IMPROVED SPEECH RECOGNITION USING QUANTIZED FREQUENCY DOMAIN FILTERS

Stephen G. Boemler and R Bradley Cope
Naval Air Warfare Center Training Systems Division
Orlando Florida

ABSTRACT

The challenge for today is to build computer systems that are very easy to use. This implies that a human and a machine should be able to act as one. It implies that the machine needs to be "trained" to understand human speech. And it implies that the machine should be able to recognize anyone's speech under any condition. This is the challenge that the speech recognition research laboratory in the Modeling & Simulation Development Branch at NAWCTSD has undertaken. This paper details the R&D effort to develop software that will enable a computer system to understand the spoken word in a noisy environment. This is an important problem that must be solved if the future training devices, as envisioned by the US Navy, are to be realized in the 21st century.

Statistical modeling used in modern independent speech recognition represents the inherent variability in human speech. Adversely, these models are highly susceptible to background noise. This research exploits the response of the statistical modeling used in recognition of speech patterns specifically to mitigate the adverse effects of noise. Time domain audio from the microphone is transformed into the frequency domain by way of the Fast Fourier Transform (FFT) algorithm. Subsequently, a mel spectrum window and cosine transform are applied to create spectral feature vectors. Models of the spoken input are created by training statistical models with the spectral feature vectors. The structure of the frequency domain is well understood in terms of the quantized method used to process and store signals as data. Computationally, it is now possible to manipulate individual frequency components to remove unwanted noise spectra. This paper describes the digital signal process which has been developed to remove frequencies that are not of interest to the information content in the audio signal and thereby describe a near noise free model and associated filter. This model and filter are integrated into the speech recognizer in a holistic method to solve the background noise problem.

ABOUT THE AUTHORS

Steve Boemler is an Electronics Engineer with the speech recognition research laboratory Simulation & Models Division, Research and Engineering Competency/Training Systems Department at NAWCTSD. He holds a BSEE from the University of Florida (8/75) and a Master of Science in Electrical Engineering from the University of South Florida (5/95). Steve's previous experience was in the science of electrical/electronic measurement at the Naval Metrology Standards Laboratory in Pensacola, Florida.

Brad Cope is an Electronics Engineer with the Simulation & Models Division of the Naval Air Warfare Center Training Systems Division. He is currently responsible for exploratory and applied speech recognition research for Navy training. He holds a BSEE from Temple University (5/83) and a Masters in Engineering Science from Penn State University (5/91). Brad was previously responsible for applied research in advanced flight control and electromagnetic effects at the Naval Air Development Center in Warminster, Pennsylvania.

IMPROVED SPEECH RECOGNITION USING QUANTIZED FREQUENCY DOMAIN FILTERS

Stephen G. Boemler and R Bradley Cope
Naval Air Warfare Center Training Systems Division
Orlando Florida

INTRODUCTION

There is a problem with noise in speech recognizers, but a new method minimizes its effect. This paper details an R&D effort to develop software that will enable a computer system to understand the spoken word in a noisy environment. This is an important problem that must be solved if the future training devices, as envisioned by the US Navy, are to be realized in the 21st century. One such training device is an Air Traffic Control (ATC) Tower Simulator, which teaches the trainee to speak properly to simulated pilots under simulated conditions. The ATC speech recognizer must be able to recognize the trainee's commands over the simulated aircraft noise plus the noise from the other nearby conversations. Generally the initial approach is to improve microphone design and then add low pass filters, however the remaining noise still interferes with speech recognition. What is needed is a software solution that is imbedded in the speech recognizer code. The approach taken in this research is from a holistic perspective. What is implied here is that the filtering process is built into the speech recognizer as a whole. This is fundamentally different from what is usually done by inserting a filter into the front end of a system as an autonomous component part. What is proposed is that the speech models are designed and built with filtered data such that the entire digital-processing scheme is based on the theoretical premise of our filtering techniques.

THEORETICAL PREMISE

In this paper we hypothesize that the frequency spectrum of a noise free speech signal contains low amplitude frequency components which are not required for recognition. By reducing the content of the frequency spectrum to only high amplitude components, and then building new models based on this reduced spectrum, the resulting system will necessarily demonstrate an improved signal to noise ratio.

This hypothesis is grounded in the mathematical approximations that are applied when the continuous transformation theory developed by Fourier is adapted for use in a digital signal processing (DSP) application. A Fourier Transformation is based on the discovery that a signal in time is composed of an infinite number of sine wave frequencies. The DSP assumption is that continuous time (t) can be separated into discrete quantities by sampling every T seconds. By quantifying time, mathematical integrals of calculus are approximated as summations Σ over an infinite number (n) of samples, and the continuous time domain signal x(t) is replaced by the discrete x(nT).

The Digital Fourier Transformation (DFT) analyzes the domain of frequencies f into an infinite summation of harmonic complex sinusoids $\exp(-j\omega nT)$ with amplitudes proportional to x(nT). The spectrum $\mathbf{X(\omega)}$ of these sinusoids is a periodic function of the continuous radial frequency $\omega = 2\pi f$.

$$\mathbf{X(\omega)} = \Sigma\ \mathbf{x(nT)}\ \ \mathbf{\exp(-j\omega nT)} \qquad (1)$$

In modern speech recognition systems with frequency bandwidths under 8 KHz, (such as the Carnegie Mellon University's SPINX II system) the continuous radial frequencies are quantized into 256 frequency bins k of

the factor W_N where n = 0,1,…,N-1 and k = 0,1,…,255. The spectrum of these frequency bins is now represented as a discrete function of k.

$$X(k) = \sum_{n=0}^{N-1} x(nT) \, W_N^{nk} \qquad (2)$$

To visualize the above equation, take for example a short 10-millisecond burst of sound. The frequency domain **X(k)** may be plotted as a bar graph with 256 bars across the horizontal axis. Each bar represents a quantum k frequency, and the height of each bar represents the total of N amplitudes. Each bar amplitude is the sum of however many signal samples occurred during the t = 10 millisecond signal (where N = t / T), and this sum is weighted by the total number of harmonics (also N) that produced the sound. The weight [given by $W_N=\exp(-j2\pi / N)$ raised to the power nk] for each bar is a factor of the phase and is a complex number (with imaginary j) which is commonly referred to as the twiddle factor .

It is premised that the information necessary for speech recognition of a noise free spectrum is contained within the 56 frequency bins displaying the largest relative amplitudes. This is because the summation over these 56 terms is normally about 97% of the value of the summation over all 256 terms. This premise is the result of observations on frequency patterns of human utterances that display energy groupings, which were correlated with small numbers of mathematical terms. The average number of terms appeared to be around 56. Although this number is arbitrary, it has been chosen based on empirical tests of various numbers of terms and has resulted in a convenient starting point. This assumed number will be the subject of further research. Our premise then implies that 97% of the energy (amplitude squared) still remains even when 200 low amplitude terms are neglected.
These terms are thereby identified with respect to their frequency bins in the spectrum and a pattern is established. If noise is now added to this speech signal, the same 200 unimportant frequency bins can be neglected irrespective of their new amplitudes. This implies that since about 78% (200/256) of the signal can be eliminated, the added noise will also be reduced by 78% (assuming white noise here – other noise such as background voices will be addressed later).

This even reduction of signal and noise frequencies produces an uneven reduction of signal and noise amplitudes. The energy distribution of white noise is uniform over the spectrum so that eliminating 200 frequencies will eliminate 78% of the noise energy but only 3% of the signal energy. This will result in a significant improvement in signal to noise ratio, which will improve the speech recognizer's ability to operate in noise.

NOISE FILTERING

The first step in designing a filter to eliminate white (or other) noise is to reprocess the output data from the Fourier Transformation software routine. This data is ordered in a frequency series of coefficients X(k) which are in a numerical format (generally floating point). We reordered this data in descending value (amplitude) so that the relatively lowest 200 amplitudes can be identified and a lowest amplitude threshold established. The data is then reassembled in the original DFT output form, except that the identified "noise" amplitudes below this threshold are set to zero. The filtered frequency domain may be thought of as a bar graph consisting of 256 discrete frequency bins on the horizontal axis, but with only 56 of these frequency bins having any height. A correlated filter is also generated and stored such that for these 56 quantized frequencies the amplitude is set to one (unity gain), and all other frequencies have zero gain. This filter will be referred to as a quantized frequency domain filter or briefly as a comb filter. Multiplying this filter by the input is equivalent to a threshold sort and reorder process.
This digital signal process is repeated every 10 milliseconds. This interval of time is chosen based on the assumption that the frequencies of human speech can be

considered stable for short periods. This is a necessary approximation for the analysis of a continually changing speech signal.

American English is analyzed into 48 linguistically distinct speech sounds commonly referred to as phonemes. In the Sphinx II system, a phoneme is modeled with 5 stationary states that are process every 10-milliseconds and are named "senomes". A unique filtering routine is preformed for each senome.

We have named this threshold sort / reorder process the "primary_format_locator" routine and inserted this code in the calculate_FFT routine of the CMU SPHINX II system. Because the CMU project is funded by DARPA, NAWCTSD has access to this software through a shared US Government research arrangement.

Since this modification of the input speech changes the characteristics of the frequency spectrum, the next step is to construct a new speech model based on these modified characteristics. The SPHINX II system uses a Hidden Markov Model (HMM).

HIDDEN MARKOV MODEL

The variability of human speech is inherent to the Hidden Markov Model. The model is built from a representative set of human subjects, each producing a set of utterances that will occur in the desired phraseology. Ideally, each possible utterance will have been spoken seven to ten times for each subject. A phonetic recognition system requires seven to ten occurrences of each phoneme in the context for which it will be used and there are forty-eight identified phonemes in the English language. Each phoneme model then represents this variability. Additionally, the number of phoneme models (48) must be cubed because the phoneme will occur with many different surrounding phonemes thus changing its sound slightly by coarticulation. This coarticulation effect creates a new sound referred to as a "triphone".

Speech recognition begins by sampling an analog microphone input with an analog-to-digital converter (A/D). The sampling rate is 16 KHz, which is more than twice the highest signal frequency, commonly known as the Nyquist frequency, and which prevents aliasing of the sampled signal. The digital audio is then transformed from the time domain to the frequency domain by way of a Fast Fourier Transform (FFT) which is the generic name for a class of computationally efficient algorithms that implement the DFT. These transforms are performed every 10 milliseconds on the input and the resulting frequency spectrum is partitioned using a set of Hamming windows. The bandwidths of these frequency windows are based on the biologically inspired mel scale which has more resolution at the lower frequencies. Subsequently, the mel spectrum is multiplied by a series of harmonically related cosine functions which are then used to characterize the cepstral energy, thus obtaining the mel frequency cepstral coefficients (MFCC's). We use a ten millisecond period because of the mechanical operation of the human articulatory organs, especially the glottis, where it is assumed that the time is short enough for the signal to be stationary. Each of the feature vectors in this system represents a ten-millisecond sound referred to as a senome or a state. Hidden Markov Models are developed by re-estimation of each possible state and establishing a distribution of the MFCC classifications that could occur for each ten millisecond period. These models use a feed forward state transition topology to model the transitions between each sub-phonetic window. The Viterbi (3), or Baum-Welch re-estimation algorithms, then compute the statistical likelihood of the model producing a given spoken input or sequence of senome sub-phonetic observations.

Finite state machine HMM's are partitioned phonetically or lexically. When the partitioning is phonetic, as is the case for our system, words are constructed by concatenating the phonetic based models together. Each ten millisecond state of the phonetic model has a probability distribution for the feature vectors that can occur for that moment in time. Initially, the probability distribution is established by aligning the acoustic signal with a prescribed phonetic topology for the expected word.

Subsequently, the probability distribution is set by re-estimating a large set of feature vectors specific to the phraseology from a variety of human subjects. The prescribed phonetic topology is defined in a phonetic dictionary. This dictionary can include many variations of a given word, which means there will be a unique set of phonemes for each possible variation.

For this research, a data set of over 20,000 recorded utterances of Air Traffic Control commands was used to construct our model. Air Traffic Control phraseology has unique concatenation of words and therefore, unique effects of coarticulation. Our HMM is composed of 10,000 senomes, and 75,000 triphones.

HOLISTIC SYSTEM

Theoretically, the combination of an information threshold on the input signal and a speech recognizer that is modeled on this data will produce a system that inherently rejects uncorrelated information (noise).

Tests at NAWCTSD have confirmed this hypothesis on the Entropic (Cambridge University) system, as reported in our 1997 I/ITSEC paper titled “Developing Speech Recognition Models for use in Training Devices” (4). The input speech signal was saturated with 12 dB of added noise, thus becoming unrecognizable (21% recognition accuracy) on the control system, but when the input data was threshold filtered and correspondingly modified models were incorporated onto the system, the accuracy improved to 74%.

As noted in this preliminary experiment, the models could not be constructed directly from the FFT output due to software licensing restrictions with the Entropic system. Therefore, the speech signal was pre-filtered on a separate computer in the frequency domain and then converted back to the time domain. This conversion is known as the Fourier synthesis transformation and should be avoided since it produces unwanted effects such as the Gibbs phenomena. The unnecessary conversion is avoided with the CMU Sphinx II system because the agreement with CMU allows NAWCTSD access to the source code. The results of the new testing on the modified Sphinx II speech recognizer will be presented with this paper at the 20th Interservice/Industry Training Systems and Education Conference.

FURTHER RESEARCH

Refinements to this approach are in process. One such refinement is to make the input filter adaptive. The first requirement for this refinement is the storage of 10,000 comb filters based on the frequency spectrum of each senome. Since memory size is no longer a restriction for modern computing systems, this requirement is feasible and indeed has been incorporated in our research speech recognizer.

These senome filters are called from memory and used in place of the threshold filter at the output of the FFT. This replacement filter is equivalent to the “primary_format_locator“ threshold routine since the former is built and stored in conjunction with the Hidden Markov Model. It is evident that this equivalence only applies if the input was recorded as previously described.

To apply these senome comb filters adaptively requires a prediction of the next senome, which is 10 milliseconds in the future! This prediction, however, is not as critical as it seems since it is not required in real time. The input signal can be temporarily stored so that if the first prediction is incorrect, another prediction can be tried. An indicator of correctness is inherent in the speech recognizer through the “scoring algorithm”. A prediction of the future senome is implicit in the HMM scheme in the form of a conditional probability.

Probability of State Transitions

Statistical modeling used in modern independent speech recognition represents the inherent variability in human speech. The CMU Sphinx II assumes a Normal

(Gausian) distribution or a mixture of Normal distributions for calculating the probability of the state transition. Figure 1 is a schematic representation of a 6 state HMM for a typical phoneme model. The pointed solid lines represent the possible transition paths. For example; state 2 may transition to state 3, or transition to state 4, or transition back to itself for the next 10 milliseconds. These transition probabilities are expressed as decimals and will always sum to 1 (i.e. one of these transitions must occur). The dotted lines indicate a relationship between a state and one or two Normal distribution curves which determine an observation probability $\mathbf{b}_j$ that a particular state will be observed.

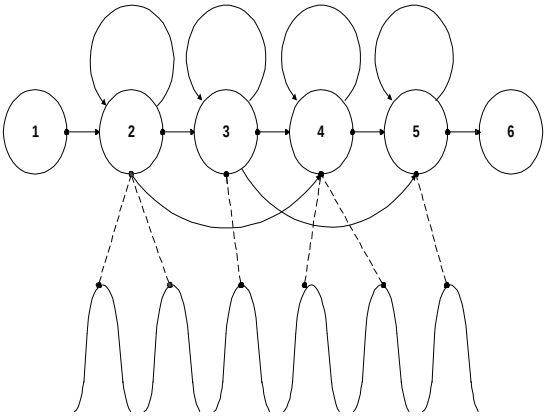

Figure 1. HMM

The input and output states are designated as null states so that the phoneme models can be connected in series to form word models. Word models may be connected in series to form sentence model and so on ad infinitum. In this way, HMM's can model any human language.

The probability $\mathbf{b}_j$ that a particular state j will be observed is:

$$\mathbf{b}_j = \prod_{s=1}^{S} \left[\sum_{m=1}^{M_s} \mathbf{c}_{jsm}\, \mathbf{N}(\mathbf{o}_{st}\,; \mu_{jsm}\,; \sigma_{jsm}) \right] \qquad (3)$$

The Normal distribution $\mathbf{N}$ is a function of :

1) the observation vector $\mathbf{o}_{st}$ where at time t there are four independent data streams s ;

2) the mean vector μ_{jsm} where integer m ranges to M = 256 for each s ;

3) the covariance matrix, which is assumed diagonal (i.e. the variance σ_{jsm}).

There are 10,000 possible states (senomes) such that j = 0,1,…,9999 and a unique comb filter is stored for each senome.

The coefficient $\mathbf{c}_{jsm}$ is calculated by applying a mel spectrum window to the frequency spectrum and then performing a cosine transform (see reference 3) to create unique Mel Frequency Cosine Coefficient. Said another way: these coefficients define the characteristic features of a senome. The MFCC can be thought of as a vector in a multi-dimensional vector space. The order of this multi-dimensional space is the product of j times s times m (10,000 X 4 X 256 = 10,240,000 dimensions). By thinking in this way, the uniqueness of the coefficient can be appreciated and the probability of observing a particular vector understood to be a very small number.

The joint probability that observation O is generated by the model M moving through the state sequence X is calculated as the product of the transition probabilities and the observation probabilities:

$$\mathbf{P(O, X|M) = a_{ij}\, b_j}$$

Where $\mathbf{a}_{ij}$ is the transition probability (i.e. $\mathbf{a}_{23}$ = transition from state 2 to state 3)

To implement this recognition in the speech recognition source code, an iterative routine referred to as the Viterbi Algorithm is employed. The most probable path (maximum log likelihood) is stored as score, and the highest score determines which model is recognized as the best match to the observations.

It is clear that the conditional probability of the next senome, given the present observation, is implicit in the mathematical theory. What is implied is that it is very likely that one can predict future utterances by what is being uttered in the present. This seems to be obvious, especially if the future is only one one-hundredth of a second away.
Therefore, we will use the Viterbi maximum log likelihood to predict the future senome.

Adaptive Filtering

We now have in place the necessary theoretical components for adapting a set of quantized frequency domain filters to a continually changing speech signal. This will improve the noise rejection of speech recognition systems. These components are implemented in the CMU Sphinx II source code as a research tool and test bed.

The filters are in the form of 10,000 files consisting of floating point ones and zeros. Each filter was created from a senome threshold frequency spectrum by the primary_format_locator routine. These filters are created at the same time that the HMMs are built and the filters are then mapped one-to-one with each senome model. Each senome model is given a name during the process of training the HMMs. This training associates the 20,000 Air traffic control spoken utterances with the corresponding written text and phoneme. The senome is stored as a 39 dimensional model vector in the form of a file consisting of 39 ordered floating point numbers.

The research speech recognizer is then designed to run on a high-speed computer in the following order. The unknown analog input speech signal is converted to a digital time domain signal (A/D) and the first 10 millisecond senome is then transformed to a digital frequency domain spectrum (FFT). This spectrum is multiplied by a filter that is selected by the Viterbi prediction. The result is transformed to an observation feature vector (mel / cosine transform) and temporarily stored in case the wrong filter was selected. This 39 dimensional observation vector is then compared to each senome model vector until the closest match is found.

Finding the closest match of a series of senomes is equivalent to recognizing a phoneme. Two vectors are said to be close when the distance between their end points is small. In the Entropic system software, mean squared distance is calculated using a Euclidean distance routine (3) and the minimum distance determines the match. Our research software uses a similar routine, but sets a maximum limit on the nearest distance as the meaning of “close”. If the distance is greater than the limit, the filter is rejected and another prediction is made. This procedure will continue until the observation matches the model according to our definition of close. The limit has been established such that the match is assured to be 95% correct.

This matching technique may require (worst case) that all of the filters are tried on a single observation and then compared with all of the senome models. This means that 100 million (10^4 X 10^4 = 10^8) operations would need to be performed per senome. Since research systems don’t operate in real time, this worst case scenario is reserved for further study, however, for faster computers using good predictions, real time systems can be envisioned.

We are pursuing research on predictions in the area of post processing. Briefly: natural language is constrained by the context in which it is used, so if one “knows” this context one will expect certain words and phrases to be used. This parallel research project employs pscholinguistics to process the data after the initial recognition in order to extract meaning from the text. This post process will be incorporated into the system as a whole in order to adapt the filter to the speech context and improve the overall predictions in a noisy environment.

Human Cognition

As has been said with respect to understanding how humans recognize language; ”Human utterance is an essence”. This implies that speech should be considered as more than the signal received

by the microphone. It contains information that is not measurable but has great influence on humans. How does one tap into this information in a systematic way such that it may be implemented into source code for predictive applications? We must be able to comprehend a meaning that transcends the mechanical vibrations of the signal. We need to expand research into the area of human cognition and understand the spirit of comprehension. This will require a theory that approaches human utterance as an essence and not as a result. This would incorporate speech as being one with human experience, where the former influences the latter as the latter creates the former in a kind of reciprocity. This type of artificial intelligence will be adapted to the filter thereby improving the ability of the speech recognizer to predict the meaning of an utterance which was obscured by noise.

SUMMARY

We are applying for a patent on this filter in which a more detailed description can be found. The results of this research should produce a speech recognizer that can be used effectively even if there is a lot of noise in the background. Our efforts so far have been directed towards Navy air traffic control trainers where aircraft noise and other conversations interfere with recognizing what the ATC trainee said. The applications for good speech recognition are unlimited if the state–of–the–art can advance to the stage where a computer can understand human language as well as humans do; even when there's a lot of noise around.

REFERENCES.

(1) Oppenheim, Alan V. and Schafer, Ronald Discrete-Time Signal Processing, Prentice Hall, Englewood Cliffs, New Jersey 07632 pg. 47

(2) Lin, Kun-Shan Digital Signal Processing Applications, (Implementation of FFT) Prentice Hall, Englewood Cliffs, New Jersey 07632 pg. 71

(3) Young, Steve, The HTK Book, (HMM Tool Kit version 2.1 March 1997) Entropic Research Laboratory, Cambridge University Technical Services Ltd.

(4) Cope,R Bradley and Boemler, Stephen G Developing Speech Recognition Models for Use in Training Devices, D. Kotick NAWCTSD, 19th Interservice/Industry Training Systems and Education Conference, 1997.