ELVIDS: Video Search System Prototype with a Three-Level Hierarchy Model

Tongjin Lee(✉) and Jun Iio

Chuo University, Hachioji-shi, Tokyo 192-0393, Japan
tongjini@gmail.com, iiojun@tamacc.chuo-u.ac.jp

Abstract. We propose a video search system with a three-level hierarchy model for searching videos from works, scenes, and shots. Videos are useful research materials; however, their scholarly use is very limited because finding the desired information is a time-consuming task for scholars. To provide solutions to this problem, we developed a system with functions to increase search efficiency. This paper mentions the reasons preventing the scholarly use of videos and describes the methods used to develop the system, focusing on database creation and the prototype's interface.

Keywords: Video database · OSS

1 Introduction

In recent years, studies have been actively conducted on the use of videos in the humanities and social sciences fields. Conversely, due to various restrictions, the scholarly use of videos in academic fields is limited compared to paper materials such as books, journals, and archives. Among the diverse restrictions hindering the use of videos in academic research, we focused on the technical aspects of accessing videos. We developed a prototype of hierarchical presentation for a video search system named ELVIDS[1]. In this paper, we first clarify the problems, which need to be solved to accelerate the scholarly use of videos; then, we describe the development method of the prototype with regard to database design and interface.

2 Issues of Video Use in Humanities and Social Science Research

The test sets used in this research were educational contents. The work stored on the database of this prototype were two videos entitled "Information poor country Nippon" and "Use Open Source Software" from Chuo University's "Corridor of Knowledge," which is an educational program based on joint production of Chuo University and Jupiter Telecommunications Co., Ltd. The videos are 30-minute programs distributed on national cable TV. All videos are uploaded on YouTube and can be streamed online.

[1] Note that this prototype is a closed caption based search system, not based on video recognition and open caption base.

L. Barolli et al. (eds.), *Advances in Network-based Information Systems*,
Lecture Notes on Data Engineering and Communications Technologies 7,
DOI 10.1007/978-3-319-65521-5_101

Two of the videos selected as test data contain interviews of an information center official and persons engaged in the development and dissemination of open source software.

Although these kinds of videos, including the videos mentioned above, can be used as educational and research material, there are few cases of their actual use. As Andreano [1] pointed out: "First, there is the difficulty scholars face in finding information about, and gaining access to, the holdings an archive's collection. Second, there is the time consuming and often frustrating task of retrieving the desired information from that collection of moving image materials."

To solve the above problem, we developed a prototype of hierarchical presentation for a video search system, which makes it possible to access specific parts of videos. In the similar structure as that of a book, such as chapters, paragraphs and sections, a video can be divided into parts such as scenes and shots. To improve video search efficiency, we implemented these structures to the video search system prototype.

However, the introduction of such a system is costly. Especially in information centers such as small archives and libraries, the budget allocated to the information system is limited. Even if a proposal to develop such a system is put forward, the introduction of the system could be difficult. In this study, to solve the problem of excessive cost, we made use of open source software. The software used in system development are the following: Ubuntu Server as the server OS, Apache as the HTTP server, SQLite 3 as the database management system, and Ruby Sinatra as the web framework.

3 Video Search System Prototype with a Three-Level Hierarchy Model

3.1 Database

A unique function of the system is to make videos accessible at the work, scene, and shot level; thereby, the metadata describing all video levels are stored in the system's database. To store the metadata, three tables were created. The tables at each level containing the logical name, physical name, and data type, are shown in Tables 1, 2, and 3, respectively.

Table 1. Column name and data type of works.

PK	Logical name	Physical name	Data type
○	id	id	integer
	title	title	text
	creator	wcrt	text
	publisher	wpub	text
	description of a work	wdesc	text
	level of unit	wlev	text
	type of video	vidtyp	text
	duration of work	wdur	integer
	identifier	idt	text
	language	lang	text

Table 2. Column name and data type of scenes.

PK	Logical name	Physical name	Data type
○	work id	work_id	integer
○	id	id	text
	contributor of a scene	sccontr	text
	description of a scene	scdesc	text
	level of unit	sclev	text
	duration of a scene	scdur	text
	start time of a scene(hour)	scsth	text
	start time of a scene(min)	scstm	text
	start time of a scene(sec)	scsts	integer
	end time of a scene(hour)	sceth	text
	end time of a scene(min)	scetm	text
	end time of a scene(sec)	scets	text

Table 3. Column name and data type of shots.

PK	Logical name	Physical name	Data type
○	work id	work_id	integer
○	scene id	scene_id	text
○	id	id	text
	description of a shot	shdesc	text
	level of unit	shlev	text
	duration of a shot	shdur	text
	start time of a shot(hour)	shsth	text
	start time of a shot(min)	shstm	text
	start time of a shot(sec)	shsts	integer
	end time of a shot(hour)	sheth	text
	end time of a shot(min)	shetm	text
	end time of a shot(sec)	shets	text

The WORKS table has nine columns: id, title, creator, publisher, description, video levels, video format, duration, and language (Table 1). The id column contains unique numbers for each record, which become the primary keys. The title is a column which stores the title of the video such as “Corridor of Knowledge-Information Poor Country Nippon” and “Corridor of Knowledge-Use Open Source Software.” The creator column stores the names of the video’s creators. The publisher column has information about individuals or organizations involved in publishing the videos. The description of the work column contains tests of video summaries. The level of the unit column on the work table has a work value without exception. The values in the column are used for searching all video units. The video format column contains either the “mp4” or “webm” value. The duration column stores the duration of videos, expressed in minutes. The language column contains information on the language used in the video.

The SCENES table has 12 columns: work id, scene id, contributor, description, video level, duration, and start time of scene (Table 2). The work id is a column that stores values used in foreign keys; the id column contains unique numbers for each record in the table. The contributor of a scene column stores the names of interviewees; the description of a scene column has text, which contains the content of what the interviews in a specific video scene. The level of the unit column in the table has a value scene without exception. The duration of the scene column stores the duration of a scene, which is expressed in seconds. The values in the columns of the start time of a scene (hour), the start time of a scene (min), the start time of a scene (sec), the end time of a scene (hour), the end time of a scene (min), and the end time of a scene (sec) are used to play a specific scene from a video.

The SHOTS table contains 12 columns: work id, scene id, id, description of a shot, unit level, shot duration, start time of a shot (hour), start time of a shot (min), start time of a shot (sec), end time of a shot (hour), end time of a shot (min), and end time of a shot (sec) (Table 3). Each column of work id and scene id has values used in foreign keys; the id column maintains unique numbers for each record in the table. The description of a scene column has text, which contains the content of what the interviewees talked about in a specific video shot. The level of the unit column in the table has a value scene without exception. The duration in the shot column stores the duration scene value, which is expressed in seconds. The values in the columns of the start time of a scene (hour), start time of a scene (min), start time of a scene (sec), end time of a scene (hour), end time of a scene (min), and end time of a scene (sec), are used to play a specific shot from a video. All video metadata, which are used as the test data set in this study, are stored in this database.

3.2 Interface

As the interface for searching and watching the video, we made 10 kinds of pages: a top page and search pages for works, scenes, and shot (Fig. 1). When searching and accessing videos, a user chooses one of the search screens prepared for each hierarchy (Fig. 2).

On either search screen, the videos will be searched by entering keywords into a search window and clicking a search button. After the user clicks the search button, the videos matching the keywords are displayed on the search result screen (Fig. 3). As shown in Fig. 3, the keywords, which the user inputs, are highlighted on the results page. For instance, if a user wants to search video shots, they get the search results by accessing a shot search screen, entering keywords, and clicking the search button.

On the search results screen, thumbnails of the top frames of scenes, the names of the interviewee, video titles, names of the creator and publishers, language, duration, time code, and interview content text are displayed. If there is a video in the search result that the user wants to watch, the transition to the video playback page (Fig. 4) is made by clicking a thumbnail or time code.

Fig. 1. The whole figure of ELVIDS.

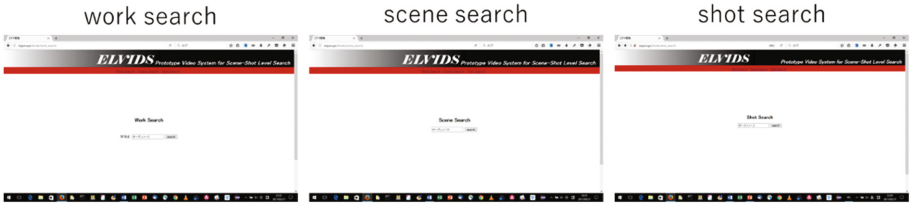

Fig. 2. Search pages of works, scenes, and shots.

Fig. 3. Search results pages of works, scenes, and shots.

Fig. 4. Playback pages of works, scenes, and shots.

4 Evaluation Experiment

4.1 Method

To confirm the effectiveness of ELVIDS, our members conducted an evaluation experiment on November 29, 2016 using video stored on the database of ELVIDS: “Information poor country Nippon” and “Use Open Source Software” from Chuo University’s “Corridor of Knowledge.” The participants of the experiments were 20 Chuo University students. The method is that the participants fill in the answer sheets (Fig. 5) while watching the videos. Each answer sheet has five questions about the contents of the video.

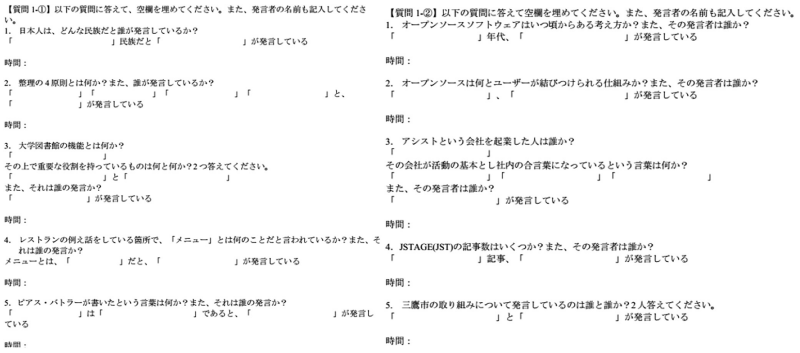
【質問 1-①】以下の質問に答えて、空欄を埋めてください。また、発言者の名前も記入してください。
1. 日本人は、どんな民族だと誰が発言しているか？
「　　」民族だと「　　」が発言している
時間：
2. 整理の 4 原則とは何か？また、誰が発言しているか？
「　　」「　　」「　　」「　　」と、
「　　」が発言している
時間：
3. 大学図書館の機能とは何か？
「　　」
その上で重要な役割を持っているものは何と何か？2 つ答えてください。
「　　」と「　　」
また、それは誰の発言か？
「　　」が発言している
時間：
4. レストランの例え話をしている箇所で、「メニュー」とは何のことだと言われているか？また、それは誰の発言か？
メニューとは、「　　」だと、「　　」が発言している
時間：
5. ピアス・バトラーが書いたという言葉は何か？また、それは誰の発言か？
「　　」は「　　」であると、「　　」が発言している
時間：
【質問 1-②】以下の質問に答えて空欄を埋めてください。また、発言者の名前も記入してください。
1. オープンソースソフトウェアはいつ頃からある考え方か？また、その発言者は誰か？
「　　」年代、「　　」が発言している
時間：
2. オープンソースは何とユーザーが結びつけられる仕組みか？また、その発言者は誰か？
「　　」、「　　」が発言している
時間：
3. アシストという会社を起業した人は誰か？
「　　」
その会社が活動の基本とし社内の合言葉になっているという言葉は何か？
「　　」「　　」「　　」
また、その発言者は誰か？
「　　」が発言している
時間：
4. JSTAGE(JST)の記事数はいくつか？また、その発言者は誰か？
「　　」記事、「　　」が発言している
時間：
5. 三鷹市の取り組みについて発言しているのは誰と誰か？2 人答えてください。
「　　」と「　　」が発言している
時間：

Fig. 5. Answer sheets of the evaluation experiment.

The experiment was conducted in the following manner. The participants were divided into two groups: eight participants in group 1 and 12 in group 2. In the first experiment, each group filled in the answer sheet while watching the video entitled “Information poor country Nippon.” In this experiment, group 1 used ELVIDS, and group 2 used YouTube. In the second experiment, each group filled in the answer sheet while watching the video entitled “Use Open Source Software.” In this experiment, group 1 used YouTube, another group used ELVIDS. After aggregating the both groups’ data on the time taken to answer, the number of answered question and the number of correct answers, we compared the results of each group. The time taken to answer in this experiment was the time until all the participants in one group finished to fill out all the blanks on the answer sheet.

4.2 Results

After completion of the experiment, we aggregated the both groups’ data on the time taken to answer, the number of answered questions and the number of correct answers. In the first experiment, the time taken to answer of group 1 was 7:52 on average; the fastest time was 6:47 and the slowest time was 10:06. The number of answered question of Group 1 was 4.88 on average and another group was 1.33. In addition,

six of eight participants of Group 1 answered the questions correctly. On the other hand, no participants of Group 2 finished to fill in all the blanks of the answer sheet and 10 of 12 participants answered zero to two questions correctly.

In the second experiment, the time taken to answer of group 2 was 7:04 on average; the fastest time was 4:30 and the slowest time was 09:09. The number of answered question of Group 1 was 2.81 on average and another group was 4.83. In addition, no participants of Group 1 answered all the questions correctly and six of eight participants answered two to three questions correctly. In contrast, all the participants of Group 2 finished to fill in all the blanks of the answer sheet and eight of 12 participants answered all the questions correctly. As a result, we concluded that the search efficiency ELVIDS can increase video search efficiency.

5 Related Work

Our approach to video content retrieval is a text-based search associated with the hierarchical work-scene-shot structure. Conversely, there are several proposals for retrieving appropriate video content.

Lu *et al.* [2] proposed SVQL, which is a video query language extended from SQL. By using SVQL, users described new conditional expressions, in the WHERE clause of SQL, to search video scenes efficiently. Li [3] discussed an XML-based system as a video database management method. In their system, metadata and scene annotation were stored in the XML file and were used as the keys to make the video database manageable and searchable.

Another approach for retrieving video content is the ontology-based approach. Daga and Ghatol [4] proposed a linguistics content extraction system to address and retrieve objects, events, and concepts from videos. In addition, Nikam and Nandwalkar's algorithm [5] allows us to realize automatic semantic content extraction from videos based on fuzzy ontology and rule based modeling. Furthermore, Ganesh *et al.* [6] proposed a method to extract semantic content from videos to retrieve video content information. Note that the text-based approach is simpler and easier compared with the ontology-based approaches.

Table 4. Usability test results

Category	Number of test items below threshold
Design	9/19
Contents	7/16
Consideration of a user environment	5/10
User burden reduction	10/18
Consideration of a handicapped user	4/8
Site management	4/5

6 Conclusions and Further Work

6.1 Conclusions

In this paper, we introduced a prototype video search system named ELVIDS. The aim of our study was to solve the problems with regard to the scholarly use of videos in the humanities and social sciences fields. The problems consist of difficulty in finding information and of the retrieval of desired information being time consuming. As a solution to these problems, we developed a three-level hierarchy model, which enables the user to retrieve videos from works, scenes, and shots. To implement the above functions, we developed a database to store the metadata for each video level. The database has three tables: works table, scenes table, and shots table. The numbers of columns in the works, scenes, and shots table are 9, 12, and 12, respectively. In addition, we also developed an interface that enables the user to find the specific work, scene, or shot. The interface has 10 pages: top page, video search page, search results page, and video playback page for each level.

6.2 Further Work

The newly developed prototype can increase video search efficiency; however, to promote the wide use of the system, its usability must be improved. After developing the prototype, the members of this project conducted a simplified usability test. The 76 test items where categorized into six groups: "Design," "Contents," "Consideration of User Environment," "User Burden Reduction," "Consideration of a handicapped user" and "Site Management." The scale of evaluation was based on four grades; "0 (poor)," "1 (fair)" and "2 (good)." The test threshold was set to "1" before conducting the test.

As shown in Table 4, the test results showed that the prototype's usability was poor. Nearly half of the test items in each category were below the threshold; namely, the items below the threshold were nine in "Design," seven in "Content," five in "Consideration of a user environment," 10 in "User burden reduction," four in "Consideration of a handicapped user," and five in "Site management." To improve the prototype's usability, we will conduct a full-scale usability test and verify the results. Then, we will develop an updated version of the prototype.

Acknowledgement. This study was funded by the Chuo University Grant for Special Research. We are grateful to the members of our laboratory for their valuable insights, feedback, and much helpful support.

References

1. Andreano, K.: The missing link: content indexing, user-created metadata, and improving scholarly access to moving image archives. Moving Image **7**(2), 82–99 (2007)
2. Lu, C., Liu, M., Wu, Z.: SVQL: A SQL extended query language for video databases. Int. J. Database Theory Appl. **8**(3), 235–248 (2015)
3. Li, Z.: An XML-based system for management and query of video databases with user identifiable and annotated scenes. Graduate theses and dissertations, Paper 14231 (2014)

4. Daga, B.S., Ghatol, A.A.: Detection of objects and activities in videos using spatial relations and ontology based approach in video database system. Int. J. Adv. Eng. Technol. **9**(6), 640–650 (2016)
5. Nikam, P.: Nandwalkar, B.R: Fuzzy ontology and rule based model for automatic semantic content extraction from videos using k-means algorithm. Int. J. Comput. Appl. **130**(13), 11–16 (2015)
6. Ganesh, K.R., Kanthavel, R., Celin, A.V.: Ontology and rule-based model for extracting semantic content in videos. J. Theoret. Appl. Inf. Technol. **62**(2), 350–355 (2014)