A method for silence removal and segmentation of speech signals, implemented in Matlab

Theodoros Giannakopoulos [1,2]

[1]*Department of Informatics and Telecommunications*
University of Athens, Greece

[2]*Computational Intelligence Laboratory (CIL)*
Insititute of Informatics and Telecommunications (IIT)
NCSR DEMOKRITOS, Greece

email: *tyiannak@gmail.com* web: *www.di.uoa.gr/∼tyiannak*

***Abstract*—This report presents a simple method for silence removal and segmentation of speech signals, implemented in Matlab. The method is based on two simple audio features, namely the signal energy and the spectral centroid. As long as the feature sequences are extracted, a simple thresholding criterion is applied in order to remove the silence areas in the audio signal.**

I. Introduction

Speech signals usually contain many areas of silence or noise. Therefore, in speech analysis it is needed to first apply a silence removal method, in order to detect "clean" speech segments. Some examples where such application may be useful are: speech-based human-computer interaction, audio-based surveillance systems and in general all automatic speech recognition systems. The method implemented here is a very simple example of how the detection of speech segments can be achieved.

II. Algorithm description

A. General

In general, the following steps are executed:

1) Two feature sequences are extracted from the whole audio signal.
2) For each sequence two thresholds are dynamically estimated.
3) A simple thresholding criterion is applied on the sequences.
4) Speech segments are detected based on the above criterion and finally a simple post-processing stage is applied.

B. Feature Extraction

In order to extract the feature sequences, the signal is first broken into non-overlapping short-term-windows (frames) of 50 mseconds length. Then for each frame, the two features, described below, are calculated, leading to two feature sequences for the whole audio signal. The adopted audio features are:

1) Signal Energy: Let $x_i(n), n = 1, \ldots, N$ the audio samples of the $i-$th frame, of length N. Then, for each frame i the energy is calculated according to the equation: $E(i) = \frac{1}{N}\sum_{n=1}^{N}|x_i(n)|^2$. This simple feature can be used for detecting silent periods in audio signals, but also for discriminating between audio classes.
2) Spectral centroid: The spectral centroid, C_i, of the i-th frame is defined as the center of "gravity" of its spectrum, i.e., $C_i = \frac{\sum_{k=1}^{N}(k+1)X_i(k)}{\sum_{k=1}^{N}X_i(k)}$. $X_i(k)$, $k = 1 \ldots, N$, is the Discrete Fourier Transform (DFT) coefficients of the i-th short-term frame, where N is the frame length. This feature is a measure of the spectral position, with high values corresponding to "brighter" sounds. Experiments have indicated that the sequence of spectral centroid is highly variated for speech segments ([1]).

The reasons that these particular features were selected (apart from their simplicity in implementation) are:

1) For simple cases, (where the level of background noise is not very high) the energy of the voiced segments is larger than the energy of the silent segments.
2) If unvoiced segments simply contain environmental sounds, then the spectral centroid for the voiced segments is again larger, since these noisy sounds tend to have lower frequencies and therefore the spectral centroid values are lower.

For more details on those audio features and their application on audio analysis one can refer to [2], [1], [3] and [4].

C. Speech Segments Detection

As long as the two feature sequences are computed, as simple threshold-based algorithm is applied, in order to extract the speech segments. At a first stage, two thresholds (one for each sequence) are computed. Towards this end, the following process is carried out, for each feature sequence:

1) Compute the histogram of the feature sequence's values.
2) Apply a smoothing filter on the histogram.
3) Detect the histogram's local maxima.
4) Let M_1 and M_2 be the positions of the first and second local maxima respectively. The threshold value is computed using the following equation: $T = \frac{W \cdot M_1 + M_2}{W+1}$. W is a user-defined parameter. Large values of W obviously lead to threshold values closer to M_1.

The above process is executed for both feature sequences, leading to two thresholds: T_1 and T_2, based on the energy sequence and the spectral centroid sequence respectively. As long as the two thresholds have been estimated, the two feature sequences are thresholded, and the segments are formed by successive frames for which the respective feature values (for both feature sequences) are larger than the computed thresholds.

D. Post processing

As a post-processing step, the detected speech segments are lengthened by 5 short term windows (i.e., 250 msecconds), on both sides. Finally, successive segments are merged.

III. Execution examples

The main implemented Matlab function is called **detectVoiced()**, and it can be called as follow:

```
[segments, fs] = detectVoiced('example.wav',1);
```

The second argument is not needed if one doesn't want to have the visualized results presented. The first argument is the path of the WAV file to be analyzed. When the above function is called, and the algorithm finished detecting the voiced segments, since the second argument has been provided, a figure is plotted that contains: 1) the energy sequence and the respective threshold 2) the spectral centroid sequence and the respective threshold and 3) the audio signal, plotted with different colours for the areas of the detected segments. At the same time, the detected voiced segments are played sequentially (one needs to press any key in order to have the next segment played). The resulted figure for the call function above, is presented in Figure 1.

The function returns:

1) Cell array "segments": each element of that cell is a vector of audio samples of the corresponding detected voiced segment.
2) The sampling frequency of the audio signal.

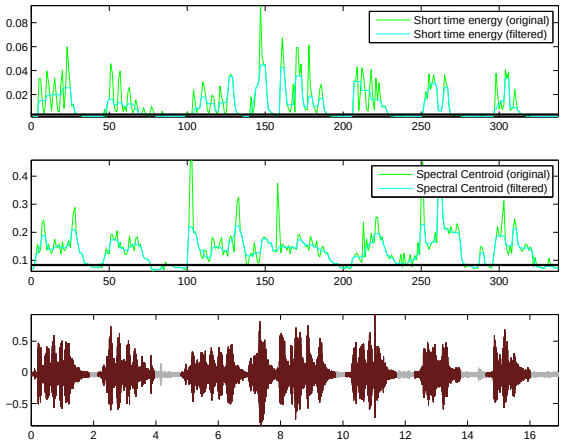

Figure 1. The presented results for an audio example. The first subfigure shows the sequence of the signal's energy. In the second subfigure the spectral centroid sequence is presented. In both cases, the respective thresholds are also shown. The third figure presents the whole audio signal. Red color represents the detected voiced segments.

For example, in order to listen to the first of the detected audio segments one has to type the command:

```
sound(segments{1}, fs);
```

For more details and possible questions, please refer to the Mathworks File Exchange web site.

REFERENCES

[1] T. Giannakopoulos, "Study and application of acoustic information for the detection of harmful content, and fusion with visual information," Ph.D. dissertation, Dpt of Informatics and Telecommunications, University of Athens, Greece, 2009.

[2] S. Theodoridis and K. Koutroumbas, *Pattern Recognition, Third Edition*. Orlando, FL, USA: Academic Press, Inc., 2008.

[3] J. Saunders, "Real-time discrimination of broadcast speech/-music," in *Proceedings of the Acoustics, Speech, and Signal Processing (ICASSP96)*, pp. 993–996.

[4] A. Pikrakis, T. Giannakopoulos, and S. Theodoridis, "An overview of speech/music discrimination techniques in the context of audio recordings," vol. 120, pp. 81–102, 2008.