Employing Smart Logic to Spot Audio in Real Time on Deeply Embedded Systems

Mario Malcangi

DSP&RTS Laboratory
DICo - Università degli Studi di Milano, Via Comelico 39,
20135 Milano, Italy
malcangi@dico.unimi.it

Abstract. Audio mining is currently the subject of several research efforts, especially because of its potential to speed up search for spoken words in audio recordings. This study explores a method for approaching the problem from the bottom. It proposes a framework based on smart logic, mainly fuzzy logic, and on an audio model applicable to any kind of audio recording, including music.

Keywords: Audio mining, audio spotting, fuzzy logic, phone unit, spoken term detection, phonetic search, keyword-spotting

1 Premises

Audio is the most powerful information medium available because of peculiarities that fit well with the requirements of the emerging technology found in embedded systems. Such technology will be very pervasive in the next generation of computing and communication devices. Because of those devices' deep level of embedding, audio will be the preferred interaction medium. There are several reasons for this.

Audio can embed much more information than any other analog medium. Not only does it represent semantic information; the same audio frame contains behavioral, environmental, psychological, and expressive information. Another important peculiarity of audio as information medium is that only audio one can be accessed in eyes-free or hands-free mode. This means that very simple human-to-machine interfaces (HMIs) could be employed in the next generation of embedded devices.

Audio is a one-dimensional signal, storage and processing are simpler and less resource-hungry than for other signal-information media, such as video. This feature is very important in developing embedded devices, because storage and processing power are always limited resources.

Researchers are very interested in using audio as interaction medium for text retrieval in spoken documents (conference speeches, broadcast news, etc.) to provide smart access to spoken audio and audio corpora, including music. Such research is targeted at several application areas, mainly spoken document retrieval, media monitoring, and personal entertainment.

2 Introduction

Using audio to access audio information is a relatively new research issue, mainly focusing on the problem of retrieving spoken documents in large audio archives. Advances in automatic speech recognition (ASR) technology have allowed for developing search engines based on the uttered word, an audio version of the text-based, word-spotting engines targeted at text documents.

ASR-based word spotting is effective but consumes vast computing resources, primarily due to the huge recognition vocabulary required. This prevents such solutions from being applicable to limited-resource systems like embedded devices. Limiting the recognizer vocabulary leads to some words not being found, which curtails practical usefulness.

A new approach to the problem of word spotting in uttered documents is based on phonetic search [1] , [2]. The goal is to spot elementary sounds like phones rather than whole words. This approach obviously minimizes problems related to vocabulary size, at the expense of very intensive computing efforts needed to implement the phonetic segmentation of utterances efficiently and reliably. Less challenging is the task of pattern matching applied to phonetic units. The main problem that arises is related to the very short duration of phonetic units compared to the duration of a word. The word-matching approach used in ASR applications (based on dynamic time warping, DTW, and hidden Markov models, HMM) is not enough robust if not supported by a smart decision logic.

A phonetic matcher based on artificial neural networks (ANNs) proves to be more reliable than word matchers based only on DTW and HMM . The disadvantage of the ANN-based solution is its need to be trained for the set of words to be spotted. This is not a disadvantage for a phonetic recognizer because training is required only once and only for a very limited set of patterns.

This research investigates the capacity of a fuzzy-logic engine to work as a phone-pattern matcher and how such capabilities can be extended to the task audio-pattern matching for application to the more general problem of the audio-spotting [3] , [4]. Audio segmentation based on fuzzy-logic processing has been successfully used to separate uttered words (end-point detection) [5], as well as to separate phone units [6] within each uttered word. The peculiarity of fuzzy-logic processing is mainly related to nature of its computing model, which is essentially data-driven [7] ,[8] . The same inference engine can be applied to a different problem when the appropriate set of rules and membership functions is available.

The aim of the research was to model a fuzzy-logic-based, audio-segmentation engine and accompanying fuzzy decision logic that would support decisions at spotting time when a search and retrieval application is to be executed on an audio stream (speech, music [9], sounds, etc.). To do this, hard-computing (digital signal processing algorithms) and soft-computing (fuzzy logic) methods were combined to yield a solution that matches the limited resources (computing power and memory footprint) available on deeply embedded systems.

3 Framework of Audio-spotting System

The whole audio-spotting system consists of an indexing engine that finds occurrences of the elementary audio units (phonemes, stationary sounds, musical tones, etc.) in a given audio pattern (uttered word, audio event, musical theme, etc.) to be spotted in an audio stream. The system consists of two main subsystems: the audio-unit-segmentation subsystem and the audio-spotting subsystem (Fig. 1).

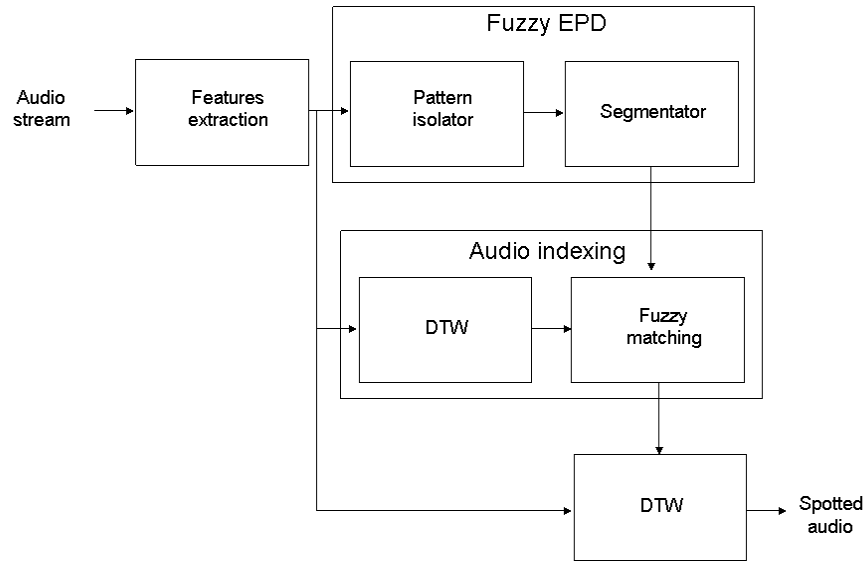

Fig. 1. Architecture of audio-spotting system.

The audio-unit-segmentation subsystem consists of a feature-extraction preprocessor and a fuzzy-logic engine trained to identify the end points of audio units in the incoming audio stream. The segmentation subsystem implements a set of digital signal-processing algorithms to measure certain key, audio-signal features (energy, dominant frequency, pitch, etc.) that help identify the separation point of the audio-units. The fuzzy-logic engine processes the (fuzzified) audio features and infers about the end-points of each audio unit embedded in the audio stream.

The audio-spotting subsystem consists of a DTW-based, pattern-matching processor and a fuzzy-logic engine tuned to infer about the spotting. The pattern-matching processor executes continuous alignment between the target audio to be spotted and a portion of the audio stream. The alignment data and the extracted audio features are fed to the fuzzy-logic engine, which evaluates whether the audio stream can be indexed as spottable.

Each time a spottable portion of audio stream is identified, full audio-spotting is then executed on the target audio pattern. This process can be run at the same time as the index processing (real-time execution) or later, after the whole stream has been indexed (off-line execution).

4 Audio feature measurements

A set of audio features [10] is measured from the audio pattern to be spotted in the audio stream. The audio features to be measured are those required by the segmentation and pattern-matching smart logic that implements the audio-spotting system.
Short-time computation is applied to the audio signal based on an N-points time window as follows:

$$w(m)=0.54-0.46\cos(2\pi m/(N-1)),\ \text{for } 0\le m\le N-1 \qquad (1)$$
$$w(m)=0,\ \text{otherwise}$$

For each window, the following feature measurements are executed: short-time energy *E(n)*, short-time zero-crossing rate *ZCR(n),* and short-time dominant frequency $P_n(k)$.

$$E(n)=\sum_{m=0}^{N-1}[s(m)w(n-m)]^2 \qquad (2)$$

$$ZCR(n)=\sum_{m=0}^{N-1}0.5\,|sign(s(m))-sign(s(m-1)|\,w(n-m) \qquad (3)$$
$$sign(s(n))=1,\ \text{for } x(n)\ge 0$$
$$sign(s(n))=\ \text{otherwise}.$$

$$P_n(k)=\sum_{m=0}^{N-1}s(m)w(n-m)s(m-k)w(n-m+k) \qquad (4)$$

Such audio features are very useful for classifying stationary audio units into classes. For example, $P_n(k)$, when the audio is an utterance, features the gender of the speaker (male (low), female (high)), as shown in the $P_n(k)$ diagram (Fig. 2).

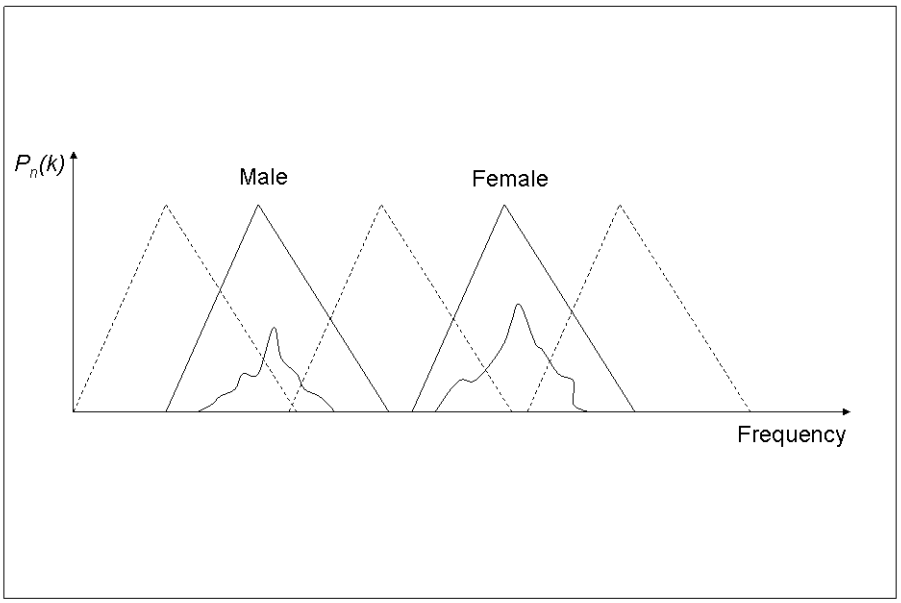

Fig. 2. Dominant frequency distribution in uttered speech maps speaker gender.

4 Audio segmentation and indexing based on fuzzy logic

Three separate tasks were implemented using fuzzy logic:

- Audio end-point detection
- Audio segmentation
- Audio indexing

The same engine was trained for each of the above tasks, generating the needed set of rules and membership functions. The engines for audio end-point detection and audio segmentation have the same set of membership functions to fuzzify their inputs (audio features) and use different sets of rules to make inferences about the audio pattern. Both evaluate the end point of the audio pattern. The former evaluates the audio boundaries of the whole pattern to be spotted, while the second evaluates the end points of the stationary audio units (e.g. phonemes, musical notes, etc.) of the end-pointed audio pattern.

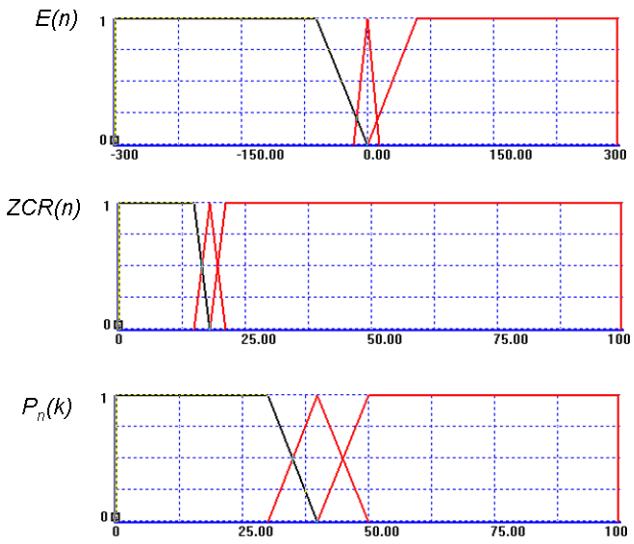

Fig. 3. Membership functions for fuzzifying audio energy, zero-crossing rate, and short-time dominant frequency.

The audio-indexing engine matches each stationary audio unit so as to index it in the processed audio stream. To accomplish this, the measured audio features and the score of the DTW audio-spotter are fuzzified and processed using an appropriately tuned set rules.

The membership functions for fuzzifying the audio features used as inputs for the fuzzy engine are derived directly from the distribution of the crisp measurements of such features.

The rules evaluate the fuzzy inputs as follows :

….

IF ZCR(n) IS Low
    AND E(n) IS Average
    AND P(n) IS High
    THEN Segment IS Vowel

….

IF ZCR(n) IS High
    AND E(n) IS Average
    AND P(n) IS Low
    THEN Segment IS Consonant

….

The defuzzifying membership functions are all of the singleton type.

5 DTW-based audio spotter

Audio spotting is executed by a DTW-based audio-pattern-matching engine, which continuously executes alignment between an audio-pattern template and the occurrence of such patterns in the audio stream. The alignment process consists of searching the audio stream for the starting point of a pattern that is the same as the template. Because the position of pattern to be spotted is unknown, pattern matching needs to be executed by sliding along the whole audio stream (Fig. 4). Using the indexing information, the pattern matcher focuses its action only on the indexed part of the audio stream, thus significantly reducing the time needed to complete the spotting process.

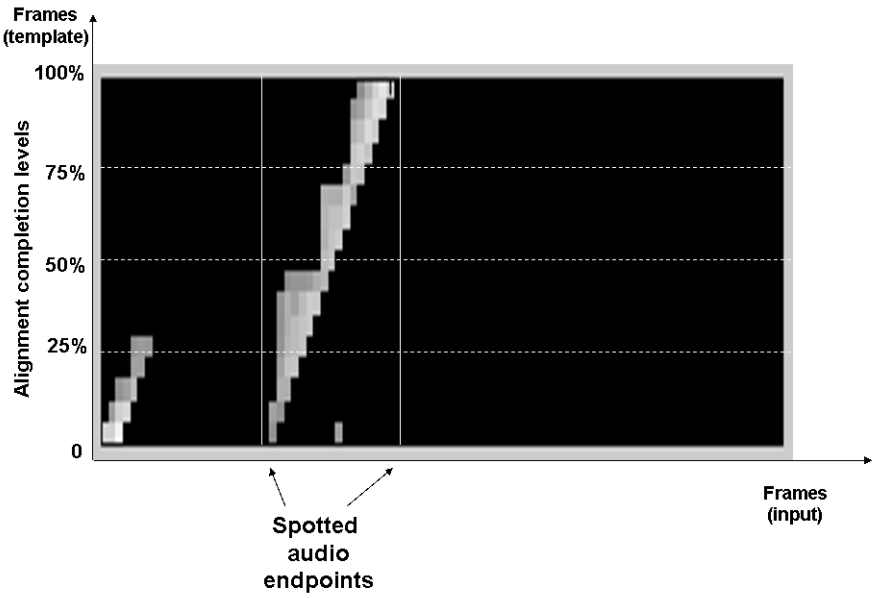

Fig. 4. Pattern matching executed by sliding along the whole audio stream to find best alignment of the template (audio pattern to be spotted) into the audio stream (input).

The DTW-based audio spotter uses dynamic-time warping to align the template pattern with the target pattern. It uses the method for similarity evaluation suggested in the work of Christiansen and Rushforth [11].

6 Conclusion

The proposed framework for smart audio spotting enables faster optimization of the method that exhaustively searches audio information for an audio pattern. It optimizes exhaustive search by indexing the audio stream in terms of the occurrence of a set of stationary audio patterns that belong to the audio pattern to be spotted. Fuzzy logic was used to implement the decision logic related to end-pointing the audio pattern to be spotted and its stationary audio segments, as well as to index audio segments throughout the audio stream.

References

1. Wallace, R.G., Vogt, R.J., Sridharan, S.: A Phonetic Search Approach to the 2006 NIST Spoken Term Detection Evaluation. In: Proceedings Interspeech 2007 - 8th Annual Conference of the International Speech Communication Association, pp. 2385--2388, Antwerp, Belgium (2007)
2. Manos, A.S., Zue, V.W. : A segment-based Wordspotter Using Phonetic Filler Models. In: Acoustics, Speech, and Signal Processing, ICASSP-97., IEEE International Conference on Issue 21-24 Apr, pp. 899 – 902, vol.2 (1997)
3. Ying, Y., Woo, P.: Speech Recognition Using Fuzzy Logic. In: Proceedings of IJCNN '99 – International Joint Conference on Neural Networks, vol. 5, pp. 2962—2964, 10-16 July (1999)
4. Hale, C., Nguyen, C.: Audio Command Recognition Using Fuzzy Logic. In: Proceedings of Wescon 95, San Francisco, CA, November 7 (1995)
5. Malcangi, M.: Improving Speech Endpoint Detection Using Fuzzy Logic-based Methodologies. In: Proceedings of the Thirteenth Turkish Symposium on Artificial Intelligence and Neural Networks, Izmir, Turkey, June 10-11 (2004)
6. Malcangi, M.: Softcomputing Approach to Segmentation of Speech in Phonetic Units. In: International Journal of Computer and Communications, Issue 3, Vol. 3, pp. 41--48 (2009)
7. Malcangi, M.: Soft-computing Approach to Fit a Speech Recognition System on a Single-chip. In: 2002 International Workshop System-on-Chip for Real-Time Applications Proceedings, Banff, Canada, July 6-7 (2002)
8. Ciota, Z.: Improvement of Speech Processing Using Fuzzy Logic Approach. In: Proceedings of IFSA World Congress and 20th NAFIPS International Conference, (2001)
9. Cano, P., Kaltenbrunner, M., Mayor, O., Batlle, E.: Statistical Significance in Song-Spotting in Audio. In: Proceedings of the International Symposium on Music Information Retrieval, Oct. (2001)
10. O'Shaughnessy, D., Speech Communication – Human and Machine. Addison-Wesley, Reading, MA (1987)
11. Christiansen, R.W.,Rushforth, C.K.: Detecting and Locating Key Words in Continuous Speech Using Linear Predictive Coding, IEEE Transactions., Vol. ASSP-25 (1977)