REVIEW ON CONTENT BASED VIDEO LECTURE RETRIEVAL

Deshmukh Bhagyashri D[1]

[1]*PVPIT, M.E. Comp. Engg., Savitribai Phule Pune University, India*

Abstract

Recent advances in multimedia technologies allow the capture and storage of video data with relatively inexpensive computers. Furthermore, the new possibilities offered by the information highways have made a large amount of video data publicly available. However, without appropriate search techniques all these data are hardly usable. Users are not satisfied with the video retrieval systems that provide analogue VCR functionality. For example, a user analyses a soccer video will ask for specific events such as goals. Content-based search and retrieval of video data becomes a challenging and important problem. Therefore, the need for tools that can be manipulate the video content in the same way as traditional databases manage numeric and textual data is significant. Therefore, a more efficient method for video retrieval in WWW or within large lecture video archives is urgently needed. This project presents an approach for automated video indexing and video search in large lecture video archives. First of all, we apply automatic video segmentation and key-frame detection to offer a visual guideline for the video content navigation. Subsequently, we extract textual metadata by applying video Optical Character Recognition (OCR) technology on key-frames and Automatic Speech Recognition on lecture audio tracks.

Keywords—*Feature extraction, video annotation, video browsing, video retrieval, video structure analysis.*

--***--

1. INTRODUCTION

In the last decade e-lecturing has become more and more popular. The more college students are trying to interact with and trying to learn from this e-lecture. Digital videos are becoming a popular storage and exchange medium due to the rapid development in recording technology. A number of universities and research institutions are taking the opportunity to record their lectures and publish them online for students to access independent of time and location. As a result, there has been a huge amount of multimedia data on the Web which is very difficult for user to judge whether a video is useful by only glancing at the title and find desired videos without a search function within video archive. The user might thus want to find the piece of information he requires without viewing the complete video. The problem becomes how to retrieve the appropriate information in a large lecture video archive more efficiently. The manually provided metadata is typically brief, high level and subjective. The next generation of video retrieval systems apply automatically generated metadata by using video analysis technologies.

Hypothesis: The relevant metadata can be automatically gathered from lecture videos by using appropriate analysis techniques. They can help a user to find and to understand lecture contents more efficiently, and the learning effectiveness can thus be improved.

Our research work mainly focus on those lecture videos produced by using the screen grabbing method. Segmenting two-scenes lecture videos can be achieved by only processing slide video streams, which contain most of the visual text metadata.

Key frame recognition by OCR technology:- The OCR technique is nothing but the Optical Character Recognition .The OCR technique is used for extracting textual metadata .By using OCR we can convert different type of documents such as paper document and image capture by digital camera into editable and searchable data. With OCR the recognized document looks like the original. The OCR software allows you to save a lot of time and efforts when creating and processing and repurchasing various document. 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.

Speech recognition by AUTOMATIC SPEECH RECOGNITION technique: The AUTOMATIC SPEECH RECOGNITION technique is nothing but the Automatic Speech Recognition .A can provide speech to text information from lecture video .in computer science speech recognization is the translation of spoken word into text .It is also known as automatic speech recognization. Speech is one of the most important carriers of information in video lectures. The poor recognition results not only limit the usability of speech transcript, but also affect the efficiency of the further indexing process.

In our research, we intended to continuously improve the AUTOMATIC SPEECH RECOGNITION result based on the open-source AUTOMATIC SPEECH RECOGNITION tool.

A large amount of textual metadata will be created by using OCR and AUTOMATIC SPEECH RECOGNITION method, which opens up the content of lecture videos. We extract metadata from visual as well as audio resources of lecture videos automatically by applying appropriate analysis techniques. For evaluation purposes we developed several automatic indexing functionalities in a large lecture video portal, which can guide both visually- and text-oriented users to navigate within lecture video.

We investigate the usability and the effectiveness of proposed video indexing features. For visual analysis, we propose a new method for slide video segmentation and apply video OCR to gather text metadata. A more flexible search function has been developed based on the structured video text. In order to overcome the solidity and consistency problems of a content-based video search system, we propose a keyword ranking method for multimodal information resources. In order to evaluate the usability, we implemented this approach in a large lecture video portal. In summary, the major contributions of this paper are the following:

a) We extract metadata from visual as well as audio resources of lecture videos automatically by applying appropriate analysis techniques. For evaluation purposes we developed several automatic indexing functionalities in a large lecture video portal, which can guide both visually- and text-oriented users to navigate within lecture video. We conducted a user study intended to verify the research hypothesis and to investigate the usability and the effectiveness of proposed video indexing features.

b) For visual analysis, we propose a new method for slide video segmentation and apply video OCR to gather text metadata. Furthermore, lecture outline is extracted from OCR transcripts by using stroke

c) We propose a solution for automatic German phonetic dictionary generation, which fills the gap in open-source ASR domain. The dictionary software and compiled speech corpus are provided for the further research use.

d) In order to overcome the solidity and consistency problems of a content-based video search system, we propose a keyword ranking method for multimodal information resources. In order to evaluate the usability, we implemented this approach in a large lecture video portal.

e) The developed video analysis methods have been evaluated by using compiled test data sets as well as opened benchmarks. All compiled test sets are publicly available from our website for the further research use.The rest of the paper is organized as follows: Section 2 reviews related work in lecture video retrieval and content-based video search domain.Section3 describes our automatic video indexing methods.

2. RELATED WORK

2.1 Lecture Video Retrieval

Wang et al. proposed an approach for lecture video indexing based on automated video segmentation and OCR analysis [9]. The proposed segmentation algorithm in their work is based on the differential ratio of text and background regions. Using thresholds they attempt to capture the slide transition. The final segmentation results are determined by synchronizing detected slide key-frames and related text books, where the text similarity between them was calculated as indicator. Grcar et al. introduced videoLectures.net in [10] which is a digital archive for multimedia presentations. Similar to [9], the authors also apply a synchronization process between the recorded lecture video and the slide file, which has to be provided by presenters. Our system contrasts to these two approaches since it directly analyzes the video, which is thus independent of any hardware or presentation technology. The constrained slide format and the synchronization with an external document are not required. Furthermore, since the animated content evolvement is often applied in the slide, but has not been considered in [9] and [10], their system might not work robustly when those effects occur in the lecture video. In [9], the final segmentation result is strongly dependent on the quality of the OCR result. It might be less efficient and imply redundancies, when only poor OCR result is obtained. Tuna et al. presented their approach for lecture video indexing and search [11]. They segment lecture videos into key frames by using global frame differencing metrics. Then standard OCR software is applied for gathering textual metadata from slide streams, in which they utilize some image transformation techniques to improve the OCR result. They developed a new video player, in which the indexing, search and captioning processes are integrated. Similar to [9], the used global differencing metrics cannot give a sufficient segmentation result when animations or content build-ups are used in the slides. In that case, many redundant segments will be created. Moreover, the used image transformations might be still not efficient enough for recognizing frames with complex content and background distributions. Making use of text detection and segmentation procedures could achieve much better results rather than applying image transformations. Jeong et al. proposed a lecture video segmentation method using Scale Invariant Feature Transform (SIFT) feature and the adaptive threshold in [12]. In their work SIFT feature is applied to measure slides with similar content. An adaptive threshold selection algorithm is used to detect slide transitions. In their evaluation, this approach achieved promising results for processing one-scene lecture videos as illustrated in Fig. 1a.Recently, collaborative tagging has become a popular functionality in lecture video portals. Sack and Waitelon is[13] and Moritz et al. [14] apply tagging data for lecture video retrieval and video search. Beyond the keyword based tagging, Yu et al. proposed an approach to annotate lecture video resources by using Linked Data. Their framework enables users to semantically annotate videos using vocabularies defined in the Linked Data cloud. Then those semantically linked educational resources are further adopted in the video browsing and video recommendation procedures. However, , the effort and cost needed by the user annotation-based approach cannot satisfy the requirements for processing large amounts of web video data with a rapid increasing speed. Here, the automatic analysis is no doubt much more suitable. Data to further automatically annotate the extracted textual metadata opens a future research direction.ASR provides speech-to-text information on spoken languages, which is thus well suited for content-based lecture video retrieval. The studies described in [5] and [15] are based on out-of-the-box commercial speech recognition software.

Concerning such commercial software, to achieve satisfying results for a special working domain an adaption process is often required, but the custom extension is rarely possible. The authors of [1] and [6] focus on English speech recognition for Technology Entertainment and Design (TED) lecture videos and webcasts. In their system, the training dictionary is created manually, which is thus hard to be extended or optimized periodically. Glass et al. proposed a solution for improving ASR results of English lectures by collecting new speech data from the rough lecture audio data [3]. Inspired by their work, we developed an approach for creating speech data from German lecture videos. Haubold and Kender focus on multi-speaker presentation videos. In their work speaker changes can be detected by applying a speech analysis method. A topic phrases extraction algorithm from highly imperfect ASR results (WER _ 75 percent) has been proposed by using lecture course-related sources such as text books and slide files [4].Overall, most of those lecture speech recognition systems have low recognition rate, the WERs of audio lectures are approximately 40-85 percent. The poor recognition results limit the further indexing efficiency. Therefore, how to continuously improve ASR accuracy for lecture videos is still an unsolved problem. The speaker-gestures-based information retrieval for lecture videos has been studied in [16]. The author equipped the lecture speaker with special gloves that enable the automatic detection and evaluation of gestures. The experimental results show that 12 percent of the lecture topic boundaries were correctly detected using speaker-gestures. However, those gesture features are highly dependent on the characteristics of speakers and topics. It might have limited use in large lecture video archives with massive amounts of speakers.

3. AUTOMATED LECTURE VIDEO INDEXING

In this chapter we will present four analysis processes for retrieving relevant metadata from the two main parts of lecture video, namely the visual screen and audio tracks. From the visual screen we firstly detect the slide transitions and extract each unique slide frame with its temporal scope considered as the video segment. Then the video OCR analysis is performed for retrieving textual metadata from slide frames. Based on OCR results, we propose a novel solution for lecture outline extraction by using stroke width and geometric information of detected text lines. In speech-to-text analysis we applied the open-source ASR software CMU Sphinx.2 To build the acoustic and language model, we collected speech training data from open-source corpora and our lecture videos. As already mentioned, it lacks method in open-source context for creating German phonetic dictionary automatically. We thus developed a solution to fill this gap and made it available for the further research use.

3.1 Slide Video Segmentation

Video browsing can be achieved by segmenting video into representative key frames. The selected key frames can provide a visual guideline for navigation in the lecture video portal. Moreover, video segmentation and key-frame selection is also often adopted as a pre-processing for other analysis tasks such as video OCR, visual concept detection, etc. often considered as a video segment. This can be roughly determined by analyzing the temporal scope of lecture. Many approaches (as, e.g., [17], [11]) make use of global pixel-level-differencing metrics for capturing slide transitions. A drawback of this kind of approach is that the salt and pepper noise of video signal can affect the segmentation accuracy. After observing the content of lecture slides, we realize that the major content as, e.g., text lines, figures, tables, etc., can be considered as Connected Components (CCs). We therefore propose to use CC instead of pixel as the basis element for the differencing analysis. We call it component-level-differencing metric. This way we are able to control the valid size of the CC, so that the salt and pepper noise can be rejected from the differencing process. For creating CCs from binary images, we apply the algorithm according to which demonstrated an excellent performance advantage. Another benefit of our segmentation method is its robustness to animated content progressive build-ups used within lecture slides. Only the most complete unique slides are captured as video segments. Those effects affect the most lecture video segmentation methods mentioned in section2. Our segmentation method consists of two steps (cf. Fig. 2):

In the first step, the entire slide video is analyzed. 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. This means that segments with a duration smaller than three seconds may be discarded in our system. Since there are very few topic segments shorter than three seconds, this setting is therefore not critical. Then we create canny edge maps for adjacent frames and build the pixel differential image from the edge maps. The CC analysis is subsequently performed on this differential image and the number of CCs is then used as a threshold for the segmentation. A new segment is captured if the number exceeds Ts1. Here we

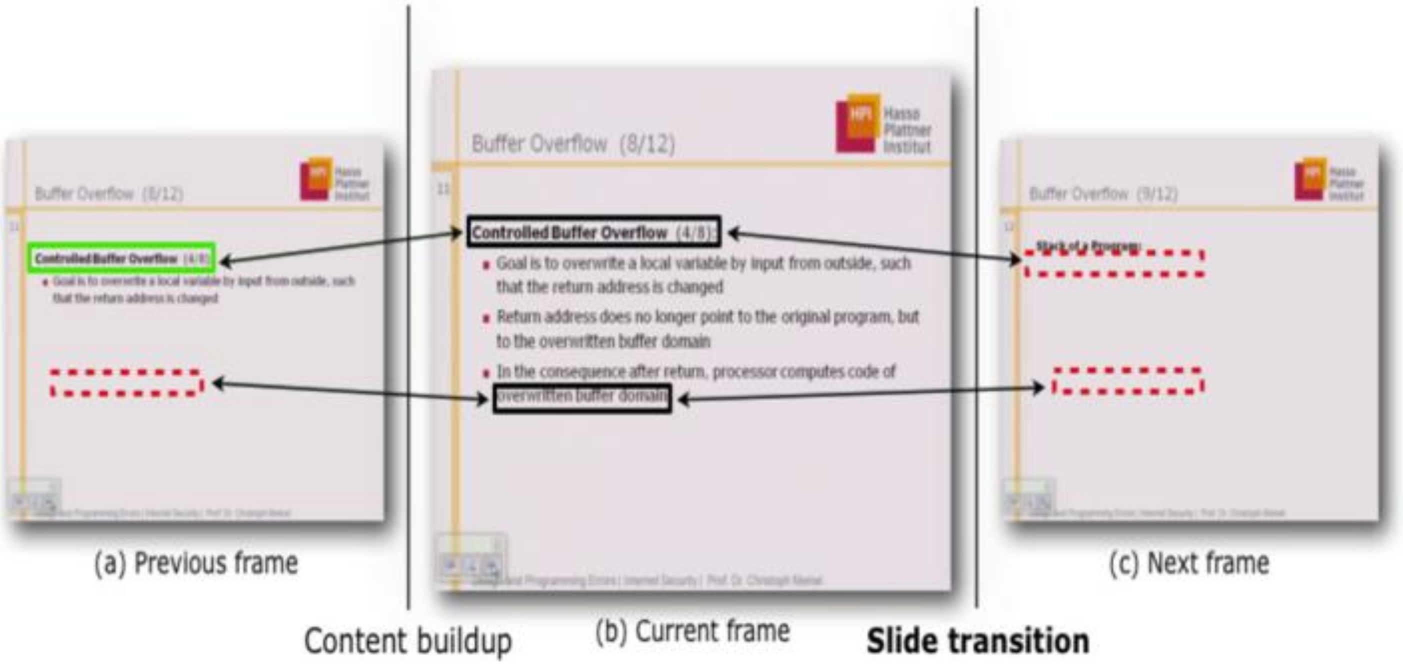
Buffer Overflow (8/12)
Hasso Plattner Institut
Controlled Buffer Overflow (4/8)
Goal is to overwrite a local variable by input from outside, such that the return address is changed
Return address does no longer point to the original program, but to the overwritten buffer domain
In the consequence after return, processor computes code of overwritten buffer domain
Buffer Overflow (9/12)
Stack of a Program
(a) Previous frame
(b) Current frame
(c) Next frame
Content buildup
Slide transition

Fig.1: we detect the first and the last object line in R_C vertically and perform the CC differencing metric on those line sequences from adjacent frames.

In frame (a) and (b), a same text line (top) can be found; whereas in frame (b) and (c), the CC-based differences of both text lines exceed the threshold T_{S2}. establish a relatively small Ts1 to ensure that each newly emerging knowledge point will be captured. Obviously, the first segmentation step is sensitive to animations and progressive build-ups. The result is thus too redundant for video indexing. Hence, the process continues with the second segmentation step based on the frames in the first step.

In the second segmentation step the real slide transitions will be captured. The title and content region of a slide frame is first defined. We established the con-tent distribution of commonly used slide styles by analyzing a large amount of lecture videos in our database. Let Rt and Rc denote the title and content region which account for 23 and 70 percent of the entire frame height respectively. In Rt we apply CC-based differencing as described above with a small threshold value of 1 for capturing the slide transi-tions. Here any small changes within the title region may cause a slide transition. For instance, two slides often differ from each other in a single chapter num-ber. If there is no difference found in Rt, then we try to detect the first and the last bounding box object in Rc vertically and perform the CC-based differenc-ing within the object regions of two adjacent frames (cf. Fig. 2). In case that the difference value of both object regions between adjacent frames exceed the threshold Ts2, a slide transition is then captured. For detecting the content progressive build-up horizon-tally, the process could be repeated in a similar man-ner. In our experiment, Ts1 ¼ 20 and Ts2 ¼ 5 have proven to serve best for our training data. Exact set-ting of these parameters is not critical.

Since the proposed method is designed for segment-ing slide videos, it might be not suitable when videos with varying genres have 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) classier and image intensity histogram features. We use the Radial Basis Function (RBF) as kernel. In order to make a comparison, we also applied the Histogram of Oriented Gradients (HOG) feature [20] in the experiment. We adapted the HOG feature with eight gradient directions, whereas the local region size was set to 64 _ 64. Moreover, in using this approach our video segmentation method is also suitable for processing such one-screen lecture videos with a frequent switch between slide- and speaker-scene.

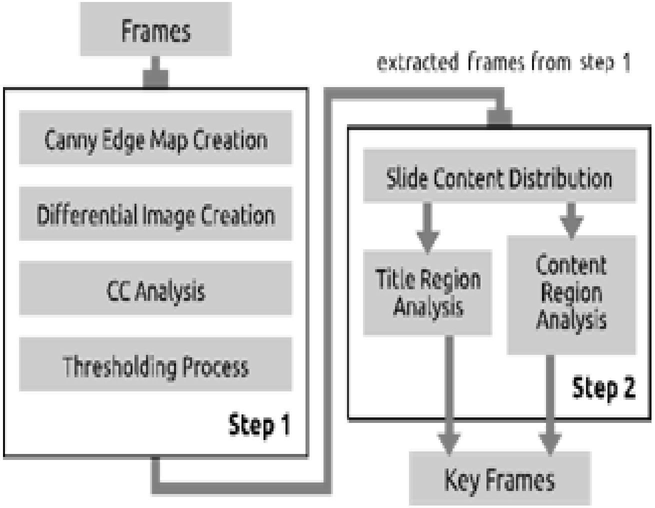

Fig. 2: Lecture video segmentation workflow. Step 1: adjacent frames are compared with each other by applying the CC analysis on their differential edge maps. Step 2: slide transitions are captured by performing title and content region analysis.

3.2 Video OCR for Lecture Videos

Texts in the lecture slides are closely related to the lecture content, can thus provide important information for the retrieval task. In our framework, we developed a novel video OCR system for gathering video text.

For text detection, we developed a new localization-verification scheme. In the detection stage, an edge-based multi-scale text detector is used to quickly localize candidate text regions with a low rejection rate. For the subsequent text area verification, an image entropy-based adaptive refinement algorithm not only serves to reject false positives that expose low edge density, but also further splits the most text- and non-text-regions into separate blocks. Then Stroke Width Transform (SWT) [22]-based verification procedures are applied to remove the non text blocks since the SWT verifier is not able to correctly identify special non-text patterns such as sphere, window-blocks, garden fence, we adopted an additional SVM classifier to sort out these non-text patterns in order to further improve the detection accuracy. For text segmentation and recognition, we developed a novel binarization approach, in which we utilize image skeleton and edge maps to identify the text pixels. The proposed method consists of three main steps: text gradient direction analysis, seed pixel selection, and seed-region growing. After the seed-region growing process, the video text images are converted into a suitable format for standard OCR engines. The subsequent spell-checking process will further sort out incorrect words from the recognition results. Our video OCR system has been evaluated by applying several test data sets. Especially by using the online evaluation of the opened benchmark ICDAR 2011 competition test sets for born-digital images [23], our text detection method achieved the second place, and our text binarization and word recognition method achieved the first place in the corresponding ranking list on their website (last check 08/2013). An in-depth discussion of the developed video OCR approach and the detailed evaluation results can be found in [24]. By applying the open-source print OCR engine tesseract-ocr,5 we achieved recognition of 92 percent of all characters and 85 percent of all words correctly for lecture video images. The compiled test data set including 180 lecture video frames and respective manual annotations is available at [21].

Generally, in the lecture slide the content of title, subtitle and key point have more significance than the normal slide text, as they summarize each slide. Due to this fact, we classify the type of text lines recognized from slide frames by using geometrical information and stroke width feature. The benefits of the structure analysis method can be summarized as follows:

The lecture outline can be extracted using classified text lines, it can provide a fast overview of a lecture video and each outline item with the time stamp can in turn be adopted for video browsing .

The structure of text lines can reflect their different significance. This information is valuable for a search/indexing engine. Similar to web search engines which make use of the explicitly pre-defined HTML structure (HTML-tags) for calculating the weight of texts in web pages, our method further opens up the video content and enables the search engine to give more accurate and flexible search results based on the structured video text.

The process begins with a title line identification procedure. A text line will be considered as a candidate title line when it localizes in the upper third part of the frame, it has more than three characters, it is one of three highest text lines and has the uppermost vertical position. Then the corresponding text line objects will be labeled as the title objects and we repeat the process on the remaining text objects in the same manner. The further detected title lines must have a similar height (the tolerance is up to 10px) and stroke width value (the tolerance is up to 5px) as the first one. For our purposes, we allow up to three title lines to be detected for each slide frame. All non-title text line objects are further classified into three classes: content text, key-point and footline. The classification is based on the height and the average stroke width of the text line object, which is described as follows:

Keypoint if $s_t > s_{mean} \wedge h_t >_{hmean}$
Footline if $s_t < s_{mean} \wedge h_t < h_{mean} \wedge y = y_{max}$

Content-text otherwise
where s_{mean} and h_{mean} denote the average stroke width and the average text line height of a slide frame, and y_{max} denotes the maximum vertical position of a text line object (starts from the top-left corner of the image).To further extract the lecture outline, we firstly apply a spell checker to sort out text line objects which do not satisfy the following conditions:

_ a valid text line object must have more than three characters,

_ a valid text line object must contain at least one noun,

_ the textual character count of a valid text line object must be more than 50 percent of the entire string length.

The remaining objects will be labeled as the outline object. Subsequently, we merge all title lines within the same slide according to their position. Other text line objects from this slide will be considered as the subitem of the title line. Then we merge text line objects from adjacent frames with the similar title content (with 90 percent content overlap).

The similarity is measured by calculating the amount of same characters and same words. After a merging process all duplicated text lines will be removed. Finally, the lecture outline is created by assigning all valid outline objects into a consecutive structure according to their occurrences.

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, as mentioned, most lecture speech recognition systems cannot achieve a sufficient recognition rate. A model-adaption process is often required. Furthermore, in the open-source context, it lacks method for generating the German phonetic dictionary automatically. Therefore, we developed a solution intended to fill this gap. We decided to build acoustic models for our special use case by applying the CMU Sphinx Toolkit6 and the German Speech Corpus by Voxforge7 as a baseline. Unlike other approaches that collect speech data by applying dictation in a quiet environment, we gathered hours of speech data from real lecture videos and created corresponding transcripts.

This way the real teaching environment can be involved in the training process. For the language model training we applied the collected text corpora from German daily news corpus (radio programs, 1996-2000), Wortschatz- Leipzig8 and the audio transcripts of the collected speech corpora. Fig. 5 depicts the workflow for creating the speech training data. First of all, the recorded audio file is segmented into smaller pieces and improper segments are sorted out.

For each remaining segment the spoken text is transcribed manually, and added to the transcript file automatically. As an intermediate step, a list of all used words in the transcript file is created. In order to obtain the phonetic dictionary, the pronunciation of each word has to be represented phonetically.

A recorded lecture audio stream yields approximately 90 minutes of speech data, which is far too long to be processed by the ASR trainer or the speech decoder at once. Shorter speech segments are thus required. Manually collecting appropriate speech segments and transcripts is rather time consuming and costly. There are a number of steps to be performed to acquire high quality input data for a ASR trainer tool. Our current approach is to fully automate segmentation (Fig. 6(2)) and partly automate selection (Fig. 6(3)) without suffering from quality drawbacks like dropped word endings. The fundamental steps can be described as follows: we first compute the absolute values of input samples to get a loudness curve, which is then down sampled (e.g., by factor 100). Then a blur filter (e.g.,radius ¼ 3) is applied to eliminate outliers and a loudness threshold determines which areas are considered quiet.Non-quiet areas with a specified maximum length (5 seconds has proven to serve best for our training data) will serve as a potential speech utterance. Finally, we retrieve speech utterances from the original audio stream and save them into files.From the experiments we have learned that all speech segments used for the acoustic model training must meet certain quality requirements. Experience has shown that approximately 50 percent of the generated audio segments have to be sorted out due to one of the following reasons:

_ the segment contains acoustical noise created by objects and humans in the evironment around the speaker, e.g., doors closing, chairs moving, students talking,

_the lecturer mispronounces some words, so that they are completely invalid from an objective point of view,

_ the speaker’s language is clear, but the segmentation algorithm cuts off parts of a spoken word so that it becomes invalid.

Therefore, the classification of audio segments in good or bad quality is not yet solvable automatically, as the term “good quality” is very subjective and strongly depends on one’s personal perception. Nevertheless, the proposed segmentation method can perform a preselection that speeds up the manual transcription process significantly. The experimental results show that the WER decreased by about 19 percent, when adding 7.2 hours of speech data from our lecture videos to the training set.

3.3 Creation of the German Phonetic Dictionary

The phonetic dictionary is an essential part of every ASR software. For each word that appears in the transcripts, it defines one or more phonetic representations. As our speech corpus is growing continuously, the extension and maintenance of the dictionary becomes a common task. An automatic generator is highly desired. Unfortunately it lacks such a tool in the open-source context. Therefore, we have built a phonetics generator by using a customized phonetic alphabet, which contains 45 phonemes used in German pronunciation.

We provide this tool and the compiled speech training data for the further research use.

The generation algorithm operates on three layers: On word level, we have defined a so-called exemption dictionary which contains foreign words in particular and whose pronunciation does not follow German rules, e.g., byte, pheanomen.

First of all, a check is made as to whether the input word can be found in the exemption dictionary. If this is the case, the phonetic representation is completely read from this dictionary and no further steps have to be performed. Otherwise, we scale down to syllable level by applying an extern hyphenation algorithm. On syllable level, we examine the result of the hyphenation algorithm, as, e.g., computer for the input word computer. For many single syllables (and also pairs of syllables),the syllable mapping describes the corresponding phonetic representation, as, e.g., au f for the German prefix auf and n i: differ for the disyllabic German prefix nieder.If there is such a representation for our input syllable, then it is added to the phonetic result immediately. Otherwise, we have to split the syllable further into its characters and proceed on character level.On character level, a set of German pronunciation rules are applied to determine how the current single character is pronounced, including character type (consonant or vowel), neighbouring characters, relative position inside the containing syllable, absolute position inside the whole word, etc., First and foremost, a heuristic checks if the current and the next 1–2 characters can be pronounced natively. If this is not the case, or the word is only one character long, the characters are pronounced as if they were spelled letter by letter, as, e.g., the abbreviations abc (for alphabet) and ZDF (a German TV channel). In the next step, we determine the character type (consonant or vowel) in order to apply the correct pronunciation rules. The conditions of each of these rules are verified, until one is true or all conditions are verified and proven to be false. If the latter is the case, the standard pronunciation is applied, which assumes closed vowels. An evaluation with 20,000 words from transcripts shows that 98.1 percent of all input words were processed correctly without any manual amendment. The 1.9 percent incorrect words mostly have an English pronunciation. They are corrected manually and added to the exemption dictionary subsequently.

4. CONCLUSION

In this paper, we presented an approach for content-based lecture video indexing and retrieval in large lecture video archives using text and audio information . In order to verify the research hypothesis we apply visual as well as audio resource of lecture videos for extracting content-based metadata automatically.

REFERENCES

[1]. E. Leeuwis, M. Federico, and M. Cettolo, “Language modelling and transcription of the ted corpus lectures,” in Proc. IEEE Int. Conf. Acoust., Speech Signal Process., 2003, pp. 232–235.

[2]. D. Lee and G. G. Lee, “A korean spoken document retrieval system for lecture search,” in Proc. ACM Special Interest Group Inf. RetrievalSearching Spontaneous Conversational Speech Workshop, 2008.

[3]. J. Glass, T. J. Hazen, L. Hetherington, and C. Wang, “Analysis and processing of lecture audio data: Preliminary investigations,” in Proc. HLT-NAACL Workshop Interdisciplinary Approaches Speech Indexing Retrieval, 2004, pp. 9–12.

[4]. A. Haubold and J. R. Kender, “Augmented segmentation and visualization for presentation videos,” in Proc. 13th Annu. ACM Int. Conf. Multimedia, 2005, pp. 51–60.

[5]. W. H€urst, T. Kreuzer, and M. Wiesenh€utter, “A qualitative study towards using large vocabulary automatic speech recognition toindex recorded presentations for search and access over the web,” in Proc. IADIS Int. Conf. WWW/Internet, 2002, pp. 135–143.

[6]. C. Munteanu, G. Penn, R. Baecker, and Y. C. Zhang, “Automatic speech recognition for webcasts: How good is good enough andwhat to do when it isn’t,” in Proc. 8th Int. Conf. Multimodal Interfaces,2006.

[7]. G. Salton and C. Buckley, “Term-weighting approaches in automatic text retrieval,” Inf. Process. Manage., vol. 24, no. 5, pp. 513–523, 1988.

[8]. G. Salton, A. Wong, and C. S. Yang. (Nov. 1975). A vector space model for automatic indexing, Commun. ACM, 18(11),pp. 613–620, [Online]. Available: http://doi.acm.org/10.1145/361219.361220

[9]. T.-C. Pong, F. Wang, and C.-W. Ngo, “Structuring low-quality videotaped lectures for cross-reference browsing by video textanalysis,” J. Pattern Recog., vol. 41, no. 10, pp. 3257–3269, 2008.

[10]. M. Grcar, D. Mladenic, and P. Kese, “Semi-automatic categorization of videos on videolectures.net,” in Proc. Eur. Conf. Mach. Learn. Knowl. Discovery Databases, 2009, pp. 730–733.

[11]. T. Tuna, J. Subhlok, L. Barker, V. Varghese, O. Johnson, and S. Shah. (2012), “Development and evaluation of indexed captioned searchable videos for stem coursework,” in Proc. 43rd ACM Tech. Symp. Comput. Sci. Educ., pp. 129–134. [Online]. Available: http://doi.acm.org/10.1145/2157136.2157177.

[12]. H. J. Jeong, T.-E. Kim, and M. H. Kim.(2012), “An accurate lecture video segmentation method by using sift and adaptive threshold,”in Proc. 10th Int. Conf. Advances Mobile Comput., pp. 285–288.[Online]. Available: http://doi.acm.org/10.1145/2428955.2429011.

[13]. H. Sack and J. Waitelonis, “Integrating social tagging and document annotation for content-based search in multimedia data,” inProc. 1st Semantic Authoring Annotation Workshop, 2006.

[14]. C. Meinel, F. Moritz, and M. Siebert, “Community tagging in tele-teaching environments,” in Proc. 2nd Int. Conf. e-Educ., e-Bus.,e-Manage. and E-Learn., 2011.

[15]. S. Repp, A. Gross, and C. Meinel, “Browsing within lecture videos based on the chain index of speech transcription,” IEEE Trans.Learn. Technol., vol. 1, no. 3, pp. 145–156, Jul. 2008.

[16]. J. Eisenstein, R. Barzilay, and R. Davis. (2007). “Turning lectures into comic books using linguistically salient gestures,” in Proc. 22nd Nat. Conf. Artif. Intell., 1, pp. 877–882. [Online]. Available: http://dl.acm.org/citation.cfm?id=1619645.1619786.

[17]. J. Adcock, M. Cooper, L. Denoue, and H. Pirsiavash, “Talkminer: A lecture webcast search engine,” in Proc. ACM Int. Conf. Multimedia, 2010, pp. 241–250.

[18]. J. Nandzik, B. Litz, N. Flores Herr, A. L€ohden, I. Konya, D.Baum, A. Bergholz, D. Sch€onfuß, C. Fey, J. Osterhoff, J.Waitelonis, H. Sack, R. K€ohler, and P. Ndjiki-Nya. (2012) “ Contentus—technologies for next generation multimedia libraries, Multimedia Tools Appl., pp. 1–43, [Online]. Available:http://dx.doi.org/10.1007/s11042-011-0971-2.

[19]. F. Chang, C.-J. Chen, and C.-J. Lu, “A linear-time component labeling algorithm using contour tracing technique,” Comput. Vis. Image Understanding, vol. 93, no. 2, pp. 206–220, Jan. 2004.

[20]. B. T. N. Dala, “Histograms of oriented gradients for humandetection,” in Proc. IEEE Conf. Comput. Vis. Pattern Recog., 2005,vol. 1, pp. 886–893.

[21]. Ground truth data. (2013). [Online]. Available: http://www.yanghaojin.com/research/videoOCR.html.

[22]. B. Epshtein, E. Ofek, and Y. Wexler, “Detecting text in natural scenes with stroke width transform,” in Proc. Int. Conf. Comput.Vis. Pattern Recog., 2010, pp. 2963–2970..

[23]. H. Yang, B. Quehl, and H. Sack. (2012), “A framework for improved video text detection and recognition,” Multimedia Tools Appl., pp. 1–29, [Online]. Available: http://dx.doi.org/10.1007/s11042-012-1250-6.

[24]. K. Toutanova, D. Klein, C. Manning, and Y. Singer, “Feature-rich part-of-speech tagging with a cyclic dependency network,” in Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Human Lang. Technol., 2003, pp. 252–259.