# Automatic Video Summary and Description

Suh-Yin Lee, Shin-Tzer Lee, Duan-Yu Chen

Department of Computer Science And Information Engineering, National Chiao Tung University 1001 Ta-Hsueh Rd, Hsinchi, Taiwan
{sylee,szlee,dychen}@csie.nctu.edu.tw

**Abstract.** Multimedia technology has been applied to many kinds of applications and the amount of multimedia data is growing dramatically. Especially the usage of digital video data is very popular today. Thus, content-based indexing and retrieval in video is getting more important in the future. In this paper, we investigate an efficient and reasonable solution to describe video contents, and propose an automatic generation of the summary of video contents with as least manual interaction as possible. Based on the summary of video contents, we extract the key features of the summary to represent its content and produce the description of the video content in MPEG-7 format using XML. We illustrate our approach by developing a system that can support content-based browsing and querying on the WWW.

## 1 Introduction

The need for efficient methods for searching, indexing and managing multimedia data is increasing due to the recent rapid proliferation of multimedia content. Therefore, content-based queries and indexing is getting more important in the future. The MPEG group recently establishes the MPEG-7 effort to standardize the multimedia content interface. The proposed interface will bridge the gap between various types of content meta-data, such as content features, annotations, relationships, and search engines.

Digital video is becoming the rising tide of multimedia. The amount of video data is growing tremendously. Thus indexing and cataloging of digital videos are more and more important for retrieval. The best way for indexing and cataloging video data is based on content. In the past, we usually describe and annotate the content of video manually. However this traditional solution is not suitable for the enormous amount of video data. We must find a mechanism that can generate the summary and description automatically and provide an efficient and flexible solution to illustrate video content for users.

In this paper, our system is divided into two parts. The first part extracts a summary of video contents. We use video processing technologies, especially the information of motion vector to get more meaningful summary about the video content. We use MPEG-2 video streams as video sources and get meaningful abstracted summary conveying significant information of the original video streams. At the same time, we use the important features of the extracted summary to represent this summary. In the second part, we use XML technology to generate extensible description schemes (**DSs**) in MPEG-7 [13]. Following the **DSs**, we can generate the descriptors (**Ds**) of the summary. When these descriptors are produced, we can use these descriptors to index and to catalog these videos. The characterizing indexing and cataloging results are more meaningful for video contents. A prototype system is implemented that can provide browsing and querying functions for users. Users can browse a video summary from WWW and download these videos that they are interested in. We aim to combine summarization and description procedures together, and generate the summary and descriptors concurrently. Figure 1 illustrates the framework of description generation flowchart. This paper is organized as follows. First, in Section 2, we present the summarizing procedure and discuss the classification of story units. In Section 3, we specify the description that we have developed in our prototype system. The applications of such video content description will be presented and reviewed in Section 4. Finally, further research direction is pointed out and conclusion is made in Section 5.

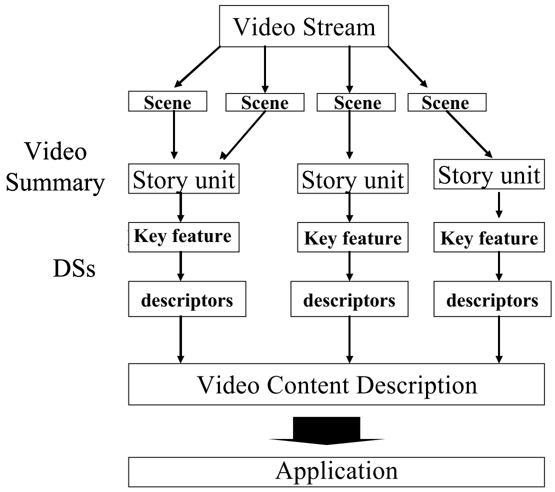


**Fig. 1.** The flowchart introduces the description generation.

## 2 Video Summary

The proposed system is a framework for using the summary of video content as the basis of descriptors in MPEG-7 to represent videos. Thus whether the summary is meaningful or not will affect the effectiveness of the system. The more significant the summary is, the more meaningful the descriptors of a video are generated. For this purpose, we must first extract useful and representative video clips from videos [2]. The summarization procedure is illustrated in the following steps. First, we dissect a video into scenes, and then cluster related scenes into story units. Finally we identify the types of story units and use the information to describe video contents.

### 2.1 Scene Change Detection

Video data is dissected into meaningful segments to serve as logical units called “shots” or “scenes”. We use GOP-based scene change detection approach [3] to segment video data. By this approach, first we detect possible occurrences of scene change GOP by GOP (inter-GOP). The difference in each consecutive GOP-pair is calculated by comparing first I-frames in each consecutive GOP-pair. If the difference of DC coefficients between these two I-frames is higher than the threshold, then there may have scene change between these two GOPs. Then, the actual scene change frame within the GOP is located. We further use the ratio of forward and backward motion vectors to find out the actual frame of scene change within the GOP [4].

### 2.2 Scenes Clustering

After the procedure of scene change detection, a lot of scenes from the video are obtained. However, they are too many and are not suitable to describe the video content. Those related scenes belonging to the same story unit (**SU**) should be clustered together. A story unit is composed of consecutive and related scenes. For example, a segment of a video is about two persons discussing. This segment is composed of several different scenes, but those different scenes should belong to a dialog story unit in a video [6][7][8][9]. Figure 2 illustrates the scene transition graph about a story unit. A, B, C and D are four different scenes, but they appear alternately and closely in a sequence. According to our definition of a story unit, these four scenes should be grouped into the same story unit. The scenes are clustered into story units and then their types are identified. We illustrate our clustering procedure by two steps. First, we have to extract the key features to represent each scene. In this paper, we select the key frame of each scene. Secondly we cluster the scenes into story units.

| | **Scene 1** | **Scene 2** | **Scene 3** | **Scene 4** | **Scene 5** | **Scene 6** | **Scene 7** | **Scene 8** |
|---|---|---|---|---|---|---|---|---|
| Cluster id: | A | B | A | B | C | D | A | B |

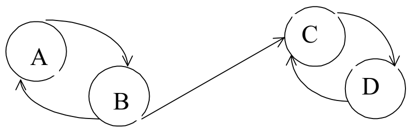


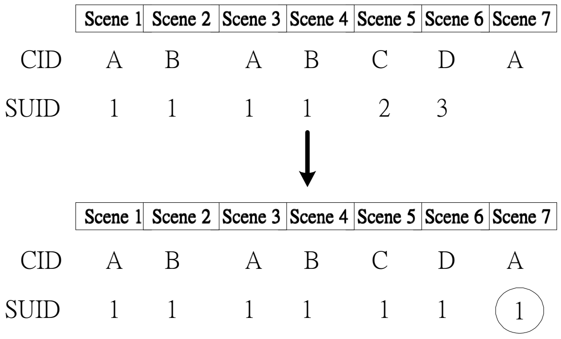
**Fig. 2.** This figure illustrates the scene transition graph in a story unit.

### Extracting the key frame of scene

We extract the first I-frame of each scene as the key frame (***KFS***), since a scene is composed of consecutive similar frames and the first I-frame is qualified to represent this scene. Furthermore, we extract the low-level information of each ***KFS***, such as luminance histogram and DC coefficients, to represent this key frame.

### Clustering related scenes into story unit

After selecting the key frame of each scene, we compare these key frames of each scene pair, and each scene will be compared with those scenes before itself within a range **R** (**R**=20 scenes here). The smallest difference value between these **R** scenes and this target scene is computed. If the smallest difference is lower than a predefined threshold, the two scenes are similar, and they will be assigned the same cluster ID (**CID**) and story unit ID (**SUID**). At the same time we will set all the scenes in between these two identified similar scenes to the same **SUID**. However, if the smallest difference is over the threshold, they are assigned different **CID** and **SUID**. All scenes are scanned sequentially, and their own **CID**s and **SUID**s are assigned. Then the scenes with the same **SUID** are clustered into the same story unit. Figure 3 illustrates the clustering procedure.



|  | Scene 1 | Scene 2 | Scene 3 | Scene 4 | Scene 5 | Scene 6 | Scene 7 |
|---|---|---|---|---|---|---|---|
| CID | A | B | A | B | C | D | A |
| SUID | 1 | 1 | 1 | 1 | 2 | 3 |  |

|  | Scene 1 | Scene 2 | Scene 3 | Scene 4 | Scene 5 | Scene 6 | Scene 7 |
|---|---|---|---|---|---|---|---|
| CID | A | B | A | B | C | D | A |
| SUID | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Fig. 3.** These scenes that are related will be assigned the same SUID.

### *Difference between key frames*

After extracting the representative key frame of each scene, we can measure the difference between two scenes. First, we generate the DC image of each key frame. Each key frame is composed of many macro-blocks (MB) in (Y, Cr, Cb) color space. We calculate the average of the DC coefficients, as the MB's DC coefficient.

The difference (**Diff**) between two scenes $KFS_i$ and $KFS_j$ is measured by their DC images as Eq. (1).

$$Diff(i,j) = \sum_{All\ Blocks\ of\ DC\ \mathrm{Im}age} KFS_i(MB_k) - KFS_j(MB_k) \qquad (1)$$

$MB_k$ is block *k* in a DC image. Each video scene can be represented by a key frame of scene *KFS*. We measure the difference (*Diff*) of every pair of *KFS* in a story unit. Then we select the *KFS* that has the smallest difference with other *KFS* in a story unit as the key frame of this story unit (*KFSU*).

### Feature extraction from the key frame

*The DC coefficients of KFSU*

We extract the DC coefficients of every macro blocks of the *KFSU* and use the average of these DC coefficients to represent key frame. We can organize these DC coefficients as a DC Image. The amount of DC coefficients can be reduced.

*The luminance histogram of KFSU*

We can extract the luminance information from *KFSU*. The luminance information is an important feature of an image, because it can roughly present the content of an image.

*The motion trajectory of SU*

Motion information is the most prominent property of a video. The most intuitive specification of motion features for human being is motion trajectory. Many new technologies can extract the motion trajectory from a video [10]. In this paper, we define the extraction of motion trajectory. It is a simple description of the region motion trajectory in SU. Figure 6 shows the motion vectors of a macro-block in a scene sequence.

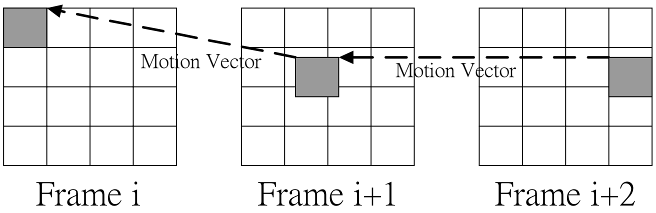


**Fig. 4.** The motion vectors of a macroblock in a scene.

In Figure 4, we can get the motion vector information from a scene sequence. Then we can use these motion vectors to generate simple MB trajectory in a scene.

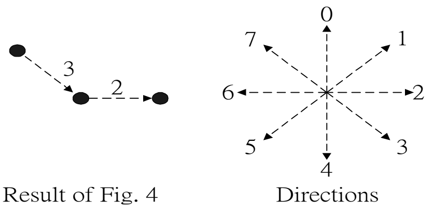


**Fig. 5.** Macro-Block Trajectory and directions

Figure 5 shows the macro-block motion directions and the MB trajectory in Figure 4. We will record all macro-blocks' trajectory by a sequence of directions and the first macro-block position. If the direction sequence is long enough, then we keep this information in video description. The information can present the trajectory of a macro-block in a scene. It could be used to trace the trajectory of scenes in **SU**.

## 2.3 Story unit type identification rules

After the related scenes are clustered into a story unit, then we can distinguish the type of each story unit as action, dialogue, variation, still and default shot type [3][6]. The definition of each story unit type is described below.

**Dialogue story unit**
A dialogue story unit is composed of several scenes with people conversation and always presents the basic structure as "ABAB", in which A and B are scenes of type A and B, respectively. A dialogue story unit is defined as several dominant scenes appear alternatively among a short time period of video.

**Action story unit**
An action story unit is a series of short-term scenes with large variance. We try to detect an action story unit by finding a sequence of short scenes in which each scene has large image variance. When the variance is larger than the threshold and this story unit is not too long, then we say this story unit is an action story unit.

**Still story unit**
A still story unit is a series of long-term scenes with not much variance. We detect the still story unit by searching the long sequential scenes that have slight image variance in this story unit. When the variance of a story unit is less than the threshold and is long enough, then we say this story unit is a still story unit.

**Variation story unit**
A variation story is a long-term story unit with much variance. The definition of variation story unit is that the story unit is long enough and each scene in this story unit is different from other scenes.

**Unidentified story unit**
If a story unit does not belong to the above four story unit types, we set it to an unidentified story unit.

## 2.4 Grouping Story Units

We have implemented the prototyping system testing the extraction of **SU** using a 2-hour long video. The 2hr video contains about 1000 **SU**. There are too many to show all **SU**. We have to group similar **SU** together to reduce the amount of data we show to users firstly. Then we present pictorial representation of SU using a **SU** poster.

**Poster of SU**
The construction of a video poster relies on the compaction of dominance measure within each video scene.

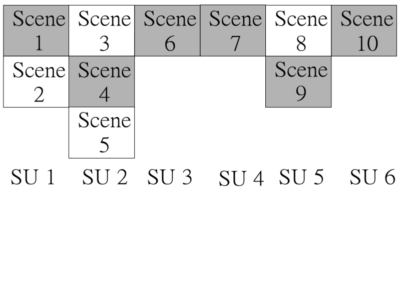
$$Dom(i) = 16 * \frac{The\ number\ of\ scenes\ in\ SU_i}{\sum_{k=1}^{M} The\ number\ of\ scenes\ in\ SU_k} \quad ,1 \le i \le M \qquad (2)$$

The parameter *M* is a predefined number that how many **SU** we want to group in one. Depending on the dominance, we can map each *KFSU* onto a layout pattern. Each poster is made up of 16 primary rectangles (more primary rectangles and more precise).
We measure the dominance of $SU_i$. Dominance measure is based on the number of scenes in this $SU_i$. Thus, if the amount of scenes in $SU_i$ is more than in $SU_j$, then the dominance of $SU_i$ will be greater than or equal to $SU_j$. Figure 6 shows the original scenes in a video sequence.

| Scene 1 | Scene 2 | Scene 3 | Scene 4 | Scene 5 | Scene 6 | Scene 7 | Scene 8 | Scene 9 | Scene 10 |
|---|---|---|---|---|---|---|---|---|---|

**Fig. 6.** The scenes in a video sequence.

Figure 7 shows the clustering of related scenes into story units. The scenes that contain the story key frames of SU are shadowed in gray color.



| | | | | | |
|---|---|---|---|---|---|
| Scene 1 | Scene 3 | Scene 6 | Scene 7 | Scene 8 | Scene 10 |
| Scene 2 | Scene 4 | | | Scene 9 | |
| | Scene 5 | | | | |
| SU 1 | SU 2 | SU 3 | SU 4 | SU 5 | SU 6 |

**Fig. 7.** Clustering Scenes into Story Units.

Figure 8 shows the compaction of a video poster (**P**) after grouping story units into a video poster (**P**). **SU**2 has the most dominance among these *M* **SU**, then its *KFSU* will occupy more space in **P**. Taking the dominance of scenes in a **SU** as a weight, we will allocate different space to each **SU**. The more number of scenes is, the more space of **SU** is allocated. We put the more important **SU** in top-left to bottom-right order.

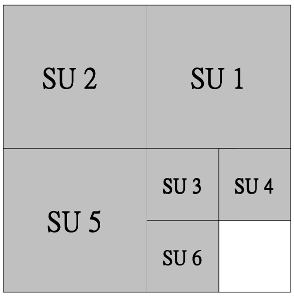


**Fig. 8.** Layout of Key Frame after Compaction.

### Poster of Posters

One level grouping may not reduce the amount of **P** to be small enough. Thus we can reduce the number of **P** in further levels continuously. We can group several posters (**P)** into a Poster of Posters (**PP**). We select the most dominant **SU** to be the key **SU** of this poster (**KSU**) and use **KSU** to present this poster (**P**). Then we group these **KSU** into one poster (**PP**). Figure 9 illustrates the hierarchical structure of a POSTER.

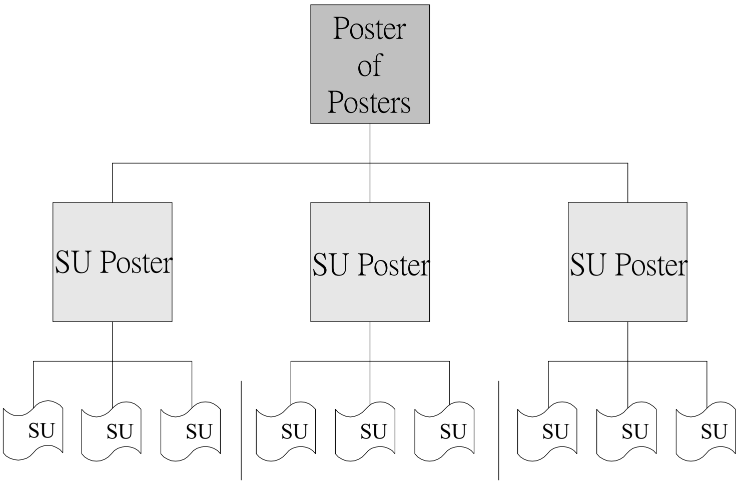


**Fig. 9.** The hierarchical structure of POSTERs.

## 3 Description of Video Content

MPEG-7 is formally named as "Multimedia Content Description Interface". It will extend the limited capabilities of current solutions in identifying multimedia content, by including more data types and specifying a standard set of description schemes and descriptors that can be used to describe various types of multimedia information. We follow MPEG-7 framework and provide a flexible description scheme of video content. We use summary as a basis for the description of the video content. The summary is a meaningful sub-stream of video content and using the features of summary to describe the video content is more reasonable and meaningful. We illustrate the video description by three steps. First, we illustrate the description definition language (**DDL**) used. Secondly we specify the description schemes (**DSs**) defined. Finally we describe the descriptors (**Ds**).

### 3.1 Grouping the story units

We use summary to represent the video content. A summary is generated by the **SU** mentioned above. **SU** are extracted from a video and more important **SU** are selected to compose into a summary. The type of each **SU** is characterized and assigned a priority. The higher priority **SU** are included in the summary of a video. Table 1 lists the priority of **SU** types.

**Table 1.** The priority of each story unit type

| Story unit type | Predefined Priority |
|---|---|
| Dialogue | 4 |
| Action | 3 |
| Still | 2 |
| Variation | 1 |
| Unidentified | 0 |

The summarizing rule is that unidentified type will never be used in a summary. The priorities of other four types will be adjusted depending on video content. Eq. (3) shows the function used to adjust the priority of SU types:

$$P_{new} = P_{pre} * \frac{NUM_T}{NUM_{SU}} \qquad (3)$$

$P_{new}$ : the adjusted priority of **SU** type in current video.

$P_{pre}$ : the predefined priority of **SU** type.

$NUM_T$ : the total number of **SU** type T in a video.

$NUM_{SU}$ : the total number of **SU** in a video

Then we can get new priority of each **SU** type and we use these new priorities in the generation of our summary. The **SU** type with higher priority will have higher proportion in a video summary.

### 3.2 Description definition language (*DDL*)

To support content-based retrieval in the browsing system and applications, we develop a standard set of video content descriptor primitives, a framework for extending the content descriptors, and a process for representing, publishing, and transmitting the content descriptors using XML. We use the extensible Markup language, or XML [11][12][13][14], as the description definition language. XML is a tagged markup language for representing hierarchical, structured data. XML is easily readable by both machines and humans. Additionally, XML is portable and extensible. Thus the content descriptions will be defined, in general, from an extensible set of content description data types.

### 3.3 Descriptor scheme (*DS*)

We use summary and its hierarchical structure to be the basis of description of video content. As shown in Figure 10, we define the description scheme that a story unit consists of six descriptors.

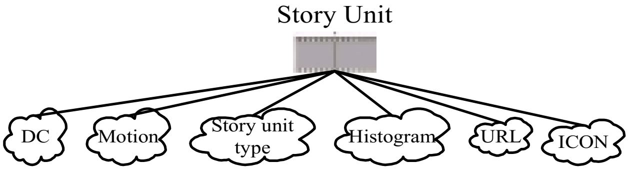


**Fig. 10.** Definition of Description Scheme.

We declare six attributes in the descriptor scheme. Each of the attributes is a descriptor (**D**). Depending on the above definition of **DS**, we can get these descriptors (**Ds**) from a video stream. Then we can use these **Ds** to describe the content of the original video, and to provide searching and browsing function in our system. ICON is the 64x64 pixels down-sampled image. We use this ICON to represent the original key frame image. Attribute URL indicates the location where users can download video summary.

## 4 Application

We are utilizing the description and summary of video content in the development of a web-based video content browsing system. The objective of this system is to catalog the video content and allow users to browse the video content by catalog. Content descriptors are used to represent the videos. Summary and descriptors can be automatically and concurrently generated. Two agents are supported in the system. First agent is text-based query service. Users can query these annotation descriptors which are movie type, director, and actor information. In addition, users can also key in a motion direction or a motion direction sequence as input and the server will find out those story units which contain similar motion sequence. Moreover, users can search for an overall view or information in a video. For example, as shown in Figure 11, users can find out dialogue type video or action type video. Second agent provides the browsing function for users. Users can hierarchically browse the video content by posters. These two kinds of agents support the function that users can download the summaries that they are interested in for viewing.

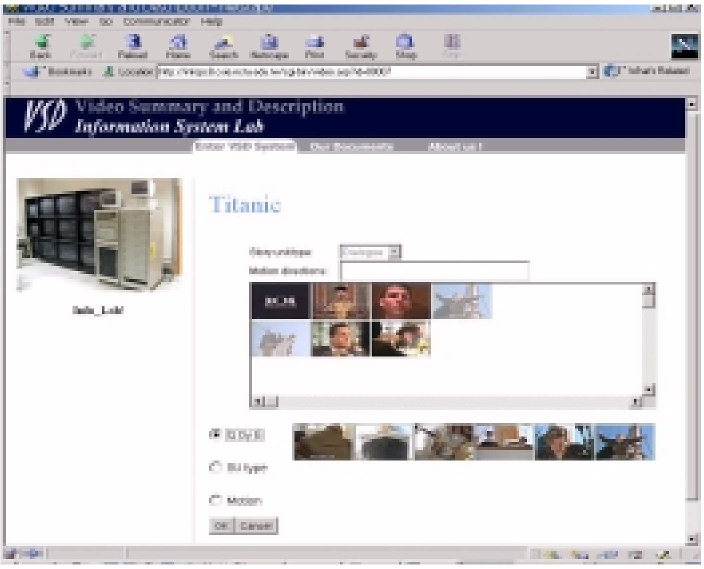


**Fig. 11.** Browse and Advanced Query Interface

## 5 Conclusion

In this paper, a mechanism for automatic extraction of more representative and significant scenes in the video content is proposed. We cluster the related scenes into story units. To accomplish the meaningful description, we first applied a summary technique to video content in order to obtain descriptors more suitable for representing video content. We implement a MPEG-7 like description by video summary and use XML to make the description scheme more flexible and extensible. Furthermore, the purposed system provides a convenient and fast method for users to browse and query the video content.

## References

[1] ISO/IEC JTC1/SC29/WG11 N2966. “MPEG-7 Generic AV Description Schemes (V0.7)”, Melbourne October 1999

[2] Nikolaos D. Doulamis, Anastasios D. Doulamis, Yannis S. Avrithis and Stefanos D. Kollias, “Video Content Representation Using Optimal Extraction of Frames and Scenes,” IEEE conference on Image processing, Vol. 1, pp. 875-879, 1998.

[3] J. L. Lian, “Video Summary and Browsing Based on Story-Unit for Video-on-Demand Service,” Master thesis, National Chiao Tung University, Dept. of CSIE, June 1999.

[4] J. Meng, Y. Juan, and S. F. Chang, “Scene Change Detection in a MPEG Compressed Video Sequence”, IS&T/SPIE’s Symposium on Electronic Imaging: Science & Technology, Vol. 2417, San Jose, CA, Feb. 1995.

[5] B.L. Yeo and B. Liu, “Rapid Scene Analysis on compressed Video”, IEEE Transactions on Circuits and Systems for Video Technology, Vol. 5, No. 6 pp.533-544, 1995.

[6] M. M. Yeung and B. L. Yeo, “Video Content Characterization and Compaction for Digital Library Applications”, In IS&T/SPIE Electronic Imaging’97: Storage and Retrieval of Image and Video Database, VI SPIE 3022, pp. 310-321, 1997.

[7] M. M. Yeung and B. L.Yeo, “Video Visualization for Compact Presentation of Pictorial Content”, IEEE Transactions on Circuits and Systems for Video Technology, Vol. 7, No. 5, pp. 771-7785, Oct. 1997.

[8] M. M. Yeung, B. L. Yeo, and B. Liu, “Extracting Story Units from Long Programs for Video Browsing and Navigation,” Proc. IEEE Conf. on Multimedia Computing and Systems, pp. 296-305, 1996.

[9] M. Yeung, B. L. Yeo, and B. Liu, “Video Browsing using clustering and scene transitions on compressed sequences,” Proc. IEEE Conf. on Multimedia Computing and Systems, 1996.

[10] Man-Kwan Shan, Shh-Yin Lee, “Content-Based Retrieval of Multimedia Document Systems”, PHD thesis, National Chiao Tung University, Dept. of CSIE, June 1998.

[11] W3 Consortium, RDF Schema Working Group, RDF schemas specification, http://www.w3.org/TR/WD-rdf-schema/, 1998.

[12] W3 Consortium, XML 1.0 Specification. http://www.w3.org/TR/REC-xml.

[13] W3 Consortium, XML Linking Language (Xlink). http://www.w3.org/TR/WD-xlink..

[14] Extensible Markup Language (XML), W3C Working Draft, http://www.w3.org/TR/WD-xml, November, 1997.