Exploiting the Chronological Semantic Structure in a Large-scale Broadcast News Video Archive for its Efficient Exploration

Ichiro Ide*¶† Tomoyoshi Kinoshita‡ Tomokazu Takahashi§
Hiroshi Mo† Norio Katayama† Shin'ichi Satoh† and Hiroshi Murase*

* Graduate School of Information Science, Nagoya University, Nagoya 464-8601 Aichi Japan
E-mail: {ide,murase}@is.nagoya-u.ac.jp
¶ Informatics Institute, University of Amsterdam, Amsterdam 1098XH Nordholland The Netherlands
E-mail: I.Ide@uva.nl
† National Institute of Informatics, Research Organization of Information and Systems, Chiyoda-ku 101-8430 Tokyo Japan
E-mail: {ide,mo,katayama,satoh}@nii.ac.jp
‡ NetCOMPASS, Ltd., Chuo-ku 104-0033 Tokyo Japan
E-mail: kino@netcompass.co.jp
§ Department of Economics and Information, Gifu Shotoku Gakuen University, Gifu 500-8288 Gifu Japan
E-mail: ttakahashi@gifu.shotoku.ac.jp

***Abstract*—Recent advance in digital storage technology has enabled us to archive more than 1,700 hours of video data from a daily Japanese news show in the last nine years. In this paper, to effectively make use of the video data in the archive, we first present a news video structuring method based on the chronological semantic relations between stories, namely the "topic thread structure". Next, we introduce an interface based on the structure, which allows users to track topics along their development and also choose video segments to visually "tell their own stories" using them as source materials. Analyses on the topic thread structures obtained by applying the proposed method to actual news footages revealed interesting relations between topics in the real world, while analyses on their size quantified the efficiency of tracking the topics and finding video materials for post-editing.**

I. Introduction

Recent increase of commercially available digital storage capacity has enabled us to accumulate a large volume of video data. Among various types of video, especially broadcast news video is a rich source of information concerning the human society, which is worth archiving and subsequently, retrieving and reusing. Thus, starting from the pioneering works in the Carnegie Mellon University's Informedia News-on-Demand project [8], many researchers have worked on indexing, retrieving, summarizing news video, and so on.

However, most of the works attempt to make use of the video data in an archive as they are. Such technology is indeed necessary, but now that a news video archive could easily house video footages from years of news shows, simply presenting a list of news stories that match a query based on the indexing and retrieving technologies, or simply showing a summary of a long video footage is insufficient. It is the time that we started handling the data according to their semantic relations together with their chronological nature, so that users could retrieve and understand the development of a news topic concerning their query, or reuse them to visually "tell stories" based on their own points-of-view, without being forced to browse through a long list of video footages which may contain redundant or noisy contents.

In this paper, we first present a news video structuring method based on the chronological semantic relations between news stories ("topic threading") in Sections III and IV, and next introduce in Section V, an interface based on the structure ("mediaWalker") which facilitates users to track news topics along the timeline, and at the same time, allows them to efficiently choose video sources so that after post-editing, they could visually tell their own stories based on the video footages. In addition, Section II introduces definitions of terms used throughout the paper, Section VI introduces related works in both the structuring and the visualization of news video, and Section VII concludes the paper.

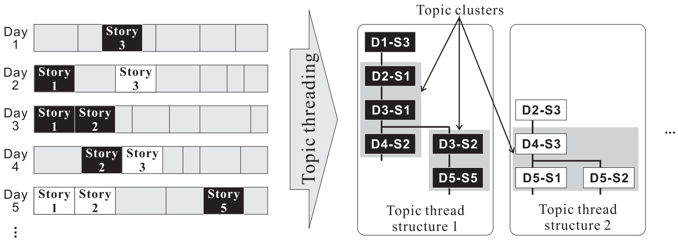
This paper is an extended version of the authors' previous publication [14].

II. Definition of terms

Some terms specific to news are used throughout this paper according to the following definitions. First, the following three terms follow the definitions in the Topic Detection and Tracking (TDT) workshop series organized by NIST [7].

- **Event**: Some incident that occurred at some specific time and place along with all necessary preconditions and unavoidable consequences.
- **Story**: A topically cohesive segment of news that includes two or more declarative independent clauses about a single event.

Proceedings of the Second APSIPA Annual Summit and Conference, pages 996–1005,
Biopolis, Singapore, 14-17 December 2010.



(a) Structure of news video per (daily) news show.

(b) Chronological semantic structures (topic thread structures) across news shows.

Fig. 1. Semantic structure in a news video archive.

- **Topic**: A seminal event or activity, along with all directly related events and activities.

Next, the following three terms are defined by the authors.

- **Topic thread**: A sequence of related stories chained chronologically. It may contain several topics.
- **Topic thread structure**: A directed graph composed of topic threads originating from a specified story.
- **Topic cluster**: A topically cohesive set of neighboring stories in the topic thread structure.

III. Analyzing the Chronological Semantic Structure of News Stories

A. Overview

Fig. 1(a) shows the general structure of news shows; Each of them is consists of several stories. Story segmentation is one of the oldest topic and a basic technology in the news contents analysis field, and the retrieval of segmented stories has also been approached from both keyword querying and content-based image retrieval. Especially, image-based retrieval has been widely studied recently in the TRECVID[1] community.

These are indeed essential functions for a news video archive, but we assume that a user would generally seek for the knowledge on the development of a topic as shown in Fig. 1(b), rather than details on individual stories. In other words, each story provides information mainly on the so-called 4Ws (When, Where, Who, and What) of an event, while a user would more likely prefer to understand the 1W1H (Why and How), which requires the browsing of the development of topics along the timeline.

Under this assumption, we introduce a method that chains individual stories into a "topic thread structure" according to their chronological and semantic relations (Fig. 1(b)). It represents not only local relations as directed edges, but also a global trend of topics as a directed graph. Fig. 2 shows an example of a large topic thread structure obtained by the proposed method. We can see that the structure not only chains obviously related topics but also topics without obvious relations, although they are actually related in the real world. The proposed method should help a user understand the actualities of such relations.

The following process is required to obtain a topic thread structure, and their details are introduced in the following sections.

1) **Story segmentation**: Each day's news video is segmented into stories.
2) **Topic threading**: Topic threads are extracted based on chronological and semantic relations between all pairs of stories.
3) **Extraction of topic clusters**: Topic clusters are extracted within a topic thread structure.

B. Story segmentation

Regardless to the existence of works making use of image features, in order to avoid dealing with the change of studio designs and editing policies in news shows, we decided to make use of only the closed-caption text (an audio transcript provided from the broadcasters) for story segmentation[2]. The following steps were applied to the closed-caption text obtained from each news show. Details of the method are found in our previous publication [10].

1) Apply morphological analysis[3] to each sentence of the closed-caption text, and extract noun compounds.
2) Classify the noun compounds into four semantic attributes; general, personal, locational/organizational, or temporal, based on our previous work [9], and create a term frequency vector for each attribute.
3) At each sentence boundary, concatenate w preceding and w succeeding term frequency vectors, and compare them in each semantic attribute. The window size w is changed from 1 to N_w, and the maximum similarity of the vectors measured in cosine measure is selected.
4) Combine the similarity in each semantic attribute as a weighted sum. In order to reflect the general difference in the importance of terms in each semantic attribute, an empirically obtained weight of (general, personal, locational/organizational, temporal) = (0.23, 0.21, 0.48, 0.08) was applied. If the combined similarity falls below a threshold θ_{seg}, a story boundary is detected there.
5) After applying steps 1) – 4) to the entire closed-caption text, create a term frequency vector for each story, and rejoin neighboring stories if the similarity between them exceeds a threshold θ_{cat}. This step is recursively applied until no more rejoining occurs.

In the following experiments, the parameters were set empirically as $N_w = 10$, $\theta_{seg} = 0.17$, $\theta_{cat} = 0.08$. An evaluation on fourteen news shows with 130 ground truth stories showed a segmentation ability of 95.4% recall and 90.5% precision when an error of plus/minus one sentence was allowed. Since

[1]See "TREC Video Retrieval Evaluation conference series," http://trecvid.nist.gov/ together with Smeaton et al.'s publication [20].

[2]The closed-caption text used in the following experiments was automatically synchronized to the audio channel at the time of archiving.

[3]A Japanese morphological analyzer JUMAN developed at Kyoto University was used. It is available for download from http://nlp.kuee.kyoto-u.ac.jp/nl-resource/juman-e.html.

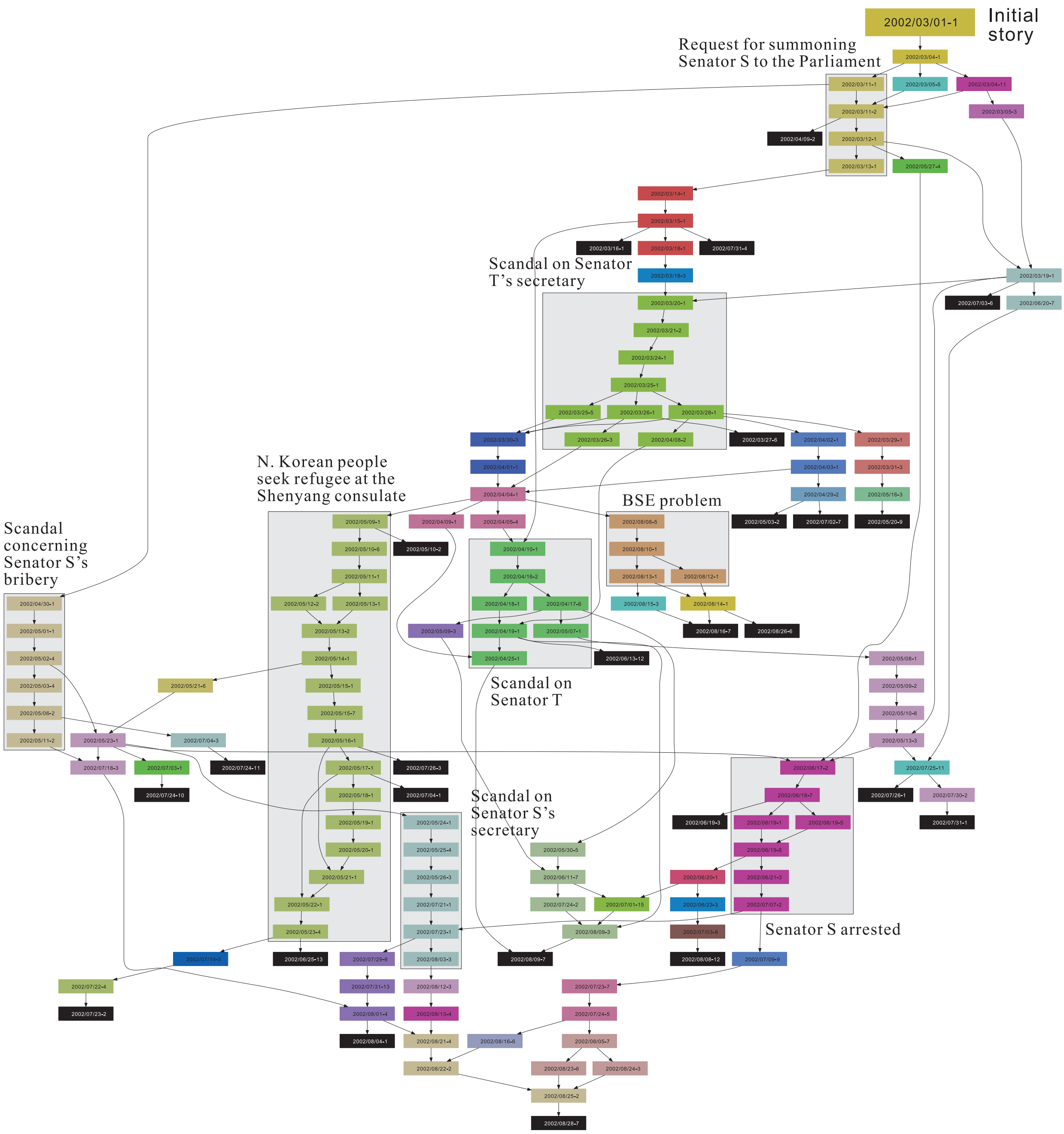


Fig. 2. Example of a topic thread structure originating from the first story on March 1, 2002 with a period of $d = 180$ days and a relation threshold $\theta_{trk} = 0.40$. Each box represent a news story broadcast on the designated date. It took 845 secs. to obtain this structure. Some large topic clusters are labeled manually by the authors to demonstrate the transition of the actual topics.

there tends to be redundant comments near story boundaries, we consider that the results are sufficient for practical use.

C. Topic threading

Next, a topic thread structure originating from a specified story is created by the topic threading method. The purpose of creating a topic thread structure is to chain related stories along the timeline in order to provide a user with paths, or "topic threads," to follow the development of topics.

The simplest solution for this task is to create a tree structure by recursively expanding related stories considering the chronological order. However, when a news video archive covers a long period, this solution results in creating a tree structure with numerous branches at each node (story), and also numerous duplicates of the same story appearing in nodes all over the tree. Tracking news topics story by story along such a structure requires watching a large number of video footages at each node to select an interesting story in the

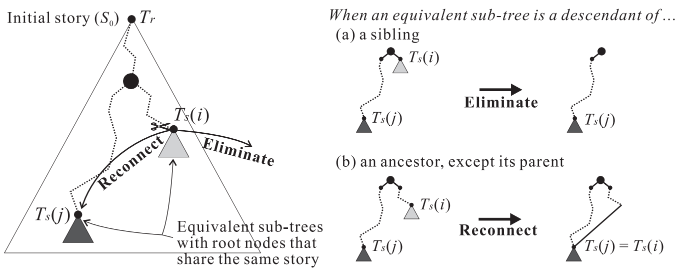


Fig. 3. Operations in the topic threading algorithm.

succeeding nodes, besides an occasional encounter with a video footage that has already been watched in a previous node due to duplicates.

The proposed topic threading method is designed so that it should avoid these problems. First, it unifies all the nodes with the same story to one node that appears in the deepest hierarchy of the tree. This eliminates the redundancy of the structure, so a user will not encounter the same video during the tracking. Next, upon unification, branches are eliminated or reconnected under certain conditions. This minimizes the number of branches at each node, so the time needed to watch video footages at each node for selection is reduced to the minimum. In the end, we obtain a directed graph structure, namely a "topic thread structure." Details of the topic threading procedure is as follows.

1) **Evaluation of the relation between stories**: The relation between two stories are measured by the cosine distance of their term frequency vectors. The vectors are created and compared in a similar way as in the story segmentation algorithm introduced in Section III-B. A pair of stories considered as related has a relation larger than a threshold θ_{trk}.
2) **Creation of a story relation tree**: Create a tree T_r originating from an initial story S_O by recursively expanding story relations under the following conditions:
    a) A child represents a story related to, and newer than the story its parent does.
    b) Among the siblings, the younger represents a newer story.
3) **Unification of nodes with the same story**: For all the sub-trees $T_s(i)$ in T_r, when there exists an equivalent sub-tree $T_s(j)$ branching with itself from an elder node, unify $T_s(i)$ to $T_s(j)$, and apply either of the following operations to the branch to the root node of $T_s(i)$ (Fig. 3):
    a) **Elimination**: When $T_s(j)$ is a descendant of $T_i(i)$'s elder sibling, eliminate the branch.
    b) **Reconnection**: When $T_s(j)$ is a descendant of $T_s(i)$'s ancestor (except its parent), reconnect the branch to the root node of $T_s(j)$.

Note that due to the elimination of an edge in step 3)-a), we can guarantee that all the siblings are independent; they are not related among themselves. This feature makes the selection among the child nodes by a user meaningful during the topic tracking process in a browsing interface, in the sense that the user would generally not need to eventually come back to a child node after tracking down a topic thread originating from its siblings, because they contain different topics than the previously selected topic thread.

Note that the above algorithm is for creating a topic thread structure in the future direction. To obtain a structure in the past direction, "newer" should be substituted with "older" in the algorithm.

D. Extraction of topic clusters

Since the topic thread structure is constructed by recursively expanding a story relation tree based only on local relations between stories, it may contain several related, but different topics that gradually develop along the time. In order to reveal such sub-structures, topic clusters which contain homogeneous stories along the topic thread structure are extracted. Note that the topic clusters are overlapped with the topic thread structure, and are not obtained by clustering all the stories that compose the topic thread structure, as it is generally the case in other works, which will result in ignoring the original topic thread structure. Details of the topic cluster extraction procedure is as follows.

1) Set story S_O as the cluster center ($C_0 = S_O$) and the story-in-focus ($S = S_O$).
2) Let the children of story S be $S_c(j)$ ($j = 1, ..., J$). When the relation between stories C_0 and $S_c(j)$ is larger than θ_{cls}, set $S_c(j)$ as a new cluster center ($C_0 = S_c(j)$).
3) Apply step 2) recursively by scanning the entire topic thread structure, shifting S one story after another.
4) Label stories between each topic cluster and the next ones with a same cluster number.

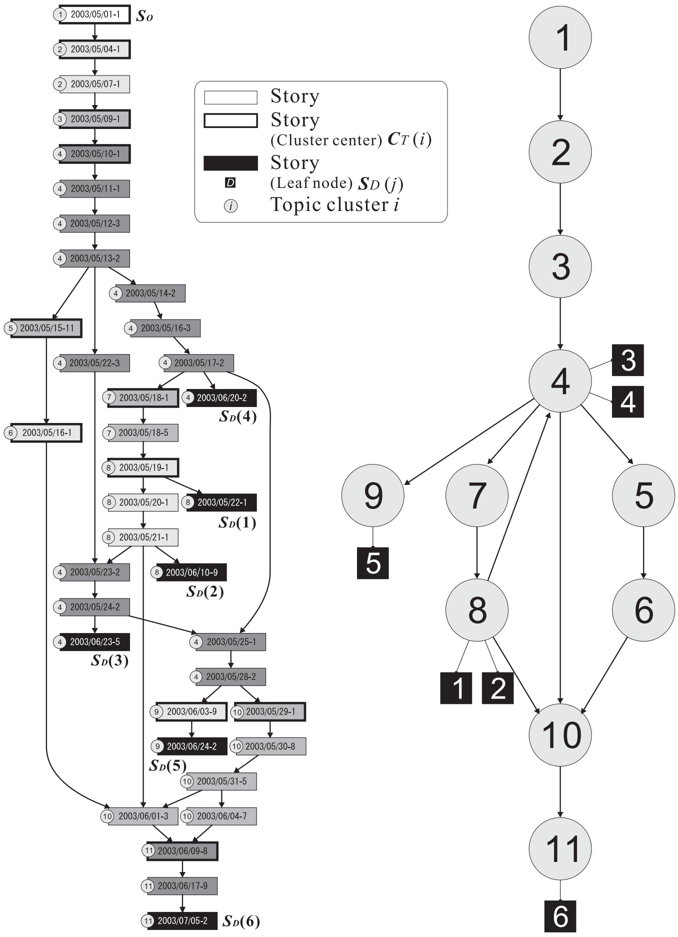
E. Example of a topic thread structure

Fig. 4(a) shows an example of a topic thread structure obtained by the proposed method. Fig. 4(b) shows a degenerated representation of the topic thread structure by topic clusters, with Table I summarizing their actual contents which were manually analyzed.

From this structure, for example, we can choose a topic thread $\{C_T(1), C_T(2), C_T(3), C_T(4), C_T(10), C_T(11)\}$, where we can follow the main-stream development of the SARS (Severe Acute Respiratory Syndrome) outbreak in 2003. Likewise, "side stories" such as a partial topic thread $\{C_T(4), C_T(5), C_T(6), C_T(10)\}$, lets us follow the development of the outbreak inside China, $\{C_T(7), C_T(8)\}$ on a false alarm in Japan.

F. Experiment

In order to analyze the effect of the proposed topic threading method, we applied it to actual broadcast news video. The stories that compose each topic thread structure were limited to a period of d days from the originating story, to limit the computation time. Table II shows the size of the data set and



(a) Topic thread structure (b) Topic thread structure represented by topic clusters

Fig. 4. A topic thread structure originating from the first story on May 1, 2003 with a period of $d = 100$ and a relation threshold of $\theta_{trk} = 0.40$.

the parameters for the process. The data set is part of the NII TV Broadcast Video Research Corpus: TV-RECS [15].

After rejecting stories with only one sentence from the results of the story segmentation, 1,431 initial story (S_O) candidates remained. Next, out of the topic thread structures (T_r) originating from all the initial story candidates, we selected 437 of them that consists of more than two stories. Under the conditions in Table II, it took several seconds to half an hour to obtain a topic thread structure on a Sun Blade 1000 work station, depending on its complexity. The following analyses are performed on these topic thread structures.

IV. Analysis of the Topic Thread Structure

A. Analysis on the contents of the structures

Among the obtained topic thread structures, there were informative ones such as those exemplified in Figs. 2 and 4. On the other hand, there were also incomprehensible structures, and moreover, many of them were actually unimportant structures involving only a few stories on a simple news topic.

TABLE I
Actual contents of the topic clusters in Fig. 4.

Cluster #	Contents
$C_T(1)$	SARS found and spreads in Beijing.
$C_T(2)$	SARS spreads all over China.
$C_T(3)$	WHO sends a mission to Beijing.
$C_T(4)$	SARS starts calming down in China, but spreads in Taiwan.
$C_T(5)$	Chinese government worries about the spread of SARS in the countryside.
$C_T(6)$	Chinese government keeps an eye on the spread of SARS in the countryside.
$C_T(7)$	Taiwanese doctor found infected to SARS after a trip to Japan.
$C_T(8)$	Search for SARS infection in Japan.
$C_T(9)$	Anti-SARS conference held in Beijing
$C_T(10)$	SARS calms down in China, but found in Toronto.
$C_T(11)$	SARS calms down in Taiwan, and WHO declares the end.

TABLE II
Data set and parameters used in the topic thread structure analysis.

News show	NHK News7 (Daily evening news in Japanese. 20 to 30 minutes long)
Period of initial stories	March 1 to June 30, 2002
Period of threading (d)	100 days; Approximately 171,600 secs. or 47.7 hours of video
Relation threshold (θ_{trk})	0.40

The former situation tends to be caused by mistakenly chaining actually unrelated stories. This occurs firstly when the story segmentation fails. Secondly, it occurs when the value of the relation threshold θ_{trk} is inappropriate for the topic; The relation tends to be high for topics on periodic events such as typical crimes or weather phenomena. The latter situation may also occur by similar reasons, in addition to the nature of the daily news show that it tends to take up a topic only when it is focused during a short period.

Since it is difficult to further evaluate the general quality of the obtained topic thread structures, we will focus on the quantification of the efficiency of browsing the contents along the structure in the next section.

B. Analyses on the size of the structures

In order to quantify the efficiency of browsing a news video archive based on the topic thread structures, we analyze them by the number of stories or clusters that compose them, and also by the video length that a user needs to watch to grasp the contents. Since the worst assumption is that a user needs to watch all the video footages in the archive during a certain period, we will observe the ratio of the video length in the structures against the archive.

1) Size of topic thread structures: First we analyze the size of the 437 topic thread structures. Fig. 5 shows the histogram of the number of stories that compose a structure. From the result, we confirmed the necessity of automatically analyzing the structures, since the result shows that they consist of many stories, which could be difficult to be analyzed manually.

2) Size of topic threads: Next we analyze the size of the 1,788 topic threads in the 437 topic thread structures. Here, a

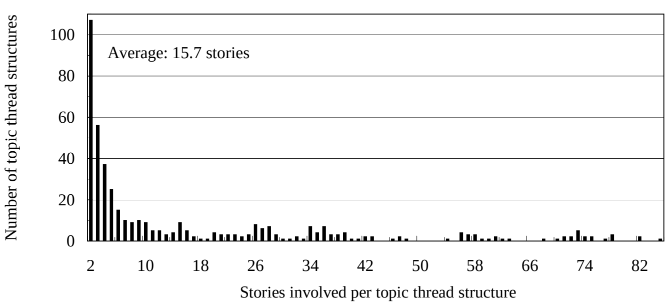


Fig. 5. Histogram of the number of stories that composes a topic thread structure.

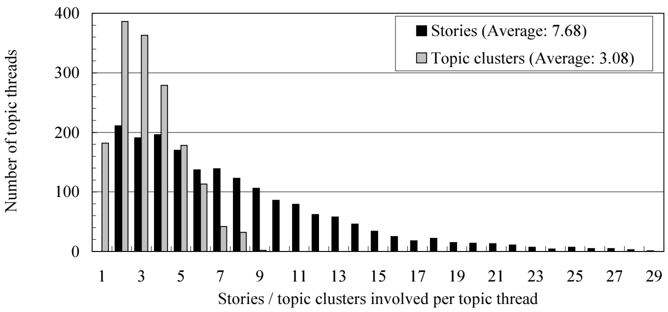


Fig. 6. Histogram of the number of stories / topic clusters that composes a topic thread. Note that topic threads that consists of only one story was filtered out.

path that connects S_O and a leaf node S_D is counted as one topic thread. For example, in the structure shown in Fig. 4, there are at least six[4] topic threads connecting S_O and $S_D(j)$ where $j = 1, ..., 6$.

Fig. 6 shows the histogram of the number of stories and topic clusters that compose a topic thread. As mentioned in Section IV-A, quite a number of topic threads consists of only a few stories, but there are a considerable number of longer ones that indicated the usage of the proposed method should be effective. Meanwhile, the maximum number of topic clusters was nine even for very long topic threads, which leads to the discussion in the next part of this section that using them could make the browsing drastically effective.

Fig. 7 shows the histogram of the duration of the topic threads. We assume that the frequency for 100 days was relatively high since there are potentially longer topic threads that were forced to terminate at d days. From this result, we realized the necessity to speed up the proposed method so that longer topic threads could be extracted in realistic computation time.

3) Efficiency of watching video footages along the topic thread structure: In this paper, we assume that a user searches and watches video footages to understand the development of certain topics-of-interest. In order to assist such searching, we provide an interface that allows a user to watch video footages

[4]Precisely speaking, there are actually more due to multiple paths that connect them.

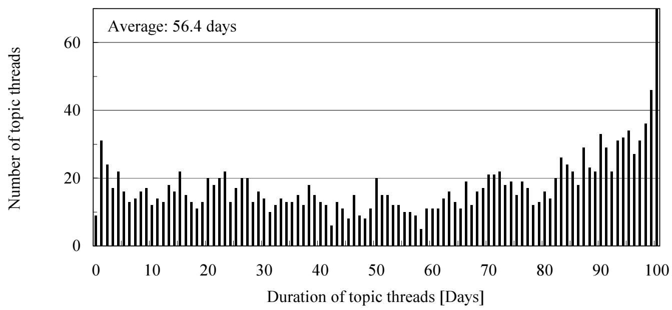


Fig. 7. Histogram of the duration of topic threads.

TABLE III
STATISTICS ON THE TOTAL VIDEO LENGTH [SECS.] OF STORIES THAT COMPOSE THE TOPIC STRUCTURES. THE NUMBER IN A PARENTHESIS INDICATES THE RATIO OF THE TOTAL VIDEO LENGTH OF THE STORIES THAT COMPOSE A TOPIC STRUCTURE TO THAT OF THE CORRESPONDING PERIOD $d = 100$ DAYS; APPROXIMATELY 47.7 HOURS.

	Minimum		Average		Maximum	
Topic thread structure	5	(0.0029%)	4,380	(2.6%)	38,125	(22%)
Topic thread	5	(0.0029%)	2,770	(1.6%)	19,038	(11%)
Topic cluster	5	(0.0029%)	1,280	(0.75%)	6,741	(3.9%)

along a topic thread structure, which will be introduced later.

Table III shows the effect of watching video footages on the user's topic-of-interest along a topic thread structure as the statistics of time required and reduced to watch them. As a worst-case baseline, we consider that a user needs to watch all the video footages during a period d. In the Table, "Topic thread structure" shows the statistics on the total time of all the video footages in a topic thread structure. It shows that a topic thread structure covers on average 2.6%, and in the worst case 22%, of all the video footages during the period. This case is considered as close to the traditional story-based video retrieval methods that require a user to watch all the video footages that match a query, though the proposed topic thread structure covers more stories than them since it also includes stories which are tracked from the stories that match the initial query.

In actual use, we consider that a user would follow a certain topic thread through the browsing interface proposed later in Section V. In order to measure the effect of watching the video footages along the topic threads, in Table III, "Topic thread" shows the statistics on the total time of all the video footages along a topic thread, and "Topic cluster" shows that of only the video footages of the stories which were the cluster centers (C_0; The first story in a topic cluster). The results show that browsing along a topic thread reduces the time needed to watch video footages related to the user's topics-of-interest by 1% on average and 11% in the worst case. Browsing by topic clusters along a topic thread, which could be considered as a digest view of the topic thread, further reduces it to 0.85% on average and 7.1% in the worst case. We consider the reductions should be extremely effective for practical use.

Fig. 8. Using the mediaWalker interface via a touch-panel display.

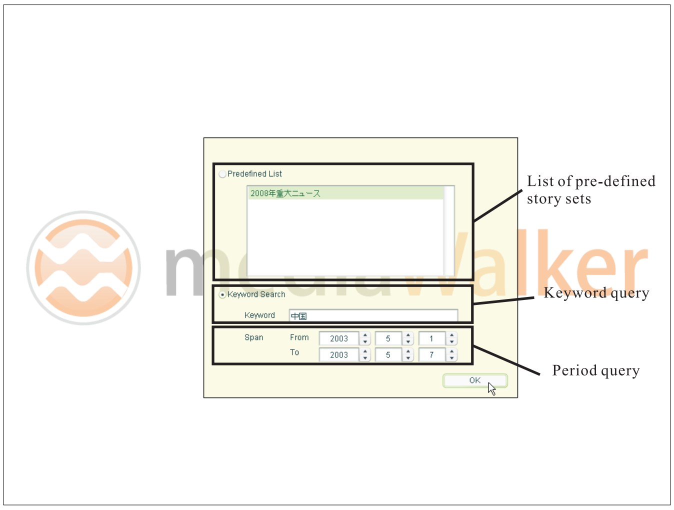


Fig. 9. The mediaWalker interface: Initial story listing.

V. MEDIAWALKER: A NEWS VIDEO BROWSING AND EDITING INTERFACE

A. Overview of the interface

We have implemented an interface for browsing and editing news video based on the topic thread structure named "mediaWalker"[5] [13]. As shown in Fig. 8, the interface could be used via a touch-panel display, so a user could interactively track up and down the chronological development of news stories. Details of the functions of the interface are described step-by-step in the following sections.

B. Initial story listing interface

To start with, a user needs to select an initial story. As shown in Fig. 9, two methods are provided through the *Initial Story Listing* interface; 1) Select a pre-defined story set, or 2) Search for a story by a combination of a query term and a period of time. Since the query term field can be left blank, listing all stories during a certain period is also possible. Stories from the selected story set or the result of the query are listed in the subsequent *Initial Story Selection* interface.

C. Initial Story Selection interface

As shown in Fig. 10, in the *Initial Story Selection* interface, thumbnail images of stories are listed. The stories listed here are those that matched the criterion specified in the *Initial Story Listing* interface. The list of thumbnail images could be rotated, and also played when clicked. Once finding an interesting story, the user sets it as the initial story by pressing either the "Past" or the "Future" buttons next to the thumbnail; the former will show the topic thread structure originating from the story towards the past, and the latter towards the future in the subsequent *Topic Thread Browsing* interface.

[5]The interface is demonstrated at the following URI: http://www.murase.m.is.nagoya-u. ac.jp/˜ide/res/mediaWalker/ .

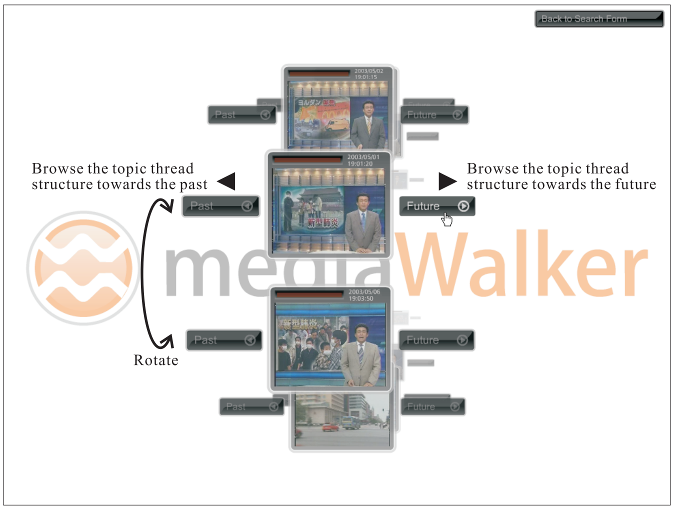


Fig. 10. The mediaWalker interface: Initial story selection.

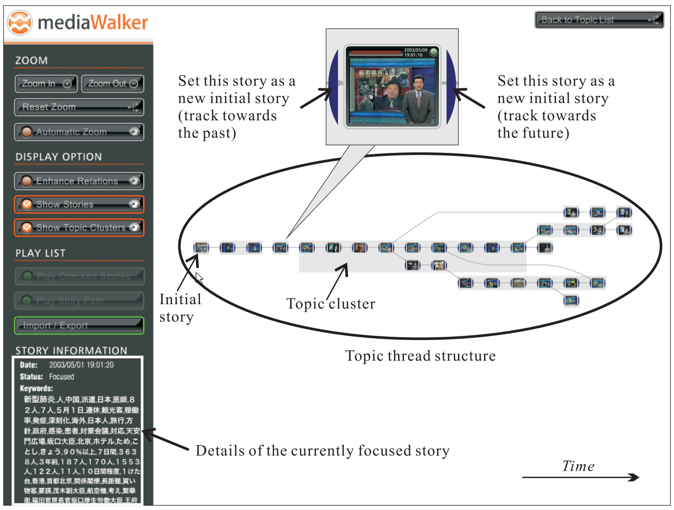
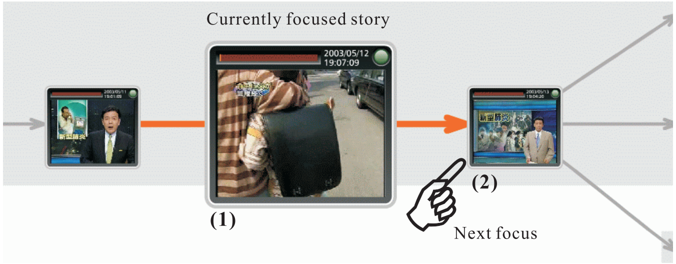
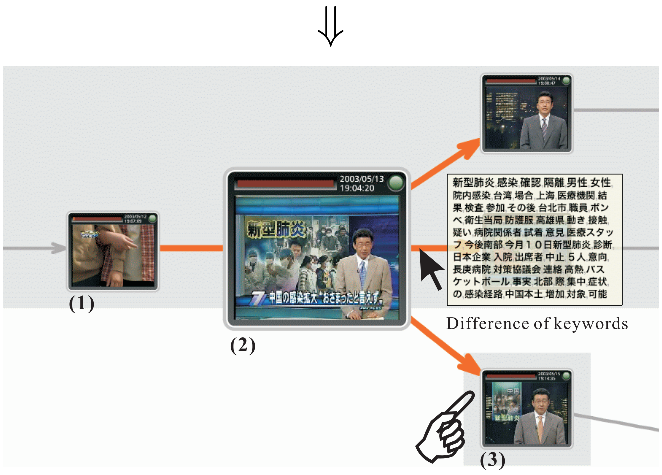


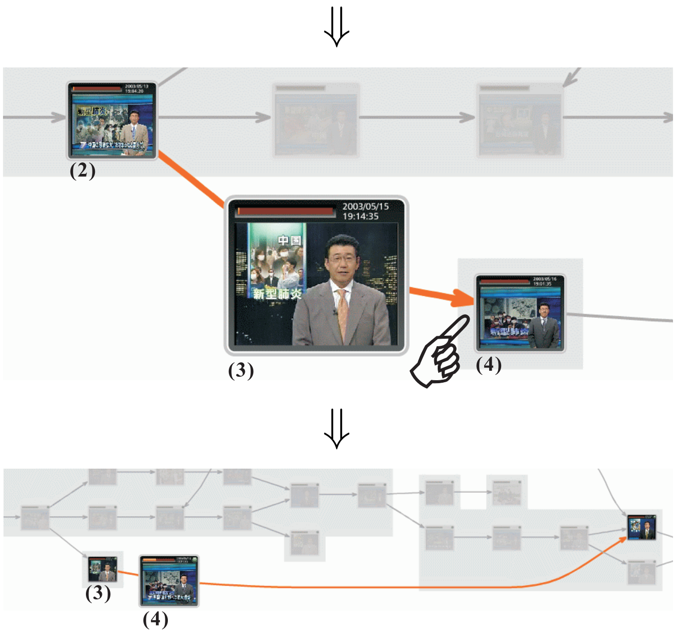
Fig. 11. The mediaWalker interface: Topic tracking and browsing.



(a) When a thumbnail image that represents a story is clicked, it is enlarged and played. The enlargement is adjusted so that the preceding and the succeeding stories should remain in the screen.



(b) When the pointer overlaps an edge, an overlayed window appears and shows the difference of keywords between the stories on both sides of the edge. This is helpful especially when selecting a thread at a diverging story.



(c) The user will keep on selecting and watching video footages of stories along a topic thread structure in this manner. This helps the understanding of the topics present in the topic thread structure.

Fig. 12. Tracking down story by story in the *Topic Thread Browsing* interface, along the topic thread structure shown in Fig. 4.

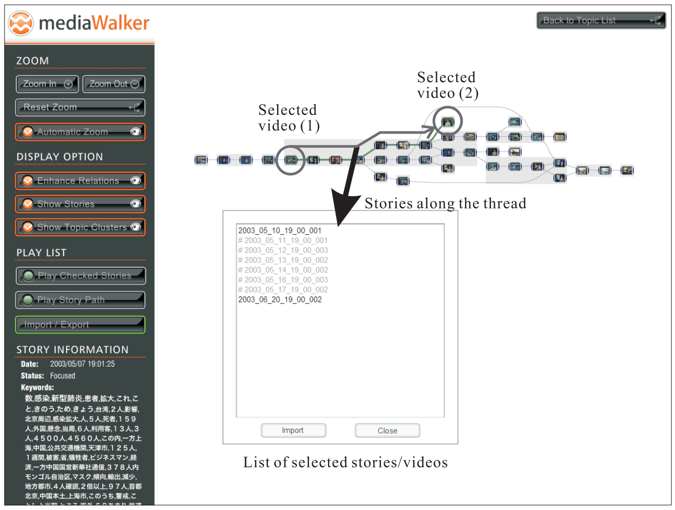


Fig. 13. The mediaWalker interface: Video editing and exporting.

D. Topic Thread Browsing interface

1) Topic tracking and playing functions: As shown in Fig. 11, the *Topic Thread Browsing* interface visualizes the topic thread structure originating from a specified story, where each node (story) is represented by a thumbnail image. Topic clusters are shown as hatched regions in the background. A user will track down a topic thread structure by clicking the thumbnail images to watch the corresponding video footage story after story. This function allows a user to efficiently explore a news video archive based on the context.

In addition, each thumbnail image is accompanied by a pair of parentheses, which allows the user to set the story as an initial story and show its topic thread structure to the past (left paranthesis) or to the future (right paranthesis). This function allows the user to freely explore the entire topic thread structure in the archive.

An example of a user tracking along a topic thread is shown in Fig. 12.

2) Video editing ("story telling") function: The *Topic Thread Browsing* interface is equipped with another function that allows a user to select stories to "tell stories" by rearranging the video footages in the archive. This function is implemented based on our policy [12] that video data in an archive should be reused effectively in an efficient manner based on the semantics.

As shown in Fig. 13, besides simply selecting each story, this function allows a user to select stories along a topic thread between two or more specified stories. The interface outputs the selected stories as a list of story/video IDs, which could be exported as an input to a video summarization system, or simply as a play-list for a video editing software.

VI. Related works

A. Works on the analysis of chronological semantic structures in news

The simplest approach to structure news stories is to first cluster related stories into a topic and then chain them linearly in chronological order, as in the "Topic detection task" defined in the TDT workshop series [7]. A majority of the existing works on news story structuring is based on this approach, such as Duygulu et al.'s work on news video tracking based on the detection of similar images and logos [5]. However, this approach is not appropriate in two senses, when the size of the data set explodes. First, since the structure tends to extend in a long chain of stories, watching video footages along the structure is extremely time consuming. Second, a linear structure could not express the simultaneous flow of multiple topic threads within a topic-of-interest.

In the final TDT workshop (TDT2004), the "Hierarchical topic detection task" [6] was introduced. It aimed at structuring a directed acyclic graph of stories instead of clusters, but it was more of a way to assign multiple topics to a story and describe relations between topics. Meanwhile, Wu et al. proposed a method that structures clustered news stories into a binary tree based on the chronological order and the local change in a topic [22]. This method partially solves the above-mentioned problems, but it still could not represent the simultaneous flow of multiple topic threads.

The method proposed in this paper structures stories into a graph so that it could represent such simultaneous flows. As related works that share the same approach, there are topic threading methods for news paper articles [16], [21], [23]. However, these methods do not consider the independence between topic threads at a diverging node, which results in the existence of redundant edges between nodes, where the proposed method eliminates such edges, for the sake of efficient tracking.

Another difference of the proposed method with the majority of the existing works, except for Nallapati et al.'s work [16], is that it does not cluster a set of stories that forms a topic beforehand. It instead forms the topic thread structure by chaining locally related stories under certain rules and later extracts topic clusters from the structure, considering the nature of news contents that a topic may gradually develop into, diverge to, or merge with other topics. This feature is essential to find unobvious relations between topics.

B. Works on the visualization of news structure

A typical method to visualize a video structure on a screen is the so-called "storyboard," which lists up an arbitrary number of representative frames from the video as thumbnail images. The interface proposed in this paper could be considered as a two-dimensional storyboard represented by a directed graph.

As for the works in visualizing news structure, most of them focus on the so-called 5W1H (When, Where, Who, What, Why, and How) attributes, and especially the first three of them (3Ws). As part of the Carnegie Mellon University's Informedia project, Christel et al. proposed a news video browsing interface that visualizes news stories based on the combinations of the 3Ws [2]. We focused on the cooccurrence of the "Who" attribute, and proposed a news video browsing interface by exploring the social network in news contents [11].

Meanwhile, the proposed interface makes use of the news contents as a whole except for the "When" attribute. It only considers the 3Ws by giving certain weights to them when evaluating the relations between stories. As a related work, Rautiainen et al. proposed a cluster-temporal browsing method [19] which visualizes the cluster of stories with similar image features. On the other hand, de Rooij et al. proposed an interface that maps semantic and chronological relations between news video footages on a sphere or a flat plane [4].

As for supporting video re-edition in an archive, Casares et al. proposed the "Silver" interface [1], which allows the re-edition of a single video, but not the recompilation of multiple video footages as in the proposed interface.

VII. Conclusion

In this paper, we proposed a topic threading method for a large amount of news video, and also introduced a video footage browsing and editing interface based on the topic thread structures. Analysis on the contents revealed interesting relations between actual topics in the real world, while analyses on the size of the topic thread structures quantified the efficiency of tracking down the development of topics-of-interest and also the selection of video footages for a further editing purpose.

As future works, we have started working on linking the structures with news shows in other languages [17], real-world contents such as web news articles, Wikipedia articles [18], and shared video footages and images on the web, together with the development of a method to summarize stories along a topic thread to compose new video contents.

Acknowledgment

Parts of the work presented in this paper were supported by the Grants-in-Aid for Scientific Research from the Japanese Ministry of Education, Culture, Sports, Science and Technology (15700116, 18049035, 18700080, 19024036, 21013022, 21700103), Kayamori Information Science Foundation (K17KenX 202), and Mitsubishi Research Institute, Inc. The work is a fruit of a joint research project between Nagoya University, NetCOMPASS Ltd., and the National Institute of Informatics.

References

[1] J. Casares, B. Myers, A. Long, R. Bhatnagar, S. Stevens, L. Dabbish, and A. Corbett, "Simplifying video editing with intelligent interaction," in *Adjunct Proc. CHI2002: Human Factors in Computing Systems*, pp. 672–673, Minneapolis, MN, USA, April 2002.

[2] M. Christel, A. Hauptmann, H. Wactler, and T. Ng, "Collages as dynamic summaries for news video," in *Proc. Tenth ACM Int. Multimedia Conf.*, pp. 561–569, Juan-les-Pins, Alps-Maritimes, France, December 2002.

[3] T.-S. Chua, S.-F. Chang, L. Chaisorn, and W. Hsu, "Story boundary detection in large broadcast news video archives —Techniques, experiences, and trends," in *Proc. Twelfth ACM Int. Multimedia Conf.*, pp. 656–659, New York, NY, USA, October 2004.

[4] O. de Rooij and M. Worring, "Browsing video along multiple threads," *IEEE Trans. Multimedia*, vol. 12, no. 2, pp. 121–130, February 2010.

[5] P. Duygulu, J.Y. Pan, and D. Forsyth, "Towards auto-documentary: Tracking the evolution of news stories," in *Proc. Twelfth ACM Int. Multimedia Conf.*, pp. 820–827, New York, NY, USA, October 2004.

[6] A. Feng and J. Allan, "Hierarchical topic detection in TDT-2004," *Technical Report, University of Massachusetts, Center for Information Retrieval*, IR-389, April 2005.

[7] J.A. Fiscus and G.R. Doddington, "Topic detection and tracking evaluation overview," in J. Allan ed., *Topic Detection and Tracking: Event-based Information Organization*, Kluwer Academic Publishers, Norwell, MA, USA, 2002, pp. 17–31.

[8] A. Hauptmann, H. Witbrock, and M. Christel, "News-on-Demand: An application of Informedia technology," *D-lib Mag.*, Corporation for National Research Initiatives, Reston, VA, USA, vol. 1, issue 3, http://www.dlib.org/dlib/september95/nod/09hauptmann1.html, September 1995.

[9] I. Ide, R. Hamada, S. Sakai, and H. Tanaka, "Semantic analysis of television news captions referring to suffixes," in *Proc. Fourth Int. Workshop on Information Retrieval with Asian Languages*, pp. 37–42, Taipei, Taiwan, November 1999.

[10] I. Ide, H. Mo, N. Katayama, and S. Satoh, "Topic-based inter-video structuring of a large-scale news video corpus," in *Proc. 2003 IEEE Int. Conf. on Multimedia and Expo*, vol. 3, pp. 305–308, Baltimore, MD, USA, July 2003.

[11] I. Ide, T. Kinoshita, H. Mo, N. Katayama, and S. Satoh, "trackThem: Exploring a large-scale news video archive by tracking human relations," in G. Lee, A. Yamada, H. Meng, and S.H. Myaeng Eds., *Information Retrieval Technology —Second Asia Information Retrieval Symposium, AIRS2005, Jeju Island, Korea, October 2005, Procs., Lecture Notes in Computer Science*, Springer-Verlag, Berlin / Heidelberg, Germany, vol. 3689, pp. 510–515, October 2005.

[12] I. Ide, H. Mo, N. Katayama, and S. Satoh, "Exploiting topic thread structures in a news video archive for the semi-automatic generation of video summaries," in *Proc. 2006 IEEE Int. Conf. on Multimedia and Expo*, pp. 1473–1476, Toronto, ON, Canada, July 2006.

[13] I. Ide, T. Kinoshita, T. Takahashi, S. Satoh, and H. Murase, "mediaWalker: A video archive explorer based on time-series semantic structure," in *Proc. Fifteenth ACM Int. Multimedia Conf.*, pp. 162–163, Augusburg, Bavaria, Germany, September 2007.

[14] I. Ide, T. Kinoshita, T. Takahashi, H. Mo, N. Katayama, S. Satoh, and H. Murase, "Proposal of an information compilation method for massive news video data based on their time-series semantic structure (in Japanese)," in *Trans. JSAI*, vol. 23, no. 5, pp. 282–292, September 2008.

[15] N. Katayama, H. Mo, I. Ide, and S. Satoh, "Mining large-scale broadcast video archives towards inter-video structuring," in K. Aizawa, Y. Nakamura, and S. Satoh Eds., *Advances in Multimedia Information Processing —PCM2004, Fifth Pacific Rim Conf. on Multimedia, Tokyo, Japan, November/December 2004, Procs. Part II, Lecture Notes in Computer Science*, Springer-Verlag, Berlin / Heidelberg, Germany, vol. 3332, pp. 489–496, December 2004.

[16] R. Nallapati, A. Feng, F. Peng, and J. Allan, "Event threading within news topics," in *Proc. Thirteenth ACM Conf. on Information and Knowledge Management*, pp. 446–453, Washington, DC, USA, November 2004.

[17] A. Ogawa, T. Takahashi, I. Ide, and H. Murase, "Cross-lingual retrieval of identical news events using image information," in S. Satoh, F. Nack, and M. Etoh Eds., *Advances in Multimedia Modeling —Fourteenth Int. Multimedia Modeling Conf., MMM2008, Kyoto, Japan, January 2008, Procs., Lecture Notes in Computer Science*, Springer-Verlag, Berlin / Heidelberg, Germany, vol. 4903, pp. 287–296, January 2008.

[18] T. Okuoka, T. Takahashi, D. Deguchi, I. Ide, and H. Murase, "Labeling news topic threads with Wikipedia entries," in *Proc. Eleventh IEEE Int. Symposium on Multimedia*, pp. 501–504, San Diego, CA, USA, December 2009.

[19] M. Rautiainen, T. Ojala, and T. Seppanen, "Cluster-temporal browsing of large news video databases," in *Proc. 2004 IEEE Int. Conf. on Multimedia and Expo*, vol. 2, pp. 751–754, Taipei, Taiwan, June 2004.

[20] A. Smeaton, P. Over, and W. Kraaij, "Evaluation campaigns and TRECVid," in *Proc. Eigth ACM Int. Workshop on Multimedia Information Retrieval*, pp. 321–330, Santa Barbara, CA, USA, October 2006.

[21] N. Uramoto and K. Takeda, "A method for relating multiple newspaper articles by using graphs, and its application to webcasting," in *Proc. Thirty-sixth Annual Meeting of the Association for Computational Linguistics*, vol. 2, pp. 1307–1313, Montreal, QC, Canada, August 1998.

[22] X. Wu, C.W. Ngo, and Q. Li, "Threading and autodocumenting news videos," *IEEE Signal Processing Mag.*, vol. 23, no. 2, pp. 59–69, March 2006.

[23] C. Yang, X. Shi, and C.P. Wei, "Tracing the evolution of terror attacks from on-line news," in H. Chen, F. Wang, C. Yang, D. Zeng, and M. Chau Eds., *Intelligent and Security Informatics —IEEE Int. Conf. on Intelligent and Security Informatics, ISI2006, San Diego CA, USA, May 2006, Procs., Lecture Notes in Computer Science*, Springer-Verlag, Berlin / Heidelberg, Germany, vol. 3975, pp. 343–354, May 2006.