IMPROVING THE AUTOMATIC RECOGNITION OF DISTORTED SPEECH

by

Jayne Angela Beauford

BS in Mathematics, Spelman College, 1992

MA in Mathematics, University of Pittsburgh, 1995

MS in Electrical Engineering, University of Pittsburgh, 1999

Submitted to the Graduate Faculty of

Swanson School of Engineering in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy in Electrical Engineering

University of Pittsburgh

2009

UNIVERSITY OF PITTSBURGH

SWANSON SCHOOL OF ENGINEERING

This dissertation was presented

by

Jayne Angela Beauford

It was defended on

September 11, 2009

and approved by

Robert Boston, PhD, Professor, Electrical Engineering

Luis Chaparro, PhD, Associate Professor, Electrical Engineering

Zhi-Hong Mao, PhD, Assistant Professor, Electrical Engineering

Patrick Loughlin, PhD, Professor, Bioengineering

Chris Lennard, PhD, Associate Professor, Mathematics

Dissertation Director: Amro El-Jaroudi, PhD, Associate Professor, Electrical Engineering

IMPROVING THE AUTOMATIC RECOGNITION OF DISTORTED SPEECH

Jayne Angela Beauford, PhD

University of Pittsburgh, 2009

Automatic speech recognition has a wide variety of uses in this technological age, yet speech distortions present many difficulties for accurate recognition. The research presented provides solutions that counter the detrimental effects that some distortions have on the accuracy of automatic speech recognition. Two types of speech distortions are focused on independently. They are distortions due to speech coding and distortions due to additive noise. Compensations for both types of distortion resulted in decreased recognition error.

Distortions due to the speech coding process are countered through recognition of the speech directly from the bitstream, thus eliminating the need for reconstruction of the speech signal and eliminating the distortion caused by it. There is a relative difference of 6.7% between the recognition error rate of uncoded speech and that of speech reconstructed from MELP encoded parameters. The relative difference between the recognition error rate for uncoded speech and that of encoded speech recognized directly from the MELP bitstream is 3.5%. This 3.2 percentage point difference is equivalent to the accurate recognition of an additional 334 words from the 12,863 words spoken.

Distortions due to noise are offset through appropriate modification of an existing noise reduction technique called minimum mean-square error log spectral amplitude enhancement. A relative difference of 28% exists between the recognition error rate of clean speech and that of speech with additive noise. Applying a speech enhancement front-end reduced this difference to 22.2%. This 5.8 percentage point difference is equivalent to the accurate recognition of an additional 540 words from the 12,863 words spoken.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1.0 INTRODUCTION

The objectives of the research presented in this document are to determine methods for reducing the independent effects of coding and additive noise on speech recognition accuracy. Experiments are performed using the BBN Byblos Speech Recognition System. Training and testing are performed on a large-vocabulary conversational speech corpus. Training is performed on unprocessed conversational speech for all experiments. Synopses of the recognition system and the corpora used for the research are provided in Section 2.4.

1.1 RESEARCH OVERVIEW

To accomplish the main objective of this research, recognition is performed on speech that has been coded, decoded, and reconstructed. Examination of the test results reveal a relative difference of 6.7% between the recognition error rate of uncoded speech and that of speech synthesized from MELP encoded parameters. We develop a front-end that permits recognition of encoded speech directly from the MELP bitstream. The relative difference between the recognition error rate for uncoded speech and that of encoded speech recognized directly from the MELP bitstream is 3.5%. These results are illustrated in Table 1. This 3.2 percentage point improvement from the synthesized input to the bitstream input is equivalent to the accurate recognition of an additional 334 words from the 12,863 words spoken. The implementation of the bitstream recognition process and the results are detailed in Chapter 4.

Table 1. Recognition of MELP Coded Speech

Input Data Format	Error
Original Waveform	73.3
MELP Synthesized Speech	78.2
MELP Bitstream	75.9

A second research objective concerns the robust recognition of noisy speech. Not surprisingly, the recognition error rate of speech with additive noise is greater than that of the clean speech. The relative difference is 28%. Applying a speech enhancement front-end designed specifically for the improvement of the recognition of noisy speech results in a relative difference in error rate of 22.2% from that of the clean speech. These results are illustrated in Table 2. This 5.8 percentage point improvement is equivalent to the accurate recognition of an additional 540 words from the 12,863 words spoken. The average signal-to-noise ratio for the noisy speech is -5.6 dB. The implementation of the speech enhancement process and the recognition results are detailed in Chapter 6.

Table 2. Recognition of Noisy Speech

Input Data Format	Error
Original Waveform	72.1
Noisy Speech Waveform	92.3
Enhanced Speech Waveform	88.1

1.2 DISSERTATION ORGANIZATION

The remainder of this document is organized as follows. Chapter 2 highlights the applications and challenges of automatic speech recognition. The general speech recognition process, with emphasis on the features of the recognition system employed for this research, is also presented. Brief descriptions of the recognition system and the corpora used for this research are provided in Section 2.4.

As background for the recognition of coded speech, Chapter 3 explains common linear predictive coding processes. Section 3.1.4 explains mixed-excitation linear prediction (MELP), which is the coding process examined in this research. The applications and problems associated with the automatic recognition of coded speech are presented as well. Chapter 4 details the current research of recognizing coded speech directly from the bitstream.

Chapter 5 explains the process of speech enhancement as well as the applications and problems associated with the automatic recognition of noisy speech. The minimum mean-square error log spectral amplitude (MMSE-LSA) enhancement algorithm used in the research presented in this document is detailed in Section 5.1.4. Chapter 6 details the current research for the improved recognition of noisy speech.

The research is summarized in Chapter 7 and possible future research is considered in Chapter 8.

2.0 AUTOMATIC SPEECH RECOGNITION

Automatic speech recognition (ASR) is defined generally as the mechanized conversion from a speech waveform to a written equivalent of the message information [Rab1978]. The goal of ASR research is to design algorithms which take a speech waveform as input to a computer and develop techniques for generating from this input an accurate text transcription of the speech [Red1976]. Advances in ASR research have led to the development and advancement of many modern electronic devices, which continue to become increasingly multi-functional and commonplace. Speech is viewed as a more user-friendly interface than keyboards and push-buttons for many communications devices, thus, accurate, easy-to-use automatic speech recognition is in high demand. Though viable technology currently exists to accomplish reasonable accuracy for such tasks, the need for further research is understandable with the use of such devices in increasingly more complex environments [Dav2002].

Some common uses of speech recognition are discussed in Section 2.1. Looking at some of the challenges to accurate recognition of speech, Section 2.2 presents the motivation for our research. Section 2.3 provides an overview of the basic components of a contemporary ASR system. Synopses of the ASR system and corpora used in this research are provided in Section 2.4.

2.1 APPLICATIONS OF SPEECH RECOGNITION

Many electronic devices employ speech recognition technology to provide a simple and natural interface for the user. The voice activated options may free users from the constraints of small keyboards, offer convenience, and increase productivity [Dav2002]. For example, voice-activated dialing for cell phone users obviates the need for the small keyboard and allows the user to conveniently operate the device while performing other tasks such as driving. Virtual assistants offer a range of information from sports scores to the weather report to driving directions through voice-activated commands [Rog2001]. Devices in the automobile which utilize speech recognition include navigation systems which offer voice destination entry and entertainment systems which permit hands-free operation. In medical care automatic dictation may be employed for efficient creation and distribution of medical reports, leading to convenience and increased productivity.

No longer limited to government, military, or even corporate uses, speech recognition has become almost commonplace as its performance has improved. As the convenience of speech-enabled devices is realized, there has been an onset and rapid emergence of hands-free devices, and verbal commands are quickly replacing push-button activation. Speech-enabled devices have become ubiquitous. Therefore, the quest for more accurate and faster recognition systems is relentlessly underway.

2.2 MOTIVATIONS FOR THE PRESENT RESEARCH

Many obstacles may lead to inaccurate recognition of speech. Among them are the difficulties posed by the speech itself, such as varying rates of speech from person to person, lack of distinct spaces between words, pronunciation differences, incomplete words and sentences in conversational speech, and unknown and similar sounding words in large vocabulary sets [Gol1999]. This variability poses problems for the effective development of word models and grammar rules. Decreased accuracy in speech recognition can also result from variations in room acoustics and microphone characteristics. These variations result in distortions in the speech signal, causing the acoustic parameters used by the recognition system to deviate significantly from those of the undistorted speech.

Distortions of the speech waveform may also arise if the waveform is processed prior to recognition, as in the case when the speech is coded for transmission or storage purposes. To recognize coded speech, the speech signal is commonly reconstructed prior to performing the recognition task. That is, recognition is performed on the reconstructed speech rather than on the original acoustic waveform. In such cases, the preservation of the exact waveform characteristics is unrealistic [Wan1991], as demonstrated in Figure 1. The figure compares the time-domain graph of an uncoded speech waveform with that of a reconstructed speech waveform. The top graphs show a waveform corresponding to a three second speech utterance. From the comparison we can see that the reconstructed speech differs from the original speech, a difference which is even more evident in the bottom graphs which focus on a 30 millisecond frame of voiced speech. This difference, or distortion, may result in decreased recognition accuracy. This particular challenge is one aspect of the research presented in this thesis, and is covered in greater detail in

Chapter 3. The approach, implementation, and results of our research on the recognition of coded speech are detailed in Chapter 4.

A second aspect of our research centers on the recognition of noisy speech. The presence of noise is another challenge to the accuracy of recognition. Many devices which use speech recognition technology are deployed in environments in which the signal is subject to auditory disturbances such as background noise and echo [Dav2002]. These disturbances result in distortion of the speech signal, as shown in Figure 2. The top graphs show a waveform corresponding to a three second speech utterance, while the bottom graphs focus on a 30 millisecond frame of voiced speech. In these graphs the noisy speech waveform takes on the temporal characteristics of the noise, resulting in a distorted signal very different from the clean signal. The distortion consequently results in decreased recognition accuracy. To combat the effects of noise on speech recognition accuracy, the speech waveform is enhanced prior to recognition. The process of speech enhancement is covered in Chapter 5. The approach, implementation, and results of our research on the recognition of noisy speech are detailed in Chapter 6.

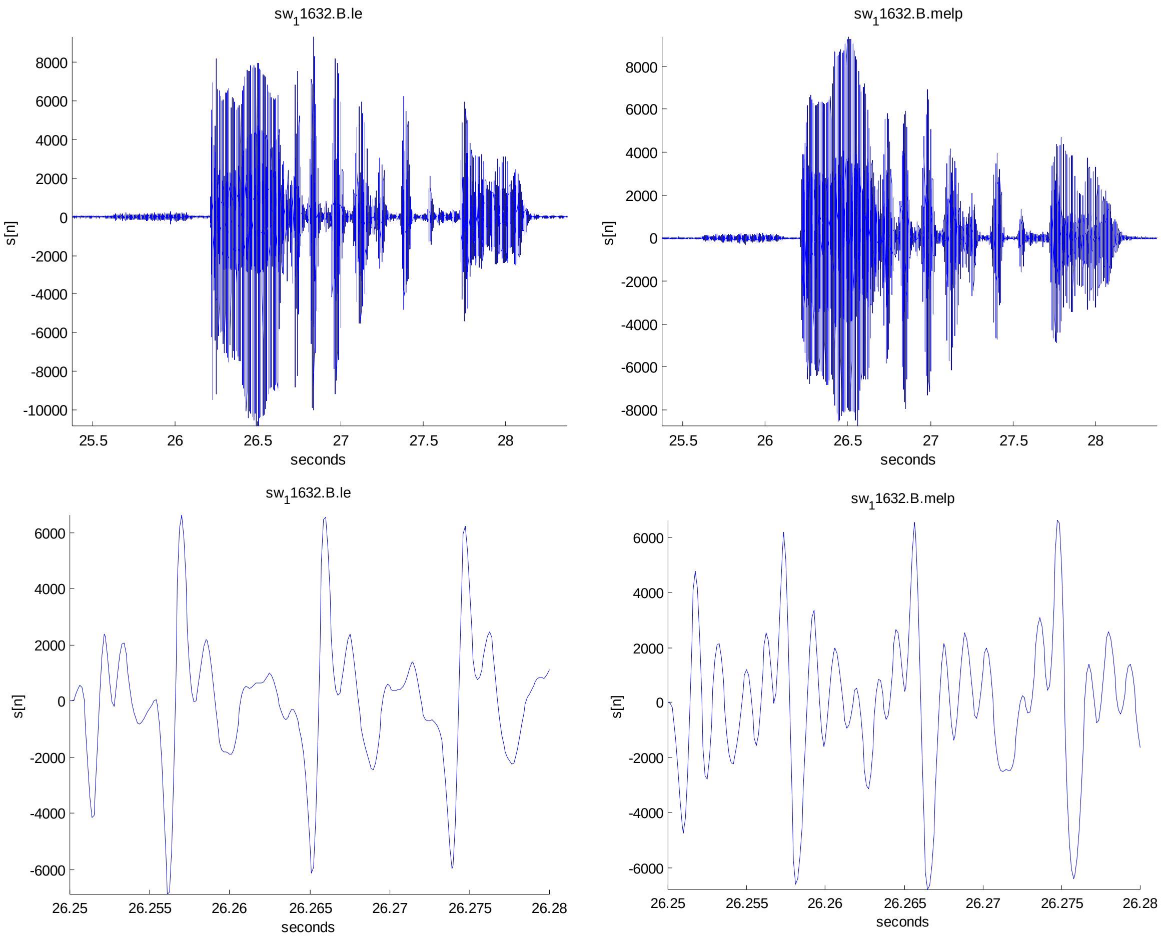


Figure 1. Effects of Coding on Time-Domain Signal
Left: Original (uncoded) speech waveform. Right: Reconstructed speech waveform
Top: 3 second speech segment. Bottom: 30ms frame of voiced speech

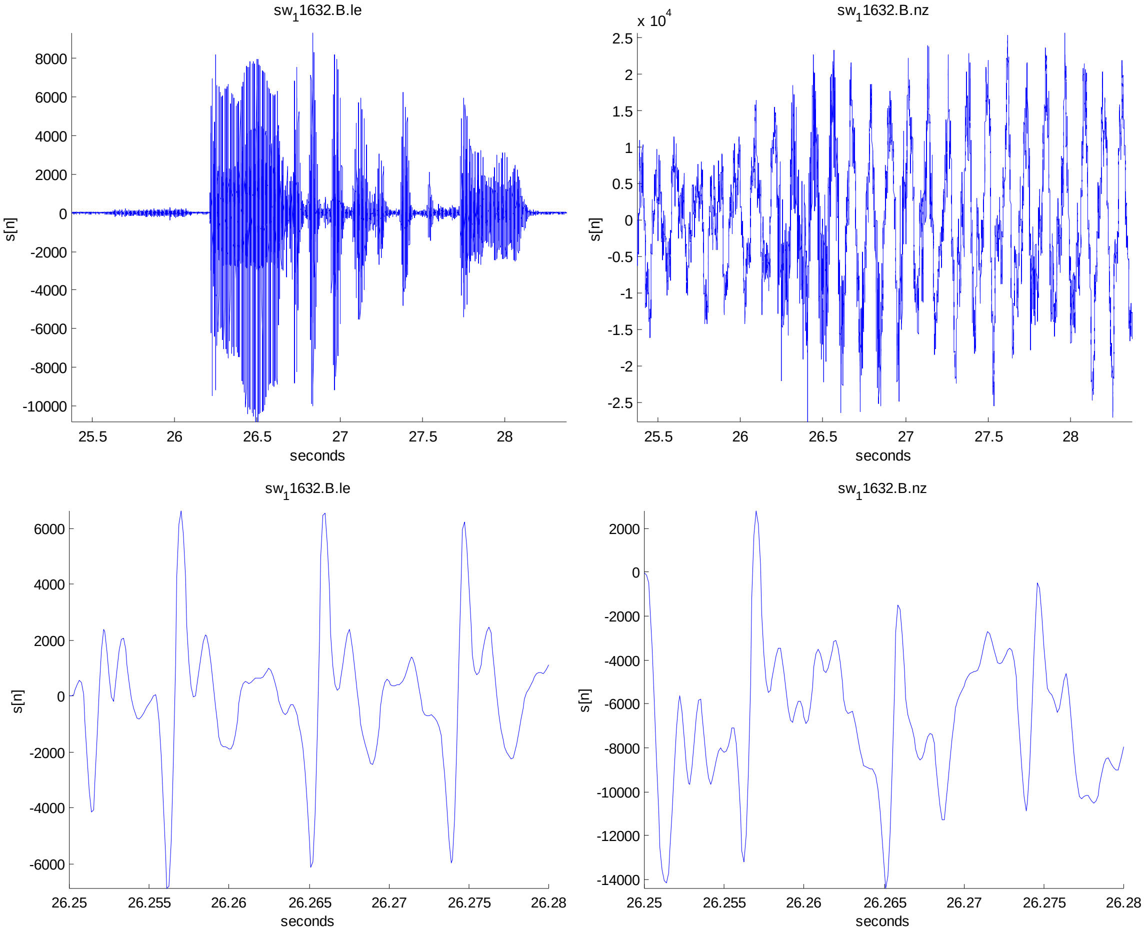


Figure 2. Effects of Noise on Time-Domain Signal
Left: Original (clean) speech waveform. Right: Noisy speech waveform
Top: 3 second speech segment. Bottom: 30ms frame of voiced speech

2.3 COMPONENTS OF A SPEECH RECOGNITION SYSTEM

Automatic speech recognition (ASR) is performed by a computer algorithm designed to take a speech waveform as input and produce as output a useful transcription of that speech. Before recognition, or decoding, of the speech can be performed, the ASR system must be trained. All current speech recognition systems perform the same fundamental operations. These steps are depicted in Figure 3 for the training and the decoding phases of recognition.

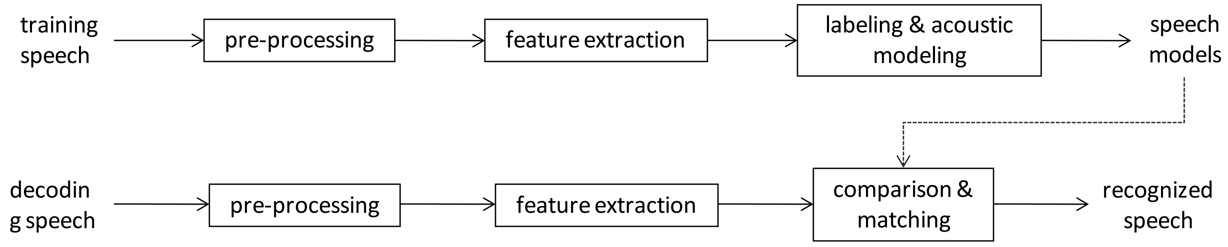


Figure 3. Speech Recognition Components
Top: Training phase. Bottom: Decoding phase.

The first step in both the training and the decoding phases of recognition is to pre-process the speech, converting it to a form that can be utilized by the recognizer. Next the desired features, those deemed essential to the recognition of speech, are extracted.

During the training phase of recognition, these features are labeled to associate each region of the speech with one or more phonetic labels. At this time an acoustic model is used to set up an equivalence relation between similar parameters to reduce the number of independent parameters.

In the decoding phase, the speech to be recognized is also pre-processed and the features are extracted. A comparison and matching technique then associates each region of the decoding speech with the training region deemed most similar.

The basic techniques used in contemporary recognition systems are explained briefly in the following sections, with emphasis on those techniques used in the Byblos ASR system. Section 2.3.1 details the pre-processing methods and Section 2.3.2 discusses feature extraction. Both of these steps are necessary for both training and decoding. Section 2.3.3 briefly describes acoustic modeling and labeling, which is needed in the training phase of recognition. Finally, Section 2.3.4 illustrates comparison and matching techniques used in the decoding phase.

2.3.1 Pre-processing

Prior to recognition the analog speech signal must be processed in order to prepare the signal for recognition. First, bandwidth limitation is necessary to avoid aliasing during sampling. Sampling digitizes the signal for easier storage and analysis. Typically the signal is sampled at 8,000 samples per second or above for the purpose of recognition. Because the statistics of a speech signal vary rapidly with time, the signal is segmented to create frames small enough to be stationary yet large enough to contain enough of the information necessary for recognition. A speech segment can be anywhere from ten to thirty milliseconds in duration, though twenty to twenty-five milliseconds is common [Lee1996]. The segments typically overlap for a frame rate of 100 frames per second, or equivalently one frame every ten milliseconds. However, the frame rate may vary from eight to twelve milliseconds. These ranges are based upon the evidence that an analysis window that exceeds twenty-five to thirty milliseconds begins to take in nonstationary segments of speech, and the human vocal tract will typically transition from one sound to another every eight to twelve milliseconds.

2.3.2 Feature extraction

Once the speech signal has been pre-processed, it is ready for the actual training or recognition (decoding) process. To perform either of these tasks, the features that are deemed necessary for the recognition of speech must be extracted from the signal. The characteristics of the desired features should not vary greatly from one utterance to the next nor from one speaker to the next for a particular phonetic unit, yet should change significantly from one phonetic unit to the next. These features routinely include the spectral characteristics of speech along with information about the energy in the signal [Gol1999]. Energy is used to classify a segment of speech as voiced, unvoiced, or silence. Spectral characteristics are used to determine pitch, formant locations, vocal tract parameters, and other parameters that may indicate the phonetic identity of the signal segment. These spectral characteristics are commonly obtained through analysis of the linear prediction coefficients $\{a_j : j = 1, \ldots, p\}$ of the speech signal. These are the coefficients of $A(z) = 1 + \sum_{j=1}^{p} a_j z^{-k}$, the denominator of an all-pole model of the speech spectrum. The computation of linear prediction coefficients is explained in detail in Section 3.1.1.

Linear prediction coefficients, the principal parameters of automatic speech recognition, are often used to convert a speech signal to a form more suitable for processing and evaluation. From the linear prediction coefficients, cepstral coefficients and line spectral frequency pairs may be computed. Both are frequently used in speech recognition systems [Liu1990][Kel1994].

Line spectral frequency (LSF) pairs are quantities which may be computed from linear prediction coefficients for the purpose of encoding the spectral information and for feature representation in speech recognition [Liu1990]. Once the linear prediction coefficients $A(z)$ have been computed, we may compute the line spectral frequency pairs as the roots of the polynomials

$$P(z) = A(z) - z^{-(p+1)} A(z^{-1})$$

$$Q(z) = A(z) + z^{-(p+1)} A(z^{-1})$$

so that $A(z) = \frac{1}{2}\left[P(z) + Q(z)\right]$. $P(z)$ and $Q(z)$ may also be written as

$$P(z) = A(z)\left[1 - R(z)\right]$$

$$Q(z) = A(z)\left[1 + R(z)\right]$$

where $R(z) = z^{-(p+1)} \frac{A(z^{-1})}{A(z)}$ is called the ratio filter.

The human vocal tract is fundamentally a tube of non-uniform cross-sectional area, and so may be modeled as the concatenation of non-uniform lossless tubes. The polynomial $P(z)$ gives a transfer function equivalent of augmenting this series of tubes with an additional section having zero cross-sectional area, i.e., that is completely closed. Conversely, $Q(z)$ gives a transfer function equivalent of augmenting the tubes with a section that has infinite area, i.e., is completely open [Rab1993].

Cepstral coefficients, which are very prominent in speech recognition, very effectively extract individual speech characteristics, such as pitch and formant location from the signal [Kel1994]. The cepstrum of a signal $x(n)$ is given as

$$c_m = \frac{1}{2\pi}\int_{-\pi}^{\pi} \log\left|X(e^{j\omega})\right| e^{j\omega m} d\omega$$

where $X(e^{j\omega})$ denotes the discrete-time Fourier transform of $x(n)$. The cepstral coefficients c_m may be computed directly from the p^{th} order linear prediction coefficients as follows [Rab1993]. Given the linear prediction coefficients $\{a_1, a_2, ..., a_p\}$ and the gain σ^2,

$$c_0 = \ln(\sigma^2)$$

$$c_1 = a_1$$

$$c_m = a_m + \sum_{k=1}^{m-1}\left(\frac{k}{m}\right) c_k a_{m-k}\ , \qquad 2 \le m \le p$$

$$c_m = \sum_{k=1}^{m-1}\left(\frac{k}{m}\right) c_k a_{m-k}\ , \qquad m > p$$

These cepstral coefficients are generally stored in vector form to help create a feature vector. Often the cepstral frequency axis is warped according to the mel scale or the bark scale. These scales convert the usual linear frequency scale to a non-uniform frequency scale in attempts to imitate the auditory system [Lee1996].

In addition to the spectral characteristics of the speech signal, the energy coefficient of each segment of speech is also computed and appended to the feature vector. The signal energy aids in the distinction between classes of speech, such as voiced, unvoiced, and silence. Voiced speech will generally have the highest signal energy while silence will have little or no energy. It is understood also that the slope and concavity of the cepstral and energy coefficients for sequential frames are useful in distinguishing between phonetic units. For this reason, the first and second derivatives of these quantities are also appended to the feature vector.

2.3.3 Labeling and acoustic modeling

In the research presented in this document the phonetic representation of speech is the phoneme modeled in a triphone context. That is, each phoneme will have multiple models, one for each possible pair of neighboring left and right phonemes. For recognition purposes speech is often modeled as the output of a hidden Markov process in which the feature vectors of the decoding speech serve as observations emitted from a collection of states. This acoustic model, called a hidden Markov model (HMM), consists of the probabilities of transitioning from one state to another as well as the probability density functions of the observations of each state [Rab1989]. Frequently these density functions are represented as mixtures of diagonal Gaussians. Each phoneme of a training utterance initially will have its own set of states and transition probabilities, creating a reference model for that utterance. Once features have been extracted from each segment of the input training data, phonetic labels must be assigned. This is done through the implementation of the Viterbi algorithm, which determines the HMM state sequence most likely to have produced a given observation.

Often the case arises when training data is insufficient so that reliable and robust determination of HMM parameters cannot be accomplished. This may arise when one model is context independent, and thus more robust, and the other context dependent, and therefore more precise. For such cases, tied mixtures may be used to set up an equivalence relation between HMM parameters in different states in order to simplify parameter estimation. There exist two types of tied mixtures. These are state clustered tied mixture (SCTM) and phonetically tied mixture (PTM). PTM is a coarser model that insists that all states of all triphones descending from the same center phoneme share a single codebook, resulting in a single set of Gaussians for each context-independent phoneme. SCTM is a more detailed model which allows context-independent phonemes to share either a single codebook or entire distributions [Siu1999]. Once a training model for each phonetic unit has been established, the models may be used for the recognition of speech.

2.3.4 Comparison and matching

Once an ASR system has been trained, a speech signal may be input for recognition. As in the training phase of recognition, features are extracted from a signal prior to decoding. These features must in some way be compared to those of the training speech to determine which phoneme most likely produced this feature vector. Many techniques exist to determine the similarity of speech patterns for recognition. One comparison technique utilizes vector quantization in which the decoding vector is compared to the centroids of vector codebooks. The decoding vector is then represented by the codebook vector closest to it [Kel1994].

Another common comparison method is template matching with dynamic time warping (DTW). With this technique acoustic feature vectors of the speech to be recognized are nonlinearly warped in time to create a best fit to each of the feature vectors from the training data. A score corresponding to the exactness of the match is assigned for each training vector. The utterance corresponding to the feature vector with the best score is considered to be the utterance actually spoken, provided that score exceeds some predetermined threshold [Osh1987].

The most common pattern matching technique in use today is hidden Markov modeling. Given an utterance to be recognized, each feature vector of the decoding speech is scored using the Viterbi algorithm to determine the model most likely to have produced it. The model with the highest probability is selected, provided the probability is above a given threshold.

2.4 SYSTEM AND CORPUS OF THE CURRENT RESEARCH

The speech recognition for this research is executed using the BBN Byblos 2001 large vocabulary automatic speech recognition system. Byblos is a speaker-independent, continuous speech, triphone context based system. Its English system has a 22,000 to 26,000 word vocabulary and uses fifty-four phonemes including seven for non-speech sounds. The forty-five dimensional feature vectors consist of the Mel-warped cepstral coefficients derived from LP coefficients, normalized frame energy and the first and second derivatives of these. The features are modeled using left-to-right hidden Markov models. The recognition system utilizes state clustered and phonetically tied mixtures as described in Section2.3.3, and is implemented using a uniform frame rate of 100 frames per second.

The speech data are taken from the Switchboard Corpus and the Call Home English Corpus. The Switchboard Corpus consists of pairs of speakers unknown to each other conversing on a predetermined topic. The Call Home Corpus consists of pairs of speakers known to each other speaking on an unspecified topic. The training data consists of 750 speakers, over ninety-nine hours of speech, from the Switchboard Corpus. The test data consists of 80 speakers from the Switchboard Corpus and the Call Home Corpus, for a total of approximately three hours of data. The total number of words in the test files is 12,863.

3.0 SPEECH CODING

The objective of speech coding is to represent a speech signal with as few bits as possible while maintaining the ability to effectively reconstruct the signal from these bits at a later time. The purpose is to be able to efficiently store or quickly transmit the speech signal. Once the bits have been transmitted or are ready for use, they are then decoded and reconstructed to produce a synthetic signal that is acoustically similar to the original. The process of speech coding is widely used in areas such as cellular communications, voice over IP, and other digital communications.

Many of the coding techniques in use today are based on linear prediction (LP) analysis. Linear prediction and the common linear predictive coding techniques are described in the remainder of this chapter. The United States Federal Standard mixed-excitation linear prediction (MELP) coder, which is implemented in this research to examine the effects of coding on recognition performance, is described in detail in Section 3.1.4.

3.1 LINEAR PREDICTION BASED CODERS

Speech coding is the process of converting a speech signal to a format that requires fewer bits for storage and transmission, yet contains enough information to adequately reconstruct a signal comparable to the original signal. The more straightforward quantization methods represent the signal using a finite number of amplitude values, so that any value in a given range, say between x_i and x_j, is assigned the same discrete value. Vector quantization improves on this technique by associating each input value with a codebook value that is closest to it in some defined sense. With these methods, however, the redundancies of a signal, such as those arising from periodicity of voiced speech, are not taken into account. Therefore, information that can be deduced is also being coded unnecessarily. One technique that takes advantage of the repetitions in speech is linear predictive coding.

Linear prediction assumes that each speech sample is not entirely unrelated to the previous samples. In fact, it assumes that each sample may be estimated as the linear combination of the previous p samples

$$\tilde{s}(n) = -\sum_{j=1}^{p} a_j s(n-j) .$$

The determination of the prediction coefficients a_j is detailed in Section 3.1.1. The use of linear prediction for speech coding is common, and three federal coding standards, linear prediction coding (LPC), code-excited linear prediction (CELP), and mixed-excitation linear prediction (MELP), are described in Section 3.1.2, Section 3.1.3, and Section 3.1.4, respectively.

3.1.1 Linear prediction of speech

Linear prediction of speech is based on the premise that a single speech sample $s(n)$ may be estimated as a linear combination of previous samples, as given by the following equation

$$\tilde{s}(n) = -\sum_{j=1}^{p} a_j s(n-j) .$$

Here, $\tilde{s}(n)$ is the predicted speech, p is the prediction order, and $\{a_j : j = 1, \ldots, p\}$ are the linear prediction coefficients. Typically the prediction order p is fixed in the range between ten and twelve for speech sampled at 8 kHz [Cha1974]. The prediction error is given by

$$e = s(n) - \tilde{s}(n)$$

and the desire is to determine the coefficients a_j that minimize the mean square error

$$\begin{aligned} mse &= E\left\{ \left[s(n) - \tilde{s}(n) \right]^2 \right\} \\ mse &= E\left\{ \left[s(n) + \sum_{j=1}^{p} a_j s(n-j) \right]^2 \right\} \end{aligned}$$

This equation is a p-dimensional surface with a unique minimum. Thus the minimum can be found by setting the partial derivative of *mse* with respect to a_k equal to zero. We then obtain the following:

$$\begin{aligned} 0 &= 2E\left\{ \left[s(n) + \sum_{j=1}^{p} a_j s(n-j) \right] s(n-k) \right\} \\ &= E\{s(n)s(n-k)\} + \sum_{j=1}^{p} a_j E\{s(n-j)s(n-k)\} \qquad k = 1, \ldots, p \\ 0 &= r(k) + \sum_{j=1}^{p} a_j r(|j-k|) \end{aligned}$$

where $r(k)$ denotes the k^{th} lag of the autocorrelation function of the signal $s(n)$. From the derivation, we see that the prediction error is minimized when $-r(k) = \sum_{j=1}^{p} a_j r(|j-k|)$ for $k = 1, \ldots, p$. This relation can be written in matrix form as

$$-\vec{r}_s = R_s \vec{a}$$

where the autocorrelation vector $\vec{r}_s$, the autocorrelation matrix R_s, and the linear prediction coefficient vector $\vec{a}$ are given by

$$\vec{r}_s = \left[r(1) \quad r(2) \quad r(3) \quad \dots \quad r(p) \right]^{\mathrm{T}}$$

$$\vec{a} = \left[a_1 \quad a_2 \quad a_3 \quad \dots \quad a_p \right]^{\mathrm{T}}$$

$$R_s = \begin{bmatrix} r(0) & r(1) & \cdots & r(p-2) & r(p-1) \\ r(1) & r(0) & \ddots & & r(p-2) \\ r(2) & r(1) & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & & \\ r(p-1) & r(p-2) & \cdots & r(1) & r(0) \end{bmatrix}$$

where $[\cdot]^{\mathrm{T}}$ denotes the matrix transpose. Given the prediction order *p* and the autocorrelation values $\{r(0) \quad r(1) \quad \cdots \quad r(p)\}$, we can compute the optimal prediction coefficients $\{a_j : j = 1, \dots, p\}$ via the Durbin Recursion algorithm, which has been deemed the most efficient method of solving this problem [Rab1978]. This recursive algorithm, which utilizes the solution to the (*p*-1)$^{\text{st}}$ order prediction to calculate the solution for order *p*, is given as follows:

```
initialization
        0.    $E^{(0)} = r(0)$

for i=1,2,..,p
        1.    $k_i = -\dfrac{r(i) + \sum_{j=1}^{i-1} \alpha_j^{(i-1)} r(|i-j|)}{E^{(i-1)}}$
        2.    $\alpha_i^{(i)} = k_i$
        3.    $\alpha_j^{(i)} = \alpha_j^{(i-1)} + k_i \alpha_{i-j}^{(i-1)}, \quad j = 1,2,\dots,i-1$
        4.    $E^{(i)} = (1 - k_i^2) E^{(i-1)}$
end
```

The linear prediction coefficients are then estimated as $a_j = \alpha_j^{(p)}$. The parameters k_i are the reflection coefficients, also called partial correlation, or PARCOR, coefficients.

3.1.2 Linear prediction coding

During speech production, air is forced by the diaphragm from the lungs through the vocal tract. During voiced speech, the air causes the vibration of the vocal cords and produces quasi-periodic pulses of air which pass through the vocal tract. When speech is unvoiced, the vocal cords do not vibrate and the waves produced by the air resemble noise. We therefore may view this process of speech production as one in which an excitation, either noise or quasi-periodic impulses, is forced through a filter which models the vocal tract [Gol1999]. Linear prediction coding assumes that a speech signal $s(n)$ may be modeled as a filtered excitation $x(n)$. This model synthesizes speech via linear prediction when the vocal tract is modeled as an all-pole filter $H(z)$ as shown in Figure 4.

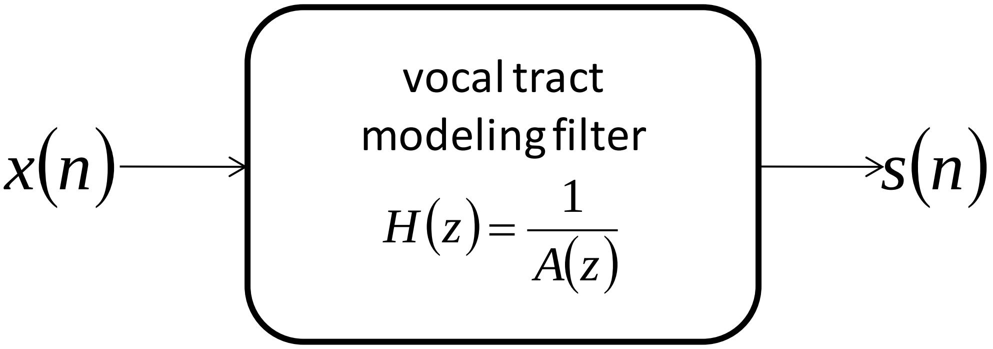


Figure 4. Simplified Speech Production Model

We may then view $s(n)$ as the output of the linear filter *H(z)* whose inverse system *A(z)* has the coefficients $\{a_j : j = 1, \ldots, p\}$. The optimal filter coefficients would be identical to the linear prediction coefficients derived in Section 3.1.1.

When the excitation $x(n)$ is incorporated into the linear prediction relation, we obtain the relationship

$$s(n) = -\sum_{j=1}^{p} a_j s(n-j) + x(n).$$

The excitation source $x(n)$ is a pulse train for voiced speech and random noise for unvoiced speech. The parameter *p* is the model or prediction order.

In order to accurately synthesize speech, other information such as gain and pitch period must be determined. The gain is the factor which multiplies the excitation to account for variations in signal energy. Thus the input to the vocal tract modeling filter is given by $x(n) = Gu(n)$, where $u(n)$ is either white noise or a unit energy impulse train and *G* is the system gain. So then the LP model is given by the equation

$$s(n) = -\sum_{j=1}^{p} a_j s(n-j) + Gu(n)$$

From the autocorrelation of the output signal we may obtain a calculation for the system gain:

$$\begin{aligned} R(k) &= E\{s(n)s(n-k)\} \\ &= E\left\{\left[-\sum_{j=1}^{p} a_j s(n-j) + Gu(n)\right] s(n-k)\right\} \\ &= -\sum_{j=1}^{p} a_j E\{s(n-j)s(n-k)\} + G \cdot E\{u(n)s(n-k)\} \end{aligned}$$

$$R(k) = \begin{cases} -\sum_{j=1}^{p} a_j R(|j-k|) & k = 1, \ldots, p \\ -\sum_{j=1}^{p} a_j R(|j-k|) + G^2 & k = 0 \end{cases}$$

Consequently, the system gain is given as the square root of

$$G^2 = R(0) + \sum_{j=1}^{p} a_j R(j).$$

Pitch period is used to determine the spacing of the impulses in the input signal for voiced speech. Because voiced speech is quasi-periodic, its autocorrelation ideally will attain a maximum value at a lag value equal to the pitch period. The human voice typically has a pitch between 50 and 500 Hertz. Restricting the maximum lag search to a range of 40 to 160 samples, at a 8 kHz sampling rate, helps to avoid unacceptable peaks that are due to vocal tract response, or formants [Chu2003].

With this gain and pitch information the speech production model is modified as shown in Figure 5. In the 1980’s linear prediction coding became the United States federal standard for speech coding. The fs-1015 linear prediction coder adopted in 1984 uses a prediction order of ten and encodes 8 kHz-sampled speech by quantizing the prediction coefficients, filter gain, voicing flag, and pitch period for each 22.5 millisecond frame of speech using 54 bits, for a total bit rate of 2400 kilobits per second. The linear prediction coefficients are quantized by line spectral frequency pairs which are explained in Section 2.3.2.

As explained above, linear prediction offers an accurate estimate of speech parameters at a very low bit rate with relative computational ease. This low bit rate is possible because linear prediction takes advantage of the redundancies and correlation in the speech signal. From the linear prediction coefficients, vocal tract parameters such as pitch, formant location, and vocal tract area may be extracted. These are the parameters which are essential to speaker identification and speech recognition. Therefore, linear prediction is among the most popular of speech coding tools. However, the LP coding scheme does have deficiencies. One such deficiency results from the strict classification of speech as either voiced or unvoiced. This classification may be inadequate for regions where the speech transitions from voiced to unvoiced. Even if a speech segment is voiced or unvoiced, the strict choice of random noise or impulses as the excitation source is not sufficient. These limitations are addressed in code-excited linear prediction (CELP) and mixed-excitation linear prediction (MELP) coding which are discussed in Section 3.1.3 and Section 3.1.4, respectively.

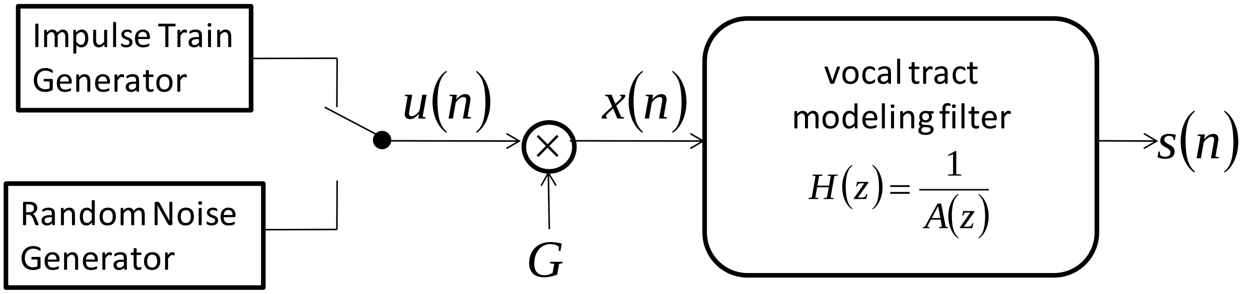


Figure 5. Speech Production Model

3.1.3 Code-excited linear prediction coding

With the standard linear predictive coder, called lpc-10, a speech signal sampled at eight kilohertz can be coded at a rate of 2.4 kilobits per second [Cia1982], a rate of less than one-half bit per sample. Although the bit rate is very satisfying, the resulting quality of speech is inadequate for many communications tasks. This assessment is not unusual for low bit rate coders. To improve the speech quality associated with linear prediction coding, a 4.8 kilobits per second code-excited linear prediction (CELP) coder was adopted as the new U.S. Federal Standard. The CELP coding offers improvements over the traditional linear prediction coder, including lower audible distortion and lower bandwidth requirement. The difference is that the excitation is represented by the succession of a fixed number of codebook sequences rather than by impulses [Sch1985]. For each segment of speech, the coder searches a preset codebook of vectors for the excitation source that best suits that input segment. The address of the code vector is then transmitted to the receiver, which contains a copy of the codebook [Gol1999]. As with lpc-10, the United States Federal Standard (fs-1016) CELP coder uses a 22.5 millisecond analysis frame of speech sampled at eight kilohertz. The linear prediction is identical to that of the old Federal Standard for lpc-10. In addition to the linear prediction coefficients, CELP also computes parameters which are obtained via analysis-by-synthesis. These adaptive codebook (ACB) and stochastic codebook (SCB) parameters are computed for each quarter frame of speech. The SCB parameters represent the residual excitation while the ACB parameters represent the periodicity of the residual excitation. Once the linear prediction coefficients have been computed, they along with possible ACB and SCB parameters are input to a synthesizer which produces a sub-frame of speech. The sub-frame is compared to the original speech via the Euclidean distance measure [Lan1991]. Prior to computing the distance, each speech segment is weighted to attenuate the error in the formant regions. The CELP coder provides adequate speech quality at a rate of 0.6 bits per sample.

3.1.4 Mixed-excitation linear prediction coding

The current U.S. Federal Standard is the 2.4 kilobits per second mixed-excitation linear prediction (MELP) coder. Like the CELP coder, MELP is based on a tenth order linear prediction model. However, in the case of MELP, as the name suggests, the excitation is computed as a mixture of random noise and quasi-periodic impulses [Sup1997]. The mixture weighting is determined by the voicing strengths calculated for each of five frequency bands. MELP also uses three classifications of speech rather than two. They are voiced, unvoiced, and jittery voiced. For frames classified as jittery voiced, a randomly generated period jitter is computed at ±25% of the pitch period. This jitter is used in voicing transition to reduce the presence of unwanted tones, which are present with the lpc-10 coder due to voicing misclassification. Other adjustments are made to produce more natural sounding synthetic speech. The MELP coder operates at a bit rate equal to that of the lpc-10 yet provides voice quality comparable to that of the Federal Standard CELP coder [Koh1997].

A block diagram of the MELP coder is shown in Figure 6. The parameters which are encoded to compose the bitstream are the linear prediction coefficients, voicing strengths, aperiodic flag, pitch period, gain, and Fourier magnitudes. The computation of the linear prediction coefficients is identical to that for the linear prediction coder as explained in Section 3.1.2. Before explaining the computation of each of the remaining parameters, we will explain the computation of the initial, or first-stage, pitch period estimate and the peakiness measure which are used in the computation of the voicing strengths and the aperiodic flag.

First-stage Pitch: The initial pitch period $T^{(1)}$ is estimated from input speech which is filtered by the first analysis filter with a passband of 0-500 Hz. A normalized autocorrelation

$r(l)$ is computed for each 180-sample frame for integer lag values of 40 to 160. The pitch period, *T*, corresponds to the lag value, *l*, for which $r(l)$ is maximized. The fractional pitch period, η, is then computed and the initial, or first-stage, pitch period is given by $T^{(1)} = T + \eta$. This initial pitch period, along with a peakiness measure, is used in the determination of the voicing strengths and the aperiodic flag.

Peakiness: Peakiness of the prediction error signal, which is defined by

$$p = \frac{\sqrt{\frac{1}{160}\sum_{n=-80}^{79} e^2(n)}}{\frac{1}{160}\sum_{n=-80}^{79} |e(n)|}$$

is designed to measure the outstanding peaks in the time-domain signal. The measure is calculated over a 160-sample frame centered at the last sample in the current frame. A frame of voiced speech is expected to have a high peakiness value due to the quasi-periodic impulse train structure. Where the pulses become sparse, as with transition frames, the peakiness measure is maximized. Noisy speech on the other hand, having no distinct peaks, will have a lower peakiness value. Peakiness is used in the determination of voicing strengths and the value of the aperiodic flag.

Voicing Strengths: Voicing strengths are computed for each of five frequency ranges with passbands defined by 0-500, 500-1000, 1000-2000, 2000-3000, and 3000-4000 Hz. From the output of the analysis filter corresponding to each frequency band, the normalized autocorrelation function r_1 is computed. A second normalized autocorrelation r_2 is computed, this time using the envelope of the bandpass signal, which may better reflect periodicity due to the fundamental pitch frequency. For each of the bands, the voicing strength is given as the

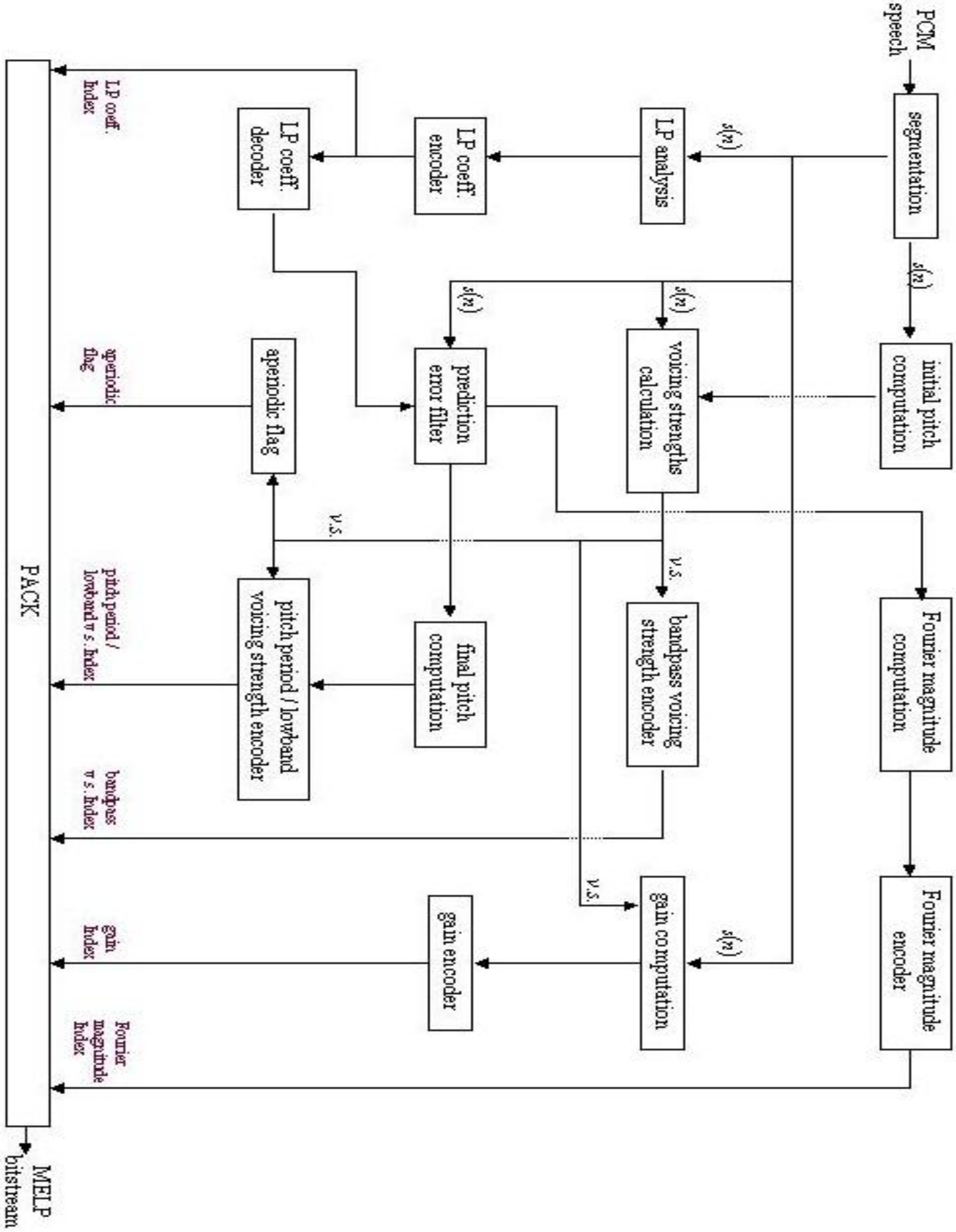


Figure 6. MELP Coder

maximum of the normalized autocorrelations r_1 and r_2 computed at a lag of $T^{(1)}$, which is the initial pitch period estimate. That is $vs_i = \max\{r_1(T^{(1)}), r_2(T^{(1)})\}$, for $i = 1,2,\ldots,5$.

The lower three voicing strengths are further modified according the peakiness measure *p*. If the peakiness is high, then the lower voicing strengths are set to unity to indicate high impulse train content in the excitation signal.

- If $p > 1.34$, then $vs_1 = 1$.
- If $p > 1.60$, then $vs_2 = 1$ and $vs_3 = 1$.

Finally, all voicing strengths are quantized to 0 or 1 for transmission. If the low-band voicing strength vs_1 is zero, then the frame is unvoiced, and all voicing strengths are set to zero. Otherwise, the voicing strengths are rounded to either 0 or 1. If $vs_i = 0$ for $i = 2,3,4$, then vs_5 is set to zero as well. The voicing strengths are used by the decoder to determine the combination of pulse excitation and noise excitation at the various frequency bands.

Aperiodic Flag: The aperiodic flag is computed from the low-band voicing strength vs_1 prior to quantization. If the value of vs_1 is low, the frame has weak periodicity and the aperiodic flag is set to one. Otherwise, it is set to zero. The aperiodic flag combined with voicing strengths are used by the decoder to determine the voicing state.

Pitch Period: The final pitch period estimation T_0 is computed from the lowpass filtered prediction error signal. The estimate is obtained in the same manner as the initial pitch period $T^{(1)}$, by searching in a neighborhood of $T^{(1)}$. The use of the prediction error results in a more accurate estimate since periodicity due to the formant structure is not present as it is in the original speech. The pitch period is used by the decoder to generate a quasi-periodic pulse excitation.

Gain: The signal gain is measured twice per frame using an identical pitch-adaptive window length for both measurements. Both gains, g_1 and g_2 are computed as

$$g = 10\log\left(0.01 + \frac{1}{L}\sum_{n} s^2(n)\right)$$

where L denotes the window length. For voiced frames, determined as those frames for which the lowband voicing strength vs_1 is greater than 0.6, the window length is given as the lowest integer multiple of the initial pitch estimate $T^{(1)}$ that is greater than 120. This determination allows for pitch-synchronization in the decoding phase and minimizes variation in the gain value with respect to window position. If the window length is greater than 320, then the value is divided by two. For unvoiced or jittery voiced frames, a default window length of 120 is used.

Fourier Magnitudes: Fourier magnitudes are the measurements of the peaks of the magnitudes of the Fourier transform. These peaks correspond to the harmonics associated with pitch frequency and are computed for voiced and jittery voiced frames from 200 samples of the prediction error signal. This 200 sample frame is zero-padded to 512 samples. The magnitude peaks are found by searching the magnitude of the Fourier transform of the error signal in the neighborhood surrounding ten first harmonics associated with pitch frequency:

$$\frac{512k}{T_0}, \quad k = 1, \ldots, 10$$

The ten peak magnitudes that are obtained are normalized to have unit root mean square value and weighted to emphasize the more important low-frequency regions. The Fourier magnitudes are used to capture the shape of the excitation pulse generated in the decoding phase, improving the auditory quality of the synthesized speech.

The above parameters are quantized and packed into a bitstream to be passed to the decoder. Most of the parameters are linearly interpolated during the decoding phase. A block diagram of the MELP decoder is shown in Figure 7. The main components of the decoder are the shaping filters, the spectral enhancement filter, the synthesis filter, and the pulse dispersion filter. Before these filters are explained, we explain parameter interpolation, the adjustment of the pitch period and the generation of the excitation pulse.

Parameter Interpolation: The set of parameters required for synthesis of a given 180-sample frame is determined from linear interpolation of the parameters from the current frame and the past frame. The interpolation factor is given by $\alpha = \frac{n_0}{180}$, where $n_0 \in [0,179]$ pertains to a given instant of time in the current frame. Each parameter P, with the exception of the gain, is interpolated as

$$\mathrm{P} = \alpha \mathrm{P}_{\mathrm{current}} + (1-\alpha)\mathrm{P}_{\mathrm{past}}.$$

Voicing strengths are not interpolated directly due to high computational cost. Rather the coefficients of the shaping filters, which are described later in this section, are interpolated. Recalling that two gain values, g_1 and g_2 are computed during the encoding phase, the gain used in the synthesis of the speech signal is interpolated according to the formula

$$g = \begin{cases} \alpha \mathrm{g}_{1,\mathrm{current}} + (1-\alpha)\mathrm{g}_{2,\mathrm{past}}, & n_0 < 90 \\ \alpha \mathrm{g}_{2,\mathrm{current}} + (1-\alpha)\mathrm{g}_{1,\mathrm{past}}, & 90 \le n_0 < 180 \end{cases}$$

where α and n_0 are as defined above.

Pitch Period Adjustment: The voicing classification for each frame of speech is determined by the decoded pitch period and low-band voicing strength values. For voiced frames, the adjusted pitch period is computed as

$$T = T_0 (1 + jitter \cdot x)$$

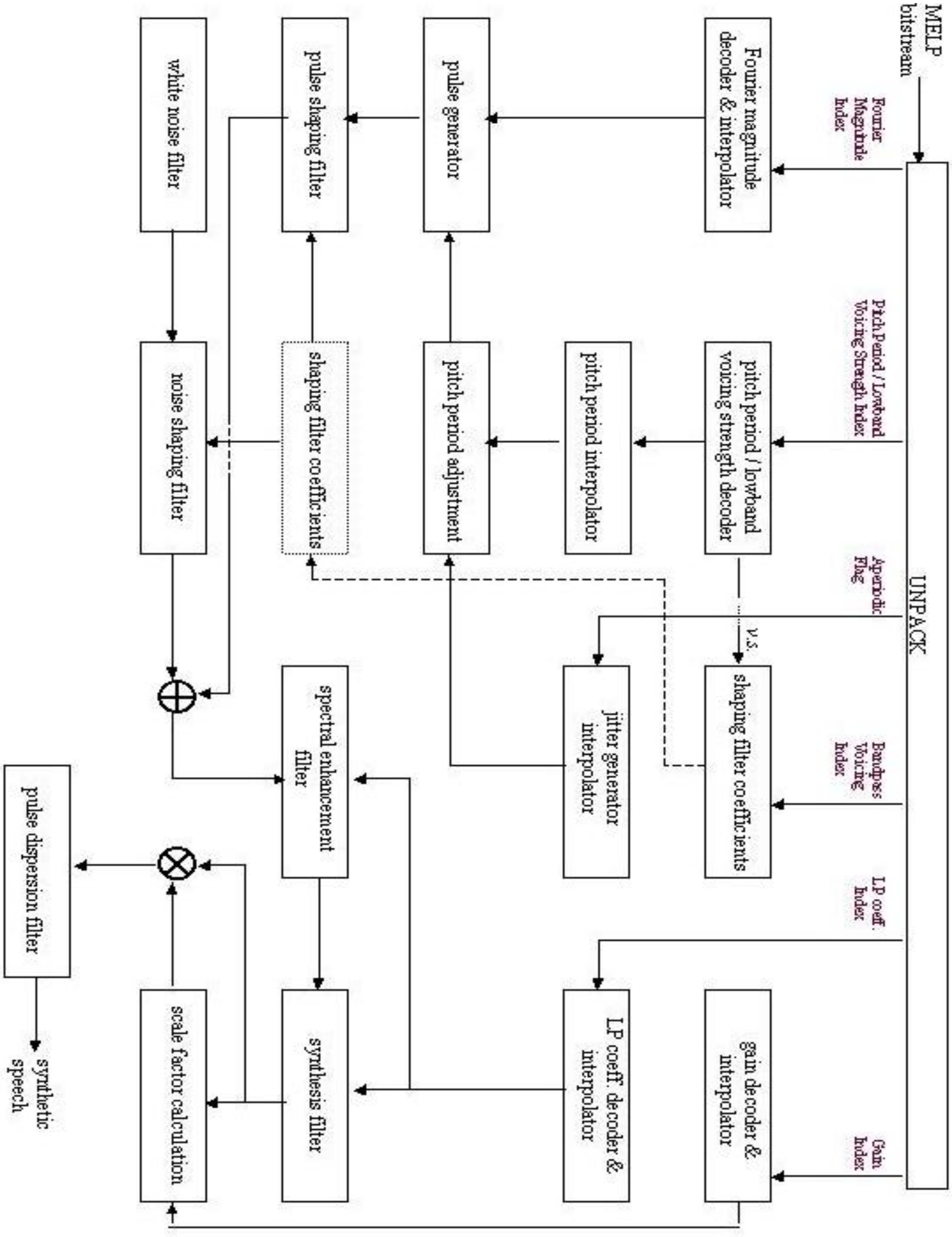

MELP bitstream
UNPACK
Fourier Magnitude Index
Pitch Period / Lowband Voicing Strength Index
Aperiodic Flag
Bandpass Voicing Index
LP coeff. Index
Gain Index
Fourier magnitude decoder & interpolator
pitch period / lowband voicing strength decoder
v.s.
shaping filter coefficients
gain decoder & interpolator
pitch period interpolator
jitter generator interpolator
LP coeff. decoder & interpolator
pulse generator
pitch period adjustment
pulse shaping filter
shaping filter coefficients
spectral enhancement filter
synthesis filter
white noise filter
noise shaping filter
scale factor calculation
pulse dispersion filter
synthetic speech

Figure 7. MELP Decoder

where T_0 is the decoded and interpolated pitch period. The jitter is set to zero if the aperiodic flag is zero, indicating a purely voiced frame of speech. Otherwise, the frame is jittery voiced , as in the case of transition frames, and the jitter is set to 0.25. The parameter *x* refers to a uniformly distributed random number between –1 and 1. Thus, the pitch period for transition frames varies at most 25% from the encoded pitch estimate. This jitter reduces the likelihood of tones common in LP coders at transition frames. For unvoiced frames, a default pitch period of 50 is used.

Excitation Generation: Once the pitch period has been determined for voiced frames, the excitation pulse is synthesized. The Fourier magnitudes F_k are symmetrically extended to T samples. That is, $Y_k = F_k$ and $Y_{T-k} = F_k$ for $k = 1,\ldots,10$. The dc-component Y_0 is set to zero. All remaining values, Y_{11} through Y_{T-11} are set to one. The excitation pulse is then given as the inverse Fourier transform of the symmetrical frequency signal. A ten sample circular shift prevents abrupt changes at the beginning of a period. The frequency distribution of this pulse train more closely matches the distribution of the original speech waveform than does the impulse train derived by the lpc-10 coder.

Shaping Filters: The MELP decoder uses two shaping filters, a pulse shaping filter for and a noise shaping filter, which control the amount of pulse and the amount of noise in the excitation signal. Each of these filters is a weighted sum of five synthesis filters connected in parallel. Each synthesis filter corresponds to one of the frequency bands {0-500, 500-1000, 1000-2000, 2000-3000, 3000-4000} Hz, and its weight is controlled by the voicing strengths vs_i . The impulse response of the pulse shaping filter is given by $h_p(n) = \sum_{i=1}^{5} vs_i h_i(n)$ and the noise shaping filter by $h_n(n) = \sum_{i=1}^{5} (1 - vs_i) h_i(n)$, were $h_i(n)$, $i = 1,\ldots,5$ denotes the impulse response of the synthesis filter. The pulse filter and noise filter mixing weights are complementary, resulting in a constant total gain.

Spectral Enhancement Filter: Since the low energy regions of a speech signal are more sensitive to noise, the spectral enhancement filter attenuates the components in the spectral valleys of the input signal to enhance the perceptual quality of the synthetic speech. Since the frequency response of the synthesis filter of an LP coder effectively follows the peaks and valleys of the speech signal, the spectral enhancement filter is derived from the synthesis filter. An all-pole filter

$$H(z) = \frac{1}{1 + \sum_{i=1}^{10} a_i \alpha^i z^{-i}}$$

effectively reduces perceived noise, but is accompanied by a lowpass spectral tilt, which results in muffled speech. Zeros are added to reduce the spectral tilt. The resultant pole-zero filter is the log difference of bandwidth expanded LP coefficients. To further reduce spectral tilt, a first order filter is added. The final spectral enhancement filter is given by

$$H(z) = \left(1 - \mu z^{-1}\right) \frac{1 + \sum_{i=1}^{10} a_i \beta^i z^{-i}}{1 + \sum_{i=1}^{10} a_i \alpha^i z^{-i}}$$

where $0 < \beta < \alpha < 1$ and $0 < \mu < 1$ are adaptive parameters whose values depend on signal characteristics. For MELP the values $\alpha = 0.8$, $\beta = 0.5$, and $\mu = 0.5$ have been found to yield good results.

Synthesis Filter: The synthesis filter is the usual LP coder vocal tract modeling filter described in Section 3.1.2. Given the interpolated gain *g* and the adjusted pitch period *T*, the output of the synthesis filter $y(n)$ is scaled by the factor

$$g_0 = \frac{10^{g/20}}{\sqrt{\frac{1}{T}\sum_n y^2(n)}}$$

so that the power of the output signal $g_0 \times y(n)$ is equal to the interpolated gain of the current period.

Pulse Dispersion Filter: The pulse dispersion filter is a nearly allpass filter derived from a spectrally flattened triangle pulse with relatively small changes in the magnitude response. The purpose is to improve the match between the reconstructed synthetic speech and the original speech in frequency bands where there is no formant resonance by decreasing the peakiness of the reconstructed synthetic waveform in frequencies away from the formants [McC1995].

3.2 AUTOMATIC RECOGNITION OF CODED SPEECH

3.2.1 Motives for coding speech prior to recognition

Recognition often is performed on recorded speech which is stored until the desired time for recognition. This storage may require much space if the signal is not first compressed. To allow for greater storage efficiency, compression or encoding techniques are used to reduce the signal size prior to recognition. Commonly the signal is then decoded and reconstructed when the recognition is to be performed. This excessive processing causes distortions in the speech signal, and hence in the signal parameters, which result in decreased recognition accuracy.

3.2.2 Problems encountered in the recognition of coded speech

Distortions of the speech waveform arise when the waveform is processed prior to recognition, as in the case when the speech is coded for transmission or storage purposes. Speech coders may introduce error in three ways. First, the quantization of the speech signal and of the encoded parameters leads to imprecise representation of both the parameters and the reconstructed speech. Second, bit errors may occur during transmission of the bitstream, yielding erroneous results at the receiving end. These bit errors, however, are minimized by error checks performed at the decoding end. Finally, the inaccuracy of the speech coder itself will result in distortion. All of these avenues for distortion will yield a reconstructed speech waveform which may differ significantly from the original speech waveform. This difference is illustrated in Figure 8 which shows the time domain graph of a 30 millisecond segment of voiced speech on the top and the corresponding magnitude spectrum on the bottom. Although acoustically similar, the graph of the uncoded speech on the left varies significantly from that of the reconstructed speech on the right. The research presented in this document focuses on reducing the distortion caused by the signal reconstruction.

A simple block diagram of the recognition process is illustrated in Figure 9. With the typical recognition process, the speech signal is coded then reconstructed. Recognition is then performed on the reconstructed waveform, which, as stated previously, may be significantly different from the original waveform. The distortion of the speech waveform results in inaccurate parameter calculations. Reducing the amount of processing on the signal will therefore result in more accurate parameter estimations.

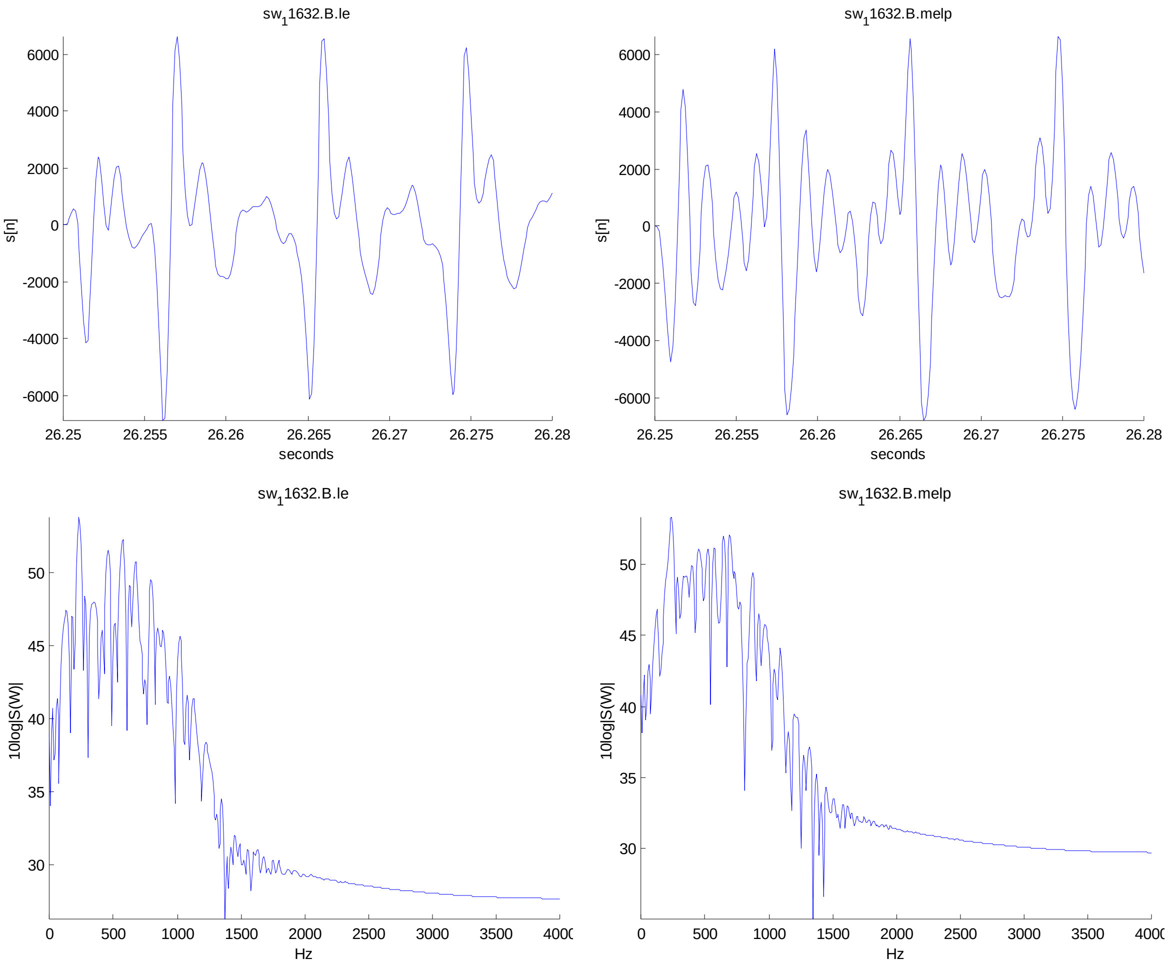


Figure 8. Effects of Coding in Time and Frequency (30ms frame of voiced speech)
<u>Left</u>: Original speech waveform. <u>Right</u>: Reconstructed speech waveform
<u>Top</u>: Time-domain representation. <u>Bottom</u>: Frequency-domain representation (log magnitude)

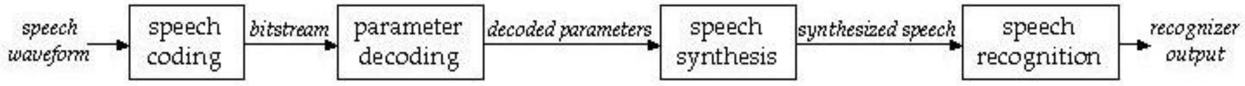


Figure 9. Recognition of Coded Speech

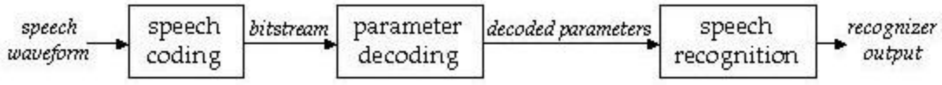


Figure 10. Recognition of Coded Speech from the Bitstream

Figure 10 shows the block diagram illustration of the research presented in this document. For the current research, the speech recognizer takes as input the encoded bitstream rather than the reconstructed signal. The parameters utilized in coding are similar, though not identical, to those used in the recognition of speech. Therefore, recognition directly from the encoded bitstream is practicable and will eliminate the processing steps necessary for reconstruction, and hence the distortion caused by it. Furthermore, this approach requires less storage space since the signal is not reconstructed, and is more time efficient than the traditional approach, avoiding the redundancy of calculating the same or similar parameters for the coding step and again for the recognition step. The detailed implementation of this approach and the results are presented in Chapter 4, but we first consider similar research on the bitstream recognition of coded speech.

3.2.3 Related research on the recognition of coded speech

Three main schemes exist for the recognition of coded speech. They are local recognition, remote recognition, and distributed recognition. These schemes indicate the location of the recognizer in a client-server framework as shown in Figure 11. In this setup the client can be any speech-enabled device and the server is some speech processing application.

Local recognition, also known as client-only recognition, embeds the recognizer in the speech coder, that is, in the client. The benefits of this method are that recognition is performed on the undistorted original speech waveform and recognition accuracy is independent of the transmission channel. However, this approach requires much additional computation at the client end. Furthermore, the technique is limited to those systems powerful enough to manage the added computation and is not ideal for systems that may require quick access to potentially large grammar databases [Dig1999].

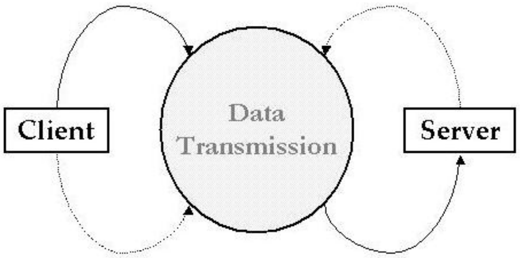


Figure 11. Client-Server Diagram

A second approach called remote, or server-only, recognition performs the recognition entirely at the server end. This approach preserves bandwidth and requires no additional computation or memory at the client end. However, signal errors due to transmission and encoding must be taken into account.

The third and final approach to speech recognition is called distributed speech recognition, or client-server recognition. This approach seeks the benefits of the former two by extracting the recognition parameters at the client end and performing the remaining computations at the server end. The additional computation required to extract the recognition parameters and the additional bandwidth required to transmit them are minimal. However, this approach would require a standardized front-end to insure that standard parameters are sent to the server for recognition [Zha2000].

For our purposes, we consider remote recognition where any device that utilizes speech coding may function as the client, while the speech decoder/synthesizer acts as the server. For this approach, the bitstream is encoded at the client end and then transmitted to the server where the encoded bitstream is decoded and the speech reconstructed.

To avoid the necessity of reconstructing the speech waveform and the distortion that results, a front-end for the recognizer must be developed that directly converts the encoded bitstream parameters into those used by the recognizer. A survey of the development of bitstream front-ends for various speech coders is given here.

GSM Coders: A bitstream front-end was developed by Juan Huerta which derived cepstral features for recognition from GSM (Global System for Mobile) parameters [Hue1998]. The recognition was performed on the NIST Resource Management Corpus (RM1), a speaker-independent, read utterance, small vocabulary, microphone quality corpus. GSM is a 13 kb/s coder that represents the speech waveform using two sets of parameters: those containing information about the LP filter and those containing information on the residual signal. Recognition performed using mel-frequency cepstral coefficients (MFCC) computed from the decoded LP coefficients resulted in an error rate of 12.1 compared to 12.3 and 10.3 for recognition performed directly on the reconstructed and the uncoded waveforms, respectively. So recognition directly from the bitstream exceeded recognition of the reconstructed waveform.

Noting that, for GSM coding, cepstral features may be extracted from both the LP filter and the residual signal components, Huerta formed the cepstral vector used for recognition from the concatenation of cepstra extracted from both sets of parameters. With this concatenation, Huerta was able to achieve a recognition error rate equal to that obtained from recognition of the uncoded speech waveform. The results are summarized in Table 3.

Table 3. Recognition from GSM Bitstream (Huerta)

Input Data Format	Error
Original Waveform	10.3
GSM Reconstructed Speech	12.3
GSM Bitstream - LPC Cepstra	12.1
GSM Bitstream - Concatenated Cepstra	10.3

Ascensión Gallardo-Antolín also investigated the recognition of GSM encoded speech directly from the bitstream [Gal2005]. Recognition results for both the full-rate 13 kb/s and the half-rate 5.6 kb/s GSM coders were examined utilizing the RM1 corpus for continuous speech. Recognition from the bitstream was performed using a technique that estimated the frame energy from four decoded parameters. The recognition error for the uncoded speech input was 9.17. For the full-rate reconstructed waveform input and the full-rate bitstream input, the error rates were 11.9 and 12.36, respectively. In this case, recognition directly from the bitstream yielded a higher error than recognition of the reconstructed waveform. For the half-rate coder, however, recognition from the bitstream yielded an 11.91 error rate while recognition of the reconstructed waveform yielded a higher error rate of 14.61. These results are summarized in Table 4.

Table 4. Recognition from GSM Bitstream (Gallardo-Antolín)

Input Data Format	Error
Original Waveform	9.17
Full-Rate GSM Reconstructed Speech	11.90
Full-Rate GSM Bitstream	12.36
Half-Rate GSM Reconstructed Speech	14.61
Half-Rate GSM Bitstream	11.91

We note that, in difference to the research of Huerta and Gallardo-Antolín, which focused on small-vocabulary, microphone-quality read speech encoded at 13 kb/s and 5.6 kb/s, our research focuses on the large-vocabulary recognition of phone-quality conversational speech encoded at the very low bit rate of 2.4 kb/s.

LP Coders: Seung Ho Choi considered the recognition of speech coded with the 8 kb/s Qualcomm Code-Excited Linear Prediction (QCELP) coder [Cho2000]. Choi incorporated weighting functions into the LSP distance measures to improve recognition accuracy. Using a connected Korean digit corpus, Choi achieved recognition error rates of 13.0 for the uncoded speech input, 21.3 for the reconstructed waveform input, and 17.4 for the bitstream input. Incorporating the use of psuedo-cepstra, the error rate for the bitstream input improved to 13.7. These results are summarized in Table 5.

Table 5. Recognition from QCELP Bitstream (Choi)

Input Data Format	Error
Original Waveform	13.0
QCELP Reconstructed Speech	21.3
QCELP Bitstream	17.4
QCELP Bitstream w/ Weighting	14.5
QCELP Bitstream w/ Psuedo-Cepstra	13.7

Hong Kook Kim investigated the recognition of speech coded using the 7.4 kb/s IS-641 coder [Kim2001]. The recognition was performed using a speaker-independent, connected digit, telephone quality corpus. For recognition from the bitstream, the cepstra were computed directly from the bitstream parameters. In addition, voicing determination from codebook gains was incorporated in the feature set. Recognition directly from the bitstream was performed under matched conditions, that is both training and testing were conducted using the encoded bitstream. The results are shown in Table 6. A recognition error rate of 3.83 is achieved from the original waveform input, 4.84 from the reconstructed input, and 4.19 from the bitstream input.

Table 6. Recognition from IS-641 Bitstream (Kim)

Input Data Format	Error
Original Waveform	3.83
IS-641 Reconstructed Speech	4.84
IS-641 Bitstream (matched conditions)	4.19

While both Kim and Choi utilized coders based on linear prediction, the research presented in this document utilizes a 2.4 kb/s coder, in contrast to the 8 kb/s and 7.4 kb/s coders used by Choi and Kim, respectively. Furthermore, both Choi and Kim performed their research using connected digit corpora, while the current research utilizes a continuous speech corpus.

MELP Coder: Experiments similar to those of Kim were conducted by Serdar Tuğaç on 2.4 kb/s Mixed-Excitation Linear Prediction (MELP) encoded speech using the TI 146 database [Tug2002]. This database is an isolated digit, small vocabulary corpus consisting of 10-digits plus 20-words spoken by sixteen speakers. The HTK speaker-dependent ASR system was used. The following three setups were considered:

- Recognition performance of an ASR that is trained and tested with the original uncoded input speech signal.
- Recognition performance of the same ASR whose input is the reconstructed speech obtained from the MELP coder.
- Recognition performance of the same ASR using a feature set extracted from the bitstream of the MELP coder.

These approaches are identical to those of the research presented in this document. However, unlike the current research, Tuğaç chose to perform recognition from bitstream parameters under like conditions. That is, both training and testing were performed using the bitstream parameters as input. In addition to the common floating-point operation MELP coder, Tuğaç also considered the fixed-point coder.

As the current research exclusively considers the floating-point MELP coder, these are the results shown in Table 7. A recognition error rate of 0.24 was achieved by Tuğaç from the original waveform input, 2.26 from the reconstructed input, and 0.26 from the bitstream input.

Table 7. Recognition from MELP Bitstream (Tuğaç)

Input Data Format	Error
Original Waveform	0.24
MELP Reconstructed Speech	2.26
MELP Bitstream (matched conditions)	0.26

Although Tuğaç recognized MELP encoded speech from the bitstream, both Tuğaç and Kim perform recognition experiments under like conditions, in contrast to the current research which uses mismatched conditions for recognition from the bitstream. Also, the isolated-digit, small vocabulary corpus and speaker-dependent ASR system differ from the conversational speech, large vocabulary corpus and speaker-independent system used for the current research.

3.2.4 Summary of experimental distinctions

All of the above experiments attempt to recognize encoded speech directly from the bitstream, without reconstructing the speech signal. The results are then compared to those obtained from the recognition of uncoded speech and the recognition of the reconstructed speech. The differences lie not only in the method used to obtain recognition parameters from the bitstream, but also in the choice of coder and corpus. These choices affect the success of the attempt.

The research of Huerta and Gallardo-Antolín focused on a small-vocabulary corpus with speech encoded at 13 kb/s and 5.6 kb/s. Kim and Choi, respectively, utilize 8 kb/s and 7.4 kb/s coders based on linear prediction and perform their research using connected digit corpora. Finally, although Tuğaç utilized the 2.4 kb/s MELP coder, he chose a isolated digit, small vocabulary corpus for use in a speaker-independent ASR system. Furthermore, both he and Kim performed recognition experiments under like conditions.

While previous research focused on small vocabulary and/or connected or isolated digit corpora, the current research utilizes conversational speech, large vocabulary corpora, containing 22,000 to 26,000 words. Also, with the exception of the research of Tuğaç, previous research considers coders with bitrates ranging from 7.4 kb/s to 13 kb/s, while the current research considers the 2.4 kb/s MELP coder. This very low bit rate leads to difficulty in the estimation of the original waveform parameters, which in turn results in higher recognition error. Although Tuğaç also considers the MELP coder, he has chosen to perform speaker-dependent recognition under matched conditions, in contrast to the current research which performs speaker-independent recognition under mismatched conditions for training and testing.

4.0 SPEECH RECOGNITION FROM MELP BITSTREAM

The objective in this research is to recognize MELP encoded speech directly from the bitstream, without reconstructing the speech, while maintaining or exceeding the accuracy obtained from the recognition of reconstructed speech. The purpose is to avoid additional processing distortion as well as additional distortion caused by reconstruction and reanalysis of the speech, and to eliminate the time required for reconstruction of the coded speech.

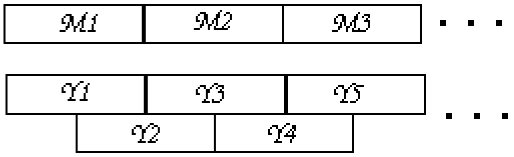
4.1 EXPERIMENT SETUP

Recognition is performed using the Byblos large vocabulary conversational speech recognition system developed by BBN. The recognition system is briefly explained in Section 2.4.

4.1.1 Recognition Training and Testing

Experiments are performed as follows. Training for all experiments is executed on uncoded speech. The training data consists of over ninety-nine hours of speech from 750 speakers from the Switchboard Corpus. The window duration for training data is 25 milliseconds with a frame rate of 10 milliseconds.

The test data consists of approximately three hours of speech from 80 speakers from the Switchboard and Call Home English corpora. The window duration is set at 22.5 milliseconds to match the standard MELP coding window duration. MELP coding uses non-overlapping frames. Therefore, a simple linear interpolation of the MELP parameters is used to approximate a 50% frame overlap during recognition. Figure 12 illustrates the interpolation process, where $\mathcal{M}$ denotes the original MELP analysis parameter and Υ denotes the interpolated MELP parameter used by the recognition system.



$$\Upsilon_1 = \mathcal{M}_1 \qquad \Upsilon_3 = \mathcal{M}_2 \qquad \Upsilon_5 = \mathcal{M}_3$$

$$\Upsilon_2 = ½\, \mathcal{M}_1 + ½\, \mathcal{M}_2 \qquad \Upsilon_4 = ½\, \mathcal{M}_2 + ½\, \mathcal{M}_3$$

Figure 12. MELP Parameter Interpolation

4.1.2 Baseline Results

Testing was performed on uncoded speech and on reconstructed speech, that is speech that is coded, decoded, and reconstructed using the MELP coding algorithm detailed in Section 3.1.4. Comparison of the results, which are shown in Table 8 , reveal an absolute difference of 4.9% and a relative difference of 6.7%.

Table 8. Baseline MELP Recognition Results

Expt #	Input Data Format	Error
4100	Original Waveform	73.3
4200	MELP Reconstructed Speech	**78.2**

The purpose of this phase of research is to recognize MELP coded speech directly from the bitstream, without the need for signal reconstruction, while maintaining or exceeding the accuracy obtained from the recognition of reconstructed speech. In addition, recognition directly from the bitstream does not require the additional time or bandwidth that recognition of reconstructed speech requires. The implementation of bitstream recognition is detailed in Section 4.2 and a summary of the results is presented in Section 4.3.

4.2 IMPLEMENTATION

The primary parameters encoded into the MELP bitstream are the line spectral frequency pairs, Fourier magnitudes and gain. From these MELP parameters we obtain those parameters utilized by Byblos for recognition, namely the frame energy, the predictor gain, and the linear prediction coefficients utilized to compute the cepstrum. The relationship between the MELP bitstream parameters and the recognition parameters is depicted in Table 9.

Table 9. MELP Bitstream and Speech Recognition Parameters

MELP Bitstream Parameter	Speech Recognition Parameter
Line Spectral Frequencies	Linear Prediction Coefficients
Fourier Magnitudes	Linear Prediction Coefficient Adjustment
Gain	Frame Energy

Because MELP uses non-overlapping frames, it is necessary to interpolate the MELP analysis parameters prior to recognition as depicted in Figure 12 of Section 4.1.1. The pitch period and the line spectral frequencies are interpolated directly from the set of decoded parameters and passed to the recognizer. Linear energy for each frame is computed from the decoded gain and interpolated prior to conversion to the log energy. Computation of the energy from the decoded linear gain is explained in Section 4.2.1. The Fourier magnitudes are interpolated only if the spectral enhancement filter coefficients are to be computed. The incorporation of the enhancement filter is covered in Section 4.2.2. The decoded and interpolated parameters require further modification as explained in Section 4.2.3.

Once the analysis parameters have been interpolated and processed as necessary, recognition is performed and the error rate is compared to that which results from the input of the unprocessed speech waveform and the MELP reconstructed speech waveform. The progression of the processing of the analysis bitstream is detailed in the remainder of this section.

4.2.1 Frame energy and spectral gain computations

The frame energy utilized by the Byblos speech recognition system is not readily available in MELP. However, we may compute the linear frame energy, $\tilde{E}$, from the log gain, g, and the pitch period, T, which are obtained directly from the MELP bitstream. Consider the portion of the decoding illustrated in Figure 13. The gain scale factor, g_0, is computed form the linear gain, $\tilde{g}$, and the pitch period, T, as

$$g_0 = \frac{\tilde{g}}{\sqrt{\frac{1}{T}\sum_n y^2(n)}}$$

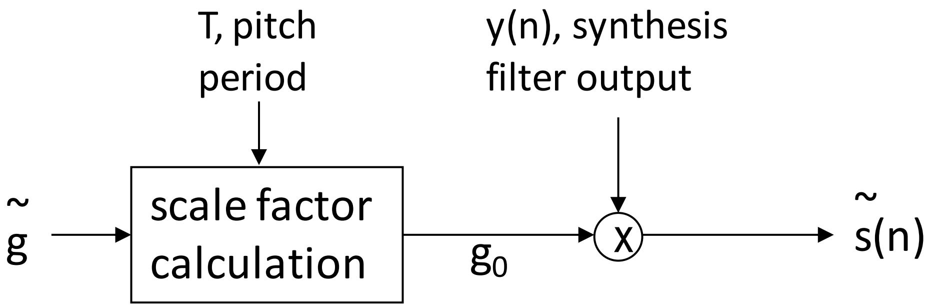


Figure 13. Scale Factor Calculation

The scaled speech signal is then given as $\tilde{s}(n) = g_0 y(n)$. Therefore, we can compute the energy of the reconstructed signal as follows:

$$g_0 = \frac{\tilde{g}}{\sqrt{\frac{1}{T}\sum_n y^2(n)}}$$

$$g_0 = \frac{\tilde{g}}{\sqrt{\frac{1}{T}\sum_n \left(\frac{\tilde{s}(n)}{g_0}\right)^2}}$$

$$g_0\sqrt{\frac{1}{T}\sum_n \left(\frac{\tilde{s}(n)}{g_0}\right)^2} = \tilde{g}$$

$${g_0}^2\left(\frac{1}{T{g_0}^2}\sum_n \tilde{s}^2(n)\right) = \tilde{g}^2$$

$$\frac{1}{T}\sum_n \tilde{s}^2(n) = \tilde{g}^2$$

$$\sum_n \tilde{s}^2(n) = T\tilde{g}^2$$

The energy of the reconstructed speech is approximated as: $\tilde{E} = T\tilde{g}^2$. This is the frame energy that we pass to the recognizer.

Next, we compute the prediction error gain, G, for use with the LP filter, $H(z) = \dfrac{G}{1 + \sum_{j=1}^{p} a_j z^{-j}}$ from which the cepstra are computed. This prediction error gain is computed from the linear prediction coefficients, a_j, and the linear energy, $\tilde{E}$, as

$$G = E \times \prod_{j=1}^{p} \left(1 - k_j\right)$$

where k_j are the reflection coefficients computed from the linear prediction coefficients.

With the energy and prediction gain computed as described above, the recognition error rate is determined to be 78.7. This value is obtained by using a default pitch period for the computation of the spectral gain for unvoiced speech. By using the encoded pitch rather than the default value, the recognition error rate decreases by 0.1 percentage point. These results from the analysis bitstream input are compared to those for the original waveform and the MELP reconstructed speech inputs in Table 10.

Table 10. MELP Bitstream Recognition

Expt #	Input Data Format	Error
4100	Original Waveform	73.3
4200	MELP Reconstructed Speech	78.2
5100	MELP Bitstream	**78.6**

In these results we note that the recognition error corresponding to the bitstream input is 0.4 percentage points greater than the error rate corresponding to reconstructed speech. The logical conclusion is that some of the processing used to reconstruct speech from the analysis bitstream would be advantageous for improved recognition. Our goal then becomes to process the bitstream parameters only as much as necessary to achieve recognition accuracy comparable to that of the reconstructed speech, without actually reconstructing the speech signal. Thus, we incorporate the MELP filters to determine their effects on the LP coefficients, and hence on the spectral parameters.

4.2.2 Filter incorporation

Given the results of Table 10, we set out to make the MELP analysis bitstream parameters as close as possible to those of the MELP reconstructed speech signal without in fact reconstructing the signal. We begin by calculating the effects of various MELP decoding filters on the linear prediction coefficients passed to Byblos. Figure 14 demonstrates a simplified MELP decoding process. From this diagram we see several decoding filters and processes. Those which would significantly affect the LSF, and consequently the linear prediction coefficient values, namely adaptive spectral enhancement filters and spectral tilt, are incorporated into the recognition front-end.

The adaptive spectral enhancement filter is a 10th order pole-zero filter given as

$$H_0(z) = \frac{1 + \sum_{i=1}^{p} a_i \beta^i z^{-i}}{1 + \sum_{i=1}^{p} a_i \alpha^i z^{-i}}$$

An additional adaptive, first-order, all-zero process compensates for spectral tilt. The spectral enhancement filter combined with the spectral tilt compensation is given by

$$H_1(z) = \left(1 - \mu z^{-1}\right) \frac{1 + \sum_{i=1}^{p} a_i \beta^i z^{-i}}{1 + \sum_{i=1}^{p} a_i \alpha^i z^{-i}}$$

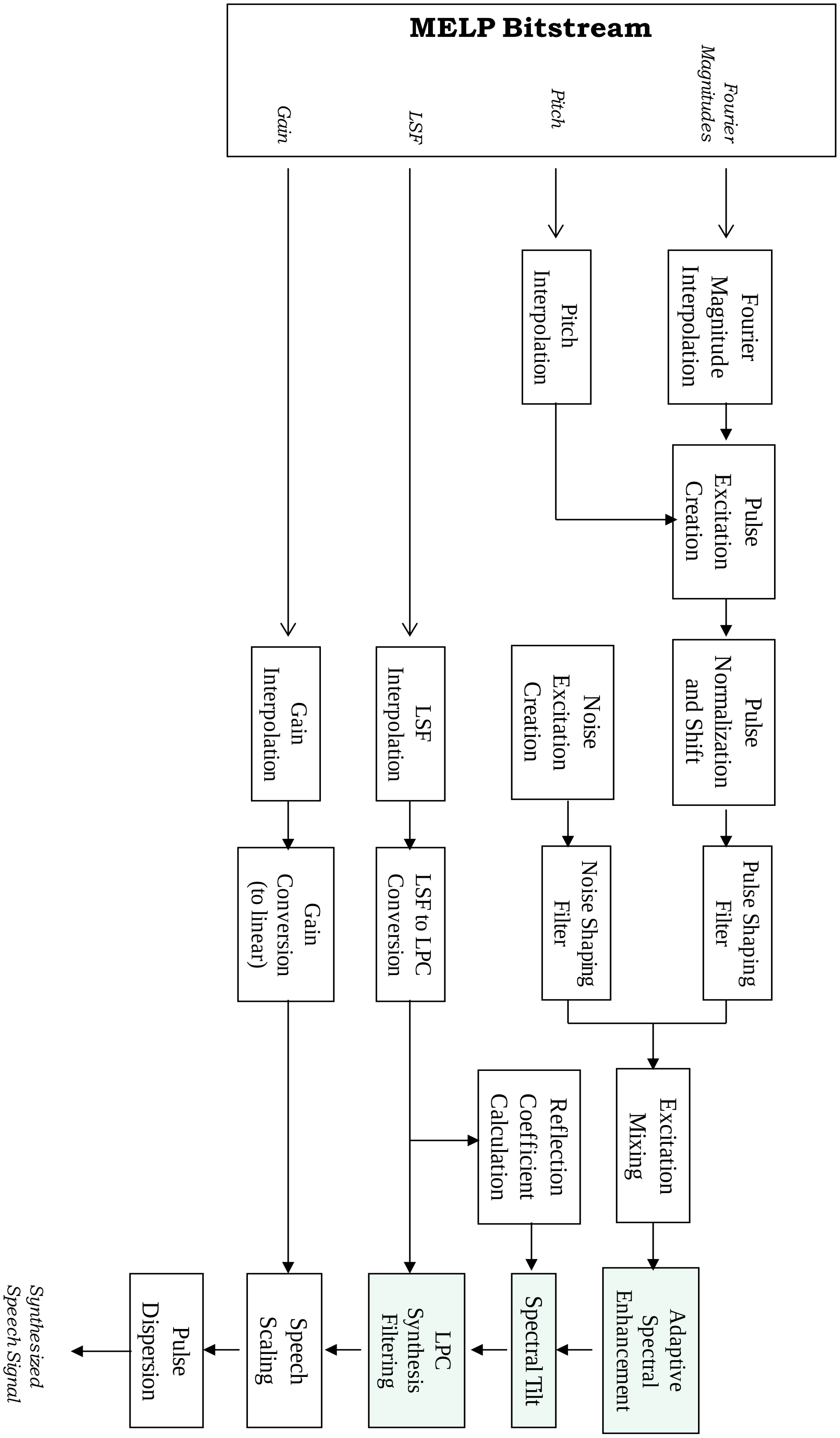


Figure 14. Parameters Used for Recognition

where a_i are the linear prediction coefficients and α, β, and μ are constants as described in Section 3.1.4.

The linear prediction filter is given by

$$H_2(z) = \frac{G}{1 + \sum_{i=1}^{p} a_i z^{-i}}$$

Combining the prediction and enhancement filters, with the spectral tilt component, we obtain

$$H(z) = H_1(z) H_2(z) = \frac{G\left(1 - \mu z^{-1}\right)\left(1 + \sum_{i=1}^{p} a_i \beta^i z^{-i}\right)}{\left(1 + \sum_{i=1}^{p} a_i z^{-i}\right)\left(1 + \sum_{i=1}^{p} a_i \alpha^i z^{-i}\right)} = \frac{G\left(1 + \sum_{i=1}^{p} \eta_i z^{-i}\right)}{1 + \sum_{j=1}^{q} \lambda_j z^{-j}}$$

where η_i and λ_j are determined as the convolution coefficients of the numerator and denominator filters, respectively. The frequency response of the combined filter is shown in Figure 15 in comparison to the frequency response of the linear prediction filter.

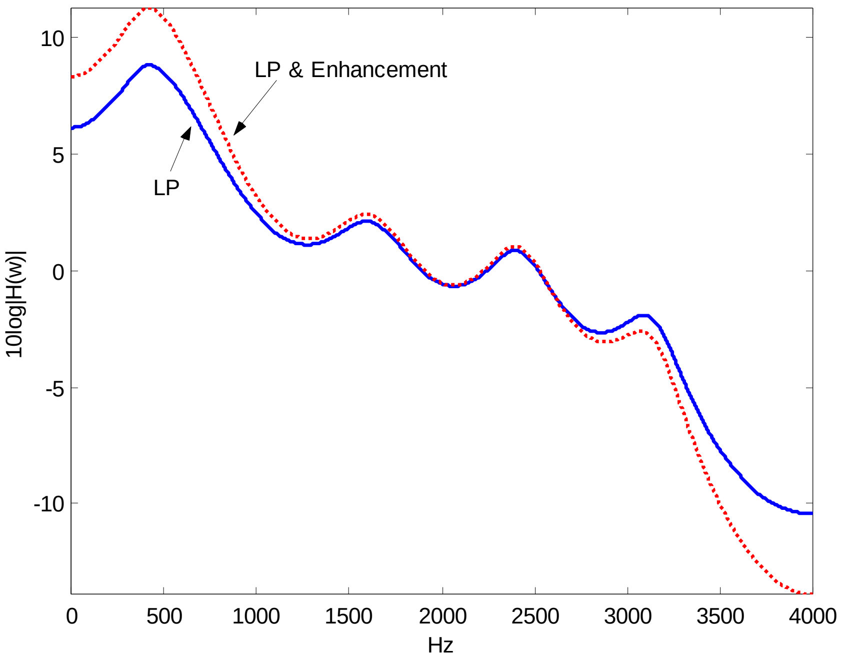


Figure 15. Effect of Spectral Enhancement Filter
LP indicates the linear prediction filter (solid line). LP&Enhancement indicates the combined linear prediction and spectral enhancement filters (dashed line)

Our need is an autoregressive approximation of this autoregressive-moving average combined filter $H(z)$. To achieve this approximation we employ Kolmogorov's theorem which assures that any finite order ARMA filter can be approximated by an infinite order AR filter via the following equation:

$$c_j = -\sum_{k=1}^{p} \eta_k c_{j-k} + \sum_{k=0}^{q} \lambda_k \delta_{j-k}$$

given that

$$\{c_{-q}, c_{-q+1}, \ldots, c_{-2}, c_{-1}\} = 0 \text{ and } \delta_{j-k} = \begin{cases} 1, & j = k \\ 0, & j \neq k \end{cases}.$$

Then we have

$$H(z) = \frac{G\left(1 + \sum_{i=1}^{p} \eta_i z^{-i}\right)}{1 + \sum_{j=1}^{q} \lambda_j z^{-j}} \approx \frac{G}{1 + \sum_{k=1}^{L} c_k z^{-k}}$$

where L is the finite order determined such that $c_n \approx 0$ for all $n > L$ [Kay1988].

The incorporation of the adaptive spectral enhancement results in an increase in recognition error when compared to the error when no filters are incorporated. The results are noted in Table 11.

Table 11. MELP Bitstream Recognition: Incorporation of Spectral Enhancement

Expt #	Input Data Format	Error
4100	Original Waveform	73.3
4200	MELP Reconstructed Speech	78.2
5100	MELP Bitstream	78.6
5201	MELP Bitstream w/ Spectral Enhancement	**79.0**

The addition of spectral tilt results in no change in recognition error rate, whether included alone or with spectral enhancement as shown in Table 12. The effect of spectral tilt on the LP filter is shown in Figure 16. Figure 17 displays the joint effect of spectral enhancement and spectral tilt.

Table 12. MELP Bitstream Recognition: Incorporation of Spectral Tilt

Expt #	**Input Data Format**	**Error**
4100	Original Waveform	73.3
4200	MELP Reconstructed Speech	78.2
5100	**MELP Bitstream**	**78.6**
5201	**MELP Bitstream w/ Spectral Enhancement**	**79.0**
5301	**MELP Bitstream w/ Spectral Tilt**	**78.7**
5302	**MELP Bitstream w/ Spectral Enhancement & Spectral Tilt**	**79.0**

The lack of improvement in recognition accuracy with the addition of spectral enhancement and spectral tilt is due to the function of these filters. The purpose of the filters is to enhance the perceptual quality of the synthetic speech. The enhancement filter is intended to suppress noise while maintaining the integrity of the speech, while the purpose of the spectral tilt application is to modify the slope of the harmonic amplitude peaks to enhance the perception of word boundaries and consonants. So then, these filters, intended for the enhancement of human perception, do not necessarily improve the signal for the purpose of automatic recognition.

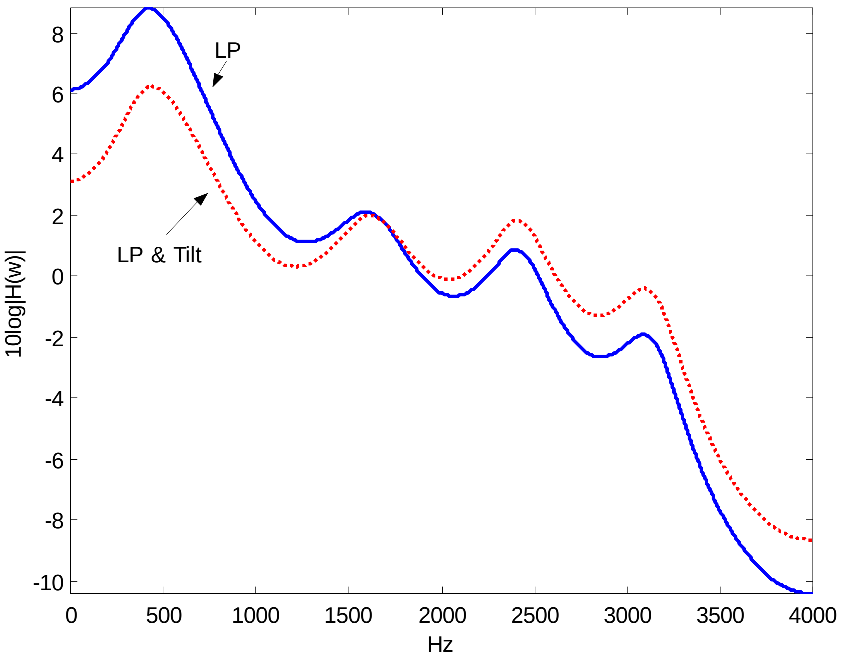


Figure 16. Effect of Spectral Tilt
LP indicates the linear prediction filter (solid line). LP&Tilt indicates the combined linear prediction and spectral tilt filters (dashed line)

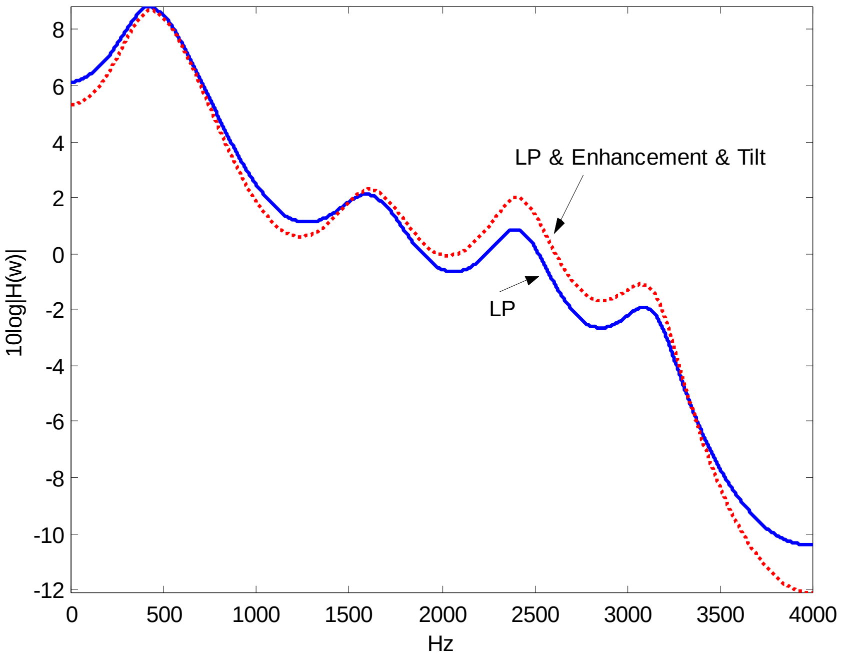


Figure 17. Effect of Spectral Enhancement and Spectral Tilt Filters
LP indicates the linear prediction filter (solid line). LP&Enhancement&Tilt indicates the combined linear prediction, spectral enhancement, and spectral tilt filters (dashed line)

4.2.3 Parameter adjustments

Due to the unsatisfactory results of incorporating spectral enhancement and spectral tilt from the decoder portion of MELP, our new direction is to undo any processing done prior to the analysis encoding that may lead to increased recognition error. Since bandwidth expansion is used during the MELP analysis phase, a bandwidth compression is employed prior to recognition. For MELP, bandwidth expansion is given as $a_k = \left(0.994^k\right)\tilde{a}_k$, where a_k are the original linear prediction coefficients and $\tilde{a}_k$ are the bandwidth expanded linear prediction coefficients. Therefore, the compression employed prior to recognition is given by $\tilde{a}_k = \frac{a_k}{0.994^k}$. This compression of the bandwidth results in a recognition error rate of 78.4, the first decrease in recognition error. The error rate is further reduced by 0.1 percentage points when these bandwidth compressed linear prediction coefficients are also used in the calculation of the reflection coefficients, which are used in the calculation of the prediction error gain. The results are compared to those for the original speech waveform and those for the MELP reconstructed speech in Table 13.

Table 13. MELP Bitstream Recognition: LP Coefficients Bandwidth Compression

Expt #	Input Data Format	Error
4100	Original Waveform	73.3
4200	MELP Reconstructed Speech	78.2
5101	MELP Bitstream	**78.3**

At this point our results are 0.1 percentage points greater than the baseline error rate of 78.2 achieved from recognition of the reconstructed waveform. Observational analysis reveals that the frame energy from the analysis bitstream differs noticeably from that of the MELP reconstructed waveform. Figure 18 shows the plots of the log energy values extracted over 500 frames from the original waveform, from the reconstructed waveform, and from the bitstream. This difference in energy levels is a result of the scale factor mentioned in Section 4.2.1. We recall that the scaled reconstructed speech signal, $\tilde{s}(n)$, is given as $\tilde{s}(n) = g_0 y(n)$, where

$$g_0 = \frac{\tilde{g}}{\sqrt{\frac{1}{T}\sum_n y^2(n)}}$$

is the scale factor which depends on the output of the synthesis filter $y(n)$, the pitch period T, and the linear gain $\tilde{g}$. Since the power, and hence the energy, of the MELP decoder excitation signal is arbitrary, this factor is necessary to insure that the power of the reconstructed signal is equal to the gain [Chu2003]. However, this factor depends on the output $y(n)$ of the synthesis filter, and calculation of this output is not desirable for bitstream recognition. Therefore, the energy difference must be otherwise compensated.

To account for this difference a polynomial mapping is determined using 5000 frames of speech from a mutually exclusive corpus to map the bitstream log energy $\tilde{E}_b$ to the reconstructed synthetic waveform energy $\tilde{E}_s$. The polynomial is computed using a least squares method, which is detailed in Appendix A. First degree through seventh degree polynomials are computed, and the root mean square (RMS) error is calculated. Due to lack of significant difference in the RMS error values and sufficiently low RMS error values for all polynomial degrees, and for computational ease, the linear mapping is chosen and is given as

$$\tilde{E}_s \approx 1.09092\tilde{E}_b - 13.53115$$

where $\tilde{E} = 10\log E$. Through cursory observation of Figure 18 the difference between the log energy extracted from the analysis bitstream appears to be a vertical shift of the log energy of the synthetic waveform corresponding to the same frame of speech. This observation is confirmed as the coefficient of $\tilde{E}_b$ is close to unity, indicating that the log energy of the reconstructed speech waveform is close to a constant negative vertical shift of the log energy extracted from the analysis bitstream. The results of the mapping are shown in Figure19.

The mapping parameters are used to determine the following mapping for the linear energy

$$E_s \approx E_b^{1.09092} \times 10^{-13.53115/10}$$

This mapping is performed prior to computation of the log energy for the purpose of interpolation, and results in an error rate of 75.9. The results are summarized in Table 14.

Table 14. MELP Bitstream Recognition: Energy Mapping

Expt #	Input Data Format	Error
4100	Original Waveform	73.3
4200	MELP Reconstructed Speech	78.2
5102	MELP Bitstream	**75.9**

At this point we have succeeded in recognizing encoded speech directly from the analysis bitstream, without the need for reconstruction of the speech signal. Furthermore, the recognition improvement of 2.6 percentage points when using bitstream input rather than reconstructed speech, shows that recognition from the bitstream is preferable. Also, we note that the difference in error rate is reduced from a 4.9 percentage point difference between original waveform input and reconstructed speech input to a 2.6 percentage point difference between original waveform input and bitstream input.

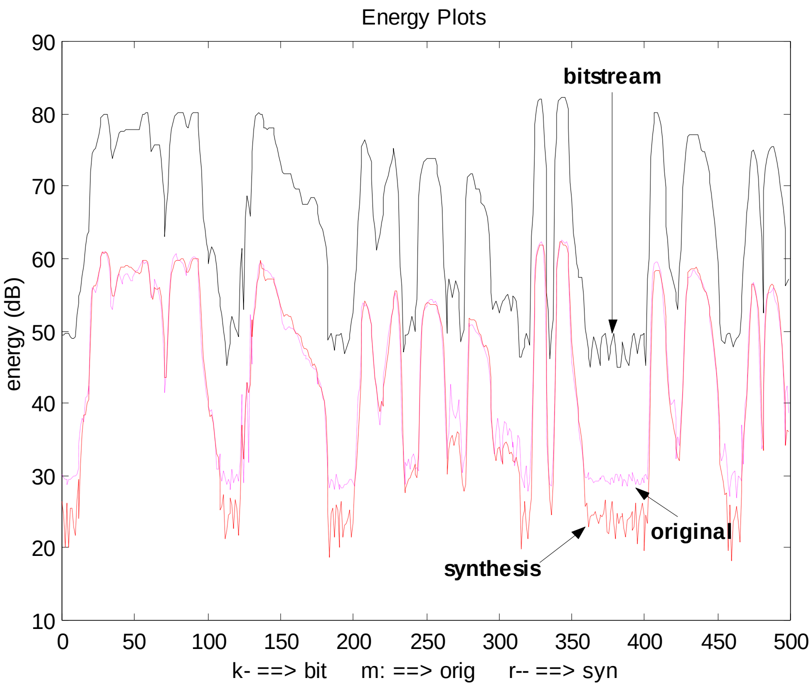


Figure 18. Log Energy Comparison
500 frames of log energy values derived from uncoded waveform,
MELP reconstructed waveform, and MELP bitstream

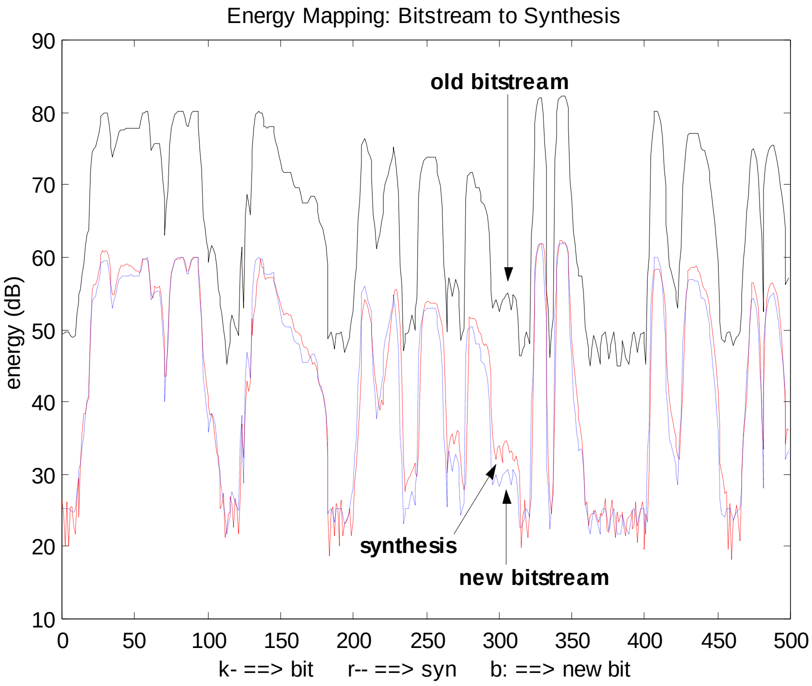


Figure 19. Log Energy Mapping Results
500 frames of log energy values derived from MELP reconstructed waveform,
MELP bitstream, and new MELP bitstream after mapping

4.3 SUMMARY OF RESULTS

Recognizing encoded speech directly from the bitstream requires less computation time and less storage and bandwidth than recognition of reconstructed speech. The time required to reconstruct the speech waveform is eliminated, and the encoded bitstream requires about one-tenth the storage space of reconstructed synthetic speech.

More important than the time and bandwidth savings, recognition of coded speech directly from the bitstream eliminates the distortions caused by decoding and reconstruction of the speech, and results in greater accuracy than recognition of the reconstructed speech. The contributions of this research concern the recognition of large vocabulary, speaker-independent, low bit-rate encoded, continuous speech under mismatched training and testing conditions. That is, the recognition system is trained on uncoded speech and recognition is performed on the 2.4 kb/s MELP encoded bitstream. The large vocabulary, the conversational speech, and the low bit rate each contribute to high recognition error. However, with the adjustments and modifications detailed in Section 4.2, the relative difference in the recognition error rate for bitstream input and that for the uncoded waveform is 3.5%. This percentage of improvement exceeded the results for the higher bit-rate coders, with the exclusion of the 13 kb/s GSM full-rate, concatenated cepstra technique of Huerta. Furthermore, the improvement for the current technique exceeded the 8.33% relative difference under like conditions for MELP presented by Tuğaç.

5.0 SPEECH ENHANCEMENT

The handling of speech in adverse conditions is a primary concern in recognition. Undesirable artifacts in a speech signal due to channel noise, reverberation, background noise, or some other interference, may alter the characteristics of the speech signal, complicating the task of recognition. The effects of noise on the speech waveform are illustrated in Figure 20. In the figure, the top left graph depicts the original clean speech waveform for a three second utterance. The top right graph depicts the same utterance with additive noise, while the noise signal itself is shown in the bottom graph. It is evident that the noisy speech signal has taken on the temporal characteristics of the noise. This alteration of the signal is indicative of the effects that the noise will have on perception and recognition. The goal of enhancement is to improve the overall quality of the speech signal by reducing background noise, reverberation, and channel noise. The goal is to remove any unwanted artifacts which may alter the spectral characteristics of the speech signal. Some common speech enhancement techniques are discussed in Section 5.1.

The enhancement algorithm employed in this research is a minimum mean-square log-spectral amplitude (MMSE-LSA) estimator developed by Ephraim and Malah. The original algorithm was written by Rainer Martin at AT&T Labs Research. MMSE-LSA enhancement is explained in detail in Section 5.1.4. The noise added to the speech signals is that of a military Blackhawk helicopter. The average signal-to-noise ratio was -5.6 dB.

Because noise in a speech signal is common, much of the automatic recognition of speech is performed on a noisy speech signal. If the speech signal is noisy and therefore distorted as shown in Figure 20, the spectral features of that signal also will be distorted and will deviate from the typical patterns on which the system has been trained. This deviation affects the accuracy of the recognizer. An appropriate noise reduction algorithm can alleviate the effects of this distortion. The automatic recognition of noisy speech is covered in Section 5.2. Section 5.3 discusses the development of an enhancement algorithm specific to the needs of a speech recognition system.

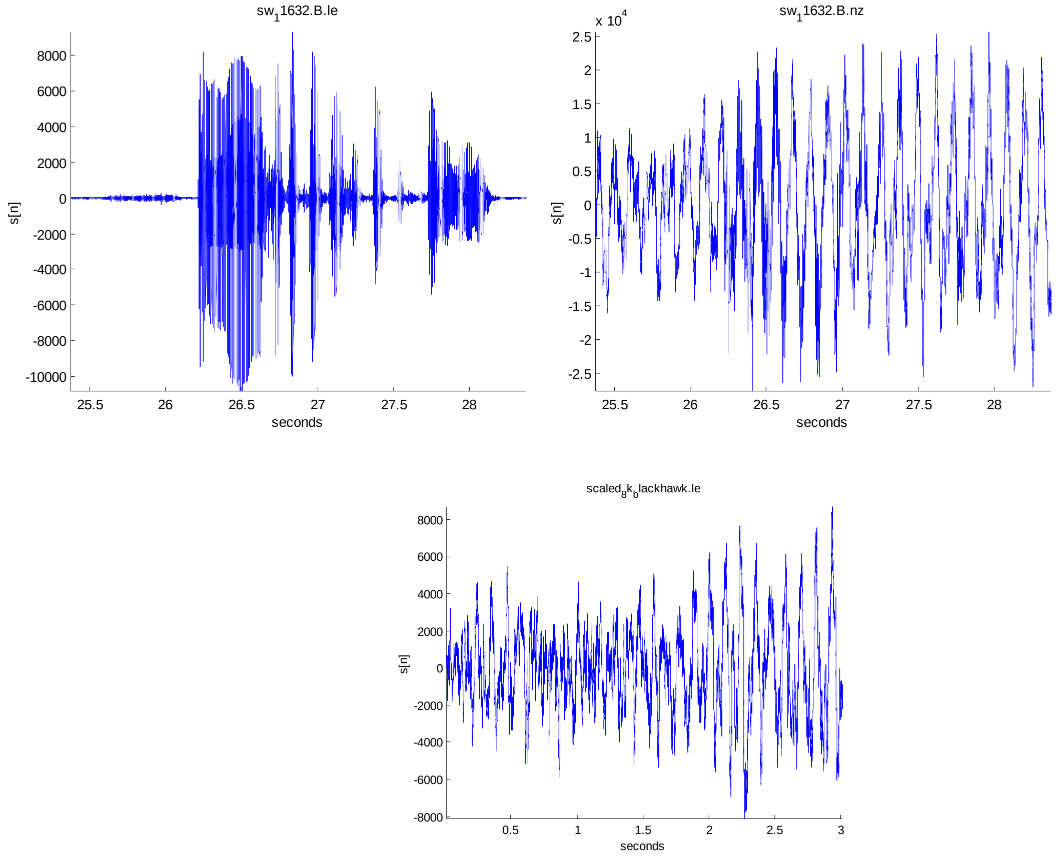


Figure 20. Effects of Noise on the Speech Waveform
Top Left: Original (clean) speech waveform. Top Right: Noisy speech waveform
Bottom: Noise signal.

5.1 COMMON SPEECH ENHANCEMENT TECHNIQUES

Prior to recognition attempts are made to reduce the effects that various types of noise may have on the performance of the recognizer. Noise may be defined as any unwanted signal and therefore is not an uncommon artifact in speech signals. Noise may occur as a result of signal interference, channel distortion during transmission, or limited measurement resolution. Various methods have been designed that attempt to suppress noise, or more specifically reduce the effects that the noise has on a particular application, such as recognition. The desire is that, without modification, the recognizer will perform as well on noisy speech as it has on noise-free speech.

We begin with a speech signal $s(n)$ degraded by additive noise $d(n)$, so that

$$x(n) = s(n) + d(n)$$

is the signal available to us for processing. For computational simplicity, we assume that both the speech and the noise are zero-mean. We focus here on additive noise, noting that multiplicative noises can be converted to additive noise by use of logarithms so that $x(n) = s(n) \cdot d(n)$ becomes $\log\{x(n)\} = \log\{s(n)\} + \log\{d(n)\}$, and convolutional noises may be converted to additive noise by use of a Fourier transform and logarithms so that $x(n) = s(n) * d(n)$ becomes $X\left(e^{j\omega}\right) = S\left(e^{j\omega}\right) \cdot D\left(e^{j\omega}\right)$ and then $\log\left\{X(e^{j\omega})\right\} = \log\left\{S(e^{j\omega})\right\} + \log\left\{D(e^{j\omega})\right\}$, where $X\left(e^{j\omega}\right)$, $S\left(e^{j\omega}\right)$, and $D\left(e^{j\omega}\right)$ denote the Fourier transforms of $x(n)$, $s(n)$, and $d(n)$, respectively.

The current enhancement methods may be categorized into three principal groups identified by the information assumed about the signal [Lim1979]. These include systems based on the periodicity of voiced speech, those using analysis-synthesis techniques, and those employing short-time spectral amplitude estimates. All of these methods assume that only the noisy speech is available for processing and that the noise is additive and uncorrelated with the speech signal.

Enhancement methods based on the periodicity of voiced speech include adaptive comb filtering and high-resolution spectrum techniques which seek to isolate the harmonics of the speech signal. These techniques are built on the premise that the short-time spectrum of speech is essential to perception, with formants being more essential than other spectral envelope features.

Analysis-synthesis enhancement techniques utilize either the all-pole or the pole-zero model of the speech production system, approximating the time-varying system as linear time-invariant for short intervals of speech. Given the model parameters of the speech signal, noise in the signal is reduced by synthesizing speech from those analysis parameters or by filtering the noisy signal with a filter designed using the analysis parameters [Lim1979].

The research presented in this text utilizes an enhancement technique which is based on the short-time spectral amplitude of the speech signal. In this category the spectral amplitude alone is enhanced and then combined with the phase of the noisy signal to create the enhanced signal. This approach is legitimate since it is commonly accepted that while the magnitude of the frequency spectrum is essential to speech intelligibility, the spectral phase is not. The short-time spectral amplitude may be estimated in the frequency domain or the signal may be passed through a Wiener filter computed from the degraded speech. Both approaches are discussed in the following sections.

5.1.1 Wiener filter

The Wiener filter method seeks to derive a minimum phase filter $W(e^{j\omega})$ that minimizes the mean-square error estimate of the signal. That is, given a speech signal $s(n)$ degraded by additive noise the Wiener filter seeks an estimate $\hat{s}(n)$ that minimizes the quantity

$$\xi = E\left\{ \left|s(n) - \hat{s}(n)\right|^2 \right\}$$

where $E\{\cdot\}$ denotes expectation. The non-causal filter that results from this minimization is given by

$$W(e^{j\omega}) = \frac{P_s(e^{j\omega})}{P_s(e^{j\omega}) + P_d(e^{j\omega})}$$

where $P_s(e^{j\omega})$ and $P_d(e^{j\omega})$ denote the power spectra of the speech signal and the noise signal, respectively [Hay1996]. Since $P_s(e^{j\omega})$ is not known, the filter is often approximated as

$$W(e^{j\omega}) = \frac{E\left\{ \left|S_w(e^{j\omega})\right|^2 \right\}}{E\left\{ \left|S_w(e^{j\omega})\right|^2 \right\} + E\left\{ \left|D_w(e^{j\omega})\right|^2 \right\}}$$

where $S_w(e^{j\omega})$ and $D_w(e^{j\omega})$ denote the Fourier transform of segments of the speech and noise signals, respectively. When $P_s\left(e^{j\omega}\right)$ is much larger than $P_d\left(e^{j\omega}\right)$, that is for high signal-to-noise ratio, $\left|W(e^{j\omega})\right| \approx 1$ so that there is little attenuation of the speech signal. On the other hand, when $P_d\left(e^{j\omega}\right)$ is much larger than $P_s\left(e^{j\omega}\right)$, that is for low signal-to-noise ratio, $\left|W(e^{j\omega})\right| \approx 0$ in order to suppress the noise.

$D_w(e^{j\omega})$ may be estimated during non-speech activity, and $E\left\{\ \left|S_w(e^{j\omega})\right|^2\right\}$ may be estimated by computing $E\left\{\ \left|X_w(e^{j\omega})\right|^2\right\}$ over several frames and subtracting from it $E\left\{\ \left|D_w(e^{j\omega})\right|^2\right\}$ [Lim1979]. The enhanced signal $\hat{S}_w(e^{j\omega})$ is then obtained by filtering

$$\hat{S}_w(e^{j\omega}) = W(e^{j\omega})X_w(e^{j\omega})$$

It is noted that since $W(e^{j\omega})$ is zero-phase, the phase of $\hat{S}_w(e^{j\omega})$ is the phase of $X(e^{j\omega})$.

5.1.2 Spectral subtraction

The enhancement algorithm involved in the research presented in this document utilizes the estimation of the short-time spectral amplitude. Unlike the Wiener filter method, this approach seeks to estimate the magnitude spectrum directly in the frequency domain. Two common techniques in this category are spectral subtraction and minimum mean-square error estimation. In either approach we begin with a signal $x(n)$, which is comprised of speech $s(n)$ degraded by additive noise $d(n)$. Spectral subtraction estimates the magnitude frequency spectrum of the speech by subtracting the magnitude frequency spectrum of the noise from that of the degraded speech [Vas1996]. The objective is to obtain an estimate $\left|\hat{S}(e^{j\omega})\right|$ of $\left|S(e^{j\omega})\right|$, where $\left|S(e^{j\omega})\right|$ is the magnitude spectrum of $s(n)$. To this end, the signal is first windowed into stationary segments to obtain

$$x_w(n) = s_w(n) + d_w(n)$$

where $s_w(n)$, $d_w(n)$, and $x_w(n)$ denote the windowed segments of the speech, noise and degraded signals, respectively. We may then take the Fourier transform to obtain the equation

$$X_w(e^{j\omega}) = S_w(e^{j\omega}) + D_w(e^{j\omega})$$

If we can estimate $\left|S_w(e^{j\omega})\right|$, this estimate then can be used to obtain the estimate $\hat{s}_w(n)$ of $s_w(n)$ via the operation

$$\hat{s}_w(n) = \mathsf{F}^{-1}\left\{ \left|\hat{S}_w(e^{j\omega})\right| e^{j\angle X_w(e^{j\omega})} \right\}$$

where $\mathsf{F}^{-1}\{\cdot\}$ denotes the inverse Fourier transform and $\angle X(e^{j\omega})$ is the phase of the degraded signal. Overlap-and-add methods may then be employed to extract the enhanced signal $\hat{s}(n)$ from the windowed estimates [Bol1979].

To estimate $\left|S_w(e^{j\omega})\right|$ we begin with the magnitude squared of the spectrum of the degraded signal, which is given by

$$\left|X_w(e^{j\omega})\right|^2 = \left|S_w(e^{j\omega})\right|^2 + S_w^*(e^{j\omega})D_w(e^{j\omega}) + S_w(e^{j\omega})D_w^*(e^{j\omega}) + \left|D_w(e^{j\omega})\right|^2$$

where $S_w^*(e^{j\omega})$ and $D_w^*(e^{j\omega})$ denote the complex conjugates of $S_w(e^{j\omega})$ and $D_w(e^{j\omega})$, respectively. The quantities $D_w(e^{j\omega})$, $S_w^*(e^{j\omega})D_w(e^{j\omega})$, and $S_w(e^{j\omega})D_w^*(e^{j\omega})$ are unknown and can be approximated using the expectation operator. Since the noise is zero-mean, both $E\{S_w^*(e^{j\omega})D_w(e^{j\omega})\}$ and $E\{S_w(e^{j\omega})D_w^*(e^{j\omega})\}$ are zero and the above equation reduces to

$$\left|X_w(e^{j\omega})\right|^2 = \left|S_w(e^{j\omega})\right|^2 + E\left\{ \left|D_w(e^{j\omega})\right|^2 \right\}$$

Finally, $\left|S_w(e^{j\omega})\right|$ is estimated as the square root of

$$\left|S_w(e^{j\omega})\right|^2 = \left|X_w(e^{j\omega})\right|^2 - E\left\{ \left|D_w(e^{j\omega})\right|^2 \right\}$$

The signal may be full-wave or half-wave rectified to account for negative values of $\left|S_w(e^{j\omega})\right|^2$. As is commonly the case, this approach requires an estimate of the noise spectrum, which may be calculated during non-speech activity.

5.1.3 Minimum mean-square error short time spectral amplitude

More related to the current research are the minimum mean-square error short-time spectral amplitude (MMSE-STSA) estimators for speech enhancement. As its name suggests, this approach exploits the magnitude spectrum of stationary segments of speech, unlike the Wiener filtering approach which is based on the long-time power spectrum. Due to the relationship between Fourier expansion coefficients and the discrete Fourier transform, Ephraim and Malah proposed the estimation of the amplitude of the complex Fourier series coefficients [Eph1984] rather than of the magnitude of the Fourier transform. Given the degraded speech signal, let $X_k = R_k e^{j\theta_k}$, $S_k = A_k e^{j\alpha_k}$, and D_k denote the kth Fourier series coefficient of *x*(n), *s*(n), and d(n), respectively. The task is to estimate the amplitude A_k by minimizing the mean square error given by $e = \left(A_k - \hat{A}_k\right)^2$. We know that the optimal linear mean-square estimation of a quantity A_k given an observation *x*(n) is the conditional mean [Hay1996]

$$\hat{A}_k = E\{A_k \mid x(n)\}$$

Operating under the assumption of statistically independent Gaussian coefficients, we obtain

$$\hat{A}_k = E\{A_k \mid X_0, X_1, \ldots\} = E\{A_k \mid X_k\}$$

where the first equality follows from the Gaussian assumption and the second from the assumption of independence. The expansion of the expectation yields the amplitude estimator [Eph1984]

$$\hat{A}_k = \frac{\sqrt{\pi v_k}}{2\gamma_k} e^{-v_k/2}\left[(1+v_k)I_0\left(\frac{v_k}{2}\right)+v_k I_1\left(\frac{v_k}{2}\right)\right]R_k$$

The quantity v_k is defined as

$$v_k = \frac{\xi_k}{1+\xi_k}\gamma_k$$

where

$$\xi_k = \frac{\lambda_s(k)}{\lambda_d(k)} \text{ and } \gamma_k = \frac{R_k}{\lambda_d(k)}$$

are the *a priori* and *a posteriori* signal-to-noise ratios, respectively. Here, $\lambda_s(k) = E\left\{\ |S_k|^2\right\}$ and $\lambda_d(k) = E\left\{\ |D_k|^2\right\}$ are the kth spectral component variances. I_0 and I_1 are the modified Bessel functions of order zero and one. The modified Bessel function is given by the integral

$$I_n(z) = \frac{1}{2\pi}\int_0^{2\pi} \cos\beta n e^{z\cos\beta} d\beta$$

5.1.4 Minimum mean-square error log spectral amplitude

Significant improvements have been made to this estimator by Ephraim, Malah and others. One such improvement involves the use of a gain modifier to account for signal uncertainty. Another uses log spectral amplitude estimation [Eph1985] with the understanding that the mean square error estimation of the log spectra is more suited for speech processing.

The minimum mean-square error log spectral amplitude (MMSE-LSA) estimator presented by Ephraim and Malah [Eph1985] seeks to minimize $E\left\{ \left(\ln A_k - \ln A_k\right)^2 \right\}$. Following the previous line of reasoning we know then that

$$\hat{A}_k = e^{E\{\ln A_k | X_k\}}$$

Under the Gaussian assumption we have

$$E\{\ln A_k \mid X_k\} = \frac{d}{d\mu} \Phi_{(\ln A_k)|X_k}(\mu)\Big|_{\mu=0}$$

where $\Phi_{(\ln A_k)|X_k}(\mu) = E\left\{A_k^{\mu} \mid X_k\right\}$ is the moment generating function of $\ln A_k$ given X_k. The evaluation of the derivative leads to the solution

$$\hat{A}_k = \frac{\xi_k}{1+\xi}\left\{ e^{\frac{1}{2}\int_{v_k}^{\infty} \frac{e^{-t}}{t} dt} \right\} R_k$$

where v_k and ξ_k are as previously defined.

To account for signal uncertainty McAulay and Malpass proposed degrees of noise suppression dependent upon how much the degraded signal power exceeds a threshold which is based on the background noise power [McA1980]. The approach employs a two-state model indicating speech presence or absence. It was observed that some existing suppression filters perform well when speech is present, but not so well during speech absence. Therefore, a binary hypothesis model was developed:

H_0 : speech absent

H_1 : speech present

So then the amplitude estimator can be represented as

$$\hat{A} = E\{A \mid X_k, H_1\} P(H_1 \mid X_k) + E\{A \mid X_k, H_0\} P(H_0 \mid X_k)$$

where $P(H_j \mid X_k)$ is the probability that speech is in state j given X_k. With this, we know that $P(H_0 \mid X_k)$ is equal to zero. Thus we have

$$\hat{A} = E\{A \mid X_k, H_1\} P(H_1 \mid X_k)$$

For the MMSE-LSA estimator, this amplitude estimation becomes

$$\ln \hat{A} = E\{\ln A \mid X_k, H_1\} P(H_1 \mid X_k)$$

Furthermore, the *a posteriori* probability for speech presence is given by [McA1980]

$$P(H_1 \mid X_k) = \frac{e^{-\xi} I_0\left[2\sqrt{\xi\left(\frac{X_k}{\lambda_k}\right)}\right]}{1 + e^{-\xi} I_0\left[2\sqrt{\xi\left(\frac{X_k}{\lambda_k}\right)}\right]}$$

As indicated by Malah, Cox, and Accardi [Mal1999], this quantity applies a fixed probability to all frequencies of each frame of input speech. However, they proposed the use of a different probability estimate for each frequency bin. The motivation for this approach is that although a particular frame may contain speech, speech may not be present in each frequency bin of voiced speech. The average number of spectral components that do not contain speech varies with time. Therefore, Malah, et. al. first developed a fixed probability *q(l)* for each analysis frame *l*, and then one value *q_k(l)* for each frequency bin *k* in that frame. They began with the amplitude estimator derived by McAulay and Malpass with one modification

$$\ln \hat{A} = E\{\ln A \mid X_k, H_1^k\} P(H_1^k \mid X_k)$$

where H_1^k is defined as the hypothesis for speech presence in the *k*th frequency bin.

To determine the quantity $P(H_1^k \mid X_k)$, we first apply Bayes' rule to obtain

$$P(H_1^k \mid X_k) = \frac{\Lambda(k)}{1 + \Lambda(k)}$$

where

$$\Lambda(k) = \frac{P\left(H_1^k\right)p\left(X_k \mid H_1^k\right)}{P\left(H_0^k\right)p\left(X_k \mid H_0^k\right)} = \frac{1-q_k}{q_k} \cdot \frac{p\left(X_k \mid H_1^k\right)}{p\left(X_k \mid H_0^k\right)}$$

The quantity q_k denotes the *a priori* probability of speech absence in the k^{th} bin. The likelihood ratio $\frac{p\left(X_k \mid H_1^k\right)}{p\left(X_k \mid H_0^k\right)}$ is given by[Mal1999]

$$\frac{p\left(X_k \mid H_1^k\right)}{p\left(X_k \mid H_0^k\right)} = \left.\frac{e^{v_k}}{1+\xi_k}\right|_{\xi_k = \frac{\eta_k}{1-q_k}}$$

where

$$\eta_k = \frac{\lambda_s(k)}{\lambda_d(k)}$$

$$\lambda_s(k) = E\left\{ \ \left|S_k\right|^2 \right\}$$

$$\lambda_d(k) = E\left\{ \ \left|D_k\right|^2 \right\}$$

and v_k, γ_k, and ξ_k are as defined above.

The initial task then is to determine a fixed value *q(l)* for each frame *l* that contains speech. To do so a binary hypothesis is set up similar to the previous

$\mathcal{H}_P : \ \eta_k \geq \eta_{\min}$ speech present in k^{th} bin

$\mathcal{H}_A \ : \ \eta_k < \eta_{\min}$ speech absent in k^{th} bin

where $\eta_{\min}$ is a threshold of η_k set typically between 0.1 and 0.2 [Mal1999]. Note that η_k parameterizes the probability density function of γ_k given by

$$p(\gamma_k) = \frac{1}{1+\eta_k} e^{-\frac{\gamma_k}{1+\eta_k}} \ ; \ \gamma_k \geq 0$$

Then given that the likelihood ratio $\frac{p(\gamma_k \mid \eta_k < \eta_{\min})}{p(\gamma_k \mid \eta_k = \eta_{\min})}$ is monotone, we may use the following hypothesis

$$\mathcal{H}_P : \gamma_k > \gamma_{TH} \text{ speech present in } k^{th} \text{ bin}$$

$$\mathcal{H}_A : \gamma_k < \gamma_{TH} \text{ speech absent in } k^{th} \text{ bin}$$

where γ_{TH} is some preset threshold, typically 0.8 [Mal1999].

If we define α_0 as the probability of rejecting $\mathcal{H}_P$ when true, then using the equation for the pdf of γ_k, we have

$$\gamma_{TH} = (1+\eta_{\min})\log\left(\frac{1}{1-\alpha_0}\right)$$

Now an estimate of the probability for speech absence for a given frame *l* would be

$$q(l) = \alpha_q q(l-1) + (1-\alpha_q)\frac{N_q(l)}{M}$$

where *M* is the number of positive frequency bins in frame *l* and $N_q(l)$ is the number of bins for which the test rejects the hypothesis $\mathcal{H}_P$. The quantity α_q is a weighting factor controlling the tradeoff between noise suppression and speech distortion. Typical values for α_q are in the range between 0.91 for less distortion and 0.98 for more aggressive suppression. This smoothing is performed in frames determined via a voice activity detector (VAD) to contain speech.

Finally, if we define the following parameter

$$I_k(l) = \begin{cases} 1 & if \quad \mathcal{H}_P \quad is \quad rejected \\ 0 & if \quad \mathcal{H}_P \quad is \quad accepted \end{cases},$$

then an estimate for the probability $q_k(l)$ for speech absence in a given frequency bin *k* of a given frame *l* is

$$q_k(l) = \alpha_q q_k(l-1) + (1-\alpha_q) I_k(l)$$

where possible threshold values are as given above. The value of *q* for each frame is computed as the arithmetic mean over *k* of the $q_k(l)$ and will change with each frame of input.

5.2 AUTOMATIC RECOGNITION OF NOISY SPEECH

Of consideration when attempting to suppress noise is the source of that noise. Although many approaches attempt to suppress noise in general, others focus on a specific type of noise, such as white noise [Lee1994]] or impulse noise [Pot2001], in an attempt to simplify the problem. Determining which approach to take when attempting to enhance noisy speech is the next step.

Through the use of an enhancement front-end, the procedure presented in this document suppresses noise with no modification of the existing speech recognizer. Experiments show that an enhancement front-end can significantly reduce the effects that speech degradation has on recognition. The problem's merit is seen in the results of experiments which monitored the improvement of noisy speech recognition when enhancement was applied [Yif2000]. These experiments show that enhancement based on spectral subtraction or based on minimum mean-square error techniques can improve recognition of noisy speech by as much as nearly twenty percent.

5.2.1 Necessity for enhancement of speech prior to recognition

Many electronic devices such as cell phones, virtual assistants, and navigation and entertainment systems offer voice-activation for hands-free operation. And many of these devices employ speech recognition technology to provide a simple and natural interface for the user. Because the devices are used in a variety of environments, noise is a common artifact in the speech signal. As has been seen in Figure 20, noise in the speech signal can cause significant distortion. For this reason, some type of enhancement prior to recognition is necessary.

5.2.2 Problems encountered in the recognition of enhanced speech

Speech enhancement algorithms are typically aimed at improving the quality of speech for the human listener. This goal may not suit speech recognition needs since the enhanced speech waveform, and hence its parameters, may differ significantly from the clean signal. As clearly seen from Figure 21, the enhancement of the speech waveform does not adequately reduce the distortion in the time-domain signal. Figure 22 shows the magnitude frequency response of a 30 ms segment of voiced speech from the same speech signal. From this illustration we can see that the spectral characteristics of the speech signal are also distorted following enhancement despite the auditory clarity of the speech. This distortion results in inaccurate calculation of the spectral parameters necessary for recognition. Hence, recognition accuracy is negatively affected.

In this research we adapt a minimum mean-square error log-spectral amplitude enhancement algorithm, which was developed to improve human perception, to render it more functional for automatic recognition systems. The result is improved recognition accuracy without modification of the existing recognition system and without the need for system retraining. Details of this phase of the research are presented in Chapter 6.

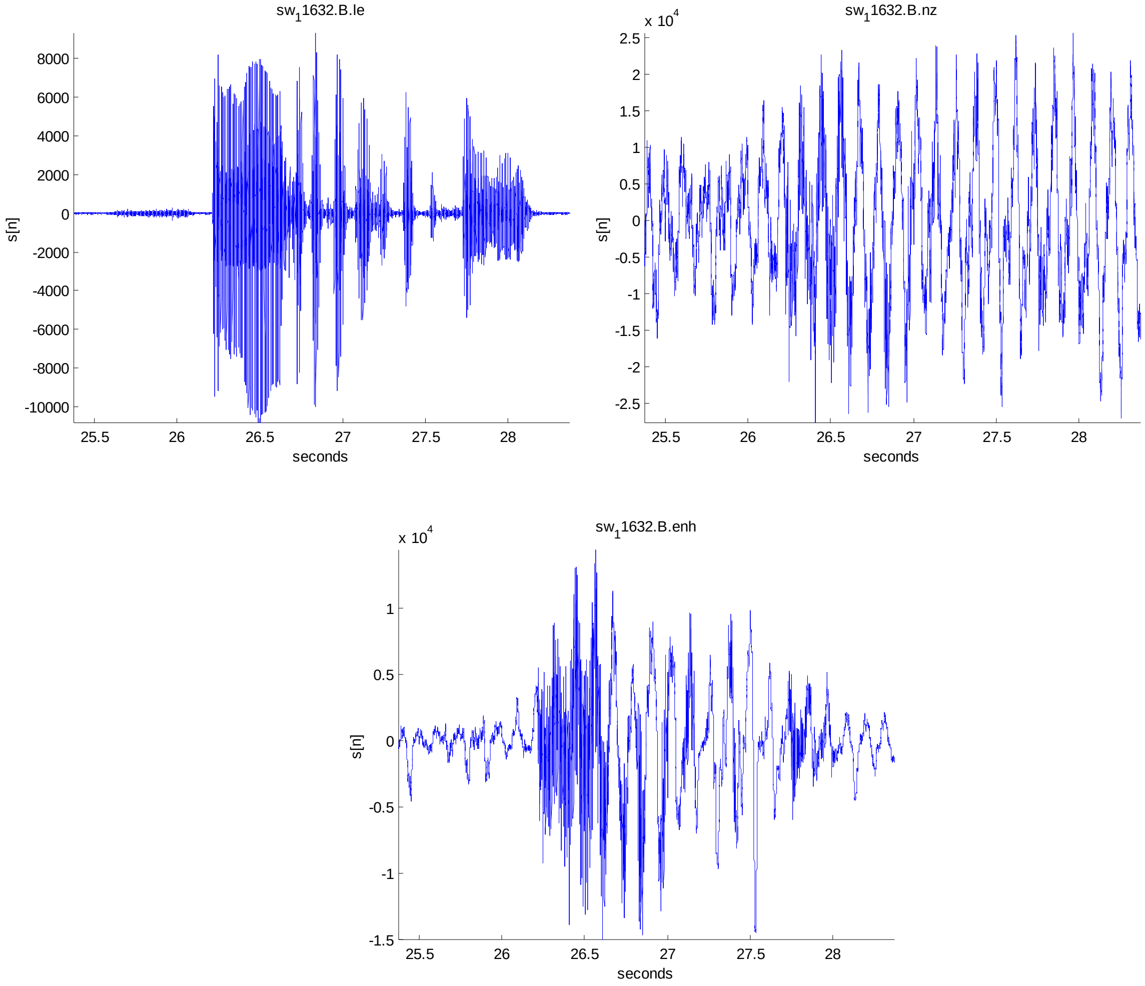


Figure 21. Enhancement of Noisy Speech: Time-Domain Comparison
Top Left: Original (clean) waveform. Top Right: Noisy waveform. Bottom: Enhanced waveform.

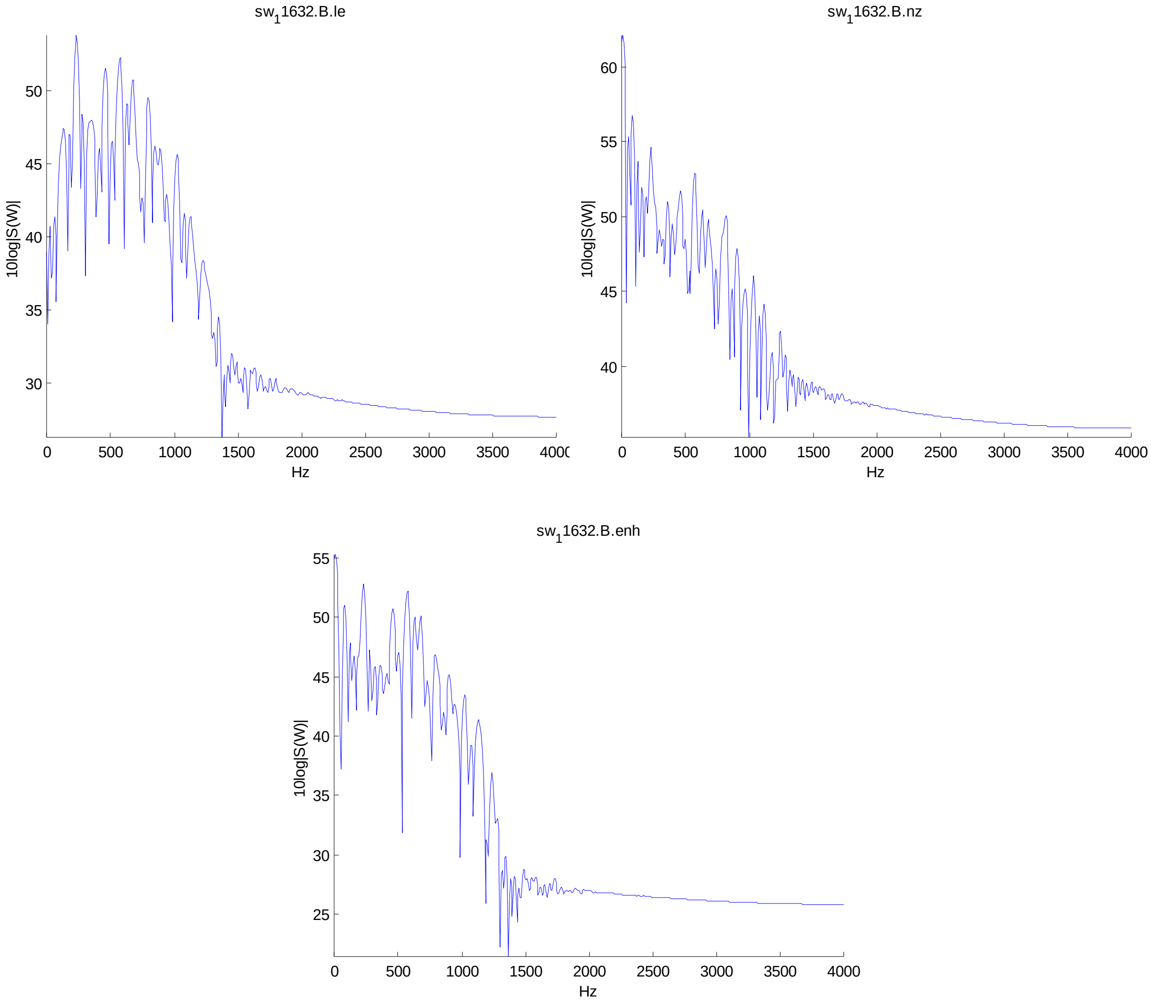


Figure 22. Enhancement of Noisy Speech: Frequency-Domain Comparison
Top Left: Original (clean) waveform. Top Right: Noisy waveform. Bottom: Enhanced waveform.
(log magnitude, 30ms frame of voiced speech)

5.2.3 Related research on the recognition of noisy speech

Many techniques have been developed to enhance noisy speech for recognition. These may be broadly categorized as those that adapt the speech recognition models to the noise conditions, those that extract noise-robust features from the signal, and those that enhance the signal prior to feature extraction.

Adapting speech models to noise conditions generally involves retraining the speech recognition system using noisy data. This requires the knowledge and availability of the noise source present in the speech to be recognized. Noise robust feature extraction aims to improve the recognition of noisy speech by improving the recognizer itself [Lu2009][Hun2009]. The goal is to determine a representation of the signal that is not significantly altered by the presence of noise. Some approaches based on this principle include perceptual linear prediction (PLP) [Her1988] and mel-frequency cepstral coefficients (MFCC) [Tia2003], which attempt to imitate the human auditory system.

The third category of enhancement utilizes a front-end system that suppresses the noise in the speech signal prior to feature extraction. The enhancement front-end may consist of one of a variety of enhancement techniques. Kalman filtering is a common approach to enhancement. As noise contamination occurs in the time domain, the Kalman filter works in the time domain to remove the noise [Lee2000] [Pop1998]. The filter succeeds in increasing the signal-to-noise ratio; however, it also succeeds in increasing recognition error [Ope1993].

Spectral subtraction techniques, which are also popular for this type of speech enhancement [Mwe1996][Xu2000][Soo2000], may generate musical noise and require an accurate measurement of noise level that is often difficult to obtain. However, spectral subtraction has been shown to be effective for the purpose of recognition [Yif2000].

The research presented in this document focuses on the use of spectral subtraction for the enhancement of noisy speech with very low signal-to-noise ratio. Spectral subtraction works on the assumption that the magnitude of the noise spectrum is lower than that of the speech spectrum. This approach is adequate for signals of high signal-to-noise ratio. However, for low SNR signals, the noise may indeed overpower the speech. Therefore, the improvements due to spectral subtraction diminish as signal-to-noise ratio decreases [Eva2006]. A method of non-linear spectral subtraction was developed that performs a non-linear compression of spectral samples [Loc1992]. The method applies less perturbation to peak spectral components since the high energy of these components masks the noise, and greater perturbation to valley components. In this way, the musical noise present in the spectral subtraction method is suppressed [Por2002]. This approach proved beneficial to human perception as well as to automatic speech recognition for signals of high to moderate signal-to-noise ratio.

For improved recognition of speech with low signal-to-noise ratio, Jounghoon Beh and Hanseok Ko proposed a method to distinguish speech-dominant segments of the signal from noise-dominant segments [Beh2003]. If the signal-to-noise ratio for a particular segment falls below a given threshold, spectral subtraction is applied as for a non-speech frame. Recognition accuracy using this method was tested on signals with signal-to-noise ratios ranging from 20 dB to -5 dB. The results of the modification on speech recognition accuracy are compared to those for spectral subtraction and non-linear spectral subtraction in Table 15 for a signal-to-noise ratio of -5 dB.

Table 15. Recognition of Noisy Speech (Beh)

Input Data Format	Error
Clean Waveform	0.98
Noisy Waveform (-5 dB)	90.75
Spectral Subtraction Enhancement	87.59
Non-linear Spectral Subtraction Enhancement	87.95
Beh Proposed Enhancement	81.81

Roberto Gemello, Franco Mana and Renato De Mori applied a soft-decision gain modification to the gain function of the spectral subtraction algorithm that would multiply the noisy acoustic parameters [Gem2004]. The method was applied using four different types of background noise with varying levels of signal-to-noise ratio and the recognition accuracy compared to that of other common techniques. The results averaged for the four noise types are shown in Table 16 for signal-to-noise ratio of -5 dB.

Table 16. Recognition of Noisy Speech (Gemello)

Input Data Format	Error
Clean Waveform	0.9
Noisy Waveform (-5 dB)	85.2
Wiener Filter Enhancement	69.4
Spectral Subtraction Enhancement	65.2
Gemello Proposed Enhancement	64.5

6.0 SPEECH ENHANCEMENT CUSTOMIZED FOR RECOGNITION

The second objective of this research is to develop a speech enhancement front-end customized for automatic speech recognition. Experiments which monitored recognition improvement due to the enhancement of noisy speech have shown that enhancement based on spectral subtraction or minimum mean-square error techniques can improve recognition accuracy by as much as nearly twenty percent [Yif2000]. Our research utilizes one such enhancement technique, the MMSE-LSA enhancement algorithm, which is described in detail in Section 5.1.4. However, as seen from Figure 21 and Figure 22 of Section 5.2.2, the enhancement of the speech waveform does not adequately reduce the visual distortion in the speech signal. The goal therefore is to determine what effects the enhancement may have on the accuracy of the recognition of the noisy speech and, based on the results, develop an enhancement algorithm specific to the needs of a speech recognition system.

6.1 EXPERIMENT SETUP

Recognition for this phase of research is performed using the Byblos large vocabulary conversational speech recognition system developed by BBN. The recognition system is briefly explained in Section 2.4.

6.1.1 Recognition training and testing

Preliminary experiments for the recognition of noisy speech are performed as follows. Training for all experiments is executed on clean, unprocessed speech. The training data consists of over ninety-nine hours of speech from 750 speakers from the Switchboard Corpus. The window duration for training data is 25 milliseconds with a frame rate of 10 milliseconds.

The test data consists of approximately three hours of speech from 80 speakers from the Switchboard and Call Home English corpora. The window duration is set at 25 milliseconds with a frame rate of 10 milliseconds.

Noise from the cockpit of a blackhawk helicopter is added to the speech signal, resulting in a noisy speech signal with a signal-to-noise ratio of -5.6 dB. The noisy speech is later enhanced using the MMSE-LSA speech enhancement algorithm described in detail in Section 5.1.4.

6.1.2 Baseline results

Testing is performed on clean speech, on noisy speech, and on noisy speech which is enhanced prior to recognition. Comparison of the results, which are shown in Table 17, reveal a relative difference of 28% between the error rate of the recognition of the clean speech and that of the noisy speech. This difference is reduced to 22.2% when the noisy speech is enhanced prior to recognition using the MMSE-LSA enhancement described in Section 5.1.4.

Table 17. Baseline Enhanced Speech Recognition Results

Input Data Format	Error
Original Waveform	72.1
Noisy Speech Waveform	92.3
Enhanced Speech Waveform	89.0

The primary purpose of this phase of research is to develop an enhancement front-end to improve the accuracy of the recognition of noisy speech. The implementation of the enhancement is detailed in Section 6.2 and a summary of the recognition improvements are presented in Section 6.3.

6.2 IMPLEMENTATION

In this phase of the research, recognition is performed on speech to which noise is added, resulting in a signal-to-noise ratio of -5.6 dB. This extremely low signal-to-noise ratio causes significant distortion in the speech signal. Hence, the parameters extracted from the noisy speech, such as energy and LP coefficients, are considerably different than those extracted from the original (clean) speech. The signal energy, for example, computed for each frame of noisy speech varies considerably from those computed from the clean speech. As shown in Table 18, the energy for a particular frame of noisy speech is 28,000 times that for the clean speech. This difference will negatively affect recognition accuracy since energy, as indicated in Section 2.3.2, is used in recognition to distinguish voiced speech segments from unvoiced and silence.

Table 18. Frame Energy Comparison: Clean and Noisy Speech

Data Format	Frame Energy
Original Waveform	489.200
Noisy Speech Waveform	14103310.650

Figure 23 shows a comparison of the LP coefficient values for a frame of clean speech and a frame of noisy speech. The relatively large differences between the coefficient values for clean and noisy speech will result in reduced recognition accuracy since the LP coefficients are used to determine spectral characteristics used for recognition as indicated in Section 2.3.2.

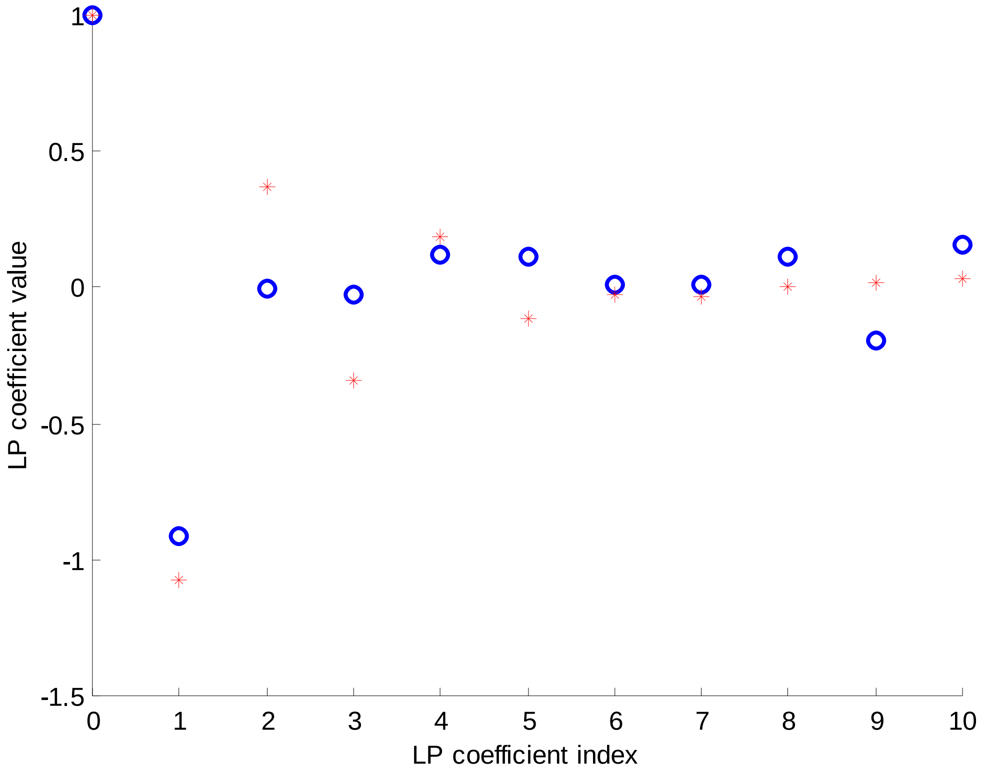


Figure 23. LP coefficient values for 20ms frame of clean and noisy speech
Circle: Original (clean) waveform. Star: Noisy waveform.

A comparison of the recognition error from the clean speech waveform and the noisy speech waveform reveals that the noise in the speech signal causes a 28% relative increase in the recognition error. A speech enhancement front-end is developed to reduce the effects of the noise on recognition accuracy.

6.2.1 Enhancement of noisy speech

To improve the recognition of the noisy speech we begin with an MMSE-LSA enhancement algorithm designed by Ephraim and Malah [Eph1985] and described in detail in Section 5.1.4. The data presented in Table 19 indicate the recognition error rates for clean, noisy, and enhanced speech. From the table we see that recognition error increases from 72.1% to 92.3% when noise is added to the speech signal. The MMSE-LSA enhancement reduces the error by 3.4 percentage points, to 89.0%.

Table 19. Recognition of Enhanced Speech

Expt #	Input Data Format	Error
7101	Original Waveform	72.1
7102	Noisy Speech Waveform	92.3
7103	Enhanced Speech Waveform	89.0

The unsatisfactory results achieved by MMSE-LSA enhancement prior to recognition indicate that, although the enhancement improves the speech from a human perception standpoint, the automatic recognition of the speech is yet greatly affected. Therefore, a more effective enhancement front-end is needed to reduce the effects that the noise has on automatic speech recognition accuracy. Two approaches are taken. One approach, detailed in Section 6.2.2, is a modification of the existing MMSE-LSA enhancement through the mapping of parameters from enhanced speech to those of clean speech. The other approach, which is explained in Section 6.2.3, couples the MMSE-LSA algorithm with MELP coding to further enhance the noisy speech prior to recognition.

6.2.2 Spectral mapping of noisy speech

The addition of noise to the clean speech signal results in distortion of the frame energy as illustrated previously in Table 18. An extreme change in the value of the frame energy may result in voicing misclassification. Furthermore, as seen previously in Figure 22 spectral distortion is still significant following enhancement. This spectral distortion is a direct result of the deviations of the LP coefficients that result from noise, as shown in Figure 23. We attempt to reduce the recognition error due to energy and spectral distortions through mappings of 7328 frames of the magnitude spectrum and of the average power. The mappings are determined independently for each of eight frequency bins: {[0 250], [250 500], [500 750], [750 1000], [1000 1250], [1250 2000], [2000 3000], [3000 4000]} Hz. First through third degree polynomial mappings are determined from each of the enhanced speech parameters to the corresponding clean speech parameter using the least squares method as detailed in Appendix A. This results in a total of six distinct mappings, three for each of the two parameters, with each mapping consisting of eight equations, one for each frequency bin. Each of the mappings in turn is incorporated into the enhancement algorithm. Enhancement is then performed on the noisy speech signals and the recognition results compared.

Of the six mappings, the greatest improvement in recognition accuracy occurs for a first degree mapping of the magnitude spectrum. This mapping results in an additional 0.3 percentage point reduction in the error rate, from 89.0% error for MMSE-LSA enhancement to 88.7% error rate for the modified enhancement. These results are depicted in Table 20.

Table 20. Recognition of Enhanced Speech w/ Mapping

Expt #	Input Data Format	Error
7101	Original Waveform	72.1
7102	Noisy Speech Waveform	92.3
7103	Enhanced Speech Waveform	89.0
7104	Enhanced Speech Waveform w/ Mapping	88.7

6.2.3 Coding of noisy speech

A second approach to improve the recognition accuracy of noisy speech involves the incorporation of MELP coding, which was detailed in Section 3.1.4. In Table 21 we summarize the results from the recognition of coded noisy speech. If the noisy speech is coded prior to recognition and recognition is performed on the bitstream parameters as described in Section 3.3.1, the error rate decreases from 92.3% for noisy speech to 91.7% for coded noisy speech. This small improvement is likely due to the perturbations of the spectral features during implementation of the MELP coder.

Table 21. Recognition of Noisy Speech from MELP Bitstream

Expt #	Input Data Format	Error
7101	Original Waveform	72.1
7102	Noisy Speech Waveform	92.3
7104	Enhanced Speech Waveform w/ Mapping	88.7
8102	MELP Bitstream: Noisy Speech Waveform	91.7

The improvement in recognition accuracy due to coding is not as significant as the improvement due to the modified enhancement with spectral mapping. However, greater improvement is achieved through the coupling of MMSE-LSA enhancement and MELP coding. That is, the noisy speech is first enhanced and then coded, with recognition again being performed directly from the bitstream. The recognition error decreases from 88.7% for enhancement with spectral mapping to 88.1% for the enhancement followed by coding. These results are displayed in Table 22.

Table 22. Recognition of Enhanced Speech from MELP Bitstream

Expt #	Input Data Format	Error
7101	Original Waveform	72.1
7102	Noisy Speech Waveform	92.3
7104	Enhanced Speech Waveform w/ Mapping	88.7
8103	MELP Bitstream: Enhanced Speech Waveform	88.1

6.3 SUMMARY OF RESULTS

The contributions of this portion of research concern the recognition of noisy speech. As stated previously, the large vocabulary and continuous speech contribute to high recognition error. Compounding the difficulty of accurate recognition is the presence of noise in the speech signal. The speech under consideration in this research has a signal-to-noise ratio of -5.6 dB. Although spectral subtraction works well to improve human perception of the speech and to improve speech recognition accuracy for high signal-to-noise ratio speech, it is not a sufficient method to improve the automatic recognition of speech having very low signal-to-noise ratio.

The lack of greater improvement in recognition accuracy at such a low signal-to-noise ratio, despite great audible improvement, may be seen in the following two examples. First, Table 23 shows the frame energy for a single frame of clean speech, noisy speech, and enhanced speech.

Table 23. Frame Energy Comparison: Clean, Noisy, and Enhanced Speech

Input Data Format	Frame Energy
Original Waveform	489.200
Noisy Speech Waveform	14103310.650
Enhanced Speech Waveform	140793.119

From the table we note that, although the enhancement lowers the frame energy, the frame energy for the enhanced speech is still 288 times that of the clean speech.

As a second example, for the same frame of speech, the LP coefficients are plotted in Figure 24 for the clean, noisy and enhanced speech signals, represented by circles, stars, and squares, respectively. It is clear from the graph that the values of the LP coefficients are not noticeably affected by enhancement. That is, the coefficients for the enhanced speech signal are nearly identical to those for the noisy speech, yet very different from those of the clean speech. The consequence is that the spectrum of the enhanced speech is quite distorted when compared to that of the clean speech, as seen previously in Figure 22.

Attempts were made to compensate for the spectral magnitude errors that persist following spectral subtraction. These modifications yielded modest gains in recognition accuracy, reducing the error rate from 89.0% for speech enhanced with spectral subtraction to 88.1% for speech enhanced with modifications and coded, as explained in Section 6.2.3. This decrease in error is equivalent to the accurate recognition of an additional 540 words from the

12,863 words spoken. This difference is comparable to that obtained by Gemello when employing a soft-decision gain modification.

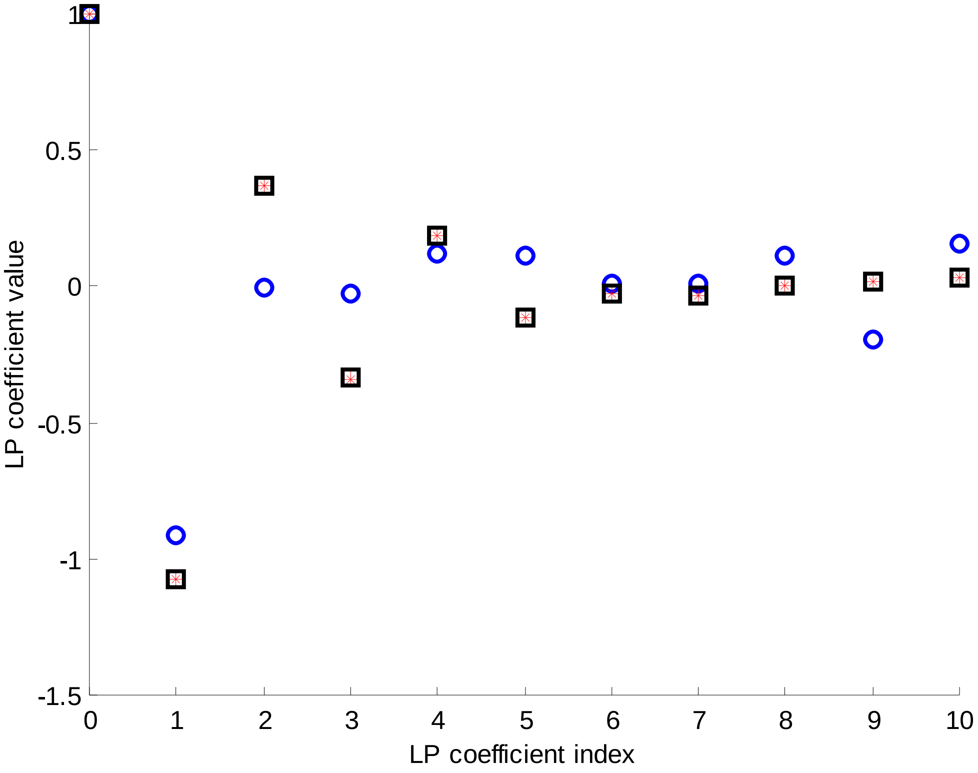


Figure 24. LP coefficient values for 20ms frame of clean, noisy, and enhanced speech
Circle: Original (clean) waveform. Star: Noisy waveform. Square: Enhanced Waveform

7.0 SUMMARY OF RESULTS

Prior to recognition, a speech signal may be coded in one form or another. The Federal Standard 2400 bits per second MELP coder extracts many of the same feature parameters that are used by speech recognizers. By eliminating the superfluous computations, we have succeeded in recognizing encoded speech under mismatched conditions directly from the analysis bitstream, without the need for reconstruction of the speech signal. Although distortion caused by coding and quantization exists, distortion caused by reconstruction of the signal is eliminated and the recognition results exceed those obtained from the reconstructed speech. Through recognition directly from the bitstream we avoid considerable distortion and thereby improve the overall efficiency and accuracy of the recognition system. The techniques presented in this document are comparable to or exceed results of similar research on higher bit-rate coders and MELP coding under like conditions.

A relative difference of 6.7% exists between the recognition error rate of uncoded speech and that of reconstructed speech. The relative difference between the recognition error rate for uncoded speech and that of encoded speech recognized directly from the MELP bitstream is 3.5%. This 3.2 percentage point improvement is equivalent to the accurate recognition of an additional 334 words from the 12,863 words spoken.

Like coding, additive noise also greatly affects the accuracy of automatic speech recognition. The performance of a speech recognizer on degraded speech is significantly poorer than its performance on noise-free speech. By enhancing speech prior to recognition, the increase in recognition error due to noise will be less substantial. In this document we offset the distortions due to noise through appropriate modification of an existing spectral subtraction technique called MMSE-LSA.

A relative difference of 28% exists between the recognition error rate of clean speech and that of the noisy speech. This relative difference was cut to 22.2% with the modified speech enhancement front-end. This 5.8 percentage point improvement is equivalent to the accurate recognition of an additional 540 words from the 12,863 words spoken. Compared to the original spectral subtraction technique, the recognition error was reduced by 0.9 percentage points. This decrease is equivalent to the accurate recognition of an additional 130 words.

8.0 FURTHER RESEARCH

8.1 RECOGNITION OF CODED SPEECH

Regarding the recognition of coded speech, satisfactory results are gained when recognition is performed directly from the encoded bitstream. The results of the recognition for several coders has been shown to exceed the results of recognition from the reconstructed waveform, as presented in Section 4.3. However, the calculations required and the results obtained were dependent upon the model and bitrate of the chosen speech coder. The variations in recognition improvement and computational methods can be eliminated by avoiding bitstream recognition altogether through the use of distributed recognition.

Distributed Recognition: Accurate recognition of encoded speech directly from the bitstream relies on the accurate estimation and quantization of parameters by the encoder. This estimation and quantization of the parameters will result in some error. Furthermore, the parameters used for coding, although similar to those used for recognition, will require some manipulation for use in a recognition system. It is therefore most advantageous if the recognition parameters are extracted simultaneously with the coding parameters from the uncoded speech. Considering the relative low cost of distributed recognition, it is foreseeable that embedded speech recognition devices may be standard in many electronic communications devices. In this way, recognition may be performed directly from the uncoded speech. This approach, however, would require the standardization of ASR systems at the remote end.

8.2 RECOGNITION OF NOISY SPEECH

Regarding the recognition of noisy speech, spectral subtraction, operating on the assumption that the magnitude of the noise spectrum is lower than that of the speech spectrum, works well on speech of moderate to high signal-to-noise ratio. The results are not as favorable for speech of low signal-to-noise ratio. It has been observed that, in addition to spectral magnitude error, phase and cross-term errors exist in the spectral subtraction estimation [Eva2006].

Phase Distortion: Although the insignificance of phase in speech recognition accuracy is commonly accepted, results from human perception experiments have indicated that phase may significantly contribute to speech intelligibility [Als2005]. Furthermore, slight improvements have been made in the recognition of noisy speech through the incorporation of phase spectrum into the acoustic features [Sch2001]. The impact of phase distortion on the accuracy of ASR systems is yet debatable, and therefore worth further examination.

Cross-Term Estimation: Although the contribution of phase error on recognition accuracy is often overlooked as negligible, the errors due to cross-term estimation in the approximation of the clean signal spectrum have been shown to be significant [Eva2006]. Considering these errors in the adjustment of the recognition parameters may significantly reduce recognition error.

APPENDIX A

POLYNOMIAL MAPPING: LEAST SQUARES METHOD

Given two $m \times 1$ vectors $\{x_1, x_2, \ldots, x_m\}$ and $\{y_1, y_2, \ldots, y_m\}$, we desire to find an nth degree polynomial mapping p such that $p(x_j) \approx y_j$ in the least squares sense for each $j = 1, \ldots, m$. Given the Vandermonde matrix

$$V = \begin{bmatrix} x_1^n & x_1^{n-1} & \cdots & x_1^2 & x_1 & 1 \\ x_2^n & x_2^{n-1} & \cdots & x_2^2 & x_2 & 1 \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ x_m^n & x_m^{n-1} & \cdots & x_m^2 & x_m & 1 \end{bmatrix}$$

we may compute the polynomial coefficients as $p = V^{-1}y$. However, because V is singular, we compute the QR-factorization such that $V = QR$, where R is a non-singular, upper triangular matrix and Q is unitary so that $Q^T Q = I_{n+1}$, where Q^T and I_{n+1} denote the transpose of Q and the $(n+1)\times(n+1)$ identity matrix, respectively. We may then compute the coefficients as $p = R^{-1}Q^T y$.

The steps to compute such a mapping are as follows:

Step 1: Compute the Vandermonde matrix V as shown above.

Step 2: Calculate the QR-factorization of V as follows:

i) Let $\{u_1, u_2, \ldots, u_{n+1}\}$ denote the columns of *V*, which are linearly independent.

ii) Using the Gram-Schmidt process, as detailed in Appendix B, transform the linearly independent set of vectors $\{u_1, u_2, \ldots, u_{n+1}\}$ into an orthonormal set $\{w_1, w_2, \ldots, w_{n+1}\}$.

iii) For each $i, j = 1, \ldots, n+1$, compute $r_{ji} = u_i \bullet w_j$

Let $Q = [w_1 \quad w_2 \quad \ldots \quad w_{n+1}]$ and $R = [r_{ij}]$. Then $V = QR$. Furthermore, Q is a unitary $m \times (n+1)$ matrix and R is a non-singular, upper triangular $(n+1) \times (n+1)$ matrix.

Step 3: Compute the polynomial coefficients as

$$p = R^{-1} Q^T y$$

The coefficients are given in descending order of degree,

$$p = [c_n \quad c_{n-1} \quad \ldots \quad c_1 \quad c_0]^T$$

so that

$$c_n x_j^n + c_{n-1} x_j^{n-1} + \cdots + c_2 x_j^2 + c_1 x_j + c_0 \approx y_j$$

for each $j = 1, \ldots, m$

APPENDIX B

GRAM-SCHMIDT PROCESS: ORTHONORMALIZATION OF A LINEARLY INDEPENDENT SET OF VECTORS

We desire to transform a linearly independent set of vectors $\{u_1, u_2, \ldots, u_m\}$into an orthonormal set of vectors $\{w_1, w_2, \ldots, w_m\}$. We begin by transforming $\{u_1, u_2, \ldots, u_m\}$ to an orthogonal set as follows:

1. Let $v_1 = u_1$.
2. Let $v_j = u_j - \sum_{k=1}^{j-1} \frac{v_k \bullet u_j}{\|v_k\|} \cdot v_k$ for $j = 2, \ldots, m$, where $\|v_k\| = v_k \bullet v_k$ denotes the vector norm.

The resulting set of vectors $\{v_1, v_2, \ldots, v_m\}$ is orthogonal. An orthonormal set of vectors can then be obtained as

3. $w_j = \frac{v_j}{\|v_j\|}$ for $j = 2, \ldots, m$.

BIBLIOGRAPHY

[Ale2008] Alexander J.M., and K.R. Kluender, “Spectral Tilt Change In Stop Consonant Perception”, The Journal of the Acoustical Society of America (January 2008), pp. 386-396.

[Als2005] Alsteris, Leigh D and Kuldip K. Paliwal, “Further Intelligibility Results From Human Listening Tests Using The Short-Time Phase Spectrum”, Speech Communication (2006), pp. 727-736.

[Beh2003] Beh, Jounghoon, and Hanseok Ko, “A Novel Spectral Subtraction Scheme For Robust Speech Recognition: Spectral Subtraction Using Spectral Harmonics Of Speech”, IEEE International Conference on Acoustics, Speech, and Signal Processing (2003), pp. I-648 - I-651.

[Bes2001] Besacier, C., et. al., "The Effect Of Speech And Audio Compression On Speech Recognition Performance", IEEE Fourth Workshop on Multimedia Signal Processing (2001), pp. 301-306.

[Bol1979] Boll, S.F., "Suppression Of Acoustic Noise In Speech Using Spectral Subtraction", IEEE Transactions on Acoustics, Speech , and Signal Processing (1979), pp. 113-120.

[Cha1974] Chandra, S. and W.C. Lin, "Experimental Comparison Between Stationary And Non-Stationary Formulations Of Linear Prediction Applied To Speech", IEEE Transactions on Acoustics, Speech , and Signal Processing (1974), pp. 403-415.

[Cho2000] Choi, Seung Ho, Hong Kook Kim, and Richard V. Cox, “Speech Recognition Using Quantized LSP Parameters and Their Transformations in Digital Communication”, Speech Communication (2000), pp. 223-233.

[Chu2003] Chu, Wai C., Speech Coding Algorithms: Foundation and Evolution of Standardized Coders (New Jersey: John Wiley & Sons, 2003).

[Cia1982] Ciaramella, A., et. al., "Implementation Of A Linear Prediction Coefficients-10 Vocoder", IEEE International Conference on Communications, Vol. 2 (1982), pp. 4G.4.1-4G.4.5.

[Dav2002] Davis, Gillian M. (ed.), Noise Reduction in Speech Applications (New York: CRC Press, 2002).

[Dig1999] Digalakis, V.V., L.G. Neumeyer, M. Perakakis, “Quantization of cepstral parameters for speech recognition over the World Wide Web”, IEEE Journal on Selected Areas in Communications, Volume 17, Issue 1 (Jan. 1999), pp. 82-90.

[Eph1984] Ephraim, Y. and D. Malah, "Speech Enhancement Using A Minimum Mean-Square Error Short-Time Spectral Amplitude Estimator", IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 32 (1984), pp. 1109-1121.

[Eph1985] Ephraim, Y. and D. Malah, "Speech Enhancement Using A Minimum Mean-Square Error Log-Spectral Amplitude Estimator", IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 33 (1985), pp. 443-445.

[Eul1994] Euler, S. and J. Zinke, "The Influence Of Speech Coding Algorithms On Automatic Speech Recognition", International Conference on Acoustics, Speech, and Signal Processing (1994), pp. 621-624.

[Eva2006] Evans, N.W.D., J.S.D. Mason, W.M. Liu, and B. Fauve, “An Assessment On The Fundamental Limitations Of Spectral Subtraction”, IEEE International Conference on Acoustics, Speech and Signal Processing (2006), pp. I-145 – I-148.

[Gal1999] Gallardo-Antolín, A., F. Diaz-de-Maria, and F. Valverde-Albacete, "Avoiding Distortions Due To Speech Coding And Transmission Errors In GSM ASR Tasks", International Conference on Acoustics, Speech, and Signal Processing (1999), pp. 277-230.

[Gal2005] Gallardo-Antolín, A., C. Pelaez-Moreno, and F. Diaz-de-Maria, “Recognizing GSM Digital Speech”, IEEE Transactions on Speech and Audio Processing (2005), pp. 1186-1205.

[Gem2004] Gemello, R., F. Mana, and R. De Mori, “A Modified Ephraim-Malah Noise Suppression Rule For Automatic Speech Recognition”, IEEE International Conference on Acoustics, Speech, and Signal Processing (2004), pp. 957-960.

[Gol1999] Gold, Ben, and Nelson Morgan, Speech and Audio Signal Processing (New York: John Wiley & Sons, 1999).

[Hay1996] Hayes, M.H., Statistical Digital Signal Processing and Modeling (New York: John Wiley & Sons, 1996).

[Her1988] Hermansky, H. and J.C. Junqua, "Optimization Of Perceptually-Based ASR Front-End", International Conference on Acoustics, Speech, and Signal Processing (1988), pp. 219-222.

[Hue1998] Huerta, Juan M., and Richard M. Stern, “Speech Recognition from GSM Codec Parameters”, Proceedings of the International Conference on Spoken Language Processing (1998), pp. 1463-1466.

[Hun2009] Hung, Jeih-weih, and Wen-hsiang Tu, “Incorporating Codebook and Utterance Information in Cepstral Statistics Normalization Techniques for Robust Speech Recognition in Additive Noise Environments”, IEEE Signal Processing Letters, Volume 16, Issue 6 (June 2009), pp. 473-476.

[Kay1988] Kay, Stephen M., Modern Spectral Estimation: Theory & Application (New Jersey: Prentice Hall, 1988).

[Kel1994] Keller, E (ed), Fundamentals of Speech Synthesis and Speech Recognition (New York: John Wiley & Sons, 1994).

[Kim2001] Kim, Hong Kook, and Richard V. Cox, “A Bitstream-Based Front-End for Wireless Speech Recognition on IS-136 Communications System”, IEEE Transactions on Speech and Audio Processing (2001), pp. 558-568.

[Koh1997] Kohler, M.A., "A Comparison Of The New 2400 BPS MELP Federal Standard With Other Standard Coders", International Conference on Acoustics, Speech, and Signal Processing (1997), pp. 1587-1590.

[Lan1991] Langi, A. and W. Kisner, "Code-Excited Linear Predictive Speech Processing For Digital Transmission And Storage", IEEE Wescanex (1991), pp. 205-209.

[Lee1994] Lee, L.M., J.K. Chen, and H.C. Wang, "Nonlinear Cepstral Equalisation Method For Noisy Speech Recognition", IEE Proceedings - Vision, Image and Signal Processing, Vol. 141, No. 6 (December, 1994), pp. 397-402.

[Lee1996] Lee, C.H., F.K. Soong, and K.K. Paliwal (ed.), Automatic Speech and Speaker Recognition (Boston: Kluwer Academic Publishers, 1996).

[Lee2000] Lee, K.Y. and J. Souhwan, "Time-Domain Approach Using Multiple Kalman Filters And EM Algorithm To Speech Enhancement With Nonstationary Noise", IEEE Transactions (May, 2000), pp. 282-291.

[Lil1996] Lilly, B.T. and K.K. Paliwal, "Effect Of Speech Coders On Speech Recognition Performance", ICSLP (1996), pp. 2344-2347.

[Lim1979] Lim, J.S., and A.V. Oppenheim, "Enhancement And Bandwidth Compression Of Noisy Speech", Proceedings of the IEEE (Vol. 67 1979), pp. 1586-1604.

[Liu1990] Liu, et.al., "Study Of Line Spectrum Frequency Pairs For Speaker Recognition", IEEE International Conference on Acoustics, Speech, and Signal Processing (1990), pp. 277-280.

[Loc1992] Lockwood, P., J. Boudy, and M. Blanchet, “Non-Linear Spectral Subtraction (NSS) And Hidden Markov Models For Robust Speech Recognition In Car Noise Environments”, IEEE International Conference on Acoustics, Speech, and Signal Processing (1992), pp. 265–268.

[Lu2009] Lu, X., S. Matsuda, M. Unoki, T. Shimizu, and S. Nakamura, “Temporal Contrast Normalization And Edge-Preserved Smoothing On Temporal Modulation Structure For Robust Speech Recognition”, IEEE International Conference on Acoustics, Speech and Signal Processing (2009), pp. 4573 – 4576.

[Mal1999] Malah, D., R.V. Cox, and A.J. Accardi, "Tracking Speech Presence Uncertainty To Improve Speech Enhancement In Non-Stationary Noise Environments", International Conference on Acoustics, Speech, and Signal Processing (1999), pp. 789-792.

[McA1980] McAulay, R.J. and M.L. Malpass, "Speech Enhancement Using A Soft-Decision Noise Suppression Filter", IEEE Transactions on Acoustics, Speech , and Signal Processing, Vol. 28 (1980), pp. 137-145.

[McC1995] McCree, A.V., and T.P. Barnwell, “A Mixed Excitation LPC Vocoder Model for Low Bit Rate Speech Coding”, International Conference on Acoustics, Speech, and Signal Processing (1995), pp. 242-245.

[Mwe1996] Mwema, W.N. and E. Mwangi, "A Spectral Subtraction Method For Noise Reduction In Speech Signals", Africon (1996), pp. 382-385.

[Ope1993] Openshaw, J.P. and J.S. Mason, "A Review Of Robust Techniques For The Analysis Of Degraded Speech", IEEE TENCON (1993), pp. 329-332.

[Osh1987] O'Shaughnessy, D., Speech Communication (Massachusetts: Addison-Wesley Publishing Company, 1987).

[Pop1998] Popescu, D.C. and I. Zelijkovic, "Kalman Filtering of Colored Noise for Speech Enhancement", International Conference on Acoustics, Speech, and Signal Processing (1998), pp. 997-1000.

[Por2002] Poruba, J., “Speech Enhancement Based On Nonlinear Spectral Subtraction”, Proceedings of the Fourth IEEE International Caracas Conference on Devices, Circuits and Systems (2002), pp. T031-1 - T031-4.

[Pot2001] Potamitis, I., N. Fakotakis, and G. Kokkinakis, "Robust Automatic Speech Recognition In The Presence Of Impulsive Noise", Electronic Letters, Vol. 37, No. 12 (June, 2001), pp. 799-800.

[Rab1978] Rabiner, L.R. and Schafer, R.W., Digital Processing of Speech Signals (New Jersey: Prentice Hall, 1978).

[Rab1993] Rabiner, L.R., and B.H.. Juang, Fundamentals of Speech Recognition (New Jersey: Prentice Hall, 1993).

[Red1976] Reddy, D.R.. “Speech recognition by machine: A review”, Proceedings of the IEEE, Vol. 64, Issue 4 (April 1976), pp.501-531.

[Rog2001] Rogoff, R., “Voice Activated GUI – The Next User Interface”, IEEE International Professional Communication Conference, Proceedings (Oct. 24-27, 2001), pp. 117-120.

[Sch1985] Schroeder, Manfred R., and Bishnu S. Atal, “Code-Extcited Linear Prediction (CELP): High-Quality Speech at Very Low Bit Rates”, IEEE International Conference on Acoustics, Speech, and Signal Processing (1985), pp. 937-940.

[Sch2001] Schlüter, Ralf and Hermann Ney, “Using Phase Spectrum Information For Improved Speech Recognition Performance”, IEEE International Conference on Acoustics, Speech, and Signal Processing (2001), pp. 133–136.

[Siu1999] Siu, M., M. Jonas, and H. Gish, "Using A Large Vocabulary Continuous Speech Recognizer For A Constrained Domain With Limited Training", IEEE International Conference on Acoustics, Speech, and Signal Processing (1999), pp. 105-108.

[Soo2000] Soon, I.Y., S.N. Koh and C.K. Yeo, "Selective Magnitude Subtraction For Speech Enhancement", High Performance Computing in the Asia-Pacific Region (2000), pp. 692-695.

[Sup1997] Supplee, L.M., R.P. Cohn, J.S. Collura, and A.V. McCree, "MELP: The New Federal Standard At 2400 BPS", International Conference on Acoustics, Speech, and Signal Processing (1997), pp. 1591-1594.

[Tia2003] Tian, Bin, Mingui Sun, Robert J. Sclabassi, Kechu Yi, “A Unified Compensation Approach for Speech Recognition in severely Adverse Environment”, IEEE Proceedings of the Fourth International Symposium on Uncertainty Modeling and Analysis (2003), pp. 256-261.

[Thi2004] Thiessen, Erik D. and Jenny R. Saffran, “Spectral Tilt As A Cue To Word Segmentation In Infancy And Adulthood”, Perception & Psychophysics (2004), pp. 779-791.

[Tug2004] Tuğaç, S., and H.G. Ilk, "Bitstream Based Wireless Speech Recognition Using Mixed Excitation Linear Prediction (MELP) Vocoder" Signal Processing and Communications Applications Conference (April 2004), pp. 414 – 417.

[Vas1996] Vaseghi, Saeed V., Advanced Signal Processing and Digital Noise Reduction (New Jersey: John Wiley & Sons, 1996).

[Wan1991] Wang, S., A. Sekey, A. Gersho, “Auditory distortion measure for speech coding”, International Conference on Acoustics, Speech, and Signal Processing (1991), pp. 493-496.

[Xu2000] Xu, J. and G. Wei, "Noise-robust Speech Recognition Based On Difference of Power Spectrum", Electronic Letters, Vol. 36, No. 14 (July, 2000), pp. 1247-1248.

[Yif2000] Yifang, Xu, Zhang Jinjie, Yao Kaisheng, Cao Zhigang, and Ma Zhengxin, "Robust Recognition Of Noisy Speech Using Speech Enhancement", Proceedings of ICSP (2000), pp. 734-737.

[Yom2006] Yoma, Néstor Becerra, Carlos Molina, Jorge Silva, and Carlos Busso, “Modeling, Estimating, and compensating Low-Bit Rate Coding Distortion in Speech Recognition”, IEEE Transactions on Audio, Speech, and Language Processing (January 2006), pp. 246-255.

[Zha2000] Zhang, WeiQi, Liang He, Yen-Lu Chow, RongZhen Yang, YePing Su, “The study on distributed speech recognition system”, IEEE International Conference on Acoustics, Speech, and Signal Processing (2000), pp. 1431-1434.