final paper on

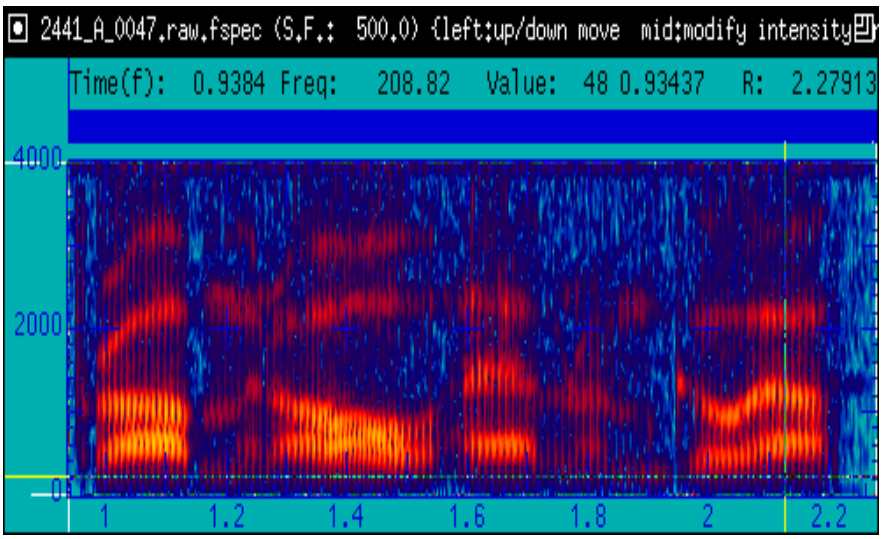
Speech Recognition using Mel cepstrum, delta cepstrum and delta-delta features

submitted to fulfill the requirements for

ECE 8993: Fundamentals of Speech Recognition

submitted to:

Dr. Joseph Picone
Department of Electrical and Computer Engineering
413 Simrall, Hardy Rd.
Mississippi State University
Mississippi State, Mississippi 39762



December 5, 1998
submitted by:

Suresh Balakrishnama
MS in Electrical Engineering Candidate
Department of Electrical and Computer Engineering
Mississippi State University
429 Simrall, Hardy Rd.
Mississippi State, Mississippi 39762

ABSTRACT

This paper explains the basic concepts of using Mel-cepstrum,delta and delta-delta features and the algorithm that uses cepstrum of the speech signal for speech recognition.

Mel scale is a better approximation of mapping perceived frequency to a linear scale. Mel-cepstral coefficients corresponding to short term correlation in speech signal are significant is obtaining a better model for the speech data. The mapping factor between the real frequency scale and the perceived Mel frequency scale is an important parameter which finds a significant use in speech recognition. A comparison of Mel-cepstral and LP derived cepstral coefficients, based on the efficiency will also be discussed.

Delta features are a measure of rate of change of a feature and useful in establishing a significant model between two frames of features. Delta-delta features are the second order derivative or the measure of rate of change of delta features. These features are very useful in speech recognition since they represent the dynamics of speech signals.

1. INTRODUCTION

All speech recognizers include an initial signal processing front end that converts a noisy and/or degraded speech waveform into features useful for further processing. The front end is required to extract important features from the speech waveform that are relatively insensitive to talker and channel variability unrelated to speech message content. This first stage also reduces the data rate into larger stages of the speech recognizer and attempts to decrease redundancy inherent in the speech waveform. The vast majority of front ends are based on standard signal processing techniques, such as filter banks, linear predictive coding (LPC), or homomorphic analysis (cepstra). There has also been interest in front ends based on known properties of the human auditory system. Some of these front ends are linear but with parameters that correspond to auditory properties (e.g., filter bank bandwidths increasing with frequency). Most of the auditory-based front ends, however, are nonlinear since this is believed to be the case for many physiological and/or perceptual processes in the auditory system. Feature selection is generally considered a process of mapping the original measurements into more effective features. If the mapping is linear, the mapping function is well defined and our task is simply to find the coefficients so as to optimize based on a criterion. If a proper criterion for evaluating the effectiveness of features is obtained. techniques of linear algebra can be used for simple criteria and iterative techniques to determine the mapping coefficients in case of complex criterion. In many applications of pattern recognition, important features are not linear functions of the original measurements but are highly nonlinear functions. The basic problem is to find a nonlinear mapping function for the given data.The selection of features becomes domain dependent in speech research. We discuss in detail the theory behind various features commonly used for pattern recognition.

2. DESCRIPTION OF FEATURES

2.1. Mel Cepstrum

Spectrum is the representation of the signal with which we can assess the “separation” of the component parts and perhaps derive needed information about the components and also, the representation of the component signals are combined linearly in the spectrum. The shape of spectrum provides the maximum information present in speech signal. Information like high frequency (high or low), resonance, noise information can be obtained

using spectrum[10]. On the other hand, the "cepstrum" represents a transformation on the speech signal with two important properties:

1. Representatives of component signals are *separated* in the cepstrum.

2. Representatives of component signals are *linearly combined* in the cepstrum.

The cepstrum provides the needed information to assess the properties of the component signals. The cepstrum derived from homomorphic processing (cepstral analysis within a general class of methods) is usually called the complex cepstrum and real part of complex cepstrum within a scale factor is called the real cepstrum.At a certain time in speech research the cepstrum features began to supplant the direct use of the LP parameters as the important feature obtained from hidden Markov modelling strategy because of two convenient enhancements that were found to improve recognition rates. First, is the ability to easily smooth the LP based spectrum using

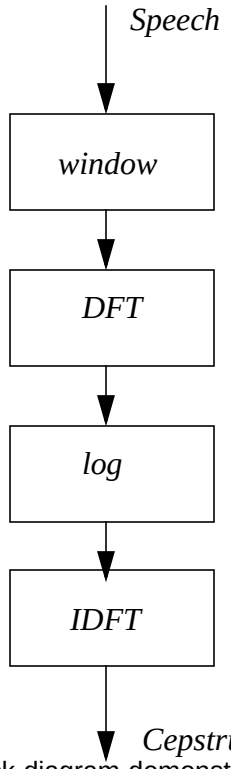


Figure 1. Block diagram demonstrating process of obtaining cepstrum features from a speech signal

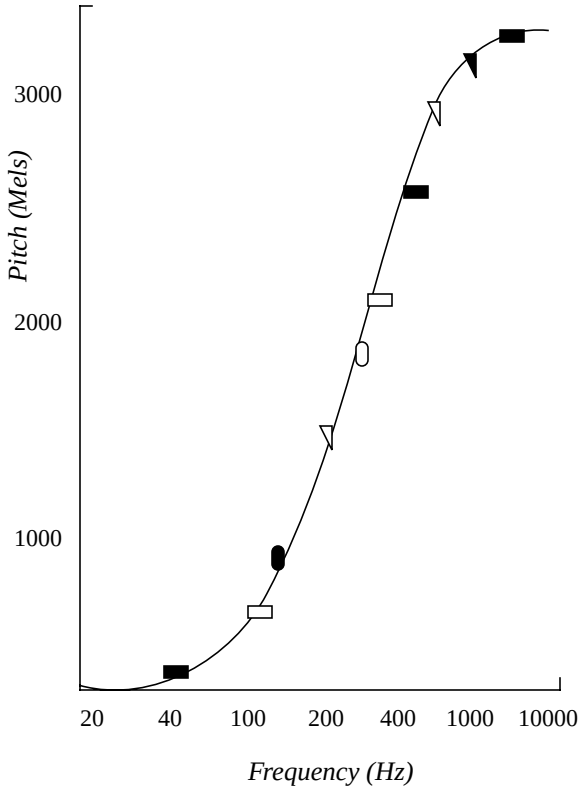


Figure 2. Mel scale illustrating the linear relationship between the real frequency scale (Hz) and the perceived frequency scale(Mels)

the liftering and weighting process. Liftering is a useful process with the real cepstrum for obtaining an estimate of the log spectrum of either of the separated components of the cepstrum. Weighting is a procedure wherein the euclidean distance between cepstral coefficients for which each term of the sum is multiplied by a predetermined weighting coefficient w_k. However when constant weighting is used this reduces to the standard cepstral distance. Triangular weighted cepstral distances comprise the subclass of weighted cepstral distance measures for which the weighting factor increases linearly with the index (k). Speaker dependent and speaker independent recognition experiments have shown that for triangular weighted cepstral distance measures recognition performance is best when the number of cepstral difference

terms are approximately equal to the order of all-pole model.This process removes the inherent variability of the LP based spectrum due to the excitation and improves recognition performance. The other method of improvement over direct use of the LP parameters is the use of so-called “Mel-based cepstrum”. A Mel is a unit of measure of perceived pitch of frequency of a tone. It does not correspond linearly to the physical frequency of the tone, as the human auditory system apparently does not perceive pitch in this linear manner. The precise meaning of the Mel scale becomes clear by examining the experiment by which it is derived. With several experiments in speech research it was concluded that a linear relationship exists between the real frequency scale (Hz) and the perceived frequency scale(Mels). The graph demonstrating the linear relationship between these parameters is shown in the figure 2. The equation illustrating the relationship between the frequency scale and perceived frequency scale is shown in equation 1.

$$F_{mel} = \frac{1000}{\log 2}\left[1 + \frac{F_{Hz}}{1000}\right] \quad (1)$$

where F_{mel} is the perceived frequency in Mels and F_{Hz} is the real frequency in Hz.

A mel filter bank (MFB) base cepstral transformation is used as a conventional control front end in most of speech recognition applications. This type of filtering involves multiplying speech waveform by a 20-ms-long Hamming window every 10ms and then computing DFT for each windowed waveform segment. In frequency domain, a vector of log energies is computed from each waveform segment by weighting the DFT coefficients by the magnitude frequency response of a filter bank. The center frequencies of the filters are spaced equally on a linear scale from 100 to 1000 Hz and equally on a logarithmic scale above 1000 Hz. Above 1000 Hz, each center frequency is 1.1 times the center frequency of the previous filter. Each filter’s magnitude frequency response has a triangular shape that is equal to unity at the center frequency and linearly decreasing to zero at the center frequencies of the two adjacent filters. Each vector of log energies is then processed by an inverse cosine transform creating a vector of mel filter bank cepstral coefficients. The cepstral coefficients are then used as input features to the speech recognizer. On a SPARCStation 2 workstation, the MFB cepstral front end operates in roughly one third real-time at a data rate of 100 frames/sec.

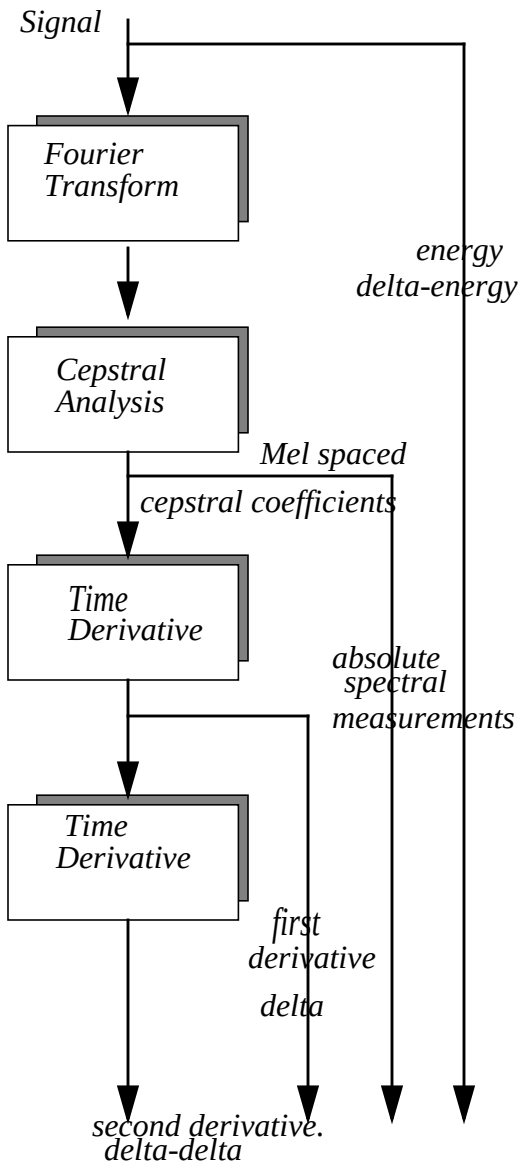


Figure 3. Flowchart exhibiting extraction of different features from a speech file

2.2. Delta cepstrum features

The performance of a speech recognition system can be greatly enhanced by adding time derivatives to the basic static parameters. The delta features can be computed using the regression formula as described in equation 2

$$\cdot d_t = \frac{\sum_{w=1}^{W} w(c_{t+w} - c_{t-w})}{2\sum_{w=1}^{W} w^2} \quad (2)$$

where d_t is a delta feature at time t computed in terms of the static parameters before and next to the current features or coefficient. W is delta window size used to configure the entire data. Since the regression formula depends on past and future speech parameter values, some modification is required to use the signal occurring at beginning and end of speech file. This can be solved by using a simple first order differences at the start and end of the speech as shown in equations 3 and 4.

$$d_t = c_{t+1} - c_t, t < W \quad (3)$$

and

$$d_t = c_t - c_{t-1}, t \geq T - W \quad (4)$$

where T is the total length of data file.

In addition to the cepstral or Mel-cepstral parameters, another popular feature used in contemporary speech recognition is the delta cepstrum. If $c(n, m)$ denotes the Mel-cepstrum feature values for the frames of the signal $s(n)$ ending at time m, the delta-cepstrum at frame m can be defined using equation 5.

$$\Delta c(n, m) = c(n, m + \delta Q) - c(n, m - \delta Q) \quad (5)$$

for all frames comprising the data. Here Q represents the number of samples by which the window is shifted for each frame. The parameter δ is chosen to smooth the estimate and typically take a value so as to look forward and backward one or two frames. A vector of such features at relatively low value of n provides information about spectral changes that have occurred since the previous frame. The delta-cepstrum can also be computed for LP based cepstral parameters. For any typical application of pattern recognition, 8-14 cepstral coefficients along with their derivatives are used in systems that employ cepstral techniques.

In areas of signal processing, computing power becomes a vital issue when considering the efficiency of algorithms. The most widely applied approximation for first order time derivative of signal $s(n)$ are:

$$s^*(n) = \frac{\partial}{\partial t} s(n) = s(n) - s(n-1) \quad (6)$$

$$s^*(n) = \frac{d}{dt} s(n) = s(n+1) - s(n) \quad (7)$$

$$s^*(n) = \frac{\partial}{\partial t} s(n) = \sum_{m=-N_d}^{N_d} m s(n+m) \quad (8)$$

Equations 6 and 7 are known as backward and forward differences respectively. The signal output form this differentiation process is defined as a delta parameter. The second order time derivative can be similarly approximated by reapplying equation 8 to the output of the first-order differentiator, The output obtained from second-order differentiation is referred as delta-delta parameter.

2.3. LP Derived Cepstral Coefficients

Linear Prediction analysis has been among the most popular methods for extracting spectral information from speech. LP analysis does not resolve the vocal-tract characteristics. Since the laryngeal characteristics vary from person

to person, and even for within person utterances of the same words, LP parameters convey some information to a speech recognizer that degrades performance, particularly for speaker-independent system.The linear prediction model is a very useful tool to compute the cepstral coefficients. If the linear prediction filter is stable (and it is guaranteed to be stable in the autocorrelation analysis), the logarithm of the inverse filter is expressed as follows[1].

$$C_{LP} = \sum_{i=0}^{N_c} C_{LP}(i) z^{-i} \quad (9)$$
$$= \log F(z)$$

The coefficients can be solved by differentiating both sides of the expression with respect to z^{-1}, and equating coefficients of the resulting polynomial. This results in the following equations.

LP error:

$$C_{LP}(1) = -a_{LP}(1) \quad (10)$$

For $2 \le i \le N_C$,

$$C_{LP}(i) = -a_{LP}(i) \quad (11)$$

$$= \sum_{j=1}^{i-1} \left(1 - \frac{j}{i}\right) a_{LP}(j) C_{LP}(i-j) \quad (12)$$

The coefficients C_{LP} are referred as LP-derived cepstral coefficients.

After LP analysis of speech is carried out. various equivalent representations of the LP parameters exist. A comparison of these parameters in terms of speaker recognition accuracy revealed that the LP cepstrum is the best when training and testing is done on a clean speech database. The problem with the LP cepstrum is that a mismatch in training and testing conditions sacrifices much performance, thereby diminishing the robustness. The LP spectrum is derived from an all-pole transfer function that describes the spectral envelope of the speech. This in particular gives information about the formants that is critical for speaker recognition to be successful. The first step involved is to transform the all-pole transfer functions derived from LP analysis into a pole-zero transfer function that gives more emphasis to the formants. The cepstrum of the pole-zero transfer function is the feature.

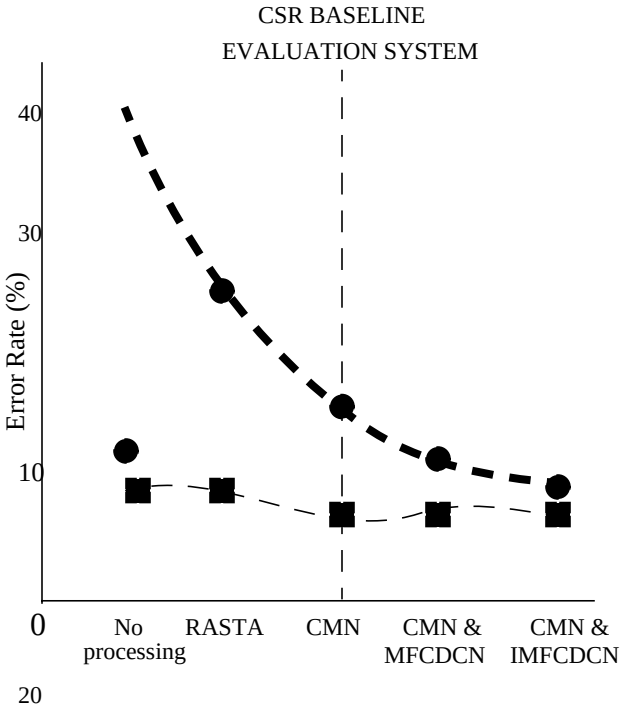


Figure 4. Comparison of the effects of MFCDCN, IMFCDCN, CMN and RASTA algorithm on recognition accuracy for DARPA CSR evaluation data

2.4. Cepstrum based Algorithms

The demand for need of a speech recognition systems and spoken language systems to be robust with respect to their acoustical environment has been increasing. The penultimate focus is on the performance of series of cepstrum-based procedures that enable

the speech recognition system to maintain a high level of recognition accuracy over a wide variety of acoustical environments. Further in the paper, we will discuss in detail different types of cepstrum-based normalization algorithms and their efficiency in terms of word error rate. Multiple fixed codeword-dependent cepstral normalization(MFCDCN) is an extension of fixed codedword-dependent cepstral normalization(FCDCN) which provides an additive environmental compensation to cepstral vectors, based on the acoustic environment. MFCDCN is less complex as far as the computational procedures are involved. It does not require domain-specific training to new acoustical environments[6].

SDCN (SNR Dependent Cepstral Normalization) - This is the simplest compensation algorithm and is applied to the correction vector in the cepstral domain that depends exclusively on the instantaneous SNR of the signal. The correction vector is the average difference in cepstra between simultaneous stereo recordings of speech samples from both the training and testing environments at each SNR of speech in the testing environment. When the SNR values are hight, the correction vector primarily compensates for differences in spectral tilt between the training and testing environments and at low SNR values the vector provides a form of noise subtraction. The SDCN algorithm is simple and effective, but it required environment-specific training[6].

FCDCN (Fixed codeword-dependent cepstral normalization) - This normalization algorithm is a form of compensation that provides greater recognition accuracy than SDCN but in a more computationally-efficient manner than the CDCN algorithm.The FCDCN algorithm applies an additive correction that depends on the instantaneous SNR of the input but that can also vary from codeword to codeword[6].

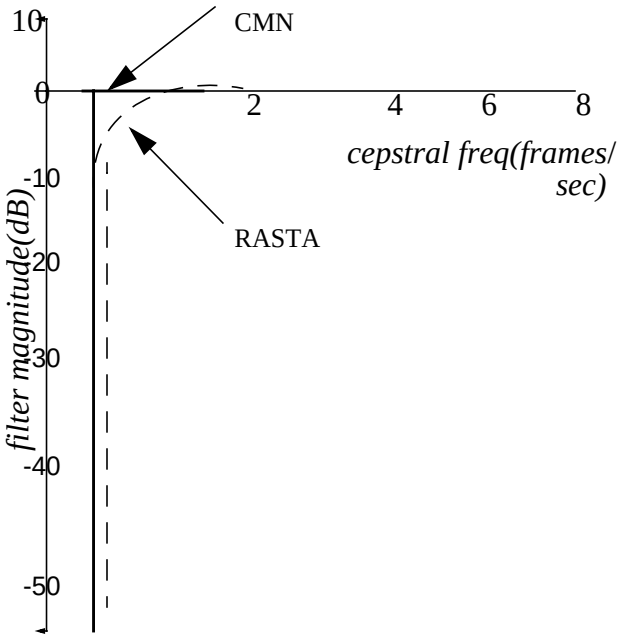


Figure 5. Comparison of the frequency response of the highpass filters implemented by RASTA algorithm as used by Stanford Research Institute (shown in dotted curve) and as implied by CMN (solid curve)

$$\hat{x} = z + r[k,l] \tag{13}$$

where for each frame $\hat{x}$ represents the estimated cepstral vector of the compensated speech, z is the cepstral vector of the incoming speech in the target environment, k is an index identifying the vector quantization codeword, l is an index identifying the SNR, and $r[k,l]$ is the correction vector.

The selection of the appropriate codeword is done at the vector quantization stage, so that the label k is chosen to minimize

$$\|z + r[k,l] - c[k]\|^2 \tag{14}$$

where the $c[k]$ are the vector quantization codewords of the speech in the training database. The new correction vectors are estimated with an EM algorithm that maximizes the likelihood of the data.

The probability density function of x is

assumed to be a mixture of Gaussian densities.

$$p(x) = \sum_{k=0}^{k=1} P[k](N_x c[k], \sum k) \quad (15)$$

The cepstra of the corrupted speech are modeled as Gaussian random vectors, whose variance depends also on the instantaneous SNR, l, of the input.

$$p(z|k,r,l) = \frac{C''''}{\sigma[l]} \exp\left(\frac{-1}{2\sigma^2}\|(z + r[k,l] - c[k])\|^2\right) \quad (16)$$

MFCDCN (Multiple fixed codeword-dependent cepstral normalization) - This algorithm is an extension of FCDCN algorithm and it does not require environment specific training. Here, the compensation vectors are precomputed in parallel for a set of target environments using the FCDCN algorithm. When an utterance from an unknown environment is input to the recognition system, compensation vectors computed using each of the possible target environments are applied successively and the environment is chosen that minimizes the average residual vector quantization distortion over the entire utterance,

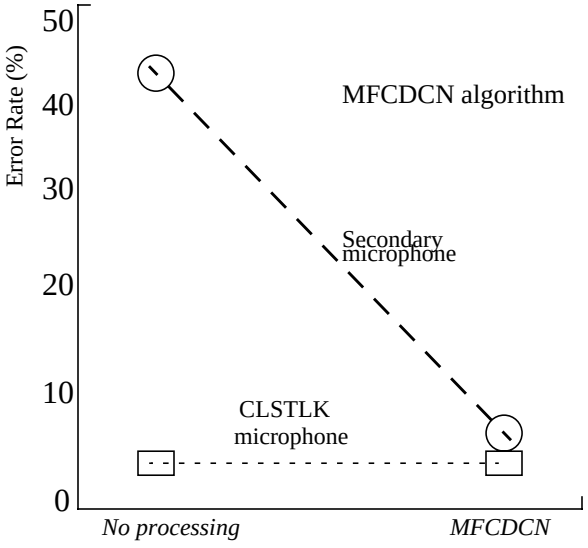


Figure 6. Performance of MFCDCN algorithm

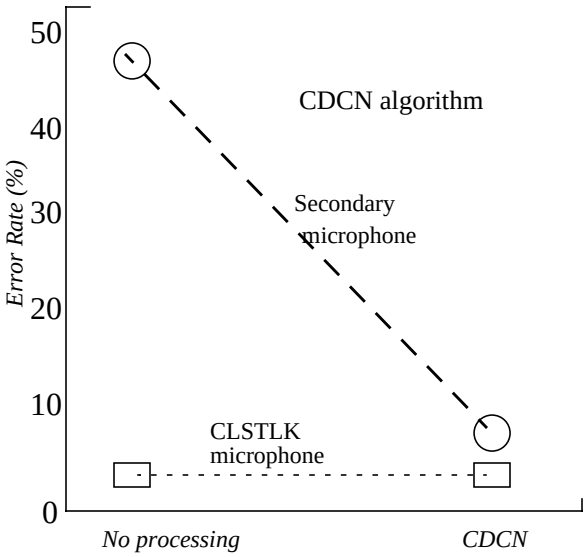


Figure 7. Performance of CDCN algorithm

$$\|z + r[k,l,m] - c[k]\|^2 \quad (17)$$

where k refers to the vector quantization codeword, l to the SNR,and m to the target environment used to train the ensemble of compensation vectors. The success of MFCDCN depends on the availability of training data with stereo pairs of speech recorded from the training environment and from a variety of possible target environments, and on the extent to which the environments in the training data are representative of what is actually encountered in testing[6].

IMFCDCN (Incremental Multiple fixed codeword-dependent cepstral normalization) - This is a unsupervised incremental adaptation algorithm. MFCDCN involves environment selection for the compensation vectors on utterance-to-utterance based whereas the probability of correct selection can be improved by allowing the classification process to make use of cepstral vectors from previous utterance[6].

CDCN (Codeword-Dependent Cepstral Normalization) - This algorithm uses Expected maximization techniques to compute ML estimates of the parameters characterizing the

contributions of additive noise and linear filtering that when applied as inverse function to the cepstra of an incoming utterance produce an ensemble of cepstral coefficients that best match the cepstral coefficients of the incoming speech in the testing environment to the locations of vector quantization codewords in the training environment[6].

RASTA - This is a filtering algorithm which provides considerable amount of environmental robustness at almost negligible cost. In RASTA a high-pass filter is applied to a log-spectral representation of speech such as the cepstral coefficients. The highpass filter can be described by the difference equation

$$y[n] = x[n] - x[n+1] + 0.97y[n-1] \quad (18)$$

where $x[n]$ and $y[n]$ are the time-varying cepstral vectors of the utterance before and after RASTA filtering, and the index n refers to the analysis frames[6].

CMN (Cepstral Mean Normalization) - This is a filtering algorithm used to obtain high-pass filter cepstral coefficients wherein the mean of cepstral vectors are subtracted from the cepstral coefficients of that utterance on a sentence-by-sentence basis.

$$y[n] = x[n] - \frac{1}{N}\sum_{n=1}^{N} x[n] \quad (19)$$

where N is the total number of frames in an utterance and $x[n]$ is the signal.

3. EXPERIMENTAL SUPPORT

The normalization and filtering algorithms were evaluated using the CMU recognition system on a data of continuous speech recognition systems using a 5000-word closed-vocabulary task consisting[6] of dictation of sentences from the Wall Street Journal. A component of that evaluation involved utterances from a set of unknown “secondary” microphones, including desktop machines, telephone handsets and speakerphones, stand mounted microphones, and lapel-mounted microphones.

3.1. Results of cepstral algorithms for DARPA evaluations

The MFCDCN algorithm was trained using 15 environments in the training set and developmental test set for this evaluation. It is observed that both CDCN and MFCDCN algorithms significantly improve the recognition accuracy obtained with secondary micropohones[6], with very little or no loss in performance when applied to speech from the close-talking(CLSTLK) microphone. The small degradation in recognition accuracy observed for speech from the CLSTLK microphone using the MFCDCN algorithm may be at least in part a consequence of errors in selecting the environment for the compensation vectors. Environment-classification errors occurred on 48.8% of the CLSTLK utterances and on 28.5% of the utterances from secondary microphone. The evaluation were repeated using MFCDCN compensation vectors obtained using only seven categories of microphones instead of original 15 acoustic environment. This modification produced only a modest increase in error rate for speech from secondary microphones (from 17.7% to 18.9%) and actually improved the error rate for speech from CLSTLK microphone (from 9.4% to 8.3%). Figure 6 and 7 illustrate the results of evaluations using cepstral algorithms.

3.2. Results of cepstral algorithms for stress-test evaluation

The data for stress-test evaluation consists of spontaneous speech, utterances containing out-of-vocabulary words and speech from unknown microphones and environments, all related to Wall Street Journal domain. The speech recognition system was trained on

13,000 speaker-independent utterances from the Wall Street Journal task and 14,000 utterances of spontaneous speech from the ATIS travel planning domain. The trigram grammar for the system was derived from 70.0 million words of text without verbalized punctuation and 11.6 million words with verbalized punctuation. The performance of baseline system was compared with system added with IMFCDCN. The baseline system achieved a a word error rate of 22.9% using only the bigram language model. The system operating with IMFCDCN reduced the error rate only to 22.7% compared to 20.8% for the stress-test system using IMFCDCN. The IMFCDCN provided only a small significant change in the results because only a small percentage of data in this test was from secondary microphones.

3.3. Results using LP derived cepstral features

The conventional feature used is the linear predictive (LP) cepstrum derived from an all-pole transfer function. A new cepstral feature based on a pole-zero function called adaptive component weighted or ACW cepstrum is considered for comparison purposes and two other features (known as PFL1 cepstrum and PFL2 cepstrum) are based on pole-zero postfilter used in speech enhancement. To test the system, a test utterance from one of the M speakers is converted to a set of test feature vectors. Each of the test feature vector is quantized by each of the M codebooks. The quantized vector is that which is closest according to some distance measure to the test feature vector. The squared euclidean distance is the best measure for pattern recognition. Hence, M different distances are recorded, one for each codebook and the process is repeated for every test feature vector. The distances are accumulated over the entire set of feature vectors. The codebook that renders the smallest accumulated distance identifies the speaker. When many utterances are tested, the success rate is the number of utterances for which the speaker is identified correctly divided by the total number of utterances tested. The codebooks are trained for one particular condition, namely, for clean speech. Different test conditions corresponding to clean and corrupted speech is used to provide a definitive and quantitative evaluation of robustness. If a feature is robust, a mismatch between the testing and training conditions causes a slight degradation in performance or success rate. Two data bases are used in the experiments - TIMIT and King data base. TIMIT comprises only clean speech, 20 speakers from the New England dialect are considered. The speech is downsampled from 16 to 8 KHz. For each speaker, there are ten sentences. The first five are used for training the vector quantizer classifier. The classifier is trained only on clean speech. The remaining five sentences are individually used for testing. King data base consists of 26 San Diego and 25 Nutley speakers. Speech is recorded over long distance telephone lines and sampled at 8 KHz. There are ten recording sessions, each having one utterance per speaker. The data is divided such that there is a big mismatch in the conditions between sessions 1 to 5 and sessions 6 to 10. This mismatch is due to a change in the recording equipment, which translates to a significantly changed environment. Training is done on session 1. Testing “within the great divide” corresponds to the utterances in sessions 2 to 5 in which there is some mismatch with session 1. Testing “across the great divide” corresponds to the utterances in sessions 6 to 10, which in turn provide a big mismatch. Training is done on session 2 while the remaining nine sessions are used for testing. The total number of test utterances considered “within the great divide” were 208 for San Diego portion and 200 for the Nutley portion whereas for “across the great divide” was 260 for San Diego

portion and 250 for the Nutley portion.

3.4.1. Testing on Clean Speech

This experiment involves testing of clean speech which is performed by using TIMIT database[8]. The performance does not always monotonically increase as the codebook gets bigger. Therefore, just using a large codebook size does not benefit in terms of performance and imposes a cost in terms of memory and search complexity. In the limit as the codebook size equals the number of vectors in the training set, a nearest neighbor classifier is obtained. Experiments show that the nearest neighbor classifier is inferior to the vector quantizer using modest size codebooks. This is because overlearning of the training data has taken place. Results on testing of clean speech are illustrated in table 1.

Table 1 Identification Success rate as a percent for clean speech (Timit database). Success rates correspond to codebook sizes of 16, 32 and 64

Features	Identification success rate
LP cepstrum	91 96 94
ACW	92 93 91
ACW2	90 96 93
PFL1	92 92 95
PFL2	89 94 95

3.5.2. Testing on Noisy Speech

Here, the test speech is degraded by additive white Gaussian noise[8]. As the SNR decreases, the mismatch between the training and test conditions becomes more glaring and the performance for all the features decreases. Results are illustrated in table 2.

Table 2 Identification Success rate as a percent for speech degraded by additive white gaussian noise (Timit database). Success rates correspond to codebook sizes of 16, 32 and 64

Feature	Test Condition		
	Noisy speech 30 dB SNR	Noisy speech 20 dB SNR	Noisy speech 10 dB SNR
LP cep-strum	79 85.3 86.3	47 56.3 61.3	18.7 24.7 21
ACW	82.3 84.7 87	57 64.7 64	26.3 26.7 23.3
ACW2	84.3 88.3 91.3	50.7 63.3 60.7	19.3 23.7 23.3
PFL1	87 83.3 86	63 67 68	27 28.3 22.7
PFL2	82.3 85 88.7	52.7 62.7 63.3	22.3 24 23

4. SUMMARY

The use of MFCDCN and phone-dependent cepstral normalization algorithms reduce the error rate by 40 percent compared to result obtained with CMN alone.

When training a system with clean speech and testing with noisy speech, LP cepstra is not suitable the performance is significantly worse than mel cepstrum.Experiments conducted using both the TIMIT and King data bases reveal that performance under mismatched training and testing conditions is a good measure of robustness. The adaptive component weighted cepstrum and the cepstrum based on the pole-zero transfer

functions perform better than the LP cepstrum.

5. ACKNOWLEDGEMENTS

I wish to thank Aravind Ganapathiraju and Neeraj Deshmukh of Institute for Signal and Information Processing (ISIP) for their constant support and guidance on this paper.

6. REFERENCES

[1] J. Picone et. al., "Signal Modeling Techniques in Speech Recognition", in *Proceedings of the 1993 IEEE Automatic Speech Recognition and Understanding Workshop*, Vol. 81, No. 9, pp. 1215-1247, September 1993.

[2] P.Loizou, "Feature extraction programs for speech Recognition", *http://giles.sgilab.ualr.edu/asd/loizou.html*, University of Arkansas at Little Rock.

[3] J. Hamaker, A. Ganapathiraju, J. Picone, and J. Godfrey, "Advances in Alphadigit Recognition Using Syllables," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Seattle, Washington, USA, May 1998.

[4] R.Stern, F.Liu, Y.Ohshima, M.Sullivan and A.Acero, "Multiple Approaches to Robust Speech Recognition", *School of Computer Science, Carnegie Mellon University*, Pittsburgh, USA.

[5] F.Liu, P.Moreno, R.Stern and A.Acero, "Signal Processing for Robust Speech Recognition", *School of Computer Science, Carnegie Mellon University*, Pittsburgh, USA.

[6] F.Liu, R.Stern, X.Huang and A.Acero, "Efficient Cepstral Normalization for Robust Speech Recognition", *School of Computer Science, Carnegie Mellon University*, Pittsburgh, USA.

[7] C.R.Jankowski, Hoang-Doan H.Vo and R.Lippmann, "A Comparison of Signal Processing Front Ends for Automatic Word Recognition", in *IEEE transactions on Speech and Audio Processing*, Vol. 3, No.4, July 1995.

[8] M.S.Zilovic, R.Ramachandran and R.Mammone, "Speaker Identification based on the use of Robust Cepstral Features obtained from Pole-Zero Transfer Functions ", *IEEE transactions on Speech and Audio Processing*, Vol 6, No.3, May 1998.

[9] A.Syrdal, R.Bennett and S.Greenspan, "Applied Speech Technology", *CRC Press, Inc*.,USA.

[10] J.Deller, J.G.Proakis and J.Hansen, "Discrete-time processing of speech signals", *Macmillan Publishing Company*, New York, USA.

[11] S.Young, J.Jansen, J.Odell, D.Ollason and P.Woodland, "HTK-Hidden MArkov Model Toolkit", *Entropic Cambridge Research Laboratory*, Cambridge CB3 OAX, England, U.K.

[12] T.H.Applebaum, B.A.Hanson and H.Wakita, "Weighted Cepstral Distance Measures in Vector Quantization based Speech Recgonizers", *Speech Technology Laboratory*, Santa Barbara, California, USA.

[13] C.R.Jankowski Jr., H.D.H.Vo and R.P.Lippmann, "A Comparison of Signal Processing Front Ends for Automatic Word Recognition", *IEEE Transactions on Speech and Audio Processing*, Vol 3, No.4, July 1995.