to\infty}[(\alpha\mathbf{P})^n f_0 + (1-\alpha)\sum_{i=1}^{n}(\alpha\mathbf{P})^{i-1}\mathbf{v}] \tag{8}$$

The first term converges to zero when n approaches infinity because each row in transition matrix P has been normalized to 1 and $0 < \alpha < 1$. According to the formula of infinite geometric series, we can obtain the closed form and unique solutions of Eqn. (8) as follows:

$$\mathbf{f}_\pi = (1-\alpha)(\mathrm{I} - \alpha\mathbf{P})^{-1}\mathbf{v} \tag{9}$$

We can see how random walk propagates the tags based on the visual similarity graph. After convergence, many key-shots will be annotated with more relevant tags for presenting helpful description to support users browsing. The visual-textual storyboard is able to benefit from the key-shot semantic mining in facilitating the comprehension of the event by users. The section of evaluation would validate the performance.

6. SUMMARIZATION

As we know, key-shots can be chronologically ordered by minimizing the time lag between each key-shot pair s_{im} and s_{jm} and the resulting sequence is $\{\lambda_l, 1 \le l \le G\}$ after minimization. For visual-textual storyboard, a summary may be constructed by the keyframes extracted from key-shots (actually key-shot processing is upon keyframes in this study) with higher informative score. For video skimming, we have to take into account the chronological order first, then *informativeness* scores. In addition, the summary has to meet user's requirements for length and duration. We denote the constraints of duration and length as T_s and F_s respectively. The strategy for summarization can be viewed as maximizing the tradeoff between the sum of relevance and time interval.

$$\begin{aligned} D = \arg\max_D (&\sum_{l\in D} if o(s_l) + \beta\tfrac{1}{|D|}\sum_{l,m\in D}\|\lambda_l - \lambda_m\|^2) \\ s.t.\ &\sum_{l\in D} length(s_l) < T \end{aligned} \tag{10}$$

where D denotes the set of the resulting key-shots and β controls the tradeoff. $|D|$ indicates the size of the set and $length(\cdot)$ reflect the duration of s_l. In experiment, we empirically set $\beta = 0.8$ for video skimming. In the case of storyboard summarization ($\beta = 0$), the maximization is subject to $\sum_{l\in D}\delta(s_l) = F_s$, where δ indicates the operation of selecting a frame from s_l. λ_l and s_l belong to the same key-shot.

We can solve the equation through a greedy algorithm, in which every step selects the key-shots with local maximal *informativeness* scores while retaining the maximal time interval. Here we conclude the detailed steps for producing both the visual-textual storyboard and dynamic video skimming as follows:

1. Key-shot groups are established by NDKs detection. Based on that, we construct the visual similarity graph and rank the key-shots according to its *informativeness* scores.

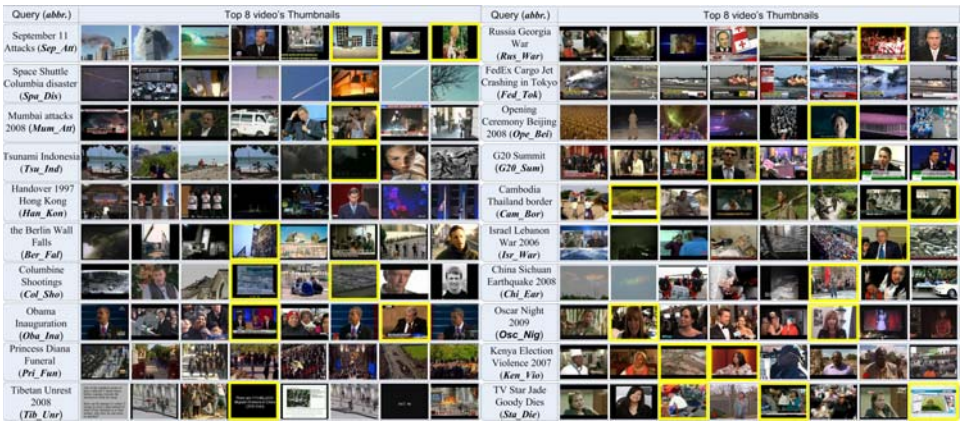


Fig. 3. The top-eight returned videos from YouTube by querying twenty event queries. These videos are in general diverse and noisy. Videos which are weakly or not relevant to the queries are boxed in yellow in the figure.

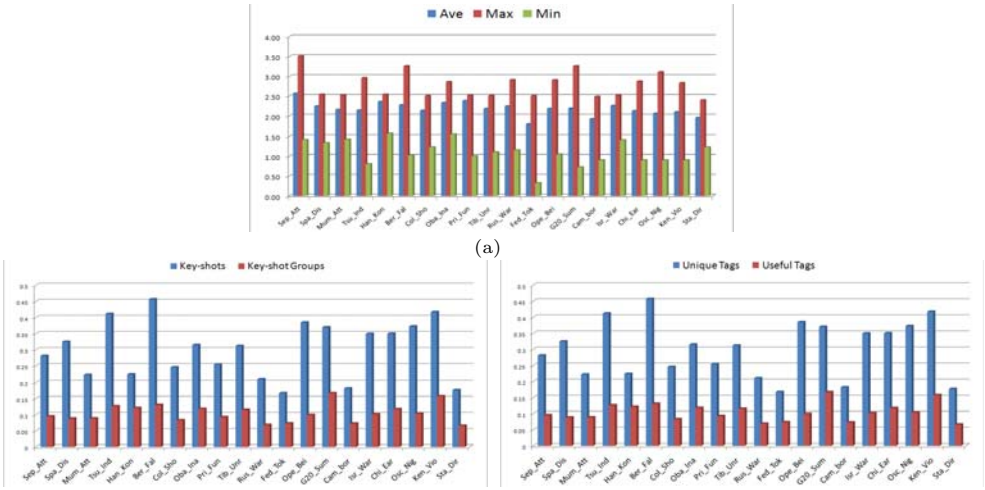
2. We thread the key-shots in chronological order by optimizing the time lag between key-shot pairs within the same group.
3. Noisy tags are filtered out by the metrics of *descriptiveness* and *representativeness*. We then propagate the tags, which are initially associated with the videos, on the graph of visual similarity between the key-shots by random walk.
4. We optimally select the key-shots for final summary according to Eqn. (10). Here β is set to control the summary as video skimming or visual-textual storyboard.

7. EVALUATION

In this section, we evaluate the feasibility and effectiveness of our proposed approaches. We first describe the dataset collected from YouTube and elaborate on the attributes of the dataset. We then analyze the performance of our approaches on key-shot detection, tagging and summarization.

7.1 Experimental Setup

The collection of videos are crawled from YouTube on April, 2009 by issuing twenty event queries. These queries cover different topics of news from historic events such as "Berlin Wall Falls" to recent news such as "Oscar Night 2009", as well as continuously evolving topics such "September 911 attack". Figure 3 shows the queries and the top-eight videos ranked by YouTube. For each query, we download either all or the top-N videos, where N is randomly decided based on the hits of the query and indicated in Table I. The associated contextual information such as tags, titles and descriptions are also crawled together with the videos. The number of videos download per query ranges from 48 to 143, with an average of 89 videos per query. Note that some videos are either inaccessible or removed by authors and thus cannot be crawled. This eventually forms a data collection of 1780 videos with a total of 155 hours, or an average of 7.5 hours per query. As seen from Figure 3,



(a) (b) (c)

Fig. 4. Analysis of the dataset. (a) the average, maximum and minimum length of videos for each query. For ease of illustration, the video duration (y-axis) is scaled with base 10 logarithm and shifted one and a half unit along the vertical axis; (b) the percentage of key-shots and key-shot groups to the number of extracted keyframes; (c) the percentage of unique tags and useful tags to the total number of tags in each query.

the video list is diverse, posting the real challenge of how to efficiently explore the list for the gist of a search topic.

In general, even though YouTube has removed most exact of the duplicates, there are still a number of near-duplicate videos, ranging from 1 to 9 per query, being found by our NDK tool [Zhao and Ngo 2008]. Empirically we set a constraint that any video with more than 90% of its keyframe being near-duplicate to another video is regarded as duplicate version. Since these videos carry no addition information, we remove them from the experiments. After removal, there are still considerable amount of near-duplicate keyframes being identified among the available set of keyframes. In total, 7.38% of keyframes are found to be NDKs, forming groups of NDKs (or key-shots) ranging from 28 to 189 per query. For tags, after removing stop words, the number of available, unique and useful tags for each query (video) are 1264 (14.2), 408 (4.2) and 128 (1.4) respectively. The useful tags are obtained by our tag filtering scheme, in which the threshold in Eqn. (6) is set to 1.25.

Figures 4 further show the detailed statistics about our video dataset. Figure 4(a) shows the minimum, maximum and average duration of videos for each query. While the average length is approximately 10 minutes, there are videos which are as short as 1 minute and as long as 100 minutes. Figure 4(b) gives the statistics about he ratios of key-shots and key-shot groups, respectively, to the set of available keyframes in a query. As seen from the figure, the ratio is somewhat dependent on the lifespan of a topic. For events which happened during a short time span and in a specific location such as "Handover 1997 Hong Kong" and "Obama inauguration", as high as 10% of keyframes are found to be key-shots. Whereas for events which have longer time span such as "Russia Georgia war" and "Kenya election violence 2007", only 3% of keyframes are regarded as key-shots since these events cover large varieties of sub-events and there are few overlaps among the sub-events. Finally, Figure 4(c) shows the ratio of unique and useful tags, respectively, to the total

available tags per query. After tag filtering, the percentage of the "useful" tags are reduced to an average of 10.5%.

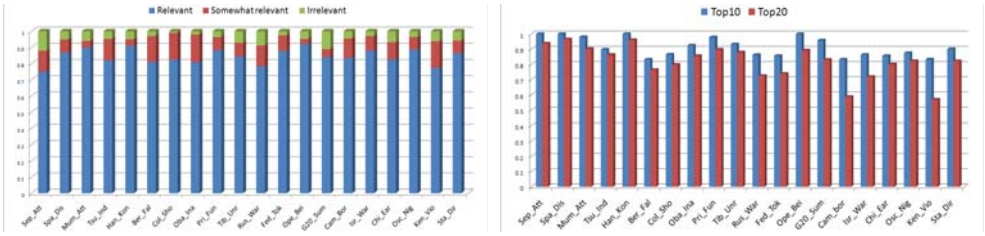
7.2 Key-shot Verification

To investigate whether the extracted key-shots are sufficient to provide the gist of news event, we conduct two user studies: 1) to assess the relevancy of key-shots generated by our approach and 2) to compare the key-shots which are automatically and manually selected. The first study is employed to judge whether the automatically established key-shots are important to the event while the second study to verify whether those "key" shots are sufficient to present the overview of the event. In the first user study, we ask eight assessors to rate the relevancy of key-shots selected by our approach. It is worthy of note that most involved assessors are graduates with computer science background. We don't put harsh terms on methodology of selecting users. However, at first, we ask all the users to comprehend the news events by Wikipedia, Google or other web knowledge. They are then involved in evaluating the system performance. We define three scales: relevant, somewhat relevant, irrelevant, for rating. Relevant indicates the key-shot is definitely related to the event. Somewhat relevant key-shots are those that cannot be judged by themselves or are weakly related to the event. Irrelevant key-shots are those that are definitely irrelevant to the event. Figure 5 depicts the examples of relevant, somewhat relevant and irrelevant key-shots for the queries "Space Shuttle Columbia Disaster" and "September 11 attacks". Figure 6(a) shows the result of user evaluation. From this figure, the average of relevant key-shots is close to 85%. If we further include "relevant" and "somewhat relevant" key shots, the percentage can be as high as 94.6%. This indeed shows that most key-shots selected by our approach are relevant to the query event.

In the second study, eight assessors are paid to generate the key-shots manually. Manually selecting key-shots by watching videos and browsing shots is expected to be a tedious job. It is likely that the assessors may simply overlook the video content and miss some key-shots. An annotation tool is designed for the assesses such that the videos are displayed together with all their shots. The assessors are asked to watch the videos of a query and then select up to twenty most representative shots as the key-shots of the query. As expected, the key-shots selected by different users are not always the same since there are biases between users. If we employ all the manually selected key-shots as ground-truth, the overlap ratio of the automatically identified top 10 and top 20 key-shots to the ground-truth is as high as 91.2% and 81.6% respectively, where the overlap ratio denotes the number of automatically identified key-shots appear in the ground-truth including the scenario of near-duplicates. From Figure 6(b), we can see that for those events that happened during a short time span and in specific locations such as "Handover 1997 Hongkong" and "Opening Ceremony Beijing 2008", all the shots from the automatically established top 10 key-shots are included in the ground-truth. Likewise most identified "key" shots are covered in the ground-truth for those events with a large number of manually selected key-shots such as "September 11 attacks" and "Space Shuttle Columbia Disaster". While for the events such as "Cambodia Thailand Border" and "Kenny Election Violence 2007", the percentages of the top 10 key-shots are comparatively low since that the small number of key-shot groups

(a) Relevant (b) Somewhat relevant (c) Irrelevant

Fig. 5. Examples from "Space Shuttle Columbia Disaster"(left) and "September 11 attacks" (right) for the defined three scales of (a) relevant, (b) somewhat relevant and (c) irrelevant.



(a) (b)

Fig. 6. Verification of key-shots. (a) automatically established key-shot verification. We can see that if we take into account the "somewhat relevant" key-shots, the ratio of these key-shots to the rest is as high as 95%; (b) subjective key-shot ground-truth verification. We can see that for subjectively selected top 10 key-shots, the average percentages of overlap with the automatically established key-shots approach to 91.2%. For top 20, the average is nevertheless as high as 81.6% though it decreases dramatically for some events due to users' bias.

lessen inclusion of top 10 key-shots in the ground-truth. Furthermore, as for top 20 key-shots of these types of events, the percentages are reduced to 58.6% and 56.9% respectively. In spite of that, the set of key-shots is sufficient for presenting an overview of the event in that top 10 shots, or even less are enough to convey the key sub-events for most queries, which will be proved by experimental results.

7.3 Key-shot Tagging Evaluation

User evaluation is conducted to validated the effectiveness of key-shot tagging. Again, eight assessors are asked to provide judgements on the set of tags that are propagated from other shots to the key-shots by visual similarity graph described in Section 5. We define three scales for assessment as follows:

1. Irrelevant (1): the tag is definitely not related to the key-shots.
2. Somewhat relevant (2): the tag is somewhat related to the semantics of the key-shots conveyed or the objects in the key-shots.
3. Relevant (3): the tag is directly related to the key-shots or the tag can be inferred from the key-shot without the context.

First the assessors are asked to browse the news events on Wikipedia[3] to have an overview about the events. Then the title and description are provided to them for understanding the content of the video. The key-shots are then presented to the evaluators randomly. Note that different number of tags can be obtained using different threshold for M (see Eqn. (6)) and the evaluation is conducted on the set of automatically generated tags after propagation. Here, we set the threshold

[3]http://www.wikipedia.com

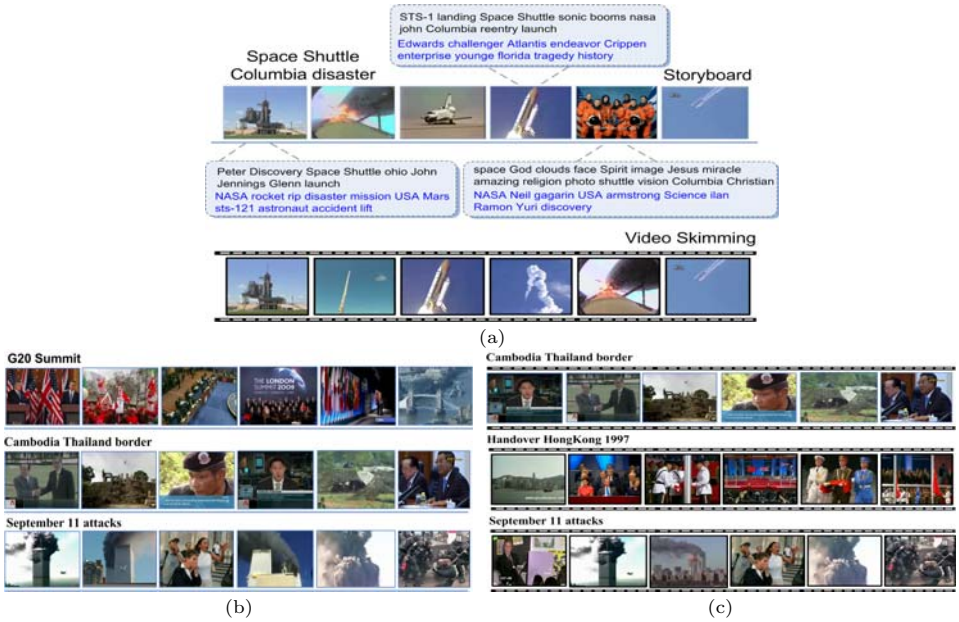


Fig. 7. Presentation of the two types of summaries. (a) comparison of video skimming and visual-textual storyboard; (b) visual-textual storyboard for the queries: "G20 summit", "Cambodia Thailand border" and "September 11 attacks"; (c) video skimming for the queries: "Cambodia Thailand border", "Handover HongKong 1997" and "September 11 attacks".

for M to 1.0 and 1.25 to study the variations of relevance pattern. The number of automatically generated tags is 875 and 653 respectively for different threshold of M. In addition, both ζ in Eqn. (6) and α in Eqn. (9) are set to 0.85. Figure 8(a) reveals that the automatically generated tags is comparatively relevant to the key-shots. By decreasing the threshold M, the average relevance values decrease due to the introduction of some noisy tags by the lower threshold.

7.4 Summarization Evaluation

In this section, we evaluate the quality of the two types of summary 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. Here, we take five queries as examples: "Columbia Space Shuttle disaster", "September 11 attacks", "Handover HongKong 1997", "Cambodia Thailand border" and "G20 summit". The result of the first query is presented in Figure 7(a) as video skimming and visual-textual storyboard. The storyboard is presented using frames from the key-shots with top six informative scores, in which the tag descriptions is embedded. The tags shown in black characters are initial tags while those in blue are automatically generated. We can see that more relevant tag descriptions are helpful for users to obtain a quick overview of the new event. Furthermore, the tags associated with the key-shots are capable of improving the precision of video search.

The video skimming for the first query is presented below the storyboard in Figure 7(a) and the key-shots are arranged in chronological order. We can see

that the space shuttle is 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. (10)). Here, for simplicity, we empirically set β in Eqn. (10) 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 7(b) shows another three examples for visual-textual storyboard. Here, we display the storyboard only for simplicity. We can see that in Figure 7(b), the second frame for "G20 summit" shows the scene of protest. As far as we know, it is a sub-event that happened during the G20 London Summit of this year. Another example is "Cambodia Thailand border", we can see that the fourth frame is an outlier of an anchor person frame. It is probably introduced because the anchor person frame predominates the source videos of this query. For the "September 11 attacks", two scenes of "airline crashed into the twin tower" are ranked highest. Other selected key-shots are capable of briefly describing the event as well. In contrast to storyboard, video skimming is constrained to time order as shown in Figure 7(c) for "Cambodia Thailand border", "Handover HongKong 1997" and "September 11 attacks". We display the video skimming and storyboard of two same queries for comparison. From Figure 7(c), we can see that the key-shots are organized in a chronological order. In general, this type of summary is capable of conveying more information about the event and thus facilitate users' comprehension for the event.

As it is difficult to objectively evaluate the proposed system, we evaluate the performance of summarization through user study. We again ask eight evaluators to assess the performance of video skimming and storyboard. 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 and the frame number of the summary. Shorter duration or number of frames indicate that the summary would include only the most relevant shots that also meet the chronological order constraints in skimming. When user increases the duration or frame number requirements, the summary will introduce more shots or frames 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. (10)). 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 performance of eight evaluators' subjective tests are illustrated in Figure 8(b) and Figure 8(c). When the duration T_s is set to 30s (see Figure 8(b)), we can see that for most queries, users think that the summaries are informative and

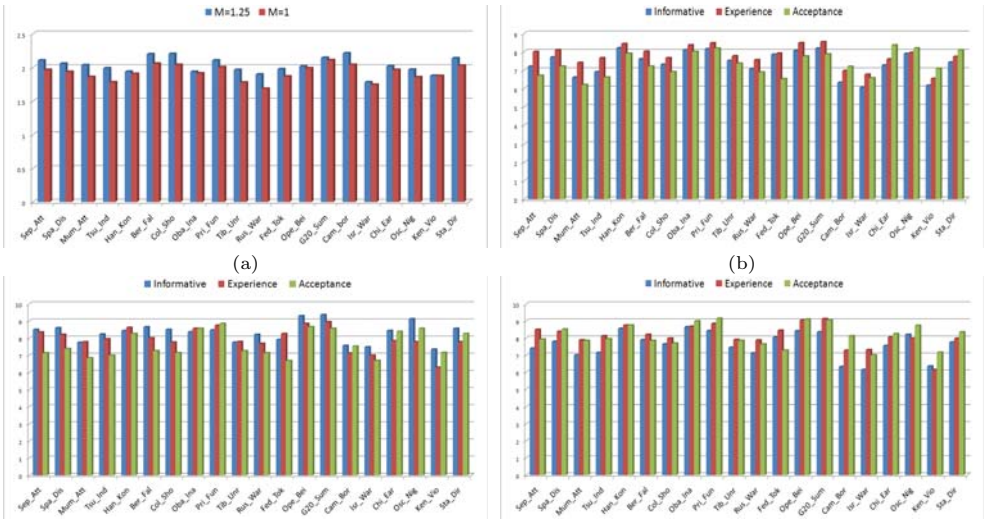


Fig. 8. User based evaluation on (a) the relevance of automatically generated tags to key-shots; (b) the video skimming with duration $T = 30$; (c) the video skimming with duration $T = 60$; (d) the visual-textual storyboard.

can help them to obtain an overview of the event. Figure 8(c) illustrates further comparison of performance, where the duration is set 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. Figure 8(d) illustrates the user based evaluation on the visual-textual storyboard. It seems that the average informative score is less than video skimming. However, the average acceptance score of storyboard is higher than video skimming. It may be because the tag descriptions embedded in the storyboard can help convey the overview of news events in a more concise way. The results show that textual information embedded storyboard is a promising solution to quick browsing in the scenario of web 2.0.

Because summarization is based on the established set of key-shots, thus the computational cost depends mainly on the NDK detection. Excluding the process of SIFT feature extraction and key-shot establishment, the summarization can be implemented near real time where approximately 57 seconds to generate 20 summaries (skimming with duration $T = 30$) for the collection of web videos of approximately 155 hours. In fact, most of the computational time has to be spent on the preprocessing of SIFT feature extraction and key-shot detection. However, these are preprocessing which can be conducted offline.

8. 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-shot related to the event 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 and mine semantics between key-shots.

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. We next performed tag filtering by the metric of linearly combined *representativeness* and *descriptiveness*. After that, useful tags are used to propagate textual descriptions between related key-shots based on random walk. Finally the first type of summary is produced by selecting the corresponding key-shots using a greed algorithm, while the second type of visual-textual storyboard is presented by the frames extracted from key-shots along with their textual descriptions.

The proposed framework can be extended to another two applications. The first is cross-modality link. Based on key-shot tagging, we can easily adapt them to the scenario of web document and construct the cross-modality link between keywords and key-shots. It would be helpful for users to browse web pages in the way of richer media. Another application is semantic propagation based on community knowledge based graph. The tags associated with near-duplicate videos, which may be categorized into exact duplicate and partial overlap, reflect the different comprehension from users. Intuitively the semantics between such near-duplicate videos may be exploited to improve the precision of searching web videos.

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 enhance the precision of key-shot tagging. However, our work points to other applications that may benefit from leveraging content redundancy. A direct way is to employ such redundancy for propagating the tags between videos from social sharing system to improve search. Furthermore, extension to other resources such as images in large sharing sites, like Flickr and Picasaweb, *etc.,* deserves further exploration.

REFERENCES

Benoit, H. and Bernard, M. 2006. Automatic video summarization. *Chapter in Interactive Video, Algorithms and Technologies ISBN: 3-540-33214-6*, 27–41.

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

Cha, M., Kwak, H., Rodriguez, P., Ahn, Y., and Moon, S. 2007. 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.

Chen, B. W., Wang, J. C., and Wang, J. F. 2003. A novel video summarization based on mining the story-structure and semantic relations among concept entities. *IEEE trans. on Multimedia 9*, 295–312.

Cheng, X., Dale, C., and Liu, J. 2007. Understanding the characteristics of internet short video sharing: Youtube as a case study. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*. San Diego, California, USA.

Cilibrasi, R. and Vitanyi, P. 2007. The google similarity distance,. *IEEE Trans. Knowledge and Data Engineering 19*, 370–383.

Duygulu, P., Pan, J.-Y., and Forsyth, D. A. 2003. Towards auto-documentary: Tracking

the evolution of news stories. In *Proceedings of the 11th ACM International Conference on Multimedia*. Berkeley, CA, USA.

Hong, R., Tang, J., Tan, H. K., Ngo, C. W., and Chua, T. S. 2009. Event driven summarization for web videos. In *Proceedings of the ACM Multimedia 2009 Workshop on Social Media*. Beijing, China.

Hsu, W. H., Kennedy, L. S., and Chang, S. F. 2007. Video search reranking through random walk over document-level context graph. In *Proceeding of ACM 14th international conference on Multimedia*. Augsburg, Germany.

Jing, Y. and Baluja, S. 2008. Pagerank for product image search. In *17th International World Wide Web Conference*.

Ke, Y., Suthankar, R., and Huston, L. 2004. Efficient near-duplicate detection and sub-image retrieval. In *Proceedings of the 12th ACM International Conference on Multimedia*. NewYork, NY, USA.

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

Money, A. G. and Agius, H. 2007. Video summarization: a conceptual framework and survey of the state of the art. *Journal of Visual Communication and Image Representation. 19*, 121–143.

Neo, S. Y., Ran, Y., Goh, H. K., Zheng, Y., and Chua, T. S. 2007. The use of topic evolution to help users browse and find answers in news video corpus. In *Proceeding of ACM 14th international conference on Multimedia*. Augsburg, Germany.

Ngo, C. W., Ma, Y. F., and Zhang, H. J. 2005. Video summarization and scene detection by graph modeling. *IEEE Trans. Circuits Syst. Video Technol. 15*, 296–315.

Pedro, J. S. and Dominguez, S. 2007. Network-aware identification of video clip fragments. In *Proceedings of the ACM International Conference on Image and Video Retrieval*. Amsterdam, Netherlands.

Peng, Y. and Ngo, C. W. 2006. Clip-based similarity measure for query-dependent clip retrieval and video summarization. *IEEE Trans. Circuits Syst. Video Technol 16*, 612–627.

Shen, J., Shepherd, J., Cui, B., , and Tan, K. 2009. A novel framework for efficient automated singer identification in large music databasesn. *ACM Trans. on Information Systems 27*, 3.

Shen, J., Tao, D., and Li, X. 2008. Modality mixture projections for semantic video event detection. *IEEE Trans. Circuits and Systems for Video Technology 18*, 1587–1596.

Siersdorfer, S., Pedro, J. S., and Sanderson, M. 2009. Automatic video tagging using content redundancy. In *Proceedings of the 32nd Annual ACM SIGIR Conference*. Boston, USA.

Truong, B. T. and Venkatesh, S. 2007. Video abstraction: A systematic review and classification. *ACM Trans. on Multmedia Computing, Communication and Application 3*, 1.

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

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

Yang, H., Chaisorn, L., Zhao, Y., Neo, S. Y., and Chua, T. S. 2003. Videoqa: question answering on news video. In *Proceedings of the 11th ACM International Conference on Multimedia*. Berkeley, CA, USA.

Zhang, D.-Q. and Chang, S.-F. 2004. Detecting image near-duplicate by stochastic attributed relational graph matching with learning. In *Proceedings of the 12th ACM International Conference on Multimedia*. NewYork, NY, USA.

Zhao, W. and Ngo, C. W. 2008. Scale-rotation invariant pattern entropy for keypoint-based near-duplicate detection. *IEEE Trans. on Image Processing 18*, 412–423.

Zhao, W., Ngo, C. W., Tan, H. K., and Wu, X. 2007. Near-duplicate keyframe identification with interest point matching and pattern learning. *IEEE Trans. on Multimedia 9*, 1037–1048.

Zhu, X., Fan, J., Elmagarmid, A. K., and Wu, X. 2003. Hierarchical video content description and summarization using unified semantic and visual similarity. *Multimedia System 9*, 31–53.

Appendix

We crawled videos using the twenty queries on April, 2009. The number of downloaded videos for each query is based on the returned video list. Table I summarizes the main information in the video collection including the number of videos (Num.), Duration (Dur.), Dupliate videos (dupV), Keyframe, Key-shot (K-S), Key-shot group (K-S Gr.), Total tags, Unique tags, Unique tags per video (UpV) and Useful tags.

Table I. Data Collection

No.	Query	Downloaded Videos			Key Shots			Tags			
		Num.	Dur. (hrs.)	dupV	Key frame	K-S	K-S Gr.	Total tags	Unique tags	UpV	Useful tags
1	***Sep_Att***	96	18.62	1	10428	535	189	1489	419	4.36	142
2	***Spa_Dis***	79	7.35	1	3647	535	168	723	235	2.97	64
3	***Mum_Att***	71	5.38	2	3551	197	65	1071	238	3.35	95
4	***Tsu_Ind***	90	6.61	2	5742	234	53	765	315	3.50	97
5	***Han_Kon***	49	5.88	1	2010	209	32	674	151	3.08	82
6	***Ber_Fal***	96	9.50	2	5163	436	113	1319	603	6.28	173
7	***Col_Sho***	63	4.53	4	2372	201	43	1058	260	4.12	88
8	***Oba_Ina***	120	13.60	9	4147	428	75	1882	593	4.94	223
9	***Pri_Fun***	99	12.54	1	6913	566	71	1120	285	2.87	104
10	***Tib_Unr***	83	6.67	3	4815	307	75	804	251	3.02	93
11	***Rus_War***	100	9.31	1	6944	174	56	2164	456	4.56	149
12	***Fed_Tok***	78	2.66	1	950	105	35	1372	230	2.94	101
13	***Ope_Bei***	100	8.03	2	5979	932	89	1234	475	4.75	123
14	***G20_Sum***	91	7.40	3	1458	96	35	1458	540	5.93	244
15	***Cam_Bor***	91	4.01	1	2245	79	28	1278	233	2.56	93
16	***Isr_War***	98	9.34	3	5969	181	54	1052	368	3.75	107
17	***Chi_Ear***	143	10.08	6	6296	481	132	1415	496	3.46	167
18	***Osc_Nig***	92	5.57	1	4836	431	78	1816	677	7.35	189
19	***Ken_Vio***	63	4.17	2	3520	89	31	563	235	3.73	89
20	***Sta_Die***	79	3.76	2	3185	153	43	2033	360	4.55	135