Search for Keywords and Vocal Elements in Audio Recordings

M. Sigmund[1]
[1]*Brno University of Technology, Faculty of Electrical Engineering and Communication, Technicka 12, CZ-61600 Brno, Czech Republic*
sigmund@feec.vutbr.cz

***Abstract*—This paper deals with search for keywords and non-verbal vocal elements in audio recordings. An efficient detection of specific words or sounds embedded in continuous speech is based on isolated word recognition approaches. The mel-frequency cepstral coefficients and more combinations of predictive coefficients and autocorrelation coefficients were evaluated. A keyword or key sound slides along the stored speech and in each of its positions a distance (i.e., similarity) to the corresponding speech segment is computed. We found an efficient distance measure for non-verbal sound search. The average detection rates achieved 93 percent in keyword search and 74 percent in non-verbal sound search. A system developed for automatic search in audio files is presented.**

***Index Terms*—Speech processing, pattern recognition, speech analysis, algorithms.**

I. INTRODUCTION

The problem of detecting keywords in continuous speech can be broadly solved in two major ways. The first one is the application of a large-vocabulary continuous speech recognizer, described, for instance, in [1], [2]. This approach is very extensive and has some difficulties in dealing with words out of vocabulary and sentences out of grammar. However, searching audio documents often has to deal with many words (proper names, acronyms and so on) that do not appear in the vocabulary of the recognition systems. The second way is to build up a word detector using the keyword models only (modeled by a phonetic string). The most currently used models are based on the hidden Markov models (HMM). For examples see [3], [4] and the references given therein. A detail description of HMM may be found, for example, in [5]. HMM-based approaches, however, raise practical concerns. They require an expensive collection of labeled training data and their training is very time-consuming. An acoustic keyword spotter based on a hybrid model formed by the HMM and artificial neural network paradigm is discussed in [6]. A new approach to keyword search based on large margin and kernel methods is proposed in [7]. The first evaluation of a significant number (10 systems in total) of keyword detection technologies was presented in [8].

Among all words, natural speech contains numerous non-verbal vocal elements known as paralanguage that accompany speech in order to communicate specific meanings. Paralanguage involves two groups of elements: vocal characterizers and vocal segregates. Vocal characterizers include things like laughing, crying, yawning, and moaning, which convey meanings to an audience. Vocal segregates include such sounds as “eh”, “hmm”, “ooh-ooh”, or “mah”, which convey messages about internal feelings. Different languages rely on disparate sets of non-verbal elements based on different cultures. Generally, some non-verbal elements can carry specific information useful for psychoanalysis. For instance, laughing in the form “ha-ha” is considered as a true (i.e., heartfelt) emotional expression while “hi-hi” betrays rather a feigned laughing [9].

Some studies have been reported about keyword searching, but works on systems for reliable detecting and investigating non-verbal vocal elements are rare in the literature. In this paper, we describe search strategies for detection of both simple keywords and non-verbal vocal elements. We compare new distance measures used for detection of chosen non-verbal sounds. The reminder of the paper is organized as follows: Sections II and III introduce the applied search strategies. Section IV describes the developed search tool. Section V presents experimental results. Finally, Section VI discusses the conclusions.

II. KEYWORD SEARCH STRATEGY

Any speech or word is naturally composed of a sequence of phonemes. Each utterance in the audio documents investigated is processed in a left-to-right search performed in a once-passed run. A spoken keyword slides step by step along the whole investigated utterance and in each position a match between the keyword and the corresponding segment of the utterance is estimated. The time shift is constant and smaller than the average duration of words because most word boundaries cannot be detected exactly.

Phonetically identical words spoken by the same speaker and in the same room acoustics differ more or less in their speech signals due to the pronunciation variability. This is especially true of continuous speech. Hence, when comparing the keyword with the utterance segments, a partial-match-only criterion is required. Figure 1 shows the

Manuscript received February 24, 2013; accepted May 29, 2013.
The research were supported by the internal grant of BUT Brno, project MOBYS. The support of the project CZ.1.07/2.3.00/20.0007 WICOMT, financed from the operational program Education for Competitiveness, is also gratefully acknowledged.

processing concept of a system for keyword searching which has been proposed and implemented in this research. At first, a sequence of feature vectors describing the phoneme sequence is used to represent the speech signals of both the keyword and the investigated utterance. Feature extraction in the audio document can be applied in advance, when the speech was recorded, as it is independent from the current keyword. To determine which utterance segment contains the keyword, the distance in the feature vector space between the keyword and the corresponding segment is computed as the match criterion. The length of the utterance segments is identical to the keyword duration. In addition, a non-linear time alignment within the strings of feature vectors improves the match.

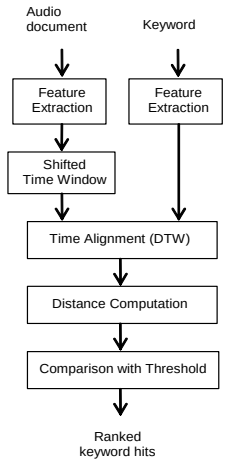

Fig. 1. Block diagram of the system for keyword search.

The final step of the search process is to compare the computed distance with an appropriately chosen threshold. The keyword is detected every time the distance falls below the threshold. Once the search is completed, overlapping keyword hits in each neighborhood are eliminated by removing any worse-score keywords that overlap partially with the best keyword. A ranked list of keyword hits is then made over the whole search space in the audio file and the information is given to the user. As an important advantage of this method, unknown words and noise that occur before or after a keyword do not disturb the detection algorithm. Moreover, no language model is used within the search.

Today‘s most effective speech recognition approaches use acoustic features inspired by models of the human auditory system. A widespread acoustic feature set of this category, the mel-frequency cepstral coefficients (MFCC) has been used in our system. Their computation is represented by the “Feature Extraction” block in Fig. 1. Every 10 ms, the Hamming window is applied to preemphasized speech frames of 20 ms and the fast Fourier transform is applied along with appropriate zero padding. The spectral magnitudes obtained are integrated within 12 triangular filters arranged on the mel-frequency scale. The filter output is the logarithm of the sum of weighted spectral magnitudes. Subsequently, a discrete cosine transform is applied to decorrelate the filter bank outputs. A more detailed description of these coefficients and their computation can be found, for example, in [10].

Dynamic time warping (DTW) was applied to perform the matching (“Time Alignment” block in Fig. 1) between the keyword and the utterance segments. The DTW algorithm is widely used for comparing time series with non-linear distortions in the time axis [11], which is characteristic of speech signals. Penalties for deletion and insertion are used in the DTW alignment algorithm. The penalties can be either fixed or phoneme dependent. In each step the keyword recognizer is only run over those signal frames that lie within matched speech intervals. To optimize the DTW match, the endpoints of utterance intervals can be extended beyond one or both boundary points of the keyword, at the cost of increasing the search space. To limit this increase, a threshold can be applied to prevent enormous extension. After the time alignment using DTW, a linear match criterion based on distance measurement was applied. For a keyword in n-th slid time position, the computed distance is

$$D(n) = \sum_{j=1}^{J} dw_j(n) = \sum_{j=1}^{J} \sum_{m=1}^{M} \left[c_m^{\mathrm{T}}(n+j) - c_m^{\mathrm{R}}(j) \right]^2, \quad (1)$$

where dw denotes local distance between correspondent frames of two words, $c^{\mathrm{T}}_m(n+j)$ is the m-th mel-cepstral coefficient of speech frame $n+j$ in the utterance being tested, $c^{\mathrm{R}}_m(j)$ is the m-th mel-cepstral coefficient of speech frame j in the reference keyword, M stands for the total number of mel-cepstral coefficients used (i.e., M = 12), and J denotes the total number of frames in the keyword. The value of the computed distance $D(n)$ is compared in each time step n with the constant threshold, and a keyword is marked if $D(n)$ is smaller than the threshold value. The decision threshold was determined experimentally [12] in training by assigning two equal error rates, the false acceptance and the false rejection of keywords.

III. VOCAL ELEMENT SEARCH

Although our system was developed primarily to search for linguistic keywords, we have tried to apply it also for detection of some non-verbal vocal elements. However, the detection rate decreases when searching for vocal elements only. Thus, a modification and optimization of the search strategy was needed. In the optimization process, we applied some specific distance measures. They are based on autocorrelation sequence $R(k)$ of speech signal $\{s(n)\}$ and autocorrelation sequence $Ra(k)$ of predictive coefficients $\{a_m\}$ which are widely known as LPC-coefficients [13]:

$$R(k) = \sum_{n=1}^{N-k} s(n)\, s(n+k), \quad (2)$$

$$Ra(k) = \sum_{m=1}^{M-k} a_m\, a_{m+k}. \quad (3)$$

Four auxiliary variables were defined by means of $Ra(k)$ and $R(k)$ for two corresponding speech frames as follows:

$$\alpha = \sum_{k=1}^{K} Ra^{\mathrm{T}}(k)\, R^{\mathrm{R}}(k), \quad (4)$$

$$\beta = \sum_{k=1}^{K} Ra^{\mathrm{R}}(k)\, R^{\mathrm{T}}(k), \quad (5)$$

$$\gamma = \sum_{k=1}^{K} Ra^{\mathrm{R}}(k)\, R^{\mathrm{R}}(k), \tag{6}$$

$$\delta = \sum_{k=1}^{K} Ra^{\mathrm{T}}(k)\, R^{\mathrm{T}}(k), \tag{7}$$

where the number of coefficients was $K = 12$. In (4) to (7), reference speech data are marked with the superscript R while test data are marked with the superscript T. The variables α, β, γ, and δ were used in three efficient local distance measures as follows:

$$de1_j = \beta / \delta, \tag{8}$$

$$de2_j = \ln[1 + x + \sqrt{x(2+x)}], \tag{9}$$

where $x = \sqrt{(\alpha\beta)/(\gamma\delta)}$.

$$de3_j = \left|(\alpha / \gamma) - (\beta / \delta)\right|. \tag{10}$$

The complete search strategy will not be detailed here since it is the same as for keyword searching presented in Section II, with the difference that the local distance dw_j was simply replaced by the local distances $de1_j$, $de2_j$ or $de3_j$.

IV. Search Tool Developed

A new system for automatic searching in audio files based on the above-described concept has been developed in our Laboratory of Signal Processing, using the C++ language. The software tool enables an easy investigation of the occurrence and position of selected words in stored speech. The main dialogue page of the tool is the starting point of working with the system. Figure 2 shows this dialogue page after the termination of a search process. The left part of the dialogue page displays the results of the search. In this part, the diagram shows the investigated speech signal in the time domain. The highlighted signal segments represent the keywords found (upper curve) in accordance with the time behavior of the computed distance $D(n)$ (lower curve) based on the partial-match-only criterion. The double straight line at the bottom of the diagram shows the decision threshold. The table displayed below the diagram in Fig. 2 summarizes the results. The following information can be seen there:

1) *Keyword*, i.e. the name of the keyword.
2) *Test file*, i.e. the name of the audio file investigated.
3) *Position*, i.e. the positions of the keywords found, expressed in time measured from the beginning of the investigated file.
4) *Position %*, i.e. the positions of the keywords found, expressed in per cent points of the total duration of the investigated file.

The table with search results can be saved as a text document or directly printed. The right part of the main dialogue page shows some additional information about the processed speech signals, e.g. sampling parameters and length of the audio files. The capability to edit the audio file and to replay the selected speech signal segments completes the tool. In the case illustrated in Fig. 2, the keyword *jitter* was searched for in the audio file *Lecture DSP* (English spoken lecture on digital signal processing) and it was found several times: for the first time in 8 minutes and 38 seconds.

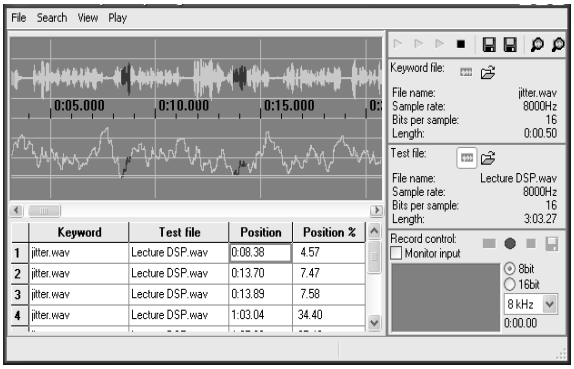

Fig. 2. Main dialogue page of the search tool, with displayed results.

A list of suggested keywords for most relevant topics was created in acoustic form and added to the set of recorded lectures in each course. This core keyword list helps the user choose the right keywords for searching through all the lectures of a course. Thus, the active keyword can be either selected from a keyword list or defined by the user.

V. Experimental Results

We conducted a series of preliminary experiments of searching in both categories simple keywords and chosen non-verbal vocal elements. Speech data used for testing and evaluating were collected from two sources: the first one contains lectures held at our university (database Lectures) representing rather informal speech as spoken in a teaching room, and the second source was a public radio transmission (database News). Hence, the speech material includes utterances by non-professional speakers such as university lecturers and indistinctly speaking politicians on the News, as well as professional newsreaders using a formal speech. The total duration of the speech signal comprised 5 hours of lectures and 12 hours of news. Approximately 4 hours of the News database were available also in transcribed form on the Internet. In this database, three topics were selected: current affairs, weather forecasts, and financial markets. The speech signal was sampled at a rate of 8 kHz and digitized with 16 bits, mono-channel. Our system was evaluated using the most frequently occurring words: 50 keywords from the Lectures database (each spoken by 4 speakers) and 60 keywords from the News database (spoken by 24 speakers in three different sub-sets as can be seen in Table I). All the keywords were selected such that they had a minimum length of two full syllables, and that none of the phonetic forms of the keywords was part of another word. The averaged test results are summarized in Table I.

It is relatively difficult to evaluate the keyword search in an application objectively because insertion errors called false alarm and omission errors become excessive as the vocabulary grows larger. The performance of our system was evaluated experimentally, using two criteria. The first one was the detection rate *DRI* giving the number of correct keywords identified N_{cor} divided by the number of keywords identified in speech N_{kw} (i.e., correct and incorrect)

$$DRI = (N_{\mathrm{cor}} / N_{\mathrm{kw}})\, 100. \tag{11}$$

The second criterion used was the detection rate *DRC* considering correct keywords only

$$DRC = (N_{\text{cor}} / N_{\text{total}})\,100, \tag{12}$$

where N_{cor} is the number of correct keywords identified and N_{total} denotes the total number of existing correct keywords (which were to be identified). The values of *DRI* and *DRC* can be a-priori influenced by setting an optimal decision threshold for the distance $D(n)$ with respect to the costs of false alarm and omission error. An interesting methodology how to evaluate a search algorithm without running the search process is proposed in [14].

TABLE I. EVALUATION OF KEYWORD SEARCHING.

Database	Test Results			
	Speakers	***Keywords***	***DRI (%)***	***DRC (%)***
Lectures	4 male	50	93	90
News: Affairs	5 male 5 female	40	93	91
News: Weather	4 male 6 female	10	98	97
News: Finances	4 male	10	95	93

Detection of non-verbal elements in speech seems to be less successful that detection of keywords. The performance of search approach is affected by some objective factors. For instance, these elements are mostly unvoiced sounds with short duration. In our non-verbal experiments, we tested three typical elements: short laughing "he-he", interjection expressing tacit approval "hmm", and sound clearing the throat "eh". All mentioned vocal sounds were simulated by 4 male speakers in continuous speech. In search process, the local distance measures *de*1 to *de*3 defined by (8) to (10) together with the one given by (1) were used for detection of all three non-verbal elements. Table II shows the achieved detection rates in details using the criterion *DRC*.

TABLE II. DETECTION RATE FOR NON-VERBAL VOCAL ELEMENTS (ALL IN %).

Sound	Number of Elements	Distance Measure			
		dw	***de1***	***de2***	***de3***
"he-he"	117	74	70	72	82
"hmm"	102	68	61	65	63
"eh"	124	73	66	68	77
Average	343	71.6	65.6	68.3	74.0

Generally, the precision of sound position identified in an audio document depends on the shift of sliding reference sounds and on the speech frame length. Time needed to perform the search process in a given audio file is proportional to the sampling frequency, signal framing, keyword (key sound) sliding, as well as the length of the current keyword or sound to be searched.

VI. CONCLUSIONS AND FUTURE WORK

A low-cost approach to open-vocabulary word and sound searching has been presented in this paper. The tool developed executes automatic keyword searching independent of language. This means that no grammar is restricting the keyword occurrence in incorrectly formed sentences. The best keyword detection rate achieved in speaker-dependent search was 97 % (average detection rate was 93 %). These results correspond to the detection rate achieved by other search strategies in English speech [8] as well as in non-English speech [15].

The strategies for keyword search and vocal element search should be optimized separately. In the case of non-verbal vocal elements, we applied three new distance measures. Depending on searched vocal elements, the detection rate varies between 63 % and 82 % by using the most efficient measure *de*3. On average, this measure gives better detection rate that the measure *dw* which satisfies well the linguistic keyword detection. In this field, no results achieved by other authors were published.

Future work will include improvements to our system for applications in adverse conditions such as speech in noise and vocal effort variability. The main goal is to improve the system performance in search for non-verbal vocal elements including both simulated and spontaneously spoken sounds. The natural next step should be an analysis of the found non-verbal sounds and their classification with respect to the speaker's psychological characterization.

REFERENCES

[1] Y. Chen, T. Hou, S. Meng, S. Zhong, J. Liu, "A new framework for large vocabulary keyword spotting using two-pass confidence measure", in *Proc. Computational Engineering in Systems Applications*, Beijing, 2006, pp. 68–71.

[2] I. Szoke, P. Schwarz, P. Matejka, L. Burget, M. Fapso, M. Karafiat, J. Cernocky, "Comparison of keyword spotting approaches for informal continuous speech", in *Proc. Joint Workshop on Multimodal Interaction and Related Machine Learning Algorithms*, Edinburgh, 2005, pp. 1–12.

[3] M. Wöllner, B. Schuller, G. Rigoll, "Keyword spotting exploiting long short-term memory", *Speech Communication*, vol. 55, no. 2, pp. 252–265, Feb. 2013. [Online]. Available: http://dx.doi.org/10.1016/j.specom.2012.08.006

[4] H. Ketabdar, J. Vepa, S. Bengio, H. Bourlard, "Posterior based keyword spotting with a priori thresholds", in *Proc. Interspeech*, Pittsburgh, 2006, pp. 1939–1942.

[5] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odel, D. Ollason, D. Povey, V. Valtchev, P. Woodland, *The HTK Book (version 3.4)*. Cambridge: University Press, 2006.

[6] J. Pinto, A. Lowitt, H. Hermansky, "Exploiting phoneme similarities in hybrid HMM-ANN keyword spotting", in *Proc. Interspeech*, Antwerp, 2007, pp. 1817–1820.

[7] J. Keshet, D. Grangier, S. Bengio, "Discriminative keyword spotting", *Speech Communication*, vol. 51, no. 4, pp. 317–329, Apr. 2009. [Online]. Available: http://dx.doi.org/10.1016/j.specom.2008.10.002

[8] J. G. Fiscus, J. Ajot, J. S. Garofolo, G. Doddington, "Results of the 2006 Spoken Term Detection Evaluation", in *Proc. Interspeech*, Antwerp, 2007, pp. 1–7.

[9] V. Birkenbihl, *Signale des Körpers*. Heidelberg: mvg-Verlag, 2007.

[10] A. C. Kelly, Ch. Gobl, "A comparison of mel-frequency cepstral coefficient (MFCC) calculation techniques", *Journal of Computing*, vol. 3, no. 10, pp. 62–66, Oct. 2011.

[11] L. R. Rabiner, B. H. Juang, *Fundamentals of Speech Recognition*. Englewood Cliffs: Prentice Hall, 1993, ch. 4.7.

[12] F. Dostal, "Keyword spotting", unpublished project report.

[13] L. R. Rabiner, R. W. Schafer, *Digital Speech Processing*. London: Prentice Hall, 2011.

[14] R. Lileikyte, L. Telksnys, "Quality estimation of speech recognition features for dynamic time warping classifier", *Information Technology and Control*, vol. 41, no. 3, pp. 268–273, 2012. [Online]. Available: http://dx.doi.org/10.5755/j01.itc.41.2.914

[15] R. Maskeliunas, K. Ratkevicius, V. Rudzionis, "Voice-based human-machine interaction modeling for automated information services", *Elektronika ir Elektrotechnika (Electronics and Electrical Engineering)*, no. 4, pp. 109–112, 2011.