Lecture Video Browsing Using Multimodal Information Resources

Haojin Yang, Franka Grünewald, Matthias Bauer, and Christoph Meinel

Hasso-Plattner-Institute (HPI), University of Potsdam, Germany
{haojin.yang,franka.gruenewald,matthias.bauer,Meinel}@hpi.uni-potsdam.de

Abstract. This paper presents an approach for automated video indexing and video search in large lecture video archives. First of all, we apply video segmentation and key-frame detection to offer a visual guideline for the video content navigation. We further extract textual metadata by applying video *Optical Character Recognition* (OCR) on detected key-frames and by performing *Automatic Speech Recognition* (ASR) on lecture audio streams, automatically. The OCR and ASR transcript as well as the detected OCR text line types are adopted in the subsequent keyword extraction process, by which both video- and segment-level keywords are extracted respectively. A user study is provided for evaluating the performance and the effectiveness of proposed indexing methods in our lecture video portal. Furthermore, we propose a novel concept for content-based video search by using multimodal information resources.

Keywords: Lecture videos, video indexing, content-based video search, lecture video archives

1 Introduction

In the last decade, digital video has become a popular storage and exchange medium due to the rapid development in recording technology, improved video compression techniques and high-speed networks. Therefore more and more universities record their lectures and publish them further online for the students to access. This results in a huge amount of multimedia data on the *World Wide Web* (WWW). How to build an efficient search function for finding lecture videos on the web or within large lecture video portals has become a challenging task.

Most of the video retrieval and video search systems such as Google, YouTube, Bing etc. reply on available textual metadata such as title, genre, person, and brief description etc. Generally, this kind of metadata has to be created by a human to ensure a high quality, but the creation step is rather time and cost consuming. Furthermore, the manually provided metadata is typically brief, high level and subjective. Therefore, beyond the current approaches, the next generation of video retrieval systems apply automatically generated metadata by using video analysis technologies. In this way, much more content-based metadata can be generated efficiently. Moreover, the temporal video information can also

be adopted for some special retrieval tasks such as lecturer action and gesture recognition.

Text is a high-level semantic feature which has often been used for the content-based information retrieval. In our framework, we have developed an entire workflow for gathering video textual information, including video segmentation/lecture slide extraction, video OCR, ASR, and keyword extraction from OCR and ASR results. By using a *Connected Component* (CC)-based segmentation method, we can detect the unique lecture slides. The detected slide key-frames are further utilized by a video OCR engine, which consists of a two-stage text detection scheme and a multi-hypotheses framework for text recognition. To obtain *speech-to-text* information we use the open-source ASR software *CMU Sphinx*[1] in combination with our acoustic and language model, which have been trained for recognizing German lecture videos.

Keywords can provide a brief summary of a document and are thereby widely used for information retrieval in video portals. We have developed an automated method for extracting segment- and video-level keywords from OCR and ASR transcripts.

To develop a content-based video search engine in a lecture video portal, the search indices will be created from different information resources, including manual annotations, OCR and ASR transcripts etc. The varying recognition accuracy of different analysis engines might result in solidity and consistency problems. Therefore, we propose a new method for ranking keywords extracted from various information resources.

In order to investigate the usability and the effectiveness of proposed indexing features, we have conducted a user study.

The rest of the paper is organized as follows: section 2 reviews related work. Section 3 describes our automatic video indexing features, while section 4 details the user study result. A conceptional discussion of video search using multimodal information resources is provided in Section 5. Section 6 concludes the paper with an outlook on future work.

2 Related Work

Wang et al. proposed an approach for lecture video search based on video segmentation and video OCR [3]. The proposed segmentation algorithm in their work is based on the differential ratio of text and background regions. Using thresholds they attempted to capture the slide transition. The final segmentation results are determined by synchronizing detected slide key-frames and related text books, where the content similarity between them has been applied as the indicator. Since the animated content evolvement has not been considered, their system might not work robustly when those effects occur in lecture slides. Furthermore, the final segmentation result is strongly dependent on the quality of the OCR result. Therefore, it might be less efficient and imply redundancies, when poor OCR results were obtained.

[1] `http://cmusphinx.sourceforge.net/`

Talkminer[2] is a lecture webcast search system which has been proposed by Adcock et al. in [1]. The system retrieved more than 37000 lecture videos from different resources as e.g., YouTube, Berkeley Webcast etc. The search indices are created based on the global metadata obtained from the video hosting website and texts extracted from slide videos by using a standard OCR engine. Since no text detection and segmentation processes have been implemented, the text recognition accuracy of their system is much lower than our system's.

Similar to Talkminer another lecture video search engine Yovisto[3] which is proposed by Sack et al. [11], utilizes an automated video segmentation method and a standard OCR engine for content-based metadata generation. Furthermore, the LOD (*Linked Open Data*) resource DBPedia [2] has been adopted to extract semantic entities from video lectures.

In the CONTENTUS [8] project, a content-based semantic multimedia retrieval system has been developed. After the digitization of media data, several analysis techniques as e.g. OCR, ASR, video segmentation, automated speaker recognition etc. have been applied for metadata generation. An entity recognition algorithm and open knowledge bases are used to extract entities from the textual metadata.

In both Yovisto and CONTENTUS a search function is provided based on the recognized semantic entities from the textual metadata. As already mentioned, searching through the information resources with various confidence scores, we have to deal with the solidity and the consistency problem. However the reviewed systems did not consider this issue.

In our previous work [13], we proposed a video visual analysis framework, which consists of a slide video segmenter, a video OCR engine, and an automatic lecture outline extraction method. In addition, in [12] we introduced a solution for improving ASR results of German lecture videos. We will thus give a general overview of our video indexing approaches in the next section.

3 Automated Lecture Video Indexing

We perform three analysis processes for the retrieval task, including visual video analysis, audio speech analysis and textual analysis.

3.1 Visual Analysis for Lecture Videos

Video browsing can be achieved by segmenting video into representative keyframes. Choosing a sufficient segmentation method is based on the definition of "video segment" and usually depends on the genre of the video. In the lecture video domain, a video sequence of an individual lecture topic is often considered as a video segment. This can be determined by analyzing lecture slide transitions. The traditional approaches utilize global pixel-differencing metrics for

[2] `http://talkminer.com/`
[3] `http://www.yovisto.com`

capturing slide segments [1]. We developed a novel CC-based method, by which the binary CCs are applied instead of image pixels as the basis element. In this way, high-frequency image noises can be removed in the frame comparison process by adjusting a valid size of CCs. Our method consists of two steps: in the first step, we try to capture every knowledge change between adjacent frames, for which we established an analysis interval of three seconds by taking both accuracy and efficiency into account. Since the result from this step may contain the progressive build-up of a complete final slide over sequence of partial versions, the progress continues with the second step intended to capture real slide transitions. Here, we apply a text line comparison method in different slide regions to determine whether two adjacent frames belong to the same slide.

Since the mentioned segmentation method is only defined for slide images, it might not be robust when videos with varying genres having been embedded in the slides and are played during the presentation. To solve this problem we have extended the original algorithm by using a *Support Vector Machine* (SVM) classifier and image intensity histogram features. The experimental results show that the achieved classification accuracy for recognizing slide frames is over 91% by using this approach.

Texts in the lecture slide are closely related to the lecture topic, can thus provide important information for the retrieval task. In our framework, *text detection* first determines whether a single frame of a video file contains text, for which a bounding box enclosing each text line is returned as the result. We have developed a two-stage approach that consists of a fast edge-based detector for coarse detection and a *Stroke Width Transform* (SWT)- and SVM-based verification procedure to remove false alarms. Then, the text segmentation process separates text regions from their background, for which we developed a novel skeleton-based binarization approach for processing video images. After this process, the text line images are converted to an acceptable format for standard OCR engines. For text recognition, we apply a multi-hypotheses framework to recognize texts from text line images. The subsequent spell-checking process will further sort out incorrect words from the recognition results. An in-depth discussion of the proposed segmentation and video OCR methods can be found in [13].

3.2 ASR for Lecture Videos

In addition to video OCR, ASR can provide speech-to-text information from lecture videos, which offers the chance to improve the quantity of automatically generated metadata dramatically. However, most lecture speech recognition systems cannot achieve a sufficient recognition rate, the WERs (*Word Error Rates*) reported from [7, 6, 5, 4] are approximately 40%–80%. Therefore, we decided to build acoustic models for our special use case by applying the CMU Sphinx Toolkit[4] and the German Speech Corpus by Voxforge[5] as a baseline. We collected

[4] `http://cmusphinx.sourceforge.net/`
[5] `http://www.voxforge.org/`

hours of speech data from our lecturers and compiled corresponding transcripts for the acoustic model training. Furthermore, we developed a method to generate the German phonetic dictionary automatically. The experimental results show that the WER decreased by about 19%, when adding 7.2 hours of speech data from our lecturers to the training set. More information of our current ASR approach and the experiment results can be found in [12].

3.3 Textual Analysis for Lecture Videos

Regarding lecture slides we can realize that contents of title, subtitle and key point have more significance than that from the slide's body, as they provide a summarization of each slide. We thus classify the type of OCR text lines by using their geometrical information and stroke width value. The defined types include *title, key-point, footline* and *normal content*. Subsequently, we extract the lecture outline by using classified text contents, which can provide an overview of the lecture to the user. Moreover, each outline item with a timestamp can in turn be used for browsing within the video.

Keywords can summarize a document and are widely used for information retrieval in digital libraries. In our framework, segment-level as well as video-level keywords are extracted from OCR and ASR results respectively. For extracting segment-level keywords, we consider each individual lecture video as a document corpus and each video segment as a single document, whereas for obtaining video-level keywords, all lecture videos in the database are processed, and each video is considered as a single document.

To extract segment-level keywords, we first arrange each ASR and OCR word to an appropriate video segment according to the timestamp. Since our system only considers nouns as keywords, we thus extract all nouns from the transcripts by using the stanford part-of-speech tagger [10]. Then we use a stemming algorithm to capture nouns with variant forms but a common meaning. To remove the spelling mistakes resulted by the OCR engine, we perform a dictionary-based filtering process.

We calculate the weighting factor for each remaining keyword by extending the standard TFIDF (*Term Frequency Inverse Document Frequency*) score [9]. The TFIDF algorithm calculates keywords only according to their statistical frequencies. It cannot represent the location information of keywords, that might be important for ranking keywords extracted from web pages or lecture slides. Therefore, we defined a new formula for calculating the TFIDF score for segment-level keywords, as shown by Eq. 1:

$$TFIDF_{seg-internal}(kw) = \frac{1}{N}(TFIDF_{ocr} \cdot \frac{1}{n}\sum_{i=1}^{n} w_i + TFIDF_{asr} \cdot w_{asr}) \quad (1)$$

where kw denotes the current keyword, $TFIDF_{ocr}$ and $TFIDF_{asr}$ denote the TFIDF score computed from OCR and ASR words respectively, w is the weighting factor, n denotes the number of various OCR text line types. N is the number

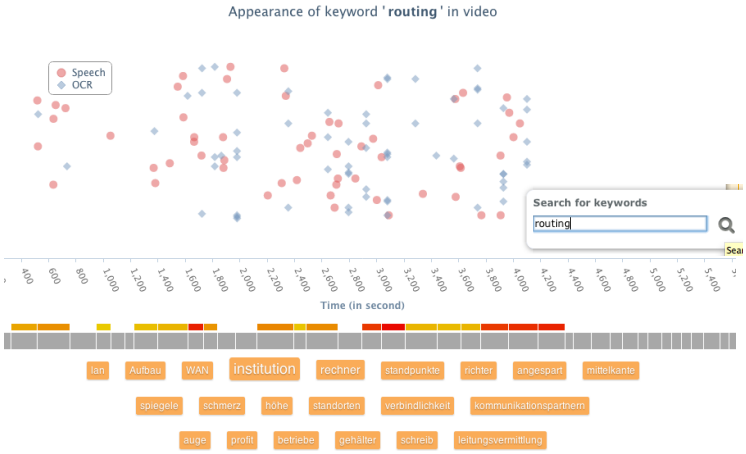

Fig. 1. Segment-level keyword browsing and keyword search function in our lecture video portal

of available information resources, in which the current keyword can be found (the corresponding TFIDF score does not equal 0).

Since OCR text lines are classified into four types in the previous analysis stage, we can calculate the corresponding weighting factor for each type and for each information resource by using their confidence score. Eq. 2 shows the formula:

$$w_i = \frac{\mu}{\sigma_i} \ (i = 1 \ldots n) \tag{2}$$

where μ is set to equal 1 and σ can be calculated by using the corresponding recognition accuracy, as shown by Eq. 3:

$$\sigma_i = 1 - Accuracy_i \ (i = 1 \ldots n) \tag{3}$$

Fig. 1 demonstrates the web GUI of the segment-level keyword browsing and search function in our lecture video portal. The detected Keywords are presented by plot-points in the scatter chart, the video will navigate to the position where the word has been spoken or appearances in the slide by clicking. Video-level keyword extraction works in a similar manner that we will discuss in section 5.

4 User Study

We conducted a user study with 12 students from our institute. These students are enrolled in computer science studies doing their bachelor's or master's degree. We wanted to identify how fast, how accurate and with the help of which video indexing tools a specific lecture topic can be found within a lecture video. Besides we meant to learn if video indexing tools could enable the learner to be more attentive and thus have a bigger learning success. Therefore, two tasks have been conducted:

- **Task 1** was to find an information in a complete lecture of about 1 hour. This task had to be done five times with the different setups, at randomly chosen but similar and non-repetitive lecture videos. The available setups to be used were
  - only the video in a seekable video player
  - video plus key-frames
  - video plus lecture outline
  - video plus keywords
  - video plus all available indexing tools.
- **Task 2** was to watch a video of 10 minutes. One time the participants were allowed to use all of the indexing tools on the video and for another video they were only allowed to use the video player without additional tools. After watching a video the students had to perform a small exam so we could measure their learning effectiveness.

During the user study each student had to work on his or her own and had an undisturbed working environment during the whole time. After a preliminary introduction, each student had one hour time to fulfill the requested tasks. All the videos were chosen carefully with similarity in complexity of the topic and findability of the required information in mind. The videos are part of computer science studies and they were held in the subjects' native language. The videos were chosen at random. No student knew about the topics before the beginning of the test procedure. Furthermore none of the students attended any of the lectures of the chosen subject (information security).

All the students who participated in the user study have never worked with the video indexing tools before the study. But even though they were new to them, every single subject was able to use them to their advantage after a small introduction. We found that the requested information were found faster and more accurate than searching without the video indexing tools whenever the students used key-frames, lecture outline or all of the available features (cf. Fig. 2 (a) and (c)). The only tool showing a slower processing speed were the keywords. In our understanding this is caused by the necessity to scroll the website down from the video player to the keywords GUI. Nevertheless, it achieved the best accuracy results as shown in Fig. 2 (c).

In the second task, we have prepared three multiple choice and a free-text question for each test video. In order to make the evaluation as general as possible, we have designed three methods for the result scoring:

- **W1** for multiple choice question: +1 for each correct answer, -2 for each incorrect/missed answer; for free-text question: +2 for each correct point.
- **W2** for multiple choice question: +1 for each correct answer, -1 for each incorrect answer, 0 for missed answers; for free-text question: +2 for each correct point.
- **Normalized W2** since the maximal number of correct answers for the multiple choice questions is different for each test video, we thus additionally built the normalized result of W2.

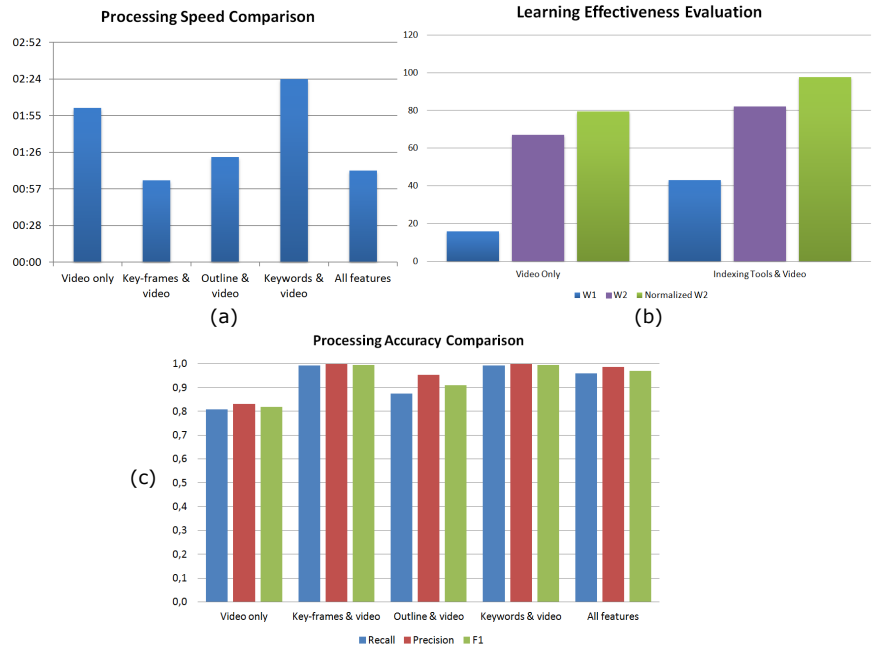

Fig. 2. (a) Processing speed evaluation results of task 1, (b) results of the learning effectiveness evaluation (task 2), where Y-axis presents the total score of the exam, (c) processing accuracy evaluation results of task 1.

From the results presented by Fig. 2 (b), we can realize that the learning effectiveness can be improved measurably by using video indexing tools.

5 Video Search Using Multimodal Resources

As already mentioned, to build a content-based video search engine by using multimodal information resources, we have to deal with solidity and consistency problems. Those information resources might be generated either by a human or by an analysis engine. For the latter case, different analysis engines may have various confidence scores. Therefore, during the ranking process we should consider both, the statistical feature of the keywords and their confidence scores as well. We have thus defined a formula for computing the video-level TFIDF score, as shown by Eq. 4:

$$TFIDF_{vid-level}(kw) = \frac{1}{N}\sum_{i=1}^{n} TFIDF_i \cdot w_i \tag{4}$$

where $TFIDF_i$ and w_i denote the TFIDF score and the corresponding weighting factor for each information resource. N is the number of available information resources, in which the current keyword can be found.

	v_1	v_2	$\cdots$	v_n
kw_1	a_{11}	a_{12}	$\cdots$	a_{1n}
kw_2	a_{21}	a_{22}	$\cdots$	a_{2n}
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
kw_m	a_{m1}	a_{m2}	$\cdots$	a_{mn}

Table 1. keyword-video matrix A

In our case, video search indices can be built from three information resources currently, including global video metadata created by a human, ASR and video OCR words. As described in Eq. 1, the OCR text lines were in turn classified into several types, let $TFIDF_g$ and w_g denote the TFIDF score and the weighting factor for global video metadata, the formula can thus be extended as:

$$TFIDF_{vid-level}(kw) = \frac{1}{N}(TFIDF_g \cdot w_g + TFIDF_{ocr} \cdot \frac{1}{n}\sum_{i=1}^{n} w_i + TFIDF_{asr} \cdot w_{asr}) \tag{5}$$

The ranked video-level keywords can directly be used by a video search engine. In addition, the video similarity can also be further computed by using a vector space model and the *cosine similarity measure*. Table 1 shows an exemplary keyword-video matrix $A_{kw \times v}$, its columns and rows correspond to video and keyword indices respectively. The value of each matrix element is the calculated $TFIDF_{vid-level}$ score of the keyword kw_i in the video v_j.

Let each column of A denote a vector d_j which corresponds to a video v_j. Here, the dimension of d_j is the number of selected keywords. Let q denote the query vector which corresponds to another video v_k, the similarity between v_j and v_k can then be calculated by using cosine similarity measure according to Eq. 6:

$$sim(d_j, q) = \frac{\sum_{i=1}^{m}(a_{ij}q_i)}{\sqrt{\sum_{i=1}^{m}(a_{ij})^2}\sqrt{\sum_{i=1}^{m}(q_i)^2}} \tag{6}$$

Furthermore, the TFIDF score for the *inter-video* segment comparison can be derived according to Eq. 7:

$$TFIDF_{seg-inter}(kw) = TFIDF_{seg-internal}(kw) \cdot TFIDF_{vid-level}(kw) \tag{7}$$

By using this score, we are able to implement a video segment-based lecture topic search/recommendation function.

6 Conclusion and Future Work

In this paper, we presented an approach for automated video indexing and video search by using multimodal information resources. To retrieve textual metadata

automatically, we developed a video OCR system and applied ASR technology. The segment- and video-level keywords are further extracted from OCR and ASR transcripts by extending the original TFIDF algorithm. We proposed a novel concept for content-based video search systems. A user study was conducted to investigate the effectiveness of proposed indexing methods. As future work, we will implement the proposed video search function in our lecture video portal.

References

1. J. Adcock, M. Cooper, L. Denoue, and H. Pirsiavash. Talkminer: A lecture webcast search engine. In *Proc. of the ACM international conference on Multimedia*, MM '10, pages 241–250, Firenze, Italy, 2010. ACM.
2. S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives. Dbpedia: a nucleus for a web of open data. In *Proceedings of the 6th international The semantic web and 2nd Asian conference on Asian semantic web conference*, ISWC'07/ASWC'07, pages 722–735, Berlin, Heidelberg, 2007. Springer-Verlag.
3. T.-C. P. F. Wang, C-W. Ngo. Structuring low-quality videotaped lectures for cross-reference browsing by video text analysis. *Journal of Pattern Recognition*, 41(10):3257–3269, 2008.
4. J. Glass, T. J. Hazen, L. Hetherington, and C. Wang. Analysis and processing of lecture audio data: Preliminary investigations. In *Proc. of the HLT-NAACL Workshop on Interdisciplinary Approaches to Speech Indexing and Retrieval*, 2004.
5. A. Haubold and J. R. Kender. Augmented segmentation and visualization for presentation videos. In *Proc. of the 13th annual ACM international conference on Multimedia*, pages 51–60. ACM, 2005.
6. D. Lee and G. G. Lee. A korean spoken document retrieval system for lecture search. In *Proc. of the SSCS speech search workshop at SIGIR*, 2008.
7. E. Leeuwis, M. Federico, and M. Cettolo. Language modeling and transcription of the ted corpus lectures. In *Proc. of the IEEE ICASSP*, pages 232–235. IEEE, 2003.
8. J. Nandzik, B. Litz, N. Flores-Herr, A. Lhden, I. Konya, D. Baum, A. Bergholz, D. Schnfuü, C. Fey, J. Osterhoff, J. Waitelonis, H. Sack, R. Khler, and P. Ndjiki-Nya. Contentusäîtechnologies for next generation multimedia libraries. *Multimedia Tools and Applications*, pages 1–43, 2012.
9. G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. In *INFORMATION PROCESSING AND MANAGEMENT*, pages 513–523, 1988.
10. K. Toutanova, D. Klein, C. Manning, and Y. Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *In Proc. of Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (HLT-NAACL 2003)*, pages 252–259, 2003.
11. J. Waitelonis and H. Sack. Towards exploratory video search using linked data. *Multimedia Tools and Applications*, 59:645–672, 2012.
12. H. Yang, C. Oehlke, and C. Meinel. An automated analysis and indexing framework for lecture video portal. In *Proc. of the 11th International Conference on Web-based Learning (ICWL 2012), September 2-4, 2012, Sinaia, Romania*, volume 7558, pages 285–294. Springer Lecture Notes in Computer Science, 2012.
13. H. Yang, H. Sack, and C. Meinel. Lecture video indexing and analysis using video ocr technology. *International Journal of Multimedia Processing and Technologies (JMPT)*, 2(4):176–196, December 2012.