

HOKKAIDO UNIVERSITY

Title	A Web video retrieval method using hierarchical structure of Web video groups
Author(s)	Harakawa, Ryosuke; Ogawa, Takahiro; Haseyama, Miki
Citation	Multimedia Tools and Applications, 75, 17059-17079 https://doi.org/10.1007/s11042-015-2976-8
Issue Date	2015-10-19
Doc URL	https://hdl.handle.net/2115/63094
Rights	This is a post-peer-review, pre-copyedit version of an article published in "Multimedia Tools and Applications". The final authenticated version is available online at: http://dx.doi.org/10.1007/s11042-015-2976-8
Type	journal article
File Information	A Web video retrieval method using hierarchical structure of Web video groups.pdf



Hokkaido University Collection of Scholarly and Academic Papers : HUSCAP

Multimed Tools Appl
DOI 10.1007/s11042-015-2976-8

A Web video retrieval method using hierarchical structure of Web video groups

Ryosuke Harakawa · Takahiro Ogawa · Miki Haseyama

Received: 17 February 2015 / Revised: 10 September 2015 / Accepted: 29 September 2015

Abstract In this paper, we propose a Web video retrieval method that uses hierarchical structure of Web video groups. Existing retrieval systems require users to input suitable queries that identify the desired contents in order to accurately retrieve Web videos; however, the proposed method enables retrieval of the desired Web videos even if users cannot input the suitable queries. Specifically, we first select representative Web videos from a target video dataset by using link relationships between Web videos obtained via metadata "related videos" and heterogeneous video features. Furthermore, by using the representative Web videos, we construct a network whose nodes and edges respectively correspond to Web videos and links between these Web videos. Then Web video groups, *i.e.*, Web video sets with similar topics are hierarchically extracted based on strongly connected components, edge betweenness and modularity. By exhibiting the obtained hierarchical structure of Web video groups, users can easily grasp the overview of many Web videos. Consequently, even if users cannot write suitable queries that identify the desired contents, it becomes feasible to accurately retrieve the desired Web videos by selecting Web video groups according to the hierarchical structure. Experimental results on actual Web videos verify the effectiveness of our method.

Keywords Web video retrieval · Web video group · hierarchical structure · strongly connected component · edge betweenness · modularity

1 Introduction

According to IDC reports [9], the digital universe in 2005 was 130 exabytes and will be 300 times larger (40 zettabytes) in 2020. Also, most of the data in the digital universe are unstructured data including Web videos[1]. Therefore, more and

R. Harakawa · T. Ogawa · M. Haseyama
Graduate School of Information Science and Technology,
Hokkaido University, Sapporo, Japan
Tel.: +81-11-706-6078
Fax: +81-11-706-7369
E-mail: {harakawa, ogawa}@lmd.ist.hokudai.ac.jp, miki@ist.hokudai.ac.jp

[1] In this paper, we call video materials on the Web "Web videos".

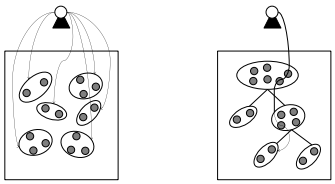
more users are retrieving Web videos to access the desired contents, *i.e.*, contents with topics that users try to find. Here, topics contained in the desired contents belong to various hierarchical levels according to each user. Given topics related to "Apple Inc." as an example, a user may desire contents with "iPhone of Apple Inc."; meanwhile, another user may desire contents with "iPhone, iPad and their related devices".

When users retrieve Web videos using existing retrieval systems such as YouTube[2], they input queries into the systems in order to narrow down the enormous amount of data to a suitable size [13]. However, it is difficult for users to write suitable queries that accurately identify the desired contents. Below, we show two examples, *i.e.*, a case when the enormous amount of data cannot be sufficiently narrowed down and another when the data is narrowed down too much. First, assume that a user desires contents with "iPhone of Apple Inc." If this user writes a query "Apple", the data cannot be narrowed down sufficiently. Second, assume that another user desires contents with "iPhone, iPad and their related devices". If this user writes a query "iPhone" or "iPad", the data is narrowed down too much. In this way, it is not necessarily easy for users to write suitable queries. In such a case, it becomes difficult for users to find the desired Web videos from the retrieval results, which are narrowed down by unsuitable queries [14].

To solve this problem, methods based on exhibition of similar Web video sets can be useful [10,16,17,31]. By using these methods, users can retrieve the desired Web videos since they can effectively browse the retrieval results via the similar Web video sets. In an article [31], a clustering method based on visual features for near-duplicate video retrieval was proposed. A work [10] proposed a clustering method that uses co-watching of Web videos and textual features of Web videos, *i.e.*, features of text attached to Web videos such as title and description. In a study [17], a clustering method of retrieval results based on visual and textual features of Web videos was proposed. Also, a paper [16] proposed a retrieval method that provides similar Web video sets on the basis of heterogeneous video features, *i.e.*, visual, audio and textual features and link relationships between Web videos. However, these clustering-based methods cannot provide results including topics at various hierarchical levels. Remember the above example, *i.e.*, a user that desires contents with "iPhone, iPad and their related devices". Here, if the obtained clusters include topics at the upper level, *e.g.*, "Apple", a user still needs to find the desired contents from the clusters. On the other hand, if the obtained clusters include topics at the lower level, *e.g.*, "iPhone", the desired contents may not exist in the clusters.

To overcome this difficulty, hierarchical clustering-based retrieval methods that can provide results including topics at various hierarchical levels are necessary [7, 23,28,30]. If hierarchical structure of Web video sets, which includes many topics, is provided, the hierarchical structure can navigate users to the desired contents (see Fig. 1). A work [28] proposed a Web video retrieval method based on exhibition of hierarchical topic structure obtained by using textual analysis and WordNet [20]. In a paper [23], a hierarchical scheme that first clusters video shots into ones with similar color and then clusters video shots into ones with similar motion to retrieve videos about sports games. The method in [7] constructs the hierarchical tree structure of semantic-sensitive video classifier for indexing and

[2] http://www.youtube.com



(a) (b)

Fig. 1 Schemes to provide Web video sets. (a) Non-hierarchical scheme: If topics contained in the Web video sets and those contained in the desired contents do not belong to the same hierarchical levels, retrieval of the desired contents becomes difficult. (b) Hierarchical scheme: The hierarchical structure can navigate users to the desired contents.

effective video access. The method in [30] performs clustering of the similar video shots in a hierarchical fashion for browsing. Although these methods can overcome the above difficulty, they utilize only a single modality, *i.e.*, textual features or visual features. Therefore, these conventional methods have the limitations in their retrieval performance due to the lack of useful information obtained from the other modalities.

Hence, the above conventional methods have the following two problems:

(Problem-i) The retrieval performance may be limited since only a single modality is used. This is the problem in the earlier works [7, 23, 28, 30].
(Problem-ii) Web video retrieval is performed without the use of the hierarchical structure of similar Web video sets. This is the problem in the conventional studies [10, 16, 17, 31].

Therefore, in this paper, we propose a Web video retrieval method that uses hierarchical structure of Web video groups. In this paper, Web video groups are defined as Web video sets with similar topics. Also, their hierarchical structure denotes the property of Web video groups being divided into the sub-groups in this paper. Our contribution of this paper is the followings:

(Contribution-i) Hierarchical structure of Web video groups is estimated by using link relationships obtained via metadata "related videos" and heterogeneous video features, *i.e.*, visual, audio and textual features.
(Contribution-ii) Web video retrieval is performed by exhibiting the hierarchical structure to users.

In (Contribution-i), we analyze a network constructed by using link relationships and heterogeneous video features on the basis of strongly connected components (SCCs) [5], edge betweenness [22] and modularity [2] to successfully estimate the hierarchical structure. In (Contribution-ii), even if users cannot write suitable queries, the hierarchical structure can navigate users to the desired contents. Finally, we note that this paper is an extended version of [12][3].

This paper is organized as follows. In Sec. 2, an outline of the proposed Web video retrieval method is presented. In Sec. 3, a method for selecting representative Web videos based on Web video features is described. A method for extracting the hierarchical structure of Web video groups is explained in Sec. 4. A Web

[3] In this paper, the method in [12] is improved by automatically deciding the number of hierarchies exhibited to users on the basis of modularity.

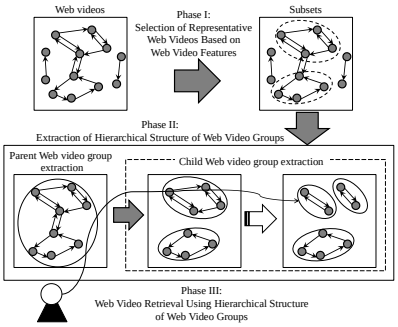


Fig. 2 Outline of the proposed Web video retrieval method using hierarchical structure of Web video groups.

video retrieval method that uses the hierarchical structure is presented in Sec. 5. In Sec. 6, experimental results on actual Web videos are shown to verify the effectiveness of our method. Limitation and future work of this paper is described in Sec. 7. Concluding remarks are given in Sec. 8.

2 Outline of Web Video Retrieval Using Hierarchical Structure of Web video groups

In this section, an outline of the proposed Web video retrieval method using hierarchical structure of Web video groups is presented. Figure 2 shows an outline of the proposed method. As shown in the figure, our method consists of three phases. An overview of them is shown below.

Phase I: Selection of Representative Web Videos Based on Web Video Features
This phase is useful for screening irrelevant contents from many Web videos. As a result, this phase enables both reduction of computational cost in Phase II and accurate retrieval of users' desired contents in Phase III.

Phase II: Extraction of Hierarchical Structure of Web Video Groups
This phase aims to provide hierarchical structure of Web video groups related to a given query. By providing the hierarchical structure, it becomes feasible for users to easily find the desired contents.

Phase III: Web Video Retrieval Using Hierarchical Structure of Web Video Groups
This phase enables to rank Web videos in each Web video group. Thus, users can retrieve Web videos in the descending order of the centrality in the selected Web video group.

The details of Phases I, II and III are presented in Secs. 3, 4 and 5, respectively.

3 Selection of Representative Web Videos Based on Web Video Features (Phase I)

In this section, a method for selecting representative Web videos based on Web video features is explained.

3.1 Definition of Web Video Features

In this subsection, we explain the definition of Web video features used in this paper. In this paper, Web video features are calculated from Web videos f_i ($i = 1, 2, \cdots, N$; N being the number of Web videos) according to a paper [16]. First, we apply the shot segmentation method [21] to each Web video f_i, and obtain several shots $s_i^{q_i}$ ($q_i = 1, 2, \cdots, M_i$; M_i being the number of shots within f_i). Then a visual feature vector $\boldsymbol{v}_i^{q_i}$, an audio feature vector $\boldsymbol{a}_i^{q_i}$ and a textual feature vector $\boldsymbol{t}_i^{q_i}$ are calculated from each shot $s_i^{q_i}$ as follows.

Visual Feature Vector (p dimensions)
We calculate the HSV color histogram with p bins every P_v frames of a Web video f_i to obtain its vector. For the frames in each shot $s_i^{q_i}$, the vector median [3] is calculated to obtain $\boldsymbol{v}_i^{q_i}$ ($= [v_i^{q_i}(1), v_i^{q_i}(2), \cdots, v_i^{q_i}(p)]^{\mathrm{T}}$). As a result, we obtain M_i visual feature vectors $\boldsymbol{v}_i^{q_i}$ ($q_i = 1, 2, \cdots, M_i$) from each Web video f_i.

Audio Feature Vector (22 dimensions)
Based on a study [25], we calculate 22 dimensional audio feature vectors $\boldsymbol{a}_i^{q_i}$ that are useful for audio signal classification for each shot $s_i^{q_i}$. For more details, refer to a paper [16].

Textual Feature Vector (r dimensions)
Let K be the total number of keywords that appear in text attached to Web videos f_i ($i = 1, 2, \cdots, N$). We apply TF-IDF [29] to the K keywords, and calculate the vector $\boldsymbol{\eta}_i$ ($= [\eta_i(1), \eta_i(2), \cdots, \eta_i(K)]^{\mathrm{T}}$) by aligning the obtained weights. Furthermore, we apply principal component analysis (PCA) to $\boldsymbol{\eta}_i$ ($i = 1, 2, \cdots, N$) to reduce their dimensions, and obtain new r-dimensional vectors $\boldsymbol{t}_i$. Thereby, for each shot $s_i^{q_i}$ in Web video f_i, we obtain M_i textual feature vectors $\boldsymbol{t}_i^{q_i}$ ($= \boldsymbol{t}_i$) ($q_i = 1, 2, \cdots, M_i$).

Thus, from each shot $s_i^{q_i}$ in the Web video f_i, the visual feature vector $\boldsymbol{v}_i^{q_i}$, the audio feature vector $\boldsymbol{a}_i^{q_i}$ and the textual feature vector $\boldsymbol{t}_i^{q_i}$ are calculated.

3.2 Selection of Representative Web Videos

In this subsection, a method for selecting representative Web videos by using the three kinds of features and link relationships between Web videos is explained. First, in order to obtain variates that can be compared between the different kinds of features, canonical correlation analysis (CCA) [24] is applied to $\boldsymbol{v}_i^{q_i}$, $\boldsymbol{a}_i^{q_i}$ and $\boldsymbol{t}_i^{q_i}$. Then we obtain $\boldsymbol{\zeta}^l_{\boldsymbol{v}_i^{q_i}}, \boldsymbol{\zeta}^l_{\boldsymbol{a}_i^{q_i}}$ and $\boldsymbol{\zeta}^l_{\boldsymbol{t}_i^{q_i}}$, which are the projection results of $\boldsymbol{v}_i^{q_i}, \boldsymbol{a}_i^{q_i}$ and $\boldsymbol{t}_i^{q_i}$ into the latent space of l ($\in \{v, a, t\}$), where v, a and t correspond to visual, audio and textual features, respectively. In the experiments shown later, the three kinds of feature vectors were projected into the latent space of textual features by setting $l = t$. For more details of this calculation, refer to a paper [16].

Next, by using $\boldsymbol{\zeta}^l_{\boldsymbol{v}_i^{q_i}}, \boldsymbol{\zeta}^l_{\boldsymbol{a}_i^{q_i}}$ and $\boldsymbol{\zeta}^l_{\boldsymbol{t}_i^{q_i}}$, we calculate similarities $S(i, j)$ between Web videos f_i and f_j by the following equations:

$$S(i,j) = \max_{q_i, q_j} \left| \frac{(\boldsymbol{\xi}_i^{q_i})^{\mathrm{T}} \boldsymbol{\xi}_j^{q_j}}{\|\boldsymbol{\xi}_i^{q_i}\| \|\boldsymbol{\xi}_j^{q_j}\|} \right|, \tag{1}$$

$$\boldsymbol{\xi}_i^{q_i} = \left[(\boldsymbol{\zeta}^l_{\boldsymbol{v}_i^{q_i}})^{\mathrm{T}}, (\boldsymbol{\zeta}^l_{\boldsymbol{a}_i^{q_i}})^{\mathrm{T}}, (\boldsymbol{\zeta}^l_{\boldsymbol{t}_i^{q_i}})^{\mathrm{T}} \right]^{\mathrm{T}}. \tag{2}$$

In this way, we calculate the similarities between Web videos by integrating the heterogeneous video features on the basis of CCA.

Furthermore, we weight the adjacency matrix that represents link relationships between Web videos and calculate the weighted adjacency matrix $\boldsymbol{L}_w$ whose (i,j)-th element $L_w(i,j)$ is as follows:

$$L_w(i,j) = \begin{cases} S(i,j) & \text{if } f_i \text{ links to } f_j \\ 0 & \text{otherwise} \end{cases}, \tag{3}$$

where $i = 1, 2, \cdots, N$ and $j = 1, 2, \cdots, N$. In the above equation, we build link relationships between Web videos based on metadata "related videos". In particular, we consider that a Web video f_i links to a Web video f_j if "related videos" of f_i include f_j. Since the metadata "related videos" are useful for obtaining Web videos related to each other [6], we introduce them into our method.

Moreover, we calculate N_g eigenvectors of $\boldsymbol{L}_w^{\mathrm{T}}\boldsymbol{L}_w$, where N_g is the number of the calculated eigenvectors. Each element of the eigenvectors of $\boldsymbol{L}_w^{\mathrm{T}}\boldsymbol{L}_w$ represents the attribution degree of each Web video to sets of Web videos with similar contents [19]. Next, we select N_m Web videos in descending order of the element values of each eigenvector, and represent these Web videos as C_q $(q = 1, 2, \cdots, N_g)$. In this paper, Web videos contained in C_q are called a "subset". By extracting the subset, we can obtain representative Web videos in the network of link relationships between Web videos since Web videos that densely link to each other can be extracted [19]. Thus, we can filter out irrelevant Web videos from many ones.

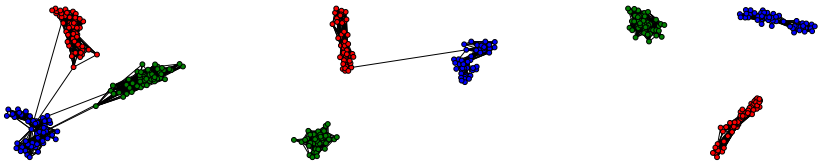
4 Extraction of Hierarchical Structure of Web Video Groups (Phase II)

In this section, a method for extracting hierarchical structure of Web video groups is explained. In our method, we summarize Web videos related to each other as Web video groups based on SCCs [5] and extract their hierarchical structure by using edge betweenness [22] and modularity [2]. Since SCCs are useful to find Web videos related to each other in the network of link relationships between Web videos, we introduce this measure to obtain "parent Web video groups". Moreover, a study [22] reports that the combination use of edge betweenness and modularity enables extraction of the significant hierarchical structure. Therefore, we adopt these two measures to extract the hierarchical structure as in [22]. Specifically, we extract the hierarchical structure by repeatedly dividing parent Web video groups into their "child Web video groups"[4] on the basis of edge betweenness and modularity. In this paper, we collectively call parent Web video groups and child Web video groups "Web video groups". Next, we explain the details of this scheme.

4.1 Construction of Parent Web Video Groups

First, from Web videos that belong to the subset C_q $(q = 1, 2, \cdots, N_g)$, a weighted and directed network $G = (V, E)$, whose nodes and edges are respectively Web

[4] When a child Web video group is divided into its child Web video groups, this child Web video group is considered to be the parent Web video group of its child Web video groups.



(a) $N_{cut} = 0$ (b) $N_{cut} = 15$ (c) $N_{cut} = 30$

Fig. 3 Simple example of Web video groups. (a) A parent Web video group is formed. (b) Two child Web video groups are formed by dividing their parent Web video group. (c) Three child Web video groups are generated from their parent Web video group.

videos and links between these videos, is constructed. When a Web video $f_i \in V$ links to $f_j \in V$, the edge weight e_{ij} from f_i to f_j is defined as follows:

$$e_{ij} = L_w(i, j). \tag{4}$$

Thus, the edge weight represents the similarity between Web videos. When $f_i \in V$ does not link to $f_j \in V$, we do not define the edge e_{ij}. In our method, parent Web video groups are obtained by extracting SCCs. SCCs are sub-networks that have a directed path between any two nodes in a directed network. A method for extracting SCCs from a network is called SCC decomposition. In a study [5], related Web pages are grouped on the basis of SCC decomposition. In the proposed method, by applying SCC decomposition to G, all SCCs are extracted. Then these obtained components are defined as parent Web video groups. By extracting parent Web video groups, we can obtain Web video sets with similar topics.

4.2 Extraction of Child Web Video Groups

Next, we extract child Web video groups by repeatedly dividing the parent Web video groups on the basis of edge betweenness. Edge betweenness is used for detecting communities in a network [22]. Edge betweenness is defined by the number of the shortest paths that go through a target edge, and it is useful for detecting influential edges in a network. Let $c_B(e)$ be edge betweenness of an edge $e \in E$, and $c_B(e)$ is defined as follows:

$$c_B(e) = \sum_{s,t \in V} \frac{\sigma(s, t|e)}{\sigma(s, t)}, \qquad e \in E, \tag{5}$$

where $\sigma(s, t|e)$ is the number of shortest paths from node s to node t that pass an edge e, and $\sigma(s, t)$ is the total number of shortest paths from node s to node t. In our method, by iteratively removing edges with the largest edge betweenness in G and re-applying SCC decomposition to G, we re-extract SCCs. In this paper, we denote the number of these iterations by N_{cut}. Here, child Web video groups are extracted when a single SCC is divided into several SCCs (see Fig. 3). By repeatedly dividing the parent Web video groups into their child Web video groups, the hierarchical structure of the Web video groups is obtained. Note that each child

Web video group becomes equivalent to each Web video when all edges in G are removed. Since it is not effective for Web video retrieval to provide all hierarchies to users, we have to decide when our method stops dividing parent Web video groups into their child Web video groups.

4.3 Introduction of Modularity

To solve the above problem, we use a quality function called modularity, which evaluates the results of division of communities in a network. Note that the node set V of $G = (V, E)$ does not contain all Web videos f_i $(i = 1, 2, \cdots, N)$ since G is constructed by Web videos that belong to the extracted subset C_q $(q = 1, 2, \cdots, N_g)$. Therefore, in order to calculate modularity of G, we define the matrix $\boldsymbol{M} = (m_{ij})$ as follows:

$$m_{ij} = \begin{cases} L_w(i,j) & \text{if } f_i \in V \text{ and } f_j \in V \\ 0 & \text{otherwise,} \end{cases} \tag{6}$$

where $i = 1, 2, \cdots, N$ and $j = 1, 2, \cdots, N$. Let $Q_{N_{cut}}$ be modularity of G where the number of cut edges is N_{cut}, and $Q_{N_{cut}}$ is defined as follows [2]:

$$Q_{N_{cut}} = \frac{1}{2m} \sum_{i=1}^{N} \sum_{j=1}^{N} (m_{ij} - \frac{m_i^{out} m_j^{in}}{2m})\ \delta(i,j), \tag{7}$$

where $2m = \sum_{i=1}^{N} \sum_{j=1}^{N} m_{ij}$, $m_i^{out} = \sum_{j=1}^{N} m_{ij}$, $m_j^{in} = \sum_{i=1}^{N} m_{ij}$ and $\delta(i,j)$ is 1 if f_i and f_j belong to the same Web video group and 0 otherwise. When significant community structure is revealed, $Q_{N_{cut}}$ becomes close to 1, *i.e.*, the maximum value of $Q_{N_{cut}}$. On the other hand, a network is divided into communities randomly, and $Q_{N_{cut}}$ becomes close to 0. In this paper, the N_{cut} value when $Q_{N_{cut}}$ is maximum is denoted by N_{cut}^{opt}. The proposed method uses the Web video groups, where N_{cut} is from 0 to N_{cut}^{opt} for Web video retrieval.

In this way, we hierarchically extract Web video groups based on SCCs, edge betweenness and modularity. Consequently, the entire contents of many Web videos can be easily grasped by obtaining the hierarchical structure of Web video groups. Algorithm 1 shows the specific procedures for extracting the hierarchical structure.

5 Web Video Retrieval Using Hierarchical Structure of Web Video Groups (Phase III)

In this section, a Web video retrieval method using the hierarchical structure of Web video groups is presented. We obtain Web video groups where N_{cut} is from 0 to N_{cut}^{opt}. In this paper, these Web video groups are represented as $Com_{N_{cut}}^{q}$ $(N_{cut} = 0, 1, \cdots, N_{cut}^{opt},\ q = 1, 2, \cdots, T_{N_{cut}}; T_{N_{cut}}$ being the number of Web video groups in which the number of cut edges is $N_{cut})$. First, each Web video f_i $(i \in \{1, 2, \cdots, N\})$

Algorithm 1 : Extraction of Hierarchical structure of Web video groups.

Input: A weighted and directed network $G = (V, E)$.
Output: Parent and child Web video groups.

$N_{cut} \leftarrow 0$
Apply SCC decomposition to G, and extract parent Web video groups.
Calculate modularity $Q_{N_{cut}}$.
while an edge of G remains **do**
  Calculate edge betweenness of remaining edges in G.
  if only an edge has the largest edge betweenness **then**
    Remove the edge with the largest edge betweenness.
  else
    Select one edge from the edges with the largest edge betweenness randomly, and remove the edge.
  end if
  $N_{cut} \leftarrow N_{cut} + 1$
  Apply SCC decomposition to G.
  if the number of SCCs is changed **then**
    Extract child Web video groups.
  end if
  Calculate modularity $Q_{N_{cut}}$.
end while
$N_{cut}^{opt} \leftarrow \arg\max_{N_{cut}} Q_{N_{cut}}$
Return parent and child Web video groups, where N_{cut} is from 0 to N_{cut}^{opt}.

that belongs to the Web video group $Com_{N_{cut}}^{q}$ is ranked in descending order of the following criterion $R_{N_{cut}}^{q}(i)$:

$$R_{N_{cut}}^{q}(i) = \sum_{j=1}^{N} m_{ji}\ \delta(i,j), \tag{8}$$

where $\delta(i,j)$ is 1 if f_i and f_j belong to the same Web video group and 0 otherwise. Hence, Web videos belonging to each Web video group are ranked on the basis of weighted links from other Web videos in the same Web video group.

Next, we exhibit the Web video groups $Com_{N_{cut}}^{q}$, where N_{cut} is from 0 to N_{cut}^{opt}. Here, the smaller N_{cut} is, the more various topics are contained in the Web video groups. On the other hand, the larger N_{cut} is, the more closely related Web videos are contained in the Web video groups. Then users select Web video groups associated with the desired contents according to the hierarchical structure. Furthermore, users retrieve the desired Web videos from the selected Web video group based on the criterion $R_{N_{cut}}^{q}(i)$. In this way, even if users cannot write suitable queries that identify the desired contents, the hierarchical structure can navigate users to the desired contents.

6 Experimental Results

In this section, the effectiveness of the proposed Web video retrieval method is verified.

Table 1 Details of datasets: K is the total number of keywords that appear in text attached to Web videos, N_g and N_m are the parameters for constructing the subsets. To compute visual feature vectors, p and P_v were set to 48 ($H = 12, S = 2, V = 2$) and 1, respectively. To extract textual features, we used title, description and comments attached to Web videos.

	Query keyword	Num. of Web videos in the dataset	K	N_g	N_m	Num. of Web videos in the subset
Dataset 1	apple	3037	76498	15	70	775
Dataset 2	galaxy	3029	87778	15	70	661
Dataset 3	jaguar	3043	88396	15	70	574
Dataset 4	sightseeing	3044	93648	15	70	762
Dataset 5	match	1064	31022	15	70	487
Dataset 6	game	1053	27588	15	70	649
Dataset 7	race	1049	33670	15	70	556
Dataset 8	Hokkaido	1053	35903	15	70	521
Dataset 9	music (in Japanese)	1037	30173	15	70	586
Dataset 10	comedy (in Japanese)	1068	19742	15	70	529

Table 2 N_{cut}^{opt} and maximum value of $Q_{N_{cut}}$ for each subset.

	N_{cut}^{opt}	Maximum value of $Q_{N_{cut}}$
Subset 1	499	0.789
Subset 2	107	0.674
Subset 3	131	0.706
Subset 4	53	0.836
Subset 5	96	0.777
Subset 6	5	0.826
Subset 7	15	0.774
Subset 8	125	0.768
Subset 9	313	0.779
Subset 10	594	0.729

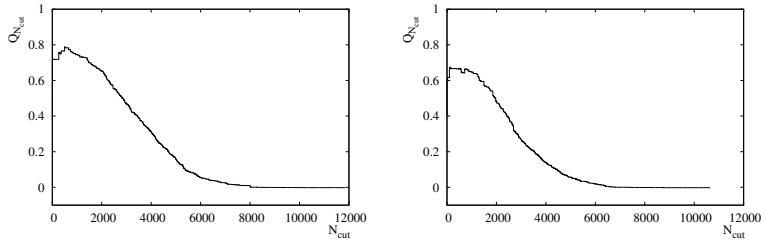
6.1 Datasets

In the proposed method, metadata "related videos" on each Web video plays an important role to extract hierarchical structure for Web video retrieval. However, standard video retrieval datasets do not have the metadata. Therefore, we constructed datasets by our own for performing a fair comparison as follows. First, by using YouTube[2], a keyword was given as a query, and the top 50 Web videos were obtained. Then we repeatedly obtained 10 Web videos contained in links, *i.e.*, "related videos"[5] of the selected Web videos, and each dataset was constructed. Here, we collected only Web videos with lengths of less than 1800 seconds. Table 1 shows the detailed conditions of each dataset.

6.2 Evaluation

From each dataset, subsets were extracted according to Sec. 3 (Phase I). Next, from each subset, we extracted the parent Web video groups and the child Web video groups according to Sec. 4 (Phase II). In Table 2, N_{cut}^{opt} and the maximum

[5] In the experiment, we used YouTube Data API v2 to obtain the links, "related videos".



(a) Result for subset 1. (b) Result for subset 2.

Fig. 4 Change in modularity $Q_{N_{cut}}$ to the number of cut edges, N_{cut}.

Table 3 Relevant Web videos used in the experiment.

	Relevant Web videos
Dataset 1	Web videos about "Product of Apple Inc., iPhone"
Dataset 2	Web videos about "Tablet of Samsung Electronics Co."
Dataset 3	Web videos about "Automobile of Jaguar Cars"
Dataset 4	Web videos about "Japan"
Dataset 5	Web videos about "Soccer game"
Dataset 6	Web videos about "Video game of Minecraft"
Dataset 7	Web videos about "Vehicle"
Dataset 8	Web videos about "Winter sport"
Dataset 9	Web videos about "Wind music"
Dataset 10	Web videos about "Japanese comedian U"

value of $Q_{N_{cut}}$ of each subset are shown, and Fig. 4 shows the change in modularity $Q_{N_{cut}}$ to N_{cut} for subsets 1 and 2. Figure 5 shows the hierarchical structure of the largest parent Web video groups of subsets 1 and 2. This figure shows that the more closely related Web videos are obtained with an increase in the number of cut edges, N_{cut}. Thus, it can be seen that the hierarchical structure of Web video groups can be estimated by our method.

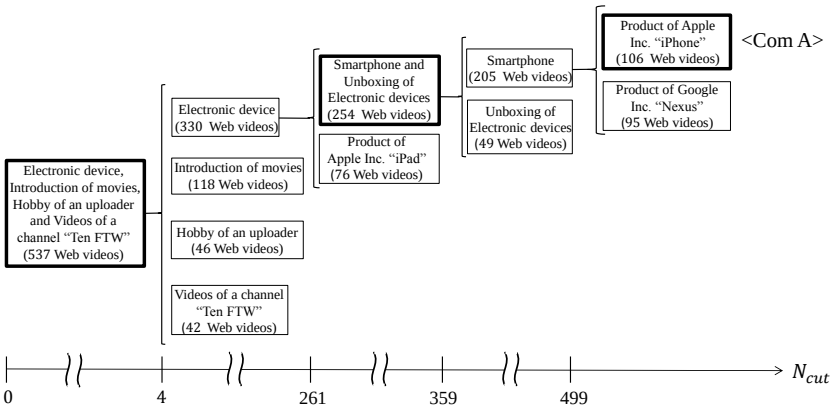
Next, we quantitatively verify the effectiveness of the proposed Web video retrieval method shown in Sec. 5 (Phase III). We used recall, precision and average precision (AP@k) defined as follows:

$$\text{Recall} = \frac{\text{Num. of correctly retrieved Web videos}}{\text{Num. of relevant Web videos}}, \tag{9}$$

$$\text{Precision} = \frac{\text{Num. of correctly retrieved Web videos}}{\text{Num. of retrieved Web videos}}, \tag{10}$$

$$\text{AP@}k = \frac{1}{R_k} \sum_{i=1}^{k} x_i \; prec_i, \tag{11}$$

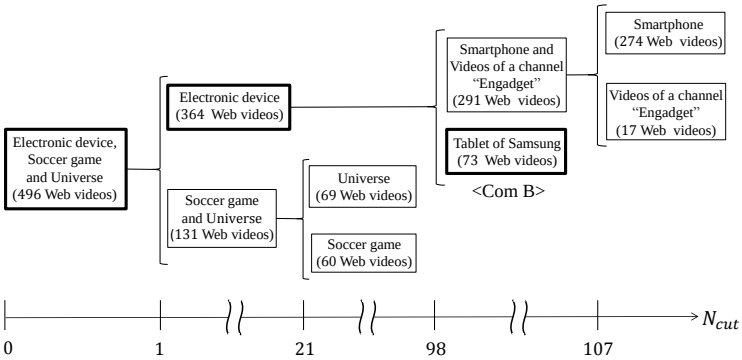
where k is the number of Web videos provided as the retrieval results, R_k is the number of "relevant Web videos" within k Web videos of the retrieval results, x_i is 1 if the i-th retrieved Web videos are "relevant Web videos" and 0 otherwise, and $prec_i$ is the precision when i Web videos are retrieved. Table 3 shows relevant Web



(a) Hierarchical structure of the largest parent Web video group of subset 1.



(b) Thumbnails of the top 10 retrieved Web videos in the order of left-to-right when "Com A" was selected.



(c) Hierarchical structure of the largest parent Web video group of subset 2.



(d) Thumbnails of the top 10 retrieved Web videos in the order of left-to-right when "Com B" was selected.

Fig. 5 Hierarchical structure of Web video groups obtained by the proposed method. Squares with thick lines show Web video groups used in calculating recall and precision in Fig. 6. We show only Web video groups containing more than 10 Web videos.

videos used in this experiment. We compare the retrieval results of the proposed method with the following reference methods.

Reference method (i)

This is a method that extracts hierarchical structure of Web video groups by using only link relationships between Web videos without the use of Web video features.

Reference method (ii)
This is a conventional method [16] that extracts Web video groups by using link relationships between Web videos and the Web video features. This method does not extract the hierarchical structure.

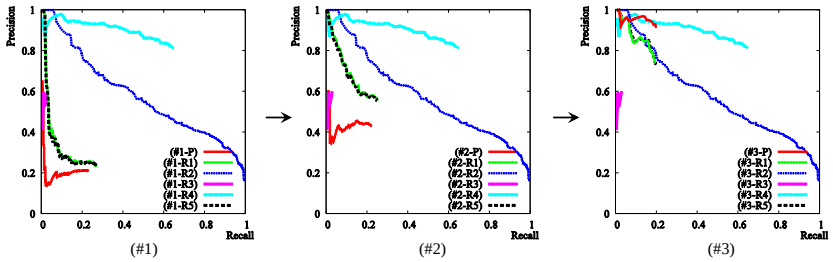
Reference method (iii)
This is a method based on [17] with a Web video group extraction scheme by affinity propagation [8]. This method uses only Web video similarities without the use of link relationships between Web videos, and does not extract the hierarchical structure. In the experiment, we did not use the original similarities presented by [17] but utilized our proposed similarities defined in Eq. (1). For the evaluation, we ranked Web videos within Web video groups obtained by this method in descending order of the sum of the above similarities.

Reference method (iv)
This is a method based on clustering via latent semantic indexing (LSI) that uses only textual features of Web videos [18], which does not exhibit hierarchical structure to users. Note that an original method in [18] utilizes information of YouTube Playlists; however, we used TF-IDF vectors obtained in the same manner as our method in this experiment. Moreover, the number of dimensions of TF-IDF vectors after LSI were set to 200 as in a paper [18]. Each Web video is ranked on the basis of the attribution degree to the belonging cluster.

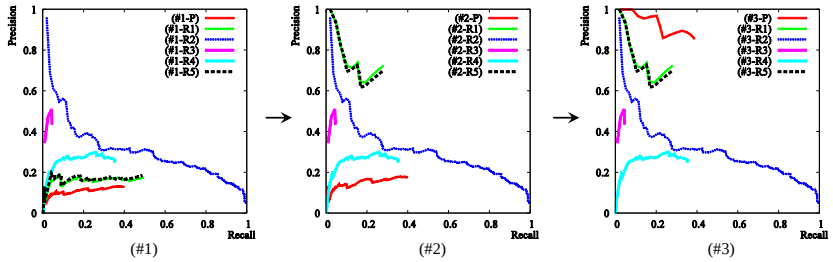
Reference method (v)
This is a method constructed by replacing our CCA-based video features by Non-negative matrix factorization (NMF)-based ones [11]. A paper [11] claims that coefficient vectors obtained by applying NMF to textual features are effective for building a latent semantic image representation. In this experiment, we used textual feature vectors $\boldsymbol{t}_i$ $(i = 1, 2, \cdots, N)$ defined in Sec. 3.1. Note that we added constant values to all elements of the textual feature vectors to keep them non-negative. Then we implemented NMF by using the obtained vectors and extracted hierarchical structure of Web video groups in the same manner as our method. Note that we set the number of dimensions of vectors after performing NMF to 1500 and 500 for datasets 1-4 and datasets 5-10, respectively.

For all methods, we selected a Web video group that includes relevant Web videos the most, and Web videos were retrieved according to the ranking of Web videos. Then recall, precision and AP@k were calculated. Figure 6 shows recall and precision calculated for datasets 1 and 2. Note that several results per dataset are shown since we calculated recall and precision at several hierarchies in the proposed method and reference methods (i) and (v). As for the proposed method and reference methods (i) and (v), the larger N_{cut} is, the higher the precision is. This means that closely related Web videos are obtained by dividing the parent Web video group into their child Web video groups.

Moreover, Table 4 and Fig. 7 respectively show AP@k and the precision-recall curves for all datasets. From these results, we can see that our method has better performance than all reference methods for most datasets. Specifically, the effectiveness of using the heterogeneous video features can be seen by comparing our method with reference methods (i), (iv) and (v). When comparing our method with reference methods (ii), (iii) and (iv), we can see that it becomes feasible to accurately retrieve the desired Web videos by using the hierarchical structure of



(a) Results for dataset 1: (#1-P), (#2-P), (#3-P) are results of the proposed method where N_{cut} are $0, 265$ and 500, respectively. (#1-R1), (#2-R1), (#3-R1) are results of reference method (i) where N_{cut} are $0, 265$ and 500, respectively. (R2), (R3) and (R4) show results of reference methods (ii), (iii) and (iv). (#1-R5), (#2-R5), (#3-R5) are results of reference method (v) where N_{cut} are $0, 265$ and 500, respectively.



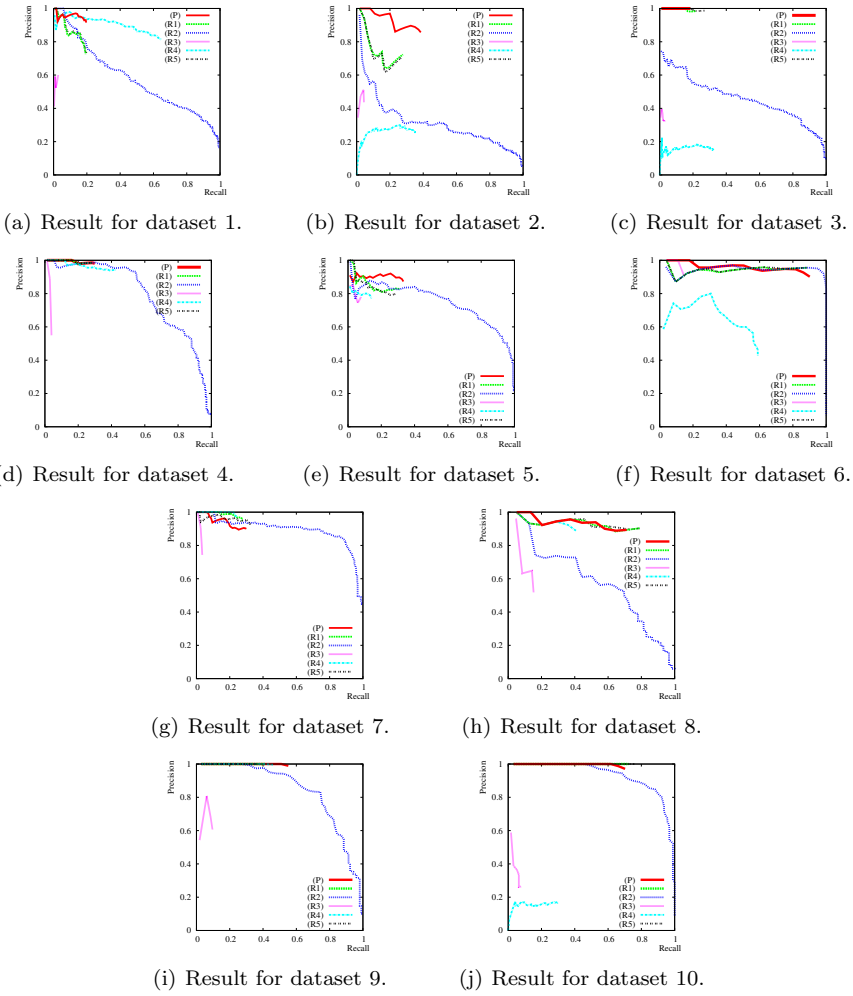
(b) Results for dataset 2: (#1-P), (#2-P), (#3-P) are results of the proposed method where N_{cut} is $0, 50$ and 100, respectively. (#1-R1), (#2-R1), (#3-R1) are results of reference method (i) where N_{cut} is $0, 50$ and 100, respectively. (R2), (R3) and (R4) show results of reference methods (ii), (iii) and (iv). (#1-R5), (#2-R5), (#3-R5) are results of reference method (v) where N_{cut} is $0, 50$ and 100, respectively.

Fig. 6 Precision-recall curves.

Table 4 Average precision (AP@k): For the proposed method and reference methods (i) and (v), the results where N_{cut} are N_{cut}^{opt} are shown. We defined the value of k as the number of Web videos in each Web video group used for retrieval. Maximum values in each row are highlighted with asterisks.

	Proposed method	Reference method (i)	Reference method (ii)	Reference method (iii)	Reference method (iv)	Reference method (v)
Dataset 1	0.958*	0.884	0.580	0.580	0.911	0.885
Dataset 2	0.937*	0.767	0.321	0.512	0.265	0.760
Dataset 3	1.00*	0.996	0.462	0.406	0.163	0.996
Dataset 4	0.993*	0.993*	0.793	0.905	0.970	0.993*
Dataset 5	0.906*	0.864	0.751	0.817	0.816	0.860
Dataset 6	0.962	0.945	0.947	0.992*	0.692	0.946
Dataset 7	0.948	0.991	0.893	0.929	0.999*	0.964
Dataset 8	0.944	0.935	0.590	0.812	0.956*	0.936
Dataset 9	0.999	1.00*	0.842	0.738	1.00*	1.00*
Dataset 10	0.998	1.00*	0.934	0.474	0.162	1.00*

Web video groups. In particular, accurate retrieval can be realized since Web video groups are refined into more similar topics according to the hierarchical structure. From the results by our method and reference methods (iii) and (iv), it can be con-



(a) Result for dataset 1. (b) Result for dataset 2. (c) Result for dataset 3.

(d) Result for dataset 4. (e) Result for dataset 5. (f) Result for dataset 6.

(g) Result for dataset 7. (h) Result for dataset 8.

(i) Result for dataset 9. (j) Result for dataset 10.

Fig. 7 Precision-recall curves: (P), (R1) and (R5) respectively correspond to our method and reference methods (i) and (v), where N_{cut} are N_{cut}^{opt}. (R2), (R3) and (R4) correspond to reference methods (ii), (iii) and (iv), respectively. Note that reference method (ii) has higher recall than other methods since only this method is not based on hard clustering and all Web videos can belong to every Web video group.However, we consider our method that has higher precision is significant since it is desirable that users can retrieve the desired contents by browsing a small number of retrieval results.

firmed that link relationships between Web videos are useful for obtaining similar Web videos accurately whereas the only use of the video features is insufficient. On the other hand, precision (Fig. 7) of our method is less than reference methods (i), (ii), (iv) and (v) for dataset 7. We guess this is because those relevant Web videos can be simply expressed by using only link relationships or textual features without the use of heterogeneous video features. To solve such a problem, in the future, we need to develop new schemes to select the effective features for retrieval of the desired contents.

7 Limitation and Future Work

In this section, we discuss limitation and future work of this paper.

7.1 Discussion on a User Interface

In this paper, we name each Web video group in the hierarchies shown in Fig. 5 by checking contents in Web video groups manually. In the future, we should develop a scheme for automatically naming the hierarchies. We consider it is necessary to reveal salient keywords in each Web video group and estimate hierarchical relationships between the keywords [4, 20].

Moreover, we explain how navigation is presented to a user for Web video retrieval. In Fig. 8, we show an example of a user interface constructed on the basis of a network drawing algorithm [15]. As shown in this figure, for a given query, we exhibit hierarchical structure of Web video groups to a user via the interface. When a user selects a Web video group, the user can grasp the overview of the selected Web video group via the visualized network and can retrieve each Web video in descending order of the rankings based on Eq. (8). In (b), (c) and (d) in Fig. 8, the left sides of each figure depict the networks visualized on the basis of an algorithm [15], and the right sides show rankings of each Web video in the selected Web video group. Then, until the desired contents are retrieved, a user repeatedly selects Web video groups in several hierarchical levels by browsing names of each Web video group.

In general, topics contained in the desired contents belong to various hierarchical levels according to each user. In our method, the exhibition of the hierarchical structure via a user interface enables navigation to the desired contents.

7.2 Discussion on Computational Cost

Here, we discuss computational cost of our method with regard a real-world deployment. In our method, computational cost of Web video similarities based on CCA is $O(N^3)$ and that of extracting the hierarchical structure is $O(M_s^2 N_s)$, where M_s and N_s are the numbers of edges and nodes in the target subset, respectively [1,22]. Also, we need to recalculate them when a new query is given by a user and new Web videos are added or deleted. Since their recalculation should be avoided for a real-world deployment, we should develop schemes to efficiently update Web video similarities and the hierarchical structure. To update Web video similarities, incremental schemes will be useful [27]. On the other hand, to keep the hierarchical structure updated and select the best N_{cut}, *i.e.*, N_{cut}^{opt}, a dynamic network analysis algorithms [26] will be necessary. In the future, we should introduce them into our method for a real-world deployment.



Fig. 8 Example of a user interface for dataset 2. This interface is built based on a network drawing algorithm [15]. (a): This interface exhibits the hierarchical structure for a given query "galaxy" to a user. (b), (c) and (d): This interface respectively exhibits details of Web video groups "Tablet of Samsung", "Universe" and "Soccer game" in Fig. 5 when a user selects them from the hierarchical structure.

8 Conclusions

In this paper, we have proposed a Web video retrieval method using hierarchical structure of Web video groups. The proposed method enabled retrieval of the

desired Web videos even if users cannot write suitable queries that identify the desired contents by the two contributions, *i.e.*, extraction of hierarchical structure of Web video groups, and Web video retrieval that uses the hierarchical structure. In the first contribution, the combination use of link relationships between Web videos and heterogeneous video features can overcome the limitation in the retrieval performance when only a single modality is used. In the second contribution, even if users cannot write suitable queries, the hierarchical structure enabled navigation to the desired contents. Experimental results have verified the effectiveness of the proposed Web video retrieval method. Future work includes developing a method for automatically naming each Web video group in the hierarchical structure. Also, in the future, we should develop schemes to efficiently update Web video similarities and the hierarchical structure when a new query is given and new Web videos are added or deleted.

Acknowledgements This work was partly supported by Grant-in-Aid for Scientific Research (B) 25280036, Japan Society for the Promotion of Science (JSPS), and Grant-in-Aid for Scientific Research on Innovative Areas 24120002 from the MEXT. We are grateful that a publisher, Springer permits us to deposit this accepted manuscript in the open access repository. The final publication is available at "http://link.springer.com/article/10.1007/s11042-015-2976-8".

References

1. Allaire, G., Kaber, S.M.: Numerical Linear Algebra. Springer-Verlag New York (2008)
2. Arenas, A., Duch, J., Fernandez, A., Gomez, S.: Size reduction of complex networks preserving modularity. New J. Phys. **9 (176)**, 604–632 (2007)
3. Astola, J., Haavisto, P., Neuvo, Y.: Vector median filters. In: Proc. IEEE, pp. 678–689 (1990)
4. Brachman, R.: What is-a is and isn't: An analysis of taxonomic links in semantic networks. IEEE Computer **16**(10), 30–36 (1983)
5. Butafogo, R.A., Schneiderman, B.: Identifying aggregates in hypertext structures. In: Proc. 3rd ACM Conf. Hypertext, pp. 63–74 (1991)
6. Cheng, X., Dale, C., Liu, J.: Statistics and social network of youtube videos. In: Proc. IEEE Int. Workshop on Quality of Service, pp. 229–238 (2008)
7. Fan, J., Elmagarmid, A.K., X. Zhu, W.G.A., Wu, L.: Classview: Hierarchical video shot classification, indexing and accessing. IEEE Trans. Multimedia **6**(1), 70–86 (2004)
8. Frey, B.J., Dueck, D.: Clustering by passing messages between data points. Science **315**(5814), 972–976 (2007)
9. Gantz, J., Reinsel, D.: The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the far east. In: IDC iView (2012)
10. Gargi, U., Wenjun, L., Vahab, M., Sangho, Y.: Large-scale community detection on youtube for topic discovery and exploration. In: Proc. Int. AAAI Conf. Weblogs and Social Media, pp. 486–489 (2011)
11. González, F.A., Caicedo, J.C., Nasraoui, O., Ben-Abdallah, J.: Nmf-based multimodal image indexing for querying by visual example. In: Proc. ACM Int. Conf. Image and Video Retrieval, pp. 366–373 (2010)
12. Harakawa, R., Hatakeyama, Y., Ogawa, T., Haseyama, M.: An extraction method of hierarchical web communities for web video retrieval. In: Proc. IEEE Int. Conf. Image Processing, pp. 4397–4401 (2013)
13. Haseyama, M., Ogawa, T.: Trial realization of human-centered multimedia navigation for video retrieval. Int. Journal of Human-Computer Interaction **29**(2), 96–109 (2013)
14. Haseyama, M., Ogawa, T., Yagi, N.: A review of video retrieval based on image and video semantic understanding. ITE Trans. MTA **1**(1), 2–9 (2013)
15. Hatakeyama, Y., Haseyama, M.: An effective visualization method based on web community extraction using hyperlink between web videos and its application to retrieval. In: Proc. Int. Tech. Conf. Circuits/Systems, Computers and Communications, pp. 371–374 (2010)

16. Hatakeyama, Y., Ogawa, T., Asamizu, S., Haseyama, M.: A novel video retrieval method based on web community extraction using features of video materials. IEICE Trans. Fundamentals **E92-A**(8), 1961–1969 (2009)
17. Hindle, A., Shao, J., Lin, D., Lu, J., Zhang, R.: Clustering web video search results based on integration of multiple features. World Wide Web **14**(1), 53–73 (2011)
18. Kamie, M., Hashimoto, T., Kitagawa, H.: Effective web video clustering using playlist information. In: in Proc. ACM Symp. Applied Computing, pp. 949–956 (2012)
19. Kleinberg, J.M.: Authoritative sources in a hyperlinked environment. Jounal of ACM **46**(5), 604–632 (1999)
20. Miller, G.A.: Wordnet: A lexical database for english. Commun. ACM **38**(11), 39–41 (1995)
21. Nagasaka, A., Tanaka, Y.: Automatic video indexing and fullvideo search for object appearances. In: Proc. IFIP 2nd Working Conf. Visual Database Systems, pp. 113–127 (1991)
22. Newman, M.E.J., Girvan, M.: Finding and evaluating community structure in networks. Phys. Rev. E **69 (026113)** (2004)
23. Ngo, C.W., Pong, T.C., Zhang, H.J.: On clustering and retrieval of video shots through temporal slices analysis. IEEE Trans. Multimedia **4**(4), 446–458 (2002)
24. Nielsen, A.: Multiset canonical correlations analysis and multispectral, truly multitemporal remote sensing data. IEEE Trans. Image Processing **11**(3), 293–305 (2002)
25. Nitanda, N., Haseyama, M.: Audio-based shot classification for audiovisual indexing using pca, mgd and fuzzy algorithm. IEICE Trans. Fundamentals **E90-A**(8), 1542–1548 (2007)
26. Papadopoulos, S., Kompatsiaris, Y., Vakali, A., Spyridonos, P.: Community detection in social media. Data Mining and Knowledge Discovery **24**(3), 515–554 (2012)
27. Ross, D.A., Lim, J., Lin, R.S., Yang, M.H.: Incremental learning for robust visual tracking. Int. J. Comput. Vis. **77**(1-3), 125–141 (2008)
28. Sang, J., Xu, C.: Browse by chunks: Topic mining and organizing on web-scale social media. ACM Trans. Multimedia Comput. Commun. Appl. **7S**(1), 30:1–30:18 (2011)
29. Sebastiani, F.: Machine learning in automated text categorization. ACM Computing Surveys **34**(1), 1–47 (2002)
30. Taskiran, C., Chen, J.Y., Albiol, A., Torres, L., Bouman, C.A., Delp, E.J.: Vibe: A compressed video database structured for active browsing and search. IEEE Trans. Multimedia **6**(1), 103–118 (2004)
31. Wang, Y., Belkhatir, M., Tahayna, B.: Near-duplicate video retrieval based on clustering by multiple sequence alignment. In: Proc. ACM Int. Conf. Multimedia, pp. 941–944 (2012)



Ryosuke Harakawa received his B.S. and M.S. degrees in Electronics and Information Engineering from Hokkaido University, Japan in 2013 and 2015, respectively. He is currently pursuing a Ph.D. degree at the Graduate School of Information Science and Technology, Hokkaido University. His research interests include audiovisual processing and Web mining. He is a student member of the IEEE, IEICE, and Institute of Image Information and Television Engineers (ITE).



Takahiro Ogawa received his B.S., M.S. and Ph.D. degrees in Electronics and Information Engineering from Hokkaido University, Japan in 2003, 2005 and 2007, respectively. He is currently an assistant professor in the Graduate School of Information Science and Technology, Hokkaido University. His research interests are multimedia signal processing and its applications. He has been an Associate Editor of ITE Transactions on Media Technology and Applications. He is a member of the IEEE, EURASIP, IEICE, and Institute of Image Information and Television Engineers (ITE).



Miki Haseyama received her B.S., M.S. and Ph.D. degrees in Electronics from Hokkaido University, Japan in 1986, 1988 and 1993, respectively. She joined the Graduate School of Information Science and Technology, Hokkaido University as an associate professor in 1994. She was a visiting associate professor of Washington University, USA from 1995 to 1996. She is currently a professor in the Graduate School of Information Science and Technology, Hokkaido University. Her research interests include image and video processing and its development into semantic analysis. She has been a Vice-President of the Institute of Image Information and Television Engineers, Japan (ITE), an Editor-in-Chief of ITE Transactions on Media Technology and Applications, a Director, International Coordination and Publicity of The Institute of Electronics, Information and Communication Engineers (IEICE). She is a member of the IEEE, IEICE, Institute of Image Information and Television Engineers (ITE) and Acoustical Society of Japan (ASJ).