AN OBJECT ORIENTED SIGNAL MODELING MODULE FOR SPEECH RECOGNITION FEATURE EXTRACTION

Submitted to

Dr. Laura Griffeth
AEE3203 — Introduction to Technical
Writing in Agricommunication

By

Richard Duncan
November 13, 1998

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

a description of

AN OBJECT ORIENTED SIGNAL MODELING MODULE FOR SPEECH RECOGNITION FEATURE EXTRACTION

submitted to:

Dr. Laura Griffeth
AEE3203 — Introduction to Technical
Writing in Agricommunication

report by:

Richard Duncan

November 12, 1998

research by:

Richard Duncan, Vishwanath Mantha, Yufeng Wu, Jie Zhao

Signal Processing Group
The Institute for Signal and Information Processing

Audience Description

The intended audience for this descriptive report is a collection of computer engineering, electrical engineering, and computer science students and professionals with an interest in signal processing and computer speech recognition. Each reader will have a strong background in digital signal processing (DSP). While not a requirement, some knowledge of continuous speech recognition systems would be beneficial.

Table of Contents

Introduction

In order for a computer to recognize human speech with current technology, the speech must first be processed into observation vectors representing events in the probability space (Picone, 1993). This process, known as signal modeling, is the function of the front-end module. Using these acoustic observation vectors and some language constraints, a network search algorithm (performed by a decoder) finds the most probable sequence of events to hypothesize the textual content of the audio signal (Picone, 1993).

Background

In order for the front-end to model useful observation vectors for speech recognition, it must extract important features from the speech waveform that are relatively insensitive to the talker and channel variability which is unrelated to the message content (Jankowski, Hoang-Doan, Lippmann, 1995). The algorithms used by the front-end are composed almost entirely of standard signal processing techniques, such as digital filter banks, linear predictive coding, and homomorphic analysis. These algorithms are successful because they model the speech signal consistently with the human auditory perceptual system —in the frequency domain (Mermelstein, Davis, 1980). Specifically, the short time spectral envelope is needed since speech is a time-variant signal (Rabiner, Juang, 1993). Furthermore, the addition of physiological knowledge of the vocal articulatory system can be applied to the problem in order to increase recognition performance (Jankowski, Hoang-Doan, Lippmann, 1995).

Purpose

The Institute for Signal and Information Processing (ISIP) has developed a standard comprehensive front-end module for a speech recognition system (Duncan, Mantha, Wu, Zhao, 1998). Several standard front-ends have been implemented, including mel cepstral, perceptual linear prediction, filter bank amplitudes, and delta features. The framework for this system was carefully designed to ensure simple integration with the speech recognition system (Deshmukh, Ganapathiraju, Hamaker, Picone, 1998). The code itself is written in tutorial fashion, with a direct correlation between algorithmic lines of code and equations in this technical paper. This report aims to describe the signal processing algorithms used in the ISIP front-end.

There are advantages and disadvantages to each algorithm described in this paper. For example, while linear prediction (LP) coefficients can generally be computed with fewer resources, the compressive nature of the transformation makes the model less robust to noise. Most current state of the art systems use one energy coefficient, twelve Fourier transform-derived cepstral coefficients, and delta and delta-delta derivatives of the first thirteen coefficients. While the ISIP front-end is capable of producing output models consistently with other state of the art systems, it can also be used to study the differences between the different algorithms. A graphical user interface (GUI) is available to facilitate such a study further.

Major Topics

This report is broken into two sections. First, an overview of the general system structure is discussed. This section focuses mainly on the pre- and post-processing, with only a cursory scan of the modeling algorithms. This section also describes how the front-end is interfaced to the full speech recognition system. The second part of the report provides an in depth look at the algorithms which form the heart of the system.

System Structure

The modular design of the front-end is shown in Figure 1. After pre-processing (windowing and pre-emphasis are not shown on the diagram), three basic operations can be performed on the speech signal. These general algorithms are filter bank amplitudes (FBA), the Fourier transform (FFT), and linear prediction (LP) (Rabiner, Juang, 1993). From the digital filter bank a power estimation may be directly computed. Perceptual linear prediction (PLP) is a post-processing step for LP coefficients, acting as a cascaded filter. The FT, LP, and PLP algorithms compute the spectrum of the signal, which is then processed into usable parameters in one of two ways. The first method is filter bank amplitudes, similar to the general FBA algorithm which operated on the original signal. It computes a reduced number of averaged sample values from the spectrum. Computing the cepstrum is an alternate method of processing this spectrum. The details of these algorithms are further described in the next section.

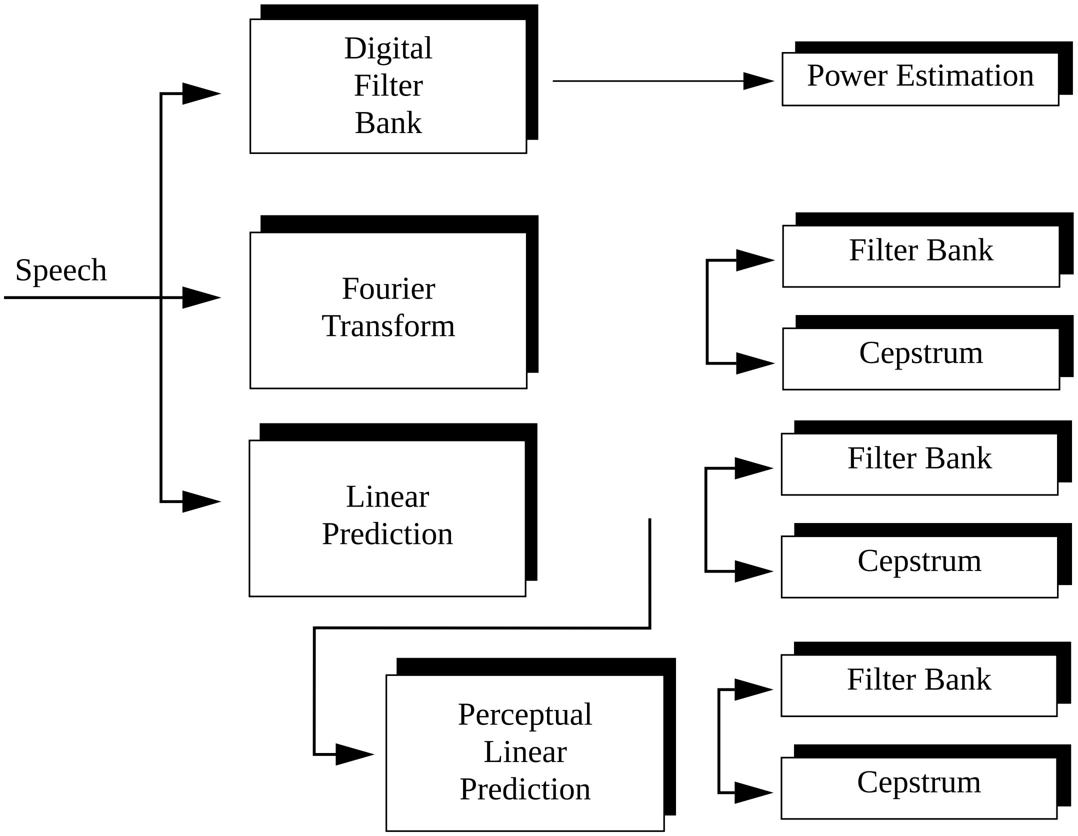

Figure 1. System block diagram.

Windowing and I/O

In order to extract short-term frequency information from a time-varying speech signal, a window function must be applied. The simplest window function is rectangular in shape; however, oftentimes more complicated shapes produce a more desirable windowed signal (Rabiner, Schafer, 1978). For speech processing, the Hamming window is used almost exclusively (Picone, 1993). The Hamming window is a special form of the general Hanning window, shown in equation (1), with $\alpha_w = 0.54$.

$$w(n) = \frac{\alpha_w - (1 - \alpha_w)\cos(2\pi n/(N_s - 1))}{\beta_w} \qquad (1)$$

The user can vary the window duration, window type, and frame duration. A physiological investigation into the human auditory system reveals the quickest movements of the vocal articulators are on the order of 10 ms. This means if the speech signal is averaged and evaluated (framed) every 10 ms, almost no information will be lost. Since the window duration is longer than the frame duration, efficient buffering algorithms reduce the I/O complexity of the task by only reading in a single frame of data at each time step. Compared to the decoding phase of speech recognition, a front-end’s computational cost is negligible (Picone, 1993). Nevertheless, poorly written code at any stage in the process can bog down a production system run in real-time.

Coefficient Concatenation

All coefficients from the various algorithms are concatenated into a single observation vector for each frame. To interpret the meaning of a number from its position, sequentially add up the number of each specified coefficient. For example, if energy and twelve FFT-derived cepstral coefficients are specified, the first number output is the energy, the fifth number is the fourth cepstral coefficient, etc. This is an efficient method for passing parameters to the network search algorithm because it decouples the signal modeling information into a vector of pure numbers for pattern recognition. The decoder need only be trained on the same coefficients as the test data.

Vector Post-Processing

Higher order time derivatives of the signal measurements can be added to better characterize temporal variations in the signal. Since the measurements previously described operate on a single window of data, they are considered zeroth order derivatives. First and second derivatives are now commonly used in speech recognition systems.

Signal Modeling Algorithms

The algorithms described in this section (with the exception of delta features) operate on a single window of speech data. The code itself is written in clear and simple form, referencing blocks of code directly to the equations described in this section where applicable. New signal modeling algorithms are added at this point in the structure.

Filter Bank Amplitudes

The digital filter bank is one of the most fundamental concepts in speech processing. A filter bank can be regarded as a crude model of the initial stages of transduction in the human auditory system. Each filter in the digital filter bank is usually implemented as a linear phase filter. The filter equations for a linear phase filter implementation can be summarized as follows:

$$s_i(n) = \sum_{j = \left(-\left(N_{FB_i} - 1\right)\right)/2}^{\left(N_{FB_i} - 1\right)/2} a_{FB_i}(j) s(n + j) , \qquad (2)$$

where $a_{FB_i}(j)$ denotes the j^{th} coefficient for the i^{th} critical band filter. The number of filter banks normally is odd when implementing linear phase filters. The basic merit of the algorithm is that certain filter outputs can be correlated with certain classes of speech sounds.

The output of filter bank analysis is a vector of power values for each frame of speech data. Usually these values are combined with other parameters, such as mean energy, to form the final signal measurement vector. Since the analysis is based entirely on linear processing, the technique is generally robust to ambient noise.

Fourier transform-derived coefficients. Simple Fourier transform-based filter banks designed for front-ends obtain the desired frequency resolution on a mel-scale (the mel-scale is described on page 6). To implement this filter bank, the window of speech data is transformed into the frequency domain by the Fourier transform. The magnitude of the spectral coefficients are then binned through correlation with triangular filters equally spaced on the mel-scale (Young, 1995). As defined here, binning means that each spectral coefficient is multiplied by the corresponding filter gain; the bin value is the accumulation of every such product. Thus, each filter bank coefficient represents the average spectral magnitude in the filter channel,

$$S_{avg}(f) = \frac{1}{N_s} \sum_{n=0}^{N_s} w_{FB}(n)|S(f)| , \tag{3}$$

where N_s represents the number of samples used to obtain the averaged value, $w_{FB}(n)$ represents a weighting function (filter gain), and $S(f)$ is the magnitude of the frequency response computed by the FFT.

Linear prediction-derived coefficients. Linear predictive (LP) analysis is an estimate of the autoregressive all-pole model $A(w)$ of the short-term power spectrum of speech $P(w)$. Alternately, LP analysis is a means for obtaining the smoothed spectral envelope of $P(w)$. The major disadvantage of the LP model in speech analysis is that $A(w)$ approximates $P(w)$ equally well at all frequencies of the analysis band. This property is in consistent with human hearing, which tends to be nonlinear above 800 Hz. Consequently, LP analysis does not preserve or discard the spectral details of $P(w)$ according to auditory prominence. The algorithms in this section improve the basic LP model.

The spectrum is computed through application of the Fourier transform to the linear prediction coefficients. Since there are fewer points in the LP model, this approach is more efficient. From this LP-derived spectrum, filter banks are applied in exactly the same way as for the FT-derived spectrum. These coefficients are known as LP-derived filter bank amplitudes.

Mel Frequency Cepstral Coefficients

A mel is a psychoacoustic unit of measure for the perceived pitch of a tone, rather than the physical frequency. The correlation of the mel to the physical frequency is not linear, as the human auditory system is a nonlinear system. A mapping between the mel scale and real frequencies was empirically determined by Stevens and Volkman in 1940 (Picone, 1993). The scale is roughly linear below 1000 Hz, then decays logarithmically. It is described mathematically as:

$$Mel(f) = 2595\log_{10}(1 + f/700)\,. \tag{4}$$

This nonlinear scale is invaluable to speech coding in that it reduces the sample space with minimal perceptual loss. In practice, filters banks are evenly spaced along the mel scale.

A homomorphic system is useful for speech processing because it offers a methodology for separating the excitation signal from the vocal tract shape (Picone, 1993). One space which offers this property is the cepstrum, computed as the inverse discrete Fourier transform (IDFT) of the log energy (Deller, Proakis, Hansen, 1993). This signal is by definition minimum phase, another useful property. Cepstral coefficients are computed by the following equation:

$$c(n) = \frac{1}{N_s}\sum_{k=0}^{N_s} \log|S_{avg}(k)|e^{j\frac{2\pi}{N_s}kn}, 0 \le n \le N_s - 1\,, \tag{5}$$

where $S_{avg}(k)$ is the average signal value in the k^{th} filter channel. In practice, the discrete cosine transform may be used in lieu of the IDFT for computational efficiency.

Perceptual Linear Prediction

Perceptual linear predictive (PLP) analysis is a relatively new method for the analysis of speech signals. It is an improvement over the widely used LP (Linear Predictive) analysis. In PLP analysis, the all-pole modeling is applied to an auditory spectrum derived by (a) convolving $P(w)$ with a critical band masking pattern, followed by (b) resampling the critical band spectrum at approximately l Bark intervals, (c) pre-emphasis by a simulated fixed equal loudness curve, and finally (d) compression of the resampled and pre-emphasized spectrum through the cubic root non-linearity, simulating the intensity-loudness power law. The low order all-pole model of such an auditory spectrum has been found to be consistent with several phenomena observed in speech perception (Hermansky, 1990). The block diagram of PLP Analysis is shown in Figure 2.

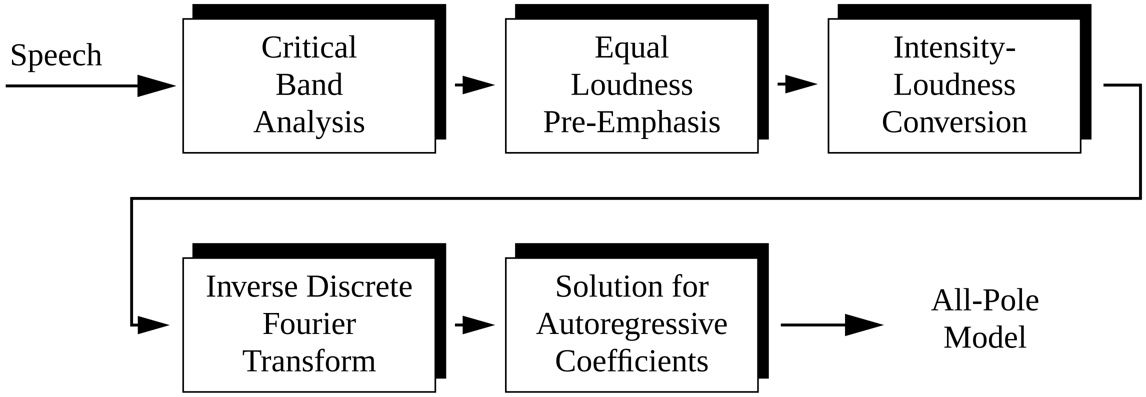
Speech
Critical Band Analysis
Equal Loudness Pre-Emphasis
Intensity-Loudness Conversion
Inverse Discrete Fourier Transform
Solution for Autoregressive Coefficients
All-Pole Model

Figure 2. Block Diagram for PLP Analysis

After windowing, the real and imaginary components of the short-term speech spectrum are squared and added to get the power spectrum,

$$P(w) = Re[S(w)]^2 + Im[S(w)]^2 . \tag{6}$$

The spectrum $P(w)$ is warped along its frequency axis into the Bark frequency Ω by

$$\Omega(w) = 6\ln\left\{(w/(1200\pi)) + [w/(1200\pi)^2 + 1]^{0.5}\right\} \tag{7}$$

where w is the angular frequency in rad/s. The resulting warped power spectrum is then convolved with the power spectrum of the simulated critical band masking curve $\psi(\Omega)$,

$$\psi(\Omega) = \begin{cases} 0 & ,\Omega < -1.3 \\ 10^{2.5(\Omega + 0.5)} & ,-1.3 \le \Omega \le -0.5 \\ 1 & ,-0.5 \le \Omega \le 0.5 \\ 10^{-1.0(\Omega - 0.5)} & ,0.5 \le \Omega \le 2.5 \\ 0 & ,2.5 < \Omega \end{cases} . \tag{8}$$

The discrete convolution of $\psi(\Omega)$ with (the even symmetric and periodic function) $P(w)$ yields samples of the critical band power spectrum,

$$\theta(\Omega_i) = \sum_{i=-1.3}^{2.5} P(\Omega - \Omega_i) \cdot \psi(\Omega) . \tag{9}$$

This convolution significantly reduces the spectral resolution of $\theta(\Omega)$ in comparison with the original $P(w)$. This also allows for down sampling.

The sampled $\theta(\Omega(w))$ is pre-emphasized by a simulated equal loudness curve, which is an approximation of the variable sensitivity of human hearing at different frequencies. This pre-emphasized function is then amplitude compressed using cubic root amplitude compression.

The final operation of PLP analysis is the approximation of $\theta(\Omega)$ by the spectrum of an all-pole model using the autocorrelation method of all-pole spectral modeling. The principle is to apply the inverse DFT to $\theta(\Omega)$ and find the dual of its autocorrelation function. These PLP coefficients can be processed through the same methods as standard LP coefficients to extract observation vectors.

Delta Features

The performance of a speech recognition system is enhanced greatly by adding time derivatives to the basic static parameters. The first-order derivatives are referred to as delta features; the second-order derivatives are referred to as delta-delta features.

In digital signal processing, there are several ways to approximate the first order time derivative of signal.

$$s^*(n) = \frac{\partial}{\partial t}s(n) = s(n) - s(n-1) \tag{10}$$

$$s^*(n) = \frac{\partial}{\partial t}(n) = s(n+1) - s(n) \tag{11}$$

$$s^*(n) = \frac{\partial}{\partial t}s(n) = \sum_{w=-N}^{N} ws(n+w) \tag{12}$$

Equations (10) and (11) are known as backward and forward differences, respectively. Equation (12) is often referred to as regression analysis. Similarly, the second-order time derivatives are approximated by reapplying these equations to the output of the first-order differentiator (Young, 1995).

Since differentiation is inherently a noisy process, computing derivatives of smoothed parameters is desirable. The regression analysis as shown in equation (12) is a popular way to achieve this result. Since this equation computes differences symmetrically placed around the sample at time n, it uses a combination of N previous samples in each direction to compute the current value. Hence some measure of smoothing is inherent.

Regression analysis is used in this front-end to compute delta features. The first formulation is simply a weighted version of equation (12):

$$d_n = \frac{\sum_{w=1}^{dw} w(c_{n+w} - c_{n-w})}{2\sum_{w=1}^{dw} w^2}, \tag{13}$$

where d_n is a delta coefficient at frame n, c_{n-w} and c_{n+w} are static parameters before and next to the current frame coefficient c_n, and dw is the delta window size. Since the regression formula depends on past and future speech parameter values, some modifications are required for the beginning and end of the speech data. The formulas shown in (14) account for these conditions.

$$d_n = \frac{\sum_{w=1}^{dw} w\left(c_{n+w} - c_0\right)}{2\sum_{w=1}^{dw} w^2}, n < dw, \quad d_n = \frac{\sum_{w=1}^{dw} w\left(c_{dw} - c_{n-w}\right)}{2\sum_{w=1}^{dw} w^2}, n > dw \tag{14}$$

Summary

The processing of speech data into observation vectors which represent events in the probability space is performed by the front-end module. Frequency domain signal analysis techniques tend to be more insensitive to talker and channel variability than time domain approaches, thus extracting more useful information for speech-to-text systems. The standard algorithms employed are mean energy, digital filter banks, the Fourier transform, linear prediction, the cepstrum, and difference equations. Physiological knowledge of the human auditory and vocal articulatory systems is applied (the mel and Bark scales, perceptual linear prediction, frame duration, etc.) to the standard signal processing techniques to better model speech and increase recognition performance.

The front-end module described in this paper interfaces directly with the ISIP speech recognition system. A public domain implementation of all algorithms examined is available on the ISIP website (Duncan, Mantha, Wu, Zhao, 1998).

References

Davis, S. B., & Mermelstein, P. (1980). Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences. IEEE Transactions on Acoustics, Speech, and Signal Processing, ASSP-28, 4, 357-366.

Deller, J., Proakis, J., & Hansen, J. (1993). Discrete-Time Processing of Speech Signals. New York: MacMillan.

Deshmukh, N., Ganapathiraju, A., Hamaker, J., & Picone, J. (1998, September). An Efficient Public Domain LVCSR Decoder. Proceedings of the Hub-5 Conversational Speech Recognition (LVCSR) Workshop. Linthicum Heights, Maryland: National Institute for Standards and Technology.

Duncan, R. J., Mantha, V., Wu, Y., & Zhao, J. (1998). *Implementation and Analysis of Speech Recognition Front-Ends* [Online]. Available: http://www.isip.msstate.edu/resources/courses/ece_4773/projects/1998/group_signal, Institute for Signal and Information Processing, [1998, November 12].

Hermansky, H. (1990). Perceptual Linear Predictive (PLP) Analysis of Speech. Journal of the Acoustical Society of America, 4, 1738-1752.

Jankowski, C. R., Hoang-Doan, H., & Lippmann, L. P. (1995). A Comparison of Signal Processing Front Ends for Automatic Word Recognition. IEEE Transactions on Speech and Audio Processing, 3, 4, 286-292.

Picone, J. (1993). Signal Modeling Techniques in Speech Recognition. IEEE Proceedings, 81, 9, 1215-1247.

Rabiner, L. R., & Juang, H. (1993). Fundamentals of Speech Recognition. Englewood Cliffs, NJ: Prentice Hall.

Rabiner, L. R., & Schafer, R. W. (1978). Digital Processing of Speech Signals. Englewood Cliffs, NJ: Prentice Hall.

Young, S. (1995). The HTK Book: for HTK Version 2.0. Cambridge: Cambridge University Press.