A&QT-R 2004 (THETA 14)
2004 IEEE-TTTC - International Conference on Automation, Quality and Testing, Robotics
May 13 – 15, 2004, Cluj-Napoca, Romania

FEATURE EXTRACTION FOR CONTINUOUS SPEECH RECOGNITION

Alexandru Căruntu, Gavril Toderean, Liviu Miclea

Technical University of Cluj – Napoca
Email: alex_caruntu@yahoo.com, toderean@cluj.astral.ro, Liviu.Miclea@aut.utcluj.ro

Speech analysis is the first step in every application that involves speech, whether is a speech recognition system or a text – to – speech one. Due to the fact that speech is analyzed on short – time intervals it is hard to make a visual interface for analysis. The program described in this paper is an attempt to realize such an interface. The application is developed in Visual C++ 6.0, and allows the extraction and visualization of speech features frame by frame.

Key words: speech processing, feature extraction, speech analysis.

1. INTRODUCTION.

Speech signals are real, continuous, finite energy waveforms. Although they vary in time, on short periods (15 to 30 ms) they can be considered stationary and their properties can be analyzed. The analysis of the speech signal aims to determine a set of parameters which describes the important characteristics of speech. The fact that speech signals must be analyzed on short intervals makes very difficult the implementation of a visual interface. Most of the existing programs on this field extract the features using commands from the command line, but these have many options hard to remember, and have a few or even none displaying tools. The program that we developed tries to overcome all these problems, using a visual interface which allows the visualization of the features of the speech signal frame by frame. This paper is organized as follows: first the preprocessing and analysis of the speech signals are described, then are given some details about the implementation, and finally conclusions and remarks about the future directions to follow are presented.

2. SPEECH SIGNAL PREPROCESSING.

First some preprocessing is applied to speech wave. Because most part of the energy of the signal lies between 50 Hz and 4 KHz a low – pass or a band – pass filtering is required. This way, low – pass components, which do not contain useful information, are eliminated. The upper constraint is necessary in order to avoid the aliasing which appears thru sampling. Next, the speech signal is digitized with the help of an analog – to - digital converter, with a resolution between 8 and 16 bits. To eliminate the effects of high frequencies attenuation, speech signal is pre – emphasized. Last step before analysis is segmentation, which is realized usually with a Hamming window (Figure 1). For better results frame overlapping is recommended.

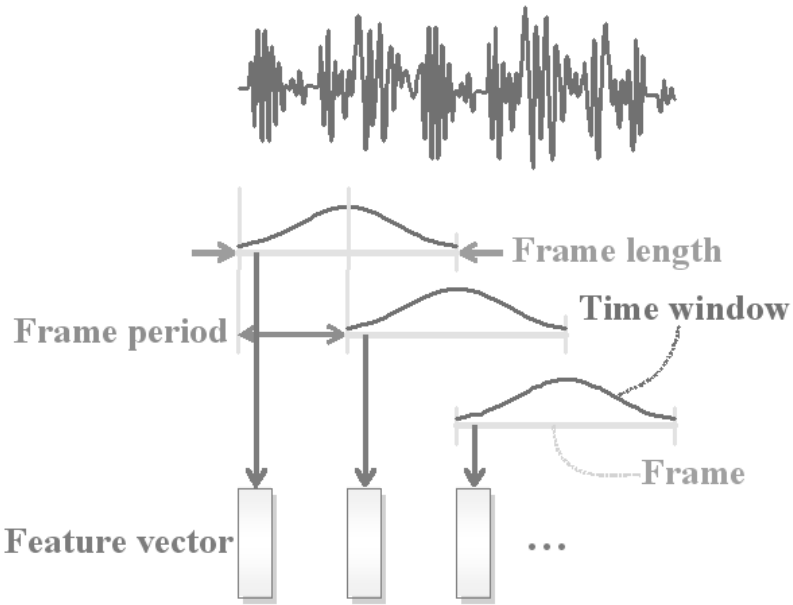


Figure 1. Speech segmentation [1].

3. SPEECH ANALYSIS.

By analyzing the signal in time domain we obtain the maximum and medium amplitude, energy, zero – crossing rate and fundamental frequency. Maximum amplitude gives us information about the voiced or unvoiced character of the speech frame. Time interval between two successive maximums corresponds to fundamental period.

Short – time energy of speech wave is defined as:

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

where *w(m)* is the window, *N* number of samples and *s(n-m)* is a sample from the signal. This parameter gives us information about the voiced (high energy) or unvoiced (low energy) nature of the speech frame. Also, it is very useful in the silence detection in isolated words recognition process.

Zero – crossing rate is calculated with the formula:

$$ZCR = \sum_{n=0}^{N-2}\frac{1-\operatorname{sgn}[s(n)]\operatorname{sgn}[s(n+1)]}{2} . \qquad (2)$$

This parameter estimates the fundamental frequency of the speech wave. For example, for a sine wave of frequency *f_0* ZCR is *$2f_0$*. Also, together with the energy, helps to silence detection.

One of the methods, which are used to determine the fundamental frequency, is based on the autocorrelation function defined as:

$$R_n(k) = \sum_{m=-\infty}^{\infty} s(m)w(n-m)s(m-k)w(n-m+k). \qquad (3)$$

For periodical signals the autocorrelation function has maximums at regular intervals, aspect which is used to determine the fundamental frequency [2].

Frequency domain analysis gives better features for processing than time domain analysis. The excitation and vocal tract can be easily separated in spectral domain. While different utterances of the same sentence can differ in time domain, in frequency domain they are similar. Also, human ear is more sensitive to aspects related to the amplitude of the speech signal than related to its phase, so spectral analysis is used most of the time to extract features that describe speech signal.

The most common method to analyze speech in frequency domain is Fast Fourier Transform. In time, a considerable number of algorithms that exploit the advantages of modern processors have been developed. In our implementation we used a radix-2 one.

Linear Predictive Coding analysis provides a good model for speech signal and is widely used in automatic speech recognition systems. This method fits the parameters of an all – pole model to the speech spectrum, although the spectrum itself is not computed explicitly. LPC coefficients can be found using either autocorrelation method, either covariance method. In our implementation we choused the first one because of its popularity and because it gives also the reflection (or PARCOR) coefficients. The autocorrelation function is evaluated first and the results are converted to LPC coefficients using Levinson Durbin algorithm:

$$\begin{aligned}
&E^0 = R(0)\\
&k_i = \left(R(i) - \sum_{j=1}^{i-1} \alpha_j^{(i-1)} \cdot R(i-j) \right) \Big/ E^{(i-1)},\ i = \overline{1,p}\\
&\alpha_i^{(i)} = k_i \qquad\qquad (4)\\
&\\
&\alpha_j^{(i)} = \alpha_j^{(i-1)} - k_i \cdot \alpha_{i-j}^{(i-1)}\\
&E^{(i)} = (1-k_i^2) \cdot E^{(i-1)}
\end{aligned}$$

where $\alpha_j = \alpha_j^{(P)}$, $1 \le j \le p$ are the LPC coefficients, k_i are PARCOR coefficients and E is frame energy.

Cepstrum is defined as the IFFT of the logarithm of the spectrum. This analysis allows separation of the contribution of the source from that of the vocal tract in the speech signal. A set of features used widely in speech representation is that of cepstrum coefficients obtained from LPC coefficients:

$$c_n = \begin{cases} 0 & n < 0 \\ \ln G & n = 0 \\ a_n + \sum_{k=1}^{n-1} \left(\frac{k}{n}\right) c_k a_{n-k} & 0 < n \le p \\ \sum_{k=n-p}^{n-1} \left(\frac{k}{n}\right) c_k a_{n-k} & n > p \end{cases} . \qquad (5)$$

This way the variability introduced by the excitation is eliminated and better results are obtained in recognition.

Mel cepstral analysis is a perceptual analysis which uses the Mel scale and a cepstral smoothing in order to obtain the final spectrum [3]. First the short – term spectrum of the speech frame is evaluated and then integrated over gradually widening frequency intervals on the Mel scale, with a triangular filter – bank as the one showed in Figure 2.

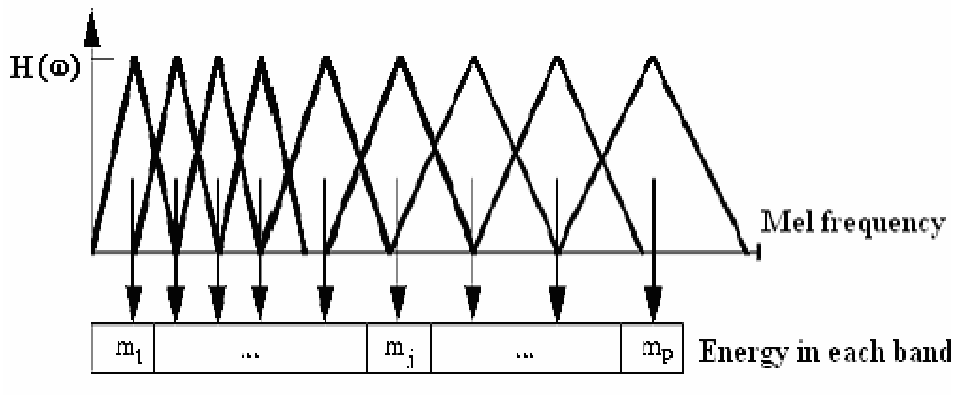


Figure 2. Mel scale triangular filter – bank.

The filter – bank is given by:

$$H_m(\omega) = \begin{cases} \dfrac{\omega - \omega_{m-1}}{\omega_m - \omega_{m-1}}, & \omega_{m-1} \le \omega \le \omega_m \\ \dfrac{\omega_{m+1} - \omega}{\omega_{m+1} - \omega_m}, & \omega_m \le \omega \le \omega_{m+1} \\ 0, & otherwise \end{cases} , \qquad (6)$$

with central frequencies:

$$\omega_m = \begin{cases} 100 \cdot m, & 0 \le m \le 10 \\ 1000 \cdot 1{,}15^{m-10}, & m > 10 \end{cases} . \qquad (7)$$

Next a vector with log energies is evaluated for each filter:

$$S_m = \ln\left[\sum_{k=0}^{N-1} |X(k)|^2 \, H_m(k)\right], \tag{8}$$

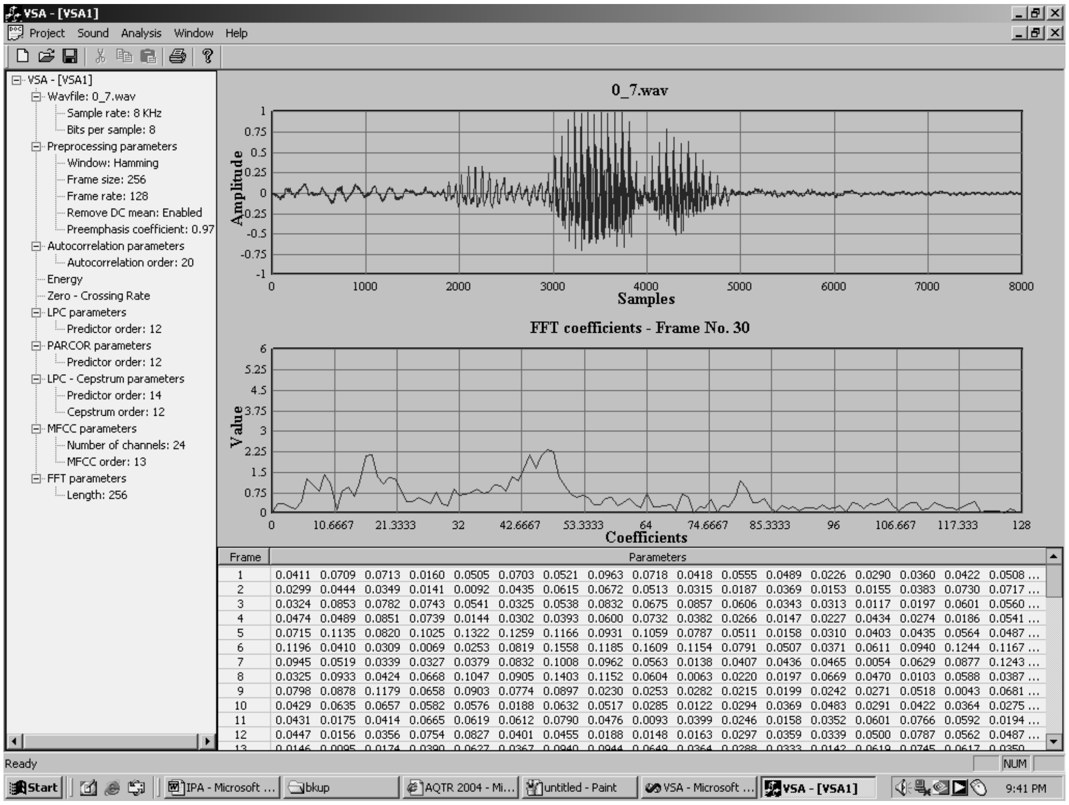
then the MFCC coefficients are obtained using a Discrete Cosine Transform:

$$c_n = \sum_{m=0}^{M-1} S_m \cos\left(\frac{\pi n(m+0{,}5)}{M}\right). \tag{9}$$

This transform is used because the coefficients obtained after the calculus of the power spectra are highly correlated and the cepstral coefficients are uncorrelated, fact that allows the number of parameters to be reduced. In practice they are using 24 to 40 filters but only the first 13 coefficients are used for speech recognition tasks.

4. IMPLEMENTATION.

The program is a MDI application developed in Visual C++ called Visual Speech Analyzer (VSA) and its interface is shown bellow.



VSA - [VSA1]
- Wavfile: 0_7.wav
  - Sample rate: 8 KHz
  - Bits per sample: 8
- Preprocessing parameters
  - Window: Hamming
  - Frame size: 256
  - Frame rate: 128
  - Remove DC mean: Enabled
  - Preemphasis coefficient: 0.97
- Autocorrelation parameters
  - Autocorrelation order: 20
- Energy
- Zero - Crossing Rate
- LPC parameters
  - Predictor order: 12
- PARCOR parameters
  - Predictor order: 12
- LPC - Cepstrum parameters
  - Predictor order: 14
  - Cepstrum order: 12
- MFCC parameters
  - Number of channels: 24
  - MFCC order: 13
- FFT parameters
  - Length: 256

Frame	Parameters																
1	0.0411	0.0709	0.0713	0.0160	0.0505	0.0703	0.0521	0.0963	0.0718	0.0418	0.0555	0.0489	0.0226	0.0290	0.0360	0.0422	0.0508 ...
2	0.0299	0.0444	0.0349	0.0141	0.0092	0.0435	0.0615	0.0672	0.0513	0.0315	0.0187	0.0369	0.0153	0.0155	0.0383	0.0730	0.0717 ...
3	0.0324	0.0853	0.0782	0.0743	0.0541	0.0325	0.0538	0.0832	0.0675	0.0857	0.0606	0.0343	0.0313	0.0117	0.0197	0.0601	0.0560 ...
4	0.0474	0.0489	0.0851	0.0739	0.0144	0.0302	0.0393	0.0600	0.0732	0.0382	0.0266	0.0147	0.0227	0.0434	0.0274	0.0186	0.0541 ...
5	0.0715	0.1135	0.0820	0.1025	0.1322	0.1259	0.1166	0.0931	0.1059	0.0787	0.0511	0.0158	0.0310	0.0403	0.0435	0.0564	0.0487 ...
6	0.1196	0.0410	0.0309	0.0069	0.0253	0.0819	0.1558	0.1185	0.1609	0.1154	0.0791	0.0507	0.0371	0.0611	0.0940	0.1244	0.1167 ...
7	0.0945	0.0519	0.0339	0.0327	0.0379	0.0832	0.1008	0.0962	0.0563	0.0138	0.0407	0.0436	0.0465	0.0054	0.0629	0.0877	0.1243 ...
8	0.0325	0.0933	0.0424	0.0668	0.1047	0.0905	0.1403	0.1152	0.0604	0.0063	0.0220	0.0197	0.0669	0.0470	0.0103	0.0588	0.0387 ...
9	0.0798	0.0878	0.1179	0.0658	0.0903	0.0774	0.0897	0.0230	0.0253	0.0282	0.0215	0.0199	0.0242	0.0271	0.0518	0.0043	0.0681 ...
10	0.0429	0.0635	0.0657	0.0582	0.0576	0.0188	0.0632	0.0517	0.0285	0.0122	0.0294	0.0369	0.0483	0.0291	0.0422	0.0364	0.0275 ...
11	0.0431	0.0175	0.0414	0.0665	0.0619	0.0612	0.0790	0.0476	0.0093	0.0399	0.0246	0.0158	0.0352	0.0601	0.0766	0.0592	0.0194 ...
12	0.0447	0.0156	0.0356	0.0754	0.0827	0.0401	0.0455	0.0188	0.0148	0.0163	0.0297	0.0359	0.0339	0.0500	0.0787	0.0562	0.0487 ...
13	0.0146	0.0095	0.0174	0.0390	0.0627	0.0367	0.0940	0.0944	0.0649	0.0364	0.0288	0.0333	0.0142	0.0619	0.0745	0.0617	0.0350

Figure 3. The interface of the program.

The Project option from the menu allows the user to create a new project for speech analysis, open an existing one or save the current one. We can open a wav file for analysis with the command Open from the menu Sound. Also the preprocessing described on section 2 is implemented here. With the command Preprocessing the user can select the window applied to the signal (Hamming or rectangular), the frame size and frame rate (both of them expressed in samples), whether to remove or not the DC mean or pre - emphasize the signal. The analysis methods described in section 3 can be found under the menu option Analysis. Their results can be saved in a file if the user validates this option.

The interface consists from a tree control which displays information about the wave that is analyzed, the preprocessing parameters and the parameters of each analysis that has been selected. The sound that is analyzed is shown on the upper graph while the features values are shown on the second one. They are displayed frame by frame as the user moves with the mouse on the graph of the signal. Finally, there is a list control which displays features values for all the frames of the signal. The user can switch between parameters by double – clicking on the parameters name from the tree control.

5. CONCLUSIONS AND FUTURE WORK.

The application described in this paper allows the features extraction of the speech signal in a visual manner. We intend to use the parameters extracted this way for an application that performs continuous speech recognition. For this we plan to add to the current program a module which permits speech labeling and segmentation.

REFERENCES:

1. Furui, S., State – of – the – Art Speech Recognition Technology.
2. Căruntu, A., (2003), Stadiul actual în domeniul recunoaşterii vorbirii continue, Referat I, p. 7 – 39.
3. Pop, P.G., Toderean, G., (2002), Comparison of Feature Parameters Used In Speaker Recognition, *Acta Tehnica Napocensis,* Vol. 43, No. 2, p. 43 – 46.