ACADEMIC PRESS

Available online at www.sciencedirect.com

SCIENCE DIRECT®

Computer Speech and Language 17 (2003) 381–402

COMPUTER SPEECH AND LANGUAGE

www.elsevier.com/locate/csl

Non-linear feature extraction for robust speech recognition in stationary and non-stationary noise ☆

Qifeng Zhu, Abeer Alwan *

Department of Electrical Engineering, The Henry Samuli School of Engineering and Applied Science, 66-147E Engr. IV, UCLA 405 Hilgard Avenue, Box 951594, Los Angeles, CA 90095-1594, USA

Received 9 August 2001; received in revised form 22 October 2002; accepted 22 March 2003

Abstract

An analysis-based non-linear feature extraction approach is proposed, inspired by a model of how speech amplitude spectra are affected by additive noise. Acoustic features are extracted based on the noise-robust parts of speech spectra without losing discriminative information. Two non-linear processing methods, harmonic demodulation and spectral peak-to-valley ratio locking, are designed to minimize mismatch between clean and noisy speech features. A previously studied method, peak isolation [IEEE Transactions on Speech and Audio Processing 5 (1997) 451], is also discussed with this model. These methods do not require noise estimation and are effective in dealing with both stationary and non-stationary noise. In the presence of additive noise, ASR experiments show that using these techniques in the computation of MFCCs improves recognition performance greatly. For the TI46 isolated digits database, the average recognition rate across several SNRs is improved from 60% (using unmodified MFCCs) to 95% (using the proposed techniques) with additive speech-shaped noise. For the Aurora 2 connected digit-string database, the average recognition rate across different noise types, including non-stationary noise background, and SNRs improves from 58% to 80%.

1. Introduction

Noise robustness is a key issue in automatic speech recognition (ASR). It can be approached either from the so called "back-end", the acoustic model (often HMM), using model adaptation

☆ Portions of this work were presented at ICSLP 2000 and Eurospeech 2001.

* Corresponding author. Tel.: +310-206-2231; fax: +310-206-4685.
E-mail addresses: qifeng@icsl.ucla.edu (Q. Zhu), alwan@icsl.ucla.edu (A. Alwan).

0885-2308/$ - see front matter
doi:10.1016/S0885-2308(03)00026-3

techniques (e.g., MLLR (Leggetter & Woodland, 1994), and MAP (Gauvain & Lee, 1994)), model compensation (e.g., PMC (Gales & Young, 1996)) or robust statistics (e.g., Renevey & Drygajlo, 2000), or handled at the "front-end" which involves acoustic feature extraction. The "back-end" approach often aims at modifying the means, μ, and variances, σ^2, for each state in the Hidden Markov Models (HMMs) trained with clean data so that they match noisy data in testing.

This paper focuses on the "front-end" approach. The goal is to find non-computationally intensive methods for noise-robust ASR feature extraction. With the current statistical pattern recognition scheme for ASR with Gaussian HMMs, there are three major criteria for a good acoustic feature: means do not shift under noise, variances relative to the feature space do not increase under noise, and the feature itself is discriminative. The third criterion can be defined parametrically the same way it is defined for Linear Discriminative Analysis (LDA) (Haeb-Umbach & Ney, 1992): inter-class variances should be large and intra-class variances should be small. The first and second criteria are more directly related to noise robustness. The third criterion contributes to noise robustness in a less direct way: if the feature is discriminative, even if it is altered by noise (mean shift and variance increase), it can tolerate the change and give correct recognition results. If diagonal covariance matrices are used in HMMs, an extra practical criterion should be that components in a feature vector are uncorrelated. This paper focuses on minimizing the mean shift without loss of discriminative information.

In studies on noise-robust features, linear representation of speech spectra was first used in frame-based feature extraction, for example, by using non-parametric FFT spectra or parametric smoothed LPC spectra. Later studies went beyond linear representations for better robustness with various signal-processing techniques. There are several important approaches in these studies and they all aim at satisfying one or more of the three criteria mentioned above, implicitly or explicitly.

The first approach to improve robustness is by using perceptually motivated features. These efforts include incorporating the Mel scale (Davis & Mermelstein, 1980) or Bark scale filter-bank analysis and equal loudness adjustment (Hermansky, 1990), which simulate the ear's frequency resolution and loudness perception, respectively. Mel Frequency Cepstral Coefficients (MFCCs) (Davis & Mermelstein, 1980) and Perceptual Linear Prediction (PLP) features (Hermansky, 1990) are two of the most widely used ASR features that use this approach. Typically, the first and second derivatives are also used in ASR. Simultaneous masking (Paliwal & Lilly, 1997) and forward masking (Strope & Alwan, 1997) in human perception can be modeled and used in feature extraction for better noise robustness. Another technique is subband-autocorrelation (SBCOR) proposed by Kajita and Itakura (1994), which incorporates information based on the periodicities in auditory nerve firings. Fast changing cues around syllable transitions, which are important cues for speech perception, are exploited using variable frame rate analysis (VFR) in (Zhu & Alwan, 2000a) and shown to improve noise robustness.

The second approach is analysis-based. The mismatch between the means and variances in the models trained with clean data and those of noisy testing data is a major cause of degradation in recognition under noise, and the goal of this approach is to decrease this mismatch. Using simple noise models and an assumption that the noise corrupting the speech is additive and stationary, several methods have been derived to minimize the mean shift and variance increase. Cepstral

Mean Subtraction (CMS) (Furui, 1981) and Spectral Subtraction (SS) (Boll, 1979; Virag, 1999) are two successful methods of this approach, where the feature mean or the mean noise estimate is subtracted from the spectra or cepstra computed from noisy data. Some modifications of these techniques include non-linear spectral subtraction (NSS) (Lockwood & Boudy, 1992), which can take into account the underlying shape of speech spectra. However, these subtraction techniques require a good estimate of the noise, which in practice may be difficult especially for non-stationary noise background. Statistics-based feature enhancement is used to decrease the mismatch in the minimal mean square error (MMSE) (Poter & Boll, 1984) sense. In Deng, Droppo, and Acero (2002) prior information of speech features is incorporated into the statistically based features.

Another way to directly alleviate the problem that the means of the noisy features shift away from those of clean features is to use a high-pass filter, instead of mean subtraction that only removes the DC component. Thus, the stationarity assumption is replaced by a weaker assumption that the noise is slowly changing, so that noise effects can be removed by high-pass filtering. RASTA (Hermansky & Morgan, 1994), for example, was introduced so that relative spectral changes can be captured and the slowly changing bias caused by noise is removed. No explicit noise estimation is required.

Some methods related to the second approach that aim directly at mismatch could be perceptually inspired. For example, in the modulation spectrum (Greenberg & Kingsbury, 1997; Kanedera, Hermansky, & Arai, 1998) technique, of which RASTA is a special case, knowledge that human speech perception is most sensitive to modulation frequencies around 14 Hz is used to design an appropriate band-pass filter.

The third approach is a subspace based approach (Ephraim & Trees, 1995). It has attracted considerable attention recently and is used for discriminative analysis or mean-shift compensation. The basic idea is to find a linear mapping that optimizes a cost function and is often implemented by multiplying the feature vector with a transformation matrix. Principal Component Analysis (PCA) (Kocsor et al., 2000), LDA (Haeb-Umbach & Ney, 1992; Lieb & Haeb-Umbach, 2000), and Independent Component Analysis (ICA) (Kocsor et al., 2000) are examples of this approach. Recent progress includes Heteroscedastic Discriminant Analysis (HDA) (Saon, Padmanabhan, Gopinath, & Chen, 2000) and multiple subspace projection (Gales, 2002).

Besides these three major approaches, other approaches exist. For example, capturing temporal information can result in noise robustness as shown in (Milner, 1996) and (Strope & Alwan, 1998), and neural network based non-linear feature transformation can also be effective (Sharma, Ellis, Kajarekar, & Hermansky, 2000).

The fourth approach, which is introduced in this paper, focuses on the mismatch problem. A general model of additive noise is first introduced. With this model we quantitatively study how noise affects speech spectra at the frame level and what parts of the speech spectra are vulnerable to noise. Several non-linear processing methods are constructed, as inspired by the model. The primary goal is to design a feature that attempts to avoid the effect of noise by not using the vulnerable parts of speech spectra, without losing important discriminative information. This approach differs from the noise removal methods, such as mean subtraction, because it does not require an estimate of the noise and does not assume a stationary or slowly changing noise background.

2. Additive noise model

2.1. Analysis

In the presence of additive noise, the speech signal, $s(k)$, is affected by noise, $n(k)$, resulting in the observed noisy signal $x(k)$

$$x(k) = s(k) + n(k).$$

This additive relation holds in the frequency domain

$$X(e^{j\omega}) = S(e^{j\omega}) + N(e^{j\omega}).$$

The same relation, however, does not hold for the amplitude spectra unless the signal and noise are in phase. Since amplitude spectra are often used in the computation of ASR features like the MFCCs, quantifying the effects of noise on speech amplitude spectra is important.

In the following analysis, without loss of generality, we consider one frequency component (ω_0) in the spectrum, and consider the speech spectrum as an unknown signal while the noise spectrum is a random variable. The complex spectra of the noise and speech at frequency ω_0 are assumed to be:

$$N(e^{j\omega_0}) = be^{j\theta_2},$$

$$S(e^{j\omega_0}) = ae^{j\theta_1},$$

where a and b are the speech and noise spectral amplitudes at ω_0, respectively, hereafter referred to as "amplitudes". For the noise, both b and θ_2 are random variables. The amplitude of the observed noisy spectrum $X(e^{j\omega})$ at ω_0 is

$$p = |ae^{j\theta_1} + be^{j\theta_2}|,$$

where p is a random variable. Depending on the relation between θ_1 and θ_2, p may be larger or smaller than the signal amplitude a.

To compute the expectation of p we first consider the amplitude of the noise, b, to be deterministic; then the only thing that is random is the phase of the noise, which is uniformly distributed between 0 and 2π

$$\begin{aligned} E(p) &= \frac{1}{2\pi}\int_0^{2\pi} |ae^{j\theta_1} + be^{j\theta_2}|\,d\theta_2 = \frac{1}{2\pi}\int_0^{2\pi} \sqrt{a^2 + b^2 + 2ab\cos(\theta)}\,d\theta \\ &= \frac{a}{2\pi}\int_0^{2\pi} \sqrt{1 + r^2 + 2r\cos(\theta)}\,d\theta = a \cdot Q(r), \end{aligned} \tag{1}$$

where

$$\theta = \theta_2 - \theta_1,$$

$$r = \frac{b}{a}$$

θ is the relative phase between the signal and the noise spectrum at ω_0; r is the ratio of the noise amplitude to the signal amplitude, as shown in Fig. 1.

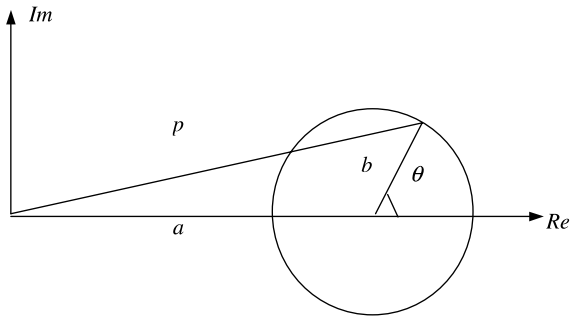

Fig. 1. An illustration of a signal with amplitude a, affected by noise with amplitude b and random phase θ.

As shown in Zhu and Alwan (2002), there is no closed-form solution for Eq. (1). Numerical solutions, however, can be computed by taking the average of a large number of uniformly distributed θ within 0–2π. Fig. 2 shows $Q(r)$ computed numerically as a function of r. $Q(r)$ lies in between $r+1$ and r, and increases slowly when r increases from 0 to 1, then it increases at a faster rate. When r is greater than 2, $Q(r)$ is approximately equal to r.

From the convex curve in Fig. 2, the most important conclusion we can draw is that, when the noise amplitude is smaller than the signal amplitude ($r < 1$), the amplitude of the observed spectrum is not far from the signal amplitude. For example if the signal amplitude is 10 and the noise amplitude is 5, the observed amplitude is expected to be 10.66. Even when the noise amplitude is the same as the signal amplitude (10), the observed amplitude is expected to be 12.76. In other words, when the linear SNR is high (higher than 1), the signal is not affected much by noise.

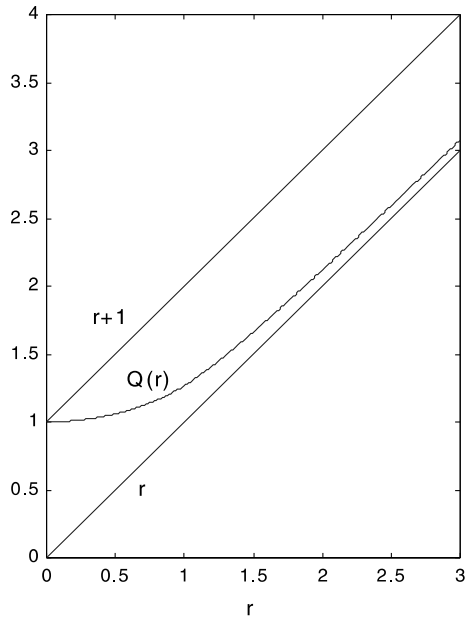

Fig. 2. Numerical solution of $Q(r)$ in the range of r from 0 to 3.

Conversely, when the noise amplitude is much higher than that of the signal, the observed amplitude is close to the noise amplitude, and increases linearly as the noise spectral amplitude increases. We will refer to this condition as "signal spectrum buried in noise".

When the amplitude of noise, b, is also a random variable, $E(p)$ would be computed as

$$E(p) = a \int_b f(b) \cdot Q(b/a)\,\mathrm{d}b,$$

where $f(b)$ is the probability density function of b. Often the variance of b is small for stationary noise, but large for non-stationary and bursty noise. The previous conclusion still holds; that is, when most of the b values are smaller than a, the effect of noise is small. In speech recognition, the logarithm is applied to simulate human perception of loudness. Thus, we have

$$\log(E(p)) = \log(a) + \log(Q(r)).$$

The logarithm operation can further decrease the log spectral change introduced to high signal amplitudes (i.e., when $Q(r)$ is close to 1, $\log(Q(r))$ is close to zero). Generally, at a certain noise level, frequencies with low energy are vulnerable to noise. For frequencies with high SNR, the effect of noise, especially after the logarithm, is obviously less dramatic.

2.2. An example of the analysis

The following example illustrates the model. Fig. 3(a) shows the amplitude spectrum of a frame from the vowel /i/, in the digit "zero" spoken by a female with computer-generated additive speech-shaped noise at 0 dB SNR. Different parts of the noise are chosen and added to the digit 150 times. For that frame, the average noisy amplitude spectrum is computed over 150 noisy frames. Note that the expectation of the noisy spectrum (computed as the average noisy spectrum) is nearly the same as the speech spectrum at harmonic peaks at the formants, where the signal spectral amplitude is about 3–7 times higher than the average noise amplitude. At frequencies where the signal spectral amplitude is low, as in the spectral valleys, the average noisy spectrum is nearly the same as the average noise spectrum itself, as shown in Fig. 3(b). The result agrees with the implication of Eq. (1).

The amplitude spectrum of a speech frame is composed of peaks and valleys. For a voiced sound, these include harmonic peaks and valleys, and formant peaks and valleys. From the example it can be seen that these valleys are vulnerable to noise (both in amplitude and general shape). The peak-to-valley ratio in the spectrum is decreased under noise due to the increase at the valleys. This will lead to a mismatch between clean and noisy speech spectra, and, hence, poor ASR results.

3. Algorithms for noise-robust feature extraction

The MFCCs are the most commonly used features for ASR. A block diagram of the different modules used in MFCC computation is shown in Fig. 4. The Mel filter bank used in our system has 26 triangular-shaped Mel filters covering 0–6250 Hz. The speech is sampled at 12 500 Hz, and a 1024-point FFT is used.

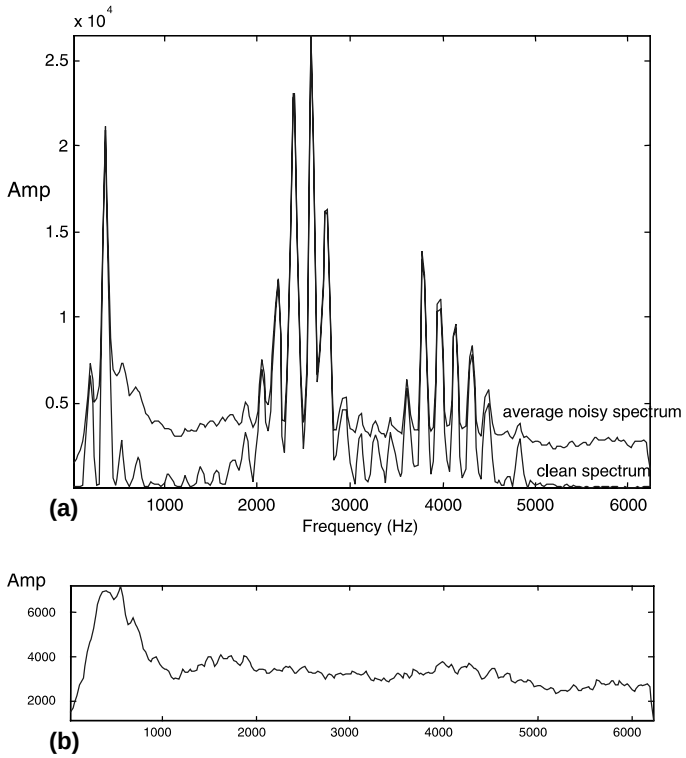

Fig. 3. The effect of noise on a speech amplitude spectrum. (a) A clean speech spectrum and the average noisy spectrum with speech-shaped noise at 0 dB SNR. (b) The average noise spectrum. Both speech and noise are passed through a first order pre-emphasis filter.

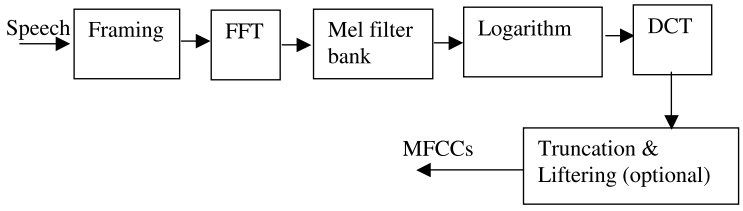

Fig. 4. A block diagram of MFCC computation.

In this section two novel algorithms, harmonic demodulation and peak-to-valley ratio locking are presented. These algorithms modify the MFCC computation for better noise robustness.

3.1. Harmonic demodulation

3.1.1. Noise effects on log Mel filter-bank output

Mel filter-bank outputs, or Mel spectra, used in MFCC computation can be affected by additive noise primarily because of the vulnerable harmonic valleys. Fig. 5 shows the first 13 Mel filters

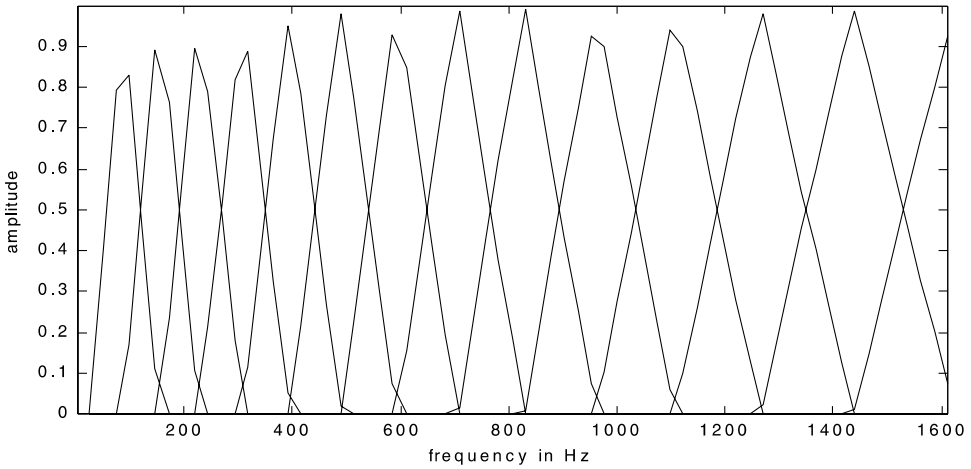

Fig. 5. The first 13 Mel filters below 1600 Hz.

covering 0–1600 Hz. Note that the filter bandwidths are around 150–300 Hz, which are somewhat comparable to the range of human pitch frequencies. Whether a filter covers a peak or a valley in the spectrum has an influence on the filter output. This means that a different pitch with the same vocal tract transfer function (VTTF) can result in different Mel filter outputs. This divergence can be handled by enough training data to cover many possible speaker conditions.

With a fixed pitch and VTTF, under noise, a Mel filter that covers mainly a spectral valley may be buried by noise and become a source of spectral mismatch. This change is difficult to compensate for by simply increasing the training data. Harmonic peaks can be about 20 dB higher than valleys. Thus, harmonic valleys should be avoided while attention should be paid to harmonic peaks, which have a higher SNR.

3.1.2. Theory of speech production and amplitude modulation

In order to avoid the spectral mismatch caused by noise at low SNR areas, and to be sure that no discriminative information is lost, we first study the speech production model. The speech spectrum contains information on both the excitation (source) signal and the VTTF. For Automatic Speech Recognition (ASR), it is important to accurately estimate the VTTF.

The linear source-filter model (Fant, 1960) of speech production views the speech waveform as the result of convolution between the excitation signal (which is either quasi-periodic, noise-like, or a combination of the two) and the impulse response of the VTTF. In the frequency domain, the speech spectrum is the result of multiplication of the source spectrum and the VTTF, as shown in Fig. 6. For voiced signals the excitation spectrum is harmonic.

The linear speech production model can also be viewed as a result of amplitude modulation (AM) in the frequency domain with the source (excitation) spectrum being a carrier and the vocal tract transfer function (VTTF) being the modulating signal. Typically, amplitude modulation refers to modulation in the time domain, where the carrier is a high frequency sinusoid and the modulating signal is a slowly varying signal.

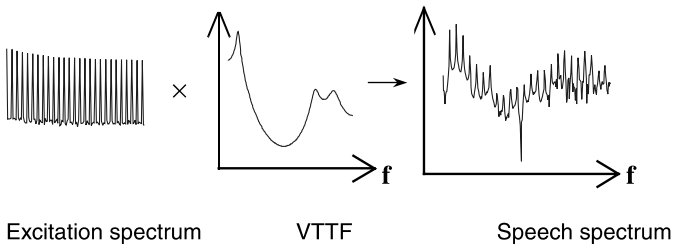

Fig. 6. The linear source-filter model of speech production in the frequency domain. The speech spectrum results from multiplying the excitation spectrum with the spectrum of the vocal tract transfer function (VTTF). The excitation spectrum in this example is harmonically related (voiced speech). The x-axis is frequency.

Notice the difference of the modulation in speech production with AM used in communication systems. In speech production, the carrier is not strictly sinusoidal, and the amplitude of the carrier is not constant. The detailed shape of the carrier also changes from speaker to speaker. Carriers in both speech production and communication systems, however, share the same property that they have a relatively slowly changing envelope, which can be modulated by the VTTF. This is also true even when the carrier is noise.

3.1.3. Demodulating the speech spectrum

Our goal is to accurately estimate the vocal tract transfer function. This is related to demodulating the speech spectrum in the frequency domain (Zhu & Alwan, 2000b).

A typical way for non-coherent amplitude demodulation used in telecommunication systems involves envelope detection using a full-wave rectifier and low-pass filter (Haykin, 1978). Fig. 7(a) illustrates the non-coherent demodulation process in the time domain after the modulated signal has been full-wave rectified. We will adopt a somewhat similar strategy but use it in the frequency domain, as shown in Fig. 7(b), where the spectrum is demodulated.

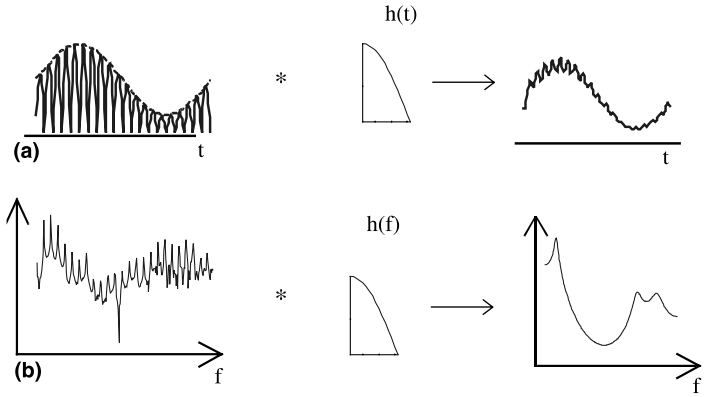

Fig. 7. (a) Envelope detector for AM demodulation. First, full-wave rectification is applied, and then the signal is convolved with the impulse response of a low-pass filter. The x-axis here is time. (b) Similar process to (a), except that it is done in the frequency domain by using the magnitude of the speech spectrum. The x-axis here is frequency.

3.1.4. Linear and non-linear envelope detection

The envelope detection method depicted in Fig. 7(b) does not ignore spectral valleys. Hence, any changes that happen at the valleys will have an impact on the resulting envelope. In this section, we introduce a non-linear envelope detection technique, to demodulate the speech spectrum, that effectively ignores valleys.

The process in Fig. 7(b) can be implemented by linear convolution between the speech spectrum and the characteristic of the low-pass filter, $h(f)$. Convolution is performed as the superposition of the convolutions between every point in the speech DFT spectrum and the response of the low-pass filter. That is

$$Y(k) = S(k) * h(k) = \sum_i [S(i)h(k-i)], \tag{2}$$

where $S(k)$ is the discrete speech spectrum, and $h(k)$ is the discrete characteristic of the low-pass filter in the frequency domain.

This process is illustrated in Fig. 8. Fig. 8(a) is a simplified representation of a speech spectrum. The highest spikes are the harmonic peaks. The other samples represent inter-harmonic frequency components, which, for simplicity, are assumed to be either 0 or half the amplitude of the harmonics. For simplicity, the envelope of the harmonics is flat in this example. We will illustrate how to recover this envelope. Fig. 8(b) shows an example of the characteristic of the low-pass filter, $h(k)$. We first compute the convolution between each point in the speech spectrum with the

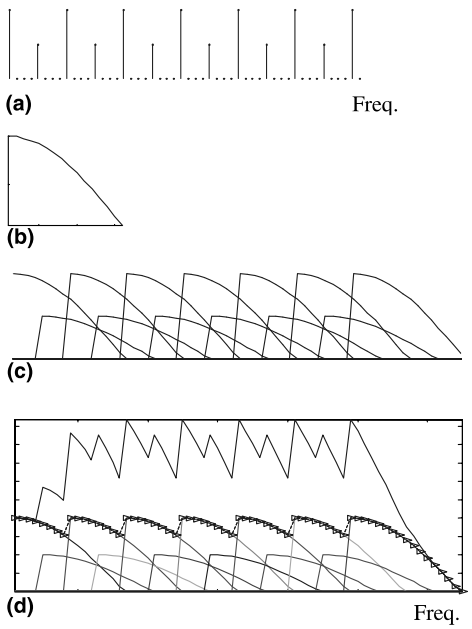

Fig. 8. Envelope detection. (a) A simplified speech spectrum. (b) The response of the low-pass filter, $h(k)$, for envelope detection. (c) Results of the convolution between every point in (a) and (b). (d) The envelope estimated by linear demodulation (LED) is shown as the solid line in the upper part of the figure, and the envelope detected using NLED is shown as the line with triangles.

characteristic of the low-pass filter, as shown in Fig. 8(c). The convolution of the whole spectrum with the filter is then obtained as the superposition of the results in Fig. 8(c), and is shown in Fig. 8(d) as the top solid line.

One problem of linear envelope detection (LED) using Eq. (2) is that it is vulnerable to inter-harmonic components through the step of superposition; hence it would not be robust to background noise. Alternatively, one can perform envelope detection in a way that is less susceptible to inter-harmonic components. We achieve this by a non-linear envelope detection technique, hereafter referred to as NLED, which effectively estimates the envelope by focusing only on spectral peaks. Instead of computing the superposition of the results in Fig. 8(c), we compute the maximum with the equation:

$$Y(k) = \max_i [S(i)h(k-i)]. \tag{3}$$

The result of demodulation using this non-linear technique for the same input spectrum is shown Fig. 8(d), as the line with triangles.

3.2. Robustness analysis

The non-linear envelope detection using Eq. (3) can ignore changes at harmonic valleys. We illustrate this with an example. Fig. 9(a) shows the same simplified speech spectrum as Fig. 8(a), except that one inter-harmonic point has increased in amplitude due to background noise (indicated by an arrow in Fig. 9(a)). With the linear envelope detection technique, this change will affect the output, as seen in the top line in Fig. 9(b). If the non-linear envelope detection is used, the output is not affected, as seen in Fig. 9(b) (line with triangles) as the change at inter-harmonic frequencies is "masked" by peaks around it.

Obviously, if the increase at inter-harmonic frequencies results in amplitudes that are higher than nearby harmonic peaks, then both the linear and non-linear techniques will not be effective. This often happens at prominent formant valleys in speech spectra.

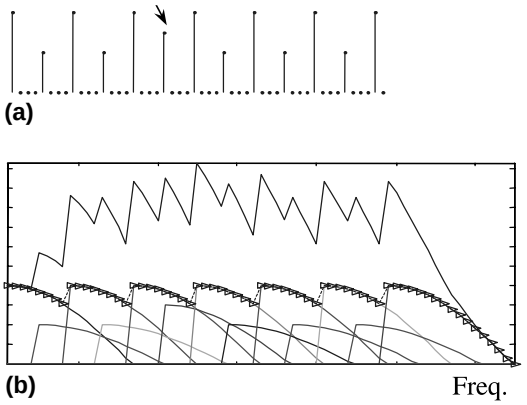

Fig. 9. (a) A simplified speech spectrum (the modulated signal) with additive noise at one point indicated by an arrow. (b) The demodulated signal. The top solid line is the envelope detected by linear convolution while the lower line (with triangles) results from NLED.

3.2.1. Noise flooring

We set a threshold for envelope detection such that values below the threshold are set to that threshold. The threshold is determined as the average spectral amplitude across all frequencies of a frame multiplied by an experimentally determined factor. For the TI46 database (an isolated digit database) and the example in this section, the factor is 0.4.

Figs. 10 and 11 show examples of these procedures of NLED and noise flooring. The speech frame is the /i/ part of the digit zero, spoken by a female, the same as that in Fig. 3. Fig. 10 shows the DFT spectra and envelopes detected with NLED for a clean and noisy speech frame with speech-shaped noise at 5 dB SNR. Note that the envelope is determined by spectral peaks only,

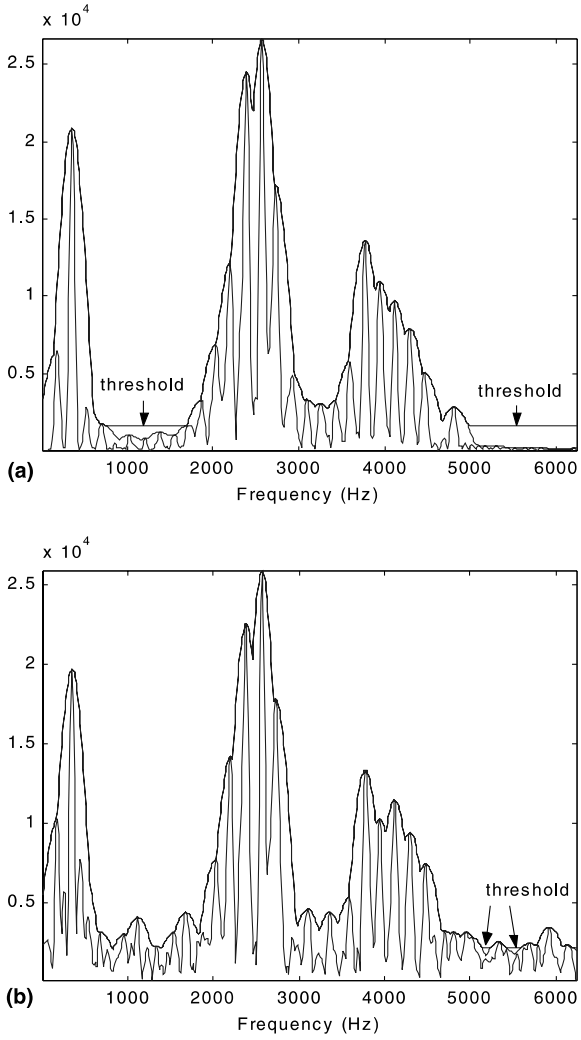

Fig. 10. (a) The spectrum of the same frame in Fig. 3 (clean), and the envelope detected with NLED with a noise flooring threshold. (b) The spectrum of the same frame under 5 dB speech-shaped noise, and the envelope detected with NLED, with a noise flooring threshold.

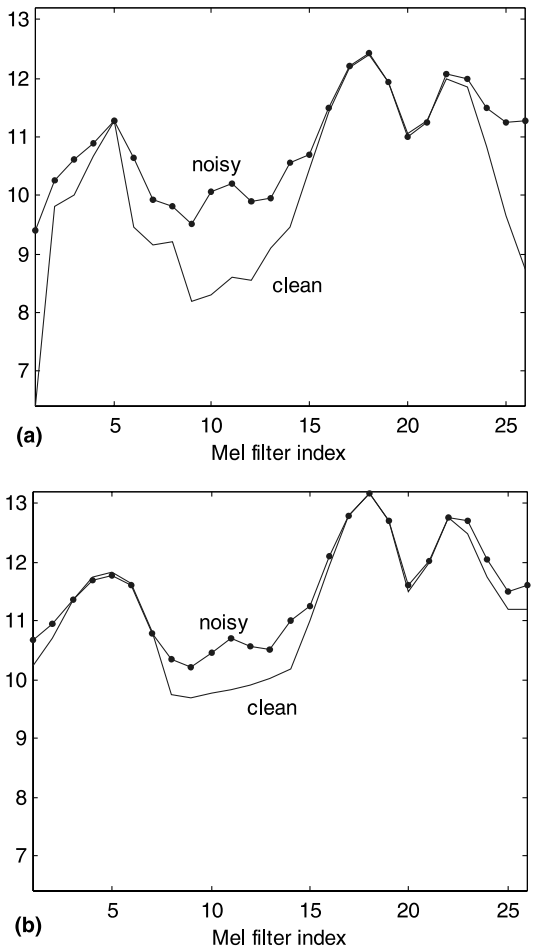

Fig. 11. (a) Log Mel filter output of the original speech spectrum for the clean and 5 dB noisy speech. (b) Log Mel filter output of the envelope detected with NLED with noise flooring for the clean and 5 dB noisy speech.

and the noise flooring threshold is effective mainly at prominent formant valleys. Fig. 11 compares the log Mel filter outputs with and without NLED and noise flooring for clean and noisy speech spectra. NLED mainly decreases the difference at formant peaks, especially the Mel filter output covering the first formant. In this example we can see that for a Mel filter index less than 7, NLED helps decrease the difference between the filter output of clean and noisy frames. Noise flooring helps decrease the difference between the outputs of filters 8–14 and 23–26, which cover prominent formant valleys.

3.3. Peak-to-valley ratio locking

3.3.1. The algorithm

With harmonic demodulation, the vulnerable inter-harmonic valleys are avoided, but not all formant valleys are. Noise flooring alleviates some of the mismatch caused by the increase at the

valleys. In Section 2 we showed that because of the difference in the noise effects on formant valleys and peaks, often the ratio of the highest peak to the lowest valley in the spectrum, i.e., the peak-to-valley ratio, in a noisy speech spectrum is lower than that in a clean speech spectrum, hence leading to a spectral mismatch.

The aim of this section is to keep the peak-to-valley ratio unchanged under noise by introducing a technique we refer to as "peak-to-valley ratio locking". The technique operates on the log Mel filter output recovered from the liftered MFCCs (Juang, Rabiner, & Wilpon, 1987). Differences in peak-to-valley ratio of the linear spectrum lead to differences in the range of the log Mel filter output. MFCCs are computed as the discrete cosine transform (DCT) of the Mel filter-bank outputs. Thus, applying inverse DCT (IDCT) on the MFCCs before liftering can go back to the log Mel filter output. As the first MFCC coefficient (C_0) is not included in the liftered MFCCs, after an IDCT on the liftered MFCCs, the average of the recovered log Mel filter output is about zero, so that formant peaks remain positive and formant valleys become negative. Fig. 12(a)

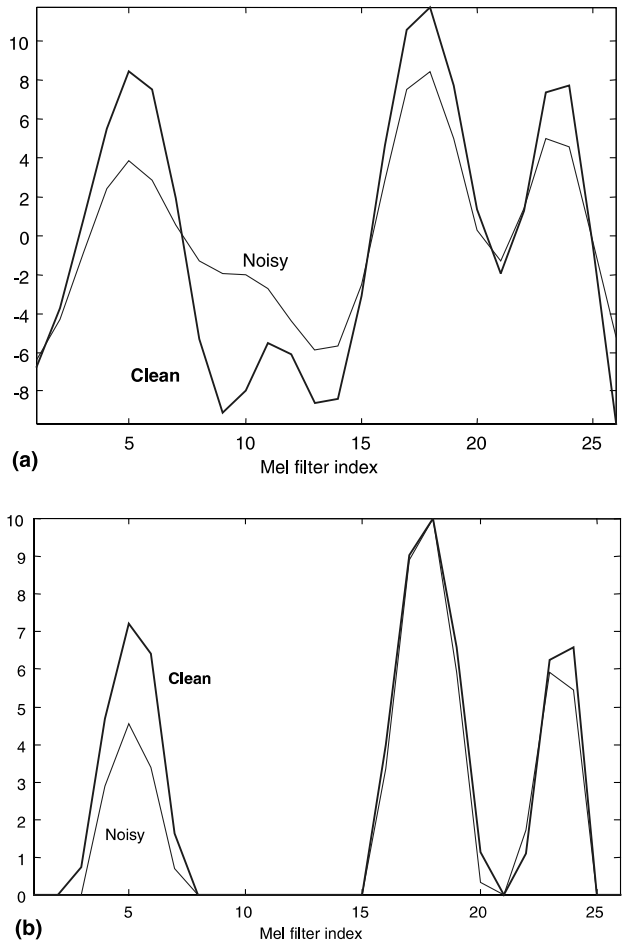

Fig. 12. (a) Log Mel filter-bank output recovered from liftered MFCCs for a clean and noisy (0 dB SNR) frames of /i/. (b) Log Mel filter-bank output after peak-to-valley ratio locking and PKISO for the same clean and noisy frames in (a).

shows an example of the log Mel spectra of a clean frame of /i/ and that of the frame with speech-shaped noise at 0 dB SNR, recovered with an IDCT from liftered MFCCs. For the noisy frame, the range from the highest peak to the lowest valley is less than that of the clean frame.

An algorithm for peak-to-valley ratio locking can be designed as follows:

$$x = \max_m D(m),$$

$$D'(m) = D(m)/x \cdot \alpha \quad (\text{for each } m),$$

where $D(m)$ is the log Mel spectrum recovered from the liftered MFCC vector of a frame, m is the filter index, x is the height of the highest peak in $D(m)$, $D'(m)$ is the log Mel spectrum after peak-to-valley ratio locking, and α is a fixed number. Thus the highest peak in the log Mel spectra is always set to α for both clean and noisy frames, and the peak-to-valley range in $D'(m)$ is about 2α. In our implementation α was set to 6 for the TI46 database and to 10 for the Aurora 2 database, which is determined roughly as the average of the highest recovered spectral peaks. A DCT is then applied to the processed log Mel spectrum, $D'(m)$, to obtain MFCCs again (Zhu, Iseli, Cui, & Alwan, 2001).

3.3.2. Combining peak-to-valley ratio locking and peak isolation

Peak-to-valley ratio locking can be implemented together with peak isolation (Strope & Alwan, 1997). Peak isolation is a noise-robust front-end processing which first computes the log Mel spectra recovered from liftered MFCCs by taking the IDCT, then half-wave rectification is applied. A DCT is then applied to obtain peak isolated MFCCs.

Peak isolation (PKISO) was inspired by the fact that formant peaks are important perceptual cues and valleys are not. Half-wave rectification removes formant valleys. With the model in Section 2, we know that the shape of the spectral valleys can be altered to the shape of the noise at low SNRs, as seen in Fig. 12(a) at the valley between the first and second formants. PKISO can effectively prevent the mismatch in the shape of formant valley areas by setting valleys to zero.

Both peak isolation and peak-to-valley ratio locking operate on log Mel spectra recovered from the IDCT of the liftered MFCCs. When combining the two, only the positive part in the recovered Mel filter output is scaled, while the negative part is set to zero. An example of the result of combining PKISO with peak-to-valley ratio locking is shown in Fig. 12(b), which uses the same frame shown in Fig. 12(a). Note that the clean and noisy spectra are close at the highest formant and at the valleys. Differences remain at other frequencies.

4. Recognition experiments

Two sets of HMM-based ASR experiments are performed with the new algorithms using HTK2.2. The first is an isolated digit recognition experiment using the TI46 database. The second is a connected digit recognition task using the Aurora 2 database (Hirsch & Pearce, 2000).

The harmonically demodulated MFCCs (HDMFCC) are computed using linear envelope detection (LED) and the non-linear envelope detection (NLED) introduced in this paper. For the TI46 database, speech is sampled at 12.5 kHz. Frame length is 25 ms, and the overlap is 15 ms. For each frame, a 1024-point FFT is computed, and only half the points are used because of the

FFT symmetry. The characteristic of the low-pass filter, $h(k)$, used in envelope detection is shown in Fig. 13. The shape of $h(k)$ is sinusoidal, and the width is 43 points, which corresponds to 525 Hz. The width of $h(k)$ of the filter was optimized to achieve high accuracy in speech recognition experiments.

For the Aurora 2 database, speech is sampled at 8 kHz, and a 512-point FFT is used. $h(k)$ has the same shape but the width is 13, which corresponds to 203 Hz.

For the first set of experiments, an HMM with 4 states and 2 mixtures is trained from 160 utterances spoken by 16 talkers (8 males and 8 females) for each digit. Training includes 2 steps of Maximum Likelihood (ML) and Expectation Maximization (EM) with 4 iterations each. A Viterbi algorithm is used for recognition using 480 different utterances. Training is done with clean signals, and recognition with noisy signals at different SNRs. The noise is additive speech-shaped noise. Both training and testing utterances have been processed first so that silence is not included. Pre-emphasis with a first order FIR filter is used.

The features evaluated in these experiments include MFCCs, and MFCCs enhanced with various modifications introduced in this paper. The first and second derivatives are used. The results are shown in Table 1, where FL refers to noise flooring and locking refers to peak-to-valley ratio locking. Clearly, each method (harmonic demodulation, peak-to-valley ratio locking, and PKISO) improves recognition performance. When LED is used with MFCCs, the

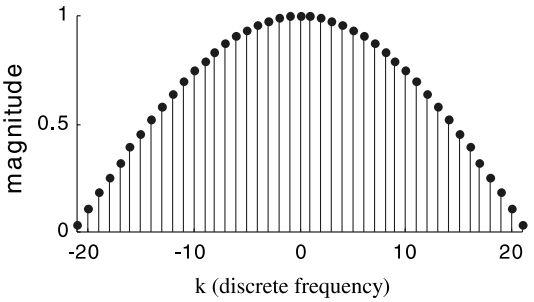

Fig. 13. $h(k)$ used in envelope detection with the TI46 database.

Table 1
Recognition results in percentage correct for the TI-46 database with additive speech-shaped noise and different front-end processing methods

SNR	0 dB	3 dB	5 dB	10 dB	20 dB	Average
MFCC	25.00	40.21	52.29	85.83	98.96	60.46
LED	28.33	45.83	57.71	83.12	98.12	62.62
Noise flooring (FL)	48.33	67.92	78.33	91.46	99.17	77.04
NLED	37.50	54.58	68.12	89.58	98.75	69.70
NLED + FL	49.17	70.62	82.92	94.79	98.75	79.25
PKISO	45.00	64.58	76.46	94.79	99.79	76.12
PKISO + NLED + FL	64.38	80.21	85.42	94.17	98.96	84.62
Locking	45.83	65.83	79.38	95.21	99.38	77.13
PKISO + locking	75.00	88.54	93.33	98.33	99.58	90.96
PKISO + locking + NLED	88.12	93.12	96.04	98.54	98.54	94.87

recognition rate is improved a little. When NLED is used, the average recognition rate increases from 60.46% to 69.70%. When noise flooring (FL) is used together with NLED, the average recognition rate reaches 79.25%. Peak-to-valley locking (Locking) itself improves the performance to 77.13%. Best results are obtained with the three techniques (NLED, Locking and PKISO) combined together, where the average recognition rate achieves 94.87%. Compared with the average recognition rate using plain MFCCs, at 60.46%, these new algorithms clearly improve noise robustness.

For the Aurora 2 experiments, training and testing follow the specifications described in (Hirsch & Pearce, 2000). A word-based ASR system for digit string recognition where each HMM word model has 16 emitting states is adopted. A three-state silence model and a one-state short pause model are used. Training is done with 8440 clean utterances from 55 male and 55 female adults. The variances in the silence model are increased by a factor of 1.2 empirically after training, because the variance the silence segments of noisy data is often higher than that of the clean training data. Testing data include eight types of realistic background noise (subway, babble, car, exhibition hall, restaurant, street, airport and train station noise) at various SNRs (clean, 20, 15, 10, 5, 0, and −5 dB). There are two test sets. Set A contains the first four types of noise and Set B contains the other four. Each of the two test sets has 4004 utterances.

The baseline front-end used is MFCC_E_D_A, which contains 12 MFCCs and log energy, and their first and second derivatives. Each feature vector thus contains 39 components. Baseline recognition results are computed with the program FE2.0, which is included with the Aurora 2 database CD.

Most algorithms discussed in this paper assume that the underlying frame is a speech frame. Thus, a speech detector is needed for the Aurora 2 database. The new algorithms are only applied to speech frames, while standard MFCCs are computed for non-speech frames.

Speech/non-speech detection is based on energy and voicing. The general idea is that if a frame has high energy and is a voiced frame or close to voiced frames, then the frame is classified as speech. The energy condition is implemented by comparing the log energy of a frame to a threshold for the corresponding utterance. A threshold (T) for an utterance is determined empirically by $T = 0.5(H + L)$, where H and L are the average of the 10 highest and 10 lowest log energy values in the utterance, respectively. Voicing detection is performed with the open source software from the SVR group at Cambridge University, Pitch_tracker 1.0, which is based on (Medan, Yair, & Chazan, 1991). If the log energy of a frame is higher than a threshold and if it is voiced or it is within 30 ms of a voiced frame, then the frame is classified as a speech frame and the algorithms are applied.

Tables 2–6 show ASR results (word accuracy) of MFCCs and MFCCs with the new algorithms, and error rate reduction compared to baseline results. Note that the average accuracy and relative error rate reduction are computed with the results between 20 and 0 dB, as suggested in (Hirsch & Pearce, 2000) for the Aurora 2 database. The average recognition results for Set A are improved from 61.34% to 72.31% with harmonic demodulation and noise flooring, to 75.77% with peak-to-valley ratio locking, and to 81.56% with harmonic demodulation with noise flooring, peak-to-valley ratio locking, and PKISO. For Set B, similar improvement is observed from 55.75% to 79.51%. The average relative error rate reductions with all the algorithms combined are 52.29% and 53.71% for Sets A and B, respectively.

Table 2
Baseline recognition results with MFCCs for the Aurora 2 database

	A					B				
	Subway	Babble	Car	Exhibition	Average	Restaurant	Street	Airport	Station	Average
Clean	98.93	99.00	98.96	99.20	99.02	98.93	99.00	98.96	99.20	99.02
20 dB	97.05	90.15	97.41	96.39	95.25	89.99	95.74	90.64	94.72	92.77
15 dB	93.49	73.76	90.04	92.04	87.33	76.24	88.45	77.01	83.65	81.34
10 dB	78.72	49.43	67.01	75.66	67.71	54.77	67.11	53.86	60.29	59.01
5 dB	52.16	26.81	34.09	44.83	39.47	31.01	38.45	30.03	27.92	31.93
0 dB	26.01	9.28	14.46	18.05	16.95	10.96	17.84	14.41	11.57	13.70
−5 dB	11.18	1.57	9.39	9.60	7.94	3.47	10.46	8.23	8.45	7.65
Average	69.49	49.89	60.60	65.39	**61.34**	52.59	61.52	53.25	55.63	**55.75**

Table 3
Recognition and relative error rate reductions results for MFCCs with harmonic demodulation and noise flooring (Sets A and B)

	A					B				
	Subway	Babble	Car	Exhibition	Average	Restaurant	Street	Airport	Station	Average
Clean	98.71	98.85	98.39	98.98	98.73	98.71	98.85	98.39	98.98	98.73
20 dB	97.64	96.92	97.32	97.13	97.25	97.18	97.58	96.93	97.38	97.27
15 dB	95.92	91.75	95.20	95.53	94.60	93.68	95.19	92.72	94.75	94.09
10 dB	88.33	74.67	85.00	88.24	84.06	78.29	85.94	78.56	83.52	81.58
5 dB	67.49	45.19	55.03	64.67	58.10	52.20	61.09	49.57	53.96	54.21
0 dB	37.52	19.47	24.75	28.45	27.55	22.84	30.65	23.53	23.30	25.08
−5 dB	14.40	6.38	8.77	8.67	9.56	7.83	10.67	8.29	8.67	8.87
Average	77.38	65.60	71.46	74.80	**72.31**	68.84	74.09	68.26	70.58	**70.44**
Relative (%)	25.87	31.36	27.56	27.19	**28.37**	34.27	32.67	32.11	33.70	**33.21**

Table 4
Recognition and relative error rate reductions results for MFCCs with peak-to-valley ratio locking (Sets A and B)

	A					B				
	Subway	Babble	Car	Exhibition	Average	Restaurant	Street	Airport	Station	Average
Clean	98.46	98.25	98.15	98.52	98.35	98.46	98.25	98.15	98.52	98.35
20 dB	96.41	96.89	97.35	96.54	96.80	97.27	96.55	97.05	97.53	97.10
15 dB	93.12	95.44	94.96	91.70	93.81	93.85	94.41	93.65	94.32	94.08
10 dB	85.05	89.78	88.73	81.58	86.29	81.95	87.64	86.28	84.70	85.14
5 dB	69.36	67.59	74.56	57.91	67.36	52.84	67.32	67.79	66.62	63.64
0 dB	39.58	26.36	41.99	30.45	34.60	19.74	39.39	33.07	36.56	32.19
−5 dB	15.17	8.86	15.45	12.06	12.89	7.06	14.90	9.28	9.56	10.20
Average	76.70	75.21	79.52	71.64	**75.77**	69.15	77.06	75.57	75.95	**74.43**
Relative (%)	23.65	50.54	48.01	18.04	**37.32**	34.92	40.39	47.74	45.79	**42.22**

Table 5
Recognition and relative error rate reductions results for MFCCs with harmonic demodulation, noise flooring and peak-to-valley ratio locking (Sets A and B)

	A					B				
	Subway	Babble	Car	Exhibition	Average	Restaurant	Street	Airport	Station	Average
Clean	98.40	97.97	97.91	98.43	98.18	98.40	97.97	97.91	98.43	98.18
20 dB	97.08	97.22	97.26	97.28	97.21	97.24	96.95	96.72	97.81	97.18
15 dB	95.18	96.13	96.03	94.48	95.46	94.72	95.62	94.42	95.53	95.07
10 dB	88.36	91.87	92.42	87.47	90.03	84.49	90.69	88.22	87.94	87.84
5 dB	73.63	71.49	81.60	67.45	73.54	57.63	73.73	72.35	72.26	68.99
0 dB	43.87	29.56	48.73	38.20	40.09	25.08	45.65	38.92	43.13	38.20
−5 dB	15.90	9.52	16.02	13.88	13.83	8.81	17.20	12.29	11.63	12.48
Average	79.62	77.25	83.21	76.98	**79.27**	71.83	80.53	78.13	79.33	**77.46**
Relative (%)	33.22	54.61	57.38	33.47	**46.36**	40.58	49.40	53.21	53.42	**49.05**

Table 6
Recognition and relative error rate reductions results for MFCCs with harmonic demodulation, noise flooring, peak-to-valley ratio locking, and PKISO (Sets A and B)

	A					B				
	Subway	Babble	Car	Exhibition	Average	Restaurant	Street	Airport	Station	Average
Clean	98.19	98.13	97.82	98.33	98.12	98.19	98.13	97.82	98.33	98.12
20 dB	96.99	97.25	97.32	96.88	97.11	97.24	96.86	97.05	97.62	97.19
15 dB	95.46	95.80	96.48	94.29	95.51	95.00	95.31	94.78	95.25	95.09
10 dB	88.98	92.99	93.38	89.08	91.11	84.99	91.44	88.97	88.89	88.57
5 dB	73.66	76.39	85.68	73.37	77.28	60.42	78.63	74.92	75.04	72.25
0 dB	43.35	35.07	62.57	46.16	46.79	27.26	52.57	46.64	51.40	44.47
−5 dB	14.92	5.99	19.74	13.79	13.61	1.44	19.01	11.18	14.44	11.52
Average	79.69	79.50	87.09	79.96	**81.56**	72.98	82.96	80.47	81.64	**79.51**
Relative (%)	33.43	59.09	67.22	42.08	**52.29**	43.01	55.72	58.23	58.62	**53.71**

5. Complexity considerations

Even though our implementation did not optimize for computational cost (but focused on optimizing recognition performance), the increase in computational complexity of PKISO, harmonic demodulation and peak-to-valley ratio locking together is not high.

For example, for the Aurora 2 database, the extra operations introduced for processing one frame in harmonic demodulation mainly comes from $13 * 256$ (13 is the length of the filter characteristics, $h(k)$, and 256 is half the FFT size) floating number multiplications. The extra operations introduced by PKISO and peak-to-valley ratio locking mainly come from the extra IDCT and DCT, which contain $23 * 12$ (23 is the number of the Mel filters and 12 is the length of the MFCC vector) floating number multiplications each.

When tested on a Sun Ultra Sparc 60 workstation, the computational load of PKISO and peak-to-valley ratio locking together is less than 4% of the total computation time of the original front-end (FE2.0) executable. Harmonic demodulation adds about 20% more computational time. These measurements only count the front-end computation time, excluding the disk I/O time. In addition, these algorithms are frame-based and thus do not introduce a delay. Speech/non-speech detection introduces a delay, the size of the utterance analyzed, when finding the log energy threshold.

6. Summary and discussion

In this paper, a quantitative analysis on how noise affects speech amplitude spectrum is presented. The analysis quantifies the vulnerability of spectral valleys to noise. Algorithms are then designed to avoid mismatch between clean and noisy spectra by relying on the noise-robust parts without loss of discriminative information, and without significant increase in computational complexity. The algorithms help prevent the mean shift of noisy features from the HMMs trained with clean data.

Harmonic demodulation is designed to estimate a spectral envelope based on harmonic peaks and, by doing so, avoids noise effects at harmonic valleys. Noise flooring is used to alleviate the problem with formant valleys, where a threshold is set to the detected envelope. When formant valleys are buried by noise, the peak-to-valley ratio of the spectrum decreases. The peak-to-valley ratio locking algorithm is designed to alleviate this type of *amplitude* mismatch. The previously studied method of peak isolation (Strope & Alwan, 1997) addresses the issue of mismatch in the *shape* of formant valleys.

With the comparative analysis of speech production and demodulation, we show that harmonic demodulation does not remove information about the VTTF. There is no loss of information in peak-to-valley ratio locking as it just scales the log Mel spectra. In PKISO, the formant valleys, which are not important for identifying speech sounds, are set to zero.

These algorithms differ from the noise removal or feature cleaning algorithms. The goal is not to remove noise effects after a vulnerable feature corrupted by noise is computed. Instead, the feature itself is more robust to noise. No noise estimate is needed, and the algorithms utilize knowledge of the underlying speech spectrum. Further, the algorithms do not assume stationary noise, and hence are effective in dealing with non-stationary additive noise as well.

One question that can be asked is: do the proposed algorithms sacrifice the discriminative power for noise robustness for a larger vocabulary ASR task? Generally, formant locations and shapes, and the relative heights of the formants carry discriminative information for articulation and perception, and this information is not lost in the proposed algorithms. Recognition experiments beyond the framework of digit string recognition might be necessary to verify this conclusion.

Acknowledgements

Work supported in part by NSF, STM, Broadcom, HRL and Mindspeed together with the state of California through the University of California MICRO program.

References

Boll, S.F., 1979. Suppression of acoustic noise in speech using spectral subtraction. IEEE Transactions on Acoustics, Speech and Signal Processing 27, 113–120.

Davis, S.B., Mermelstein, P., 1980. Comparison of parametric representations for monosyllabic world recognition in continuously spoken sentences. IEEE Transactions on Acoustics, Speech and Signal Processing 28, 357–366.

Deng, L., Droppo, J., Acero, A., 2002. A Bayesian approach to speech feature enhancement using the dynamic cepstral prior. Proceedings of the International Conference on Acoustics, Speech and Signal Processing 1, 829–832.

Ephraim, Y., Trees, H.L., 1995. A signal subspace approach for speech enhancement. IEEE Transactions on Speech and Audio Processing 3, 257–266.

Fant, G., 1960. The Acoustic Theory of Speech Production. S'Gravenhage, Mouton.

Furui, S., 1981. Cepstral analysis technique for automatic speaker verification. IEEE Transactions on Acoustics, Speech and Signal Processing 29, 254–272.

Gales, M.J.F., 2002. Maximum likelihood multiple subspace projections for hidden Markov models. IEEE Transactions on Speech and Audio Processing 10, 37–47.

Gales, M.J.F., Young, S.J., 1996. Robust continuous speech recognition using parallel model combination. IEEE Transactions on Speech and Audio Processing 4, 352–359.

Gauvain, J.L., Lee, C.H., 1994. Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains. IEEE Transactions on Speech and Audio Processing 2, 291–298.

Greenberg, S., Kingsbury, B., 1997. The modulation spectrogram: in pursuit of an invariant representation of speech. Proceedings of the International Conference on Acoustics, Speech and Signal Processing 3, 1647–1650.

Haeb-Umbach, R., Ney, H., 1992. Linear discriminant analysis for improved large vocabulary continuous speech recognition. Proceedings of the International Conference on Acoustics, Speech and Signal Processing 1, 13–16.

Haykin, S., 1978. Communication Systems. Wiley, New York.

Hermansky, H., 1990. Perceptual linear predictive (PLP) analysis of speech. Journal of Acoustical Society of America 87, 1738–1952.

Hermansky, H., Morgan, N., 1994. RASTA processing of speech. IEEE Transactions on Speech and Audio Processing 2, 578–589.

Hirsch, H.G., Pearce, D., 2000. The AURORA experimental framework for the performance evaluations of speech recognition systems under noisy condition. Automatic speech recognition: challenges for the next millennium.

Juang, B., Rabiner, L., Wilpon, J., 1987. On the use of bandpass liftering in speech recognition. IEEE Transactions on Acoustics, Speech and Signal Processing 35, 947–954.

Kajita, S., Itakura, F., 1994. Speech analysis and speech recognition using subband-autocorrelation analysis. Journal of the Acoustical Society 15, 329–338.

Kanedera, N., Hermansky, H., Arai, T., 1998. On properties of modulation spectrum for robust automatic speech recognition. Proceedings of the International Conference on Acoustics, Speech and Signal Processing 2, 613–616.

Kocsor, A., Toth, L., Kuba, A., Kovacs, K., Jelasity, M., Gyimothy, T., Csirik, J., 2000. A comparative study of several feature transformation and learning methods for phoneme classification. International Journal of Speech Technology 3, 263–276.

Leggetter, C., Woodland, P., 1994. Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models. Proceedings of International Conference on Spoken Language Processing 2, 451–454.

Lieb, M., Haeb-Umbach, R., 2000. LDA derived cepstral trajectory filters in adverse environmental conditions. Proceedings of the International Conference on Acoustics, Speech and Signal Processing 2, 105–108.

Lockwood, P., Boudy, J., 1992. Experiments with a nonlinear spectral subtractor (NSS), hidden Markov models and the projection of robust speech recognition in cars. Speech Communication 11, 215–228.

Medan, Y., Yair, E., Chazan, D., 1991. Super resolution pitch determination of speech signals. IEEE Transactions on Signal Processing 39, 40–48.

Milner, B., 1996. Inclusion of temporal information into features for speech recognition. Proceedings of International Conference on Spoken Language Processing 1, 256–259.

Paliwal, K.K., Lilly, B.T., 1997. Auditory masking based acoustic front-end for robust speech recognition. Proceedings of IEEE TENCON 1, 165–168.

Poter, J.E., Boll, S.F., 1984. Optimal estimators for spectral restoration of noisy speech. Proceedings of International Conference on Spoken Language Processing 3, 18A.2.1.

Renevey, P., Drygajlo, A., 2000. Statistical estimation of unreliable features for robust speech recognition. Proceedings of the International Conference on Acoustics, Speech and Signal Processing 3, 1731–1734.

Saon, G., Padmanabhan, M., Gopinath, R., Chen, S., 2000. Maximum likelihood discriminant feature spaces. Proceedings of the International Conference on Acoustics, Speech and Signal Processing 2, 1129–1132.

Sharma, S., Ellis, D., Kajarekar, S., Hermansky, H., 2000. Feature extraction using nonlinear transformation for robust speech recognition on the Aurora database. Proceedings of the International Conference on Acoustics, Speech and Signal Processing 2, 1117–1120.

Strope, B., Alwan, A., 1997. A model of dynamic auditory perception and its application to robust word recognition. IEEE Transactions on Speech and Audio Processing 5, 451–464.

Strope, B., Alwan, A., 1998. Robust word recognition using threaded spectral peaks. Proceedings of the International Conference on Acoustics, Speech and Signal Processing 2, 625–629.

Virag, N., 1999. Single channel speech enhancement based on masking properties of the human auditory system. IEEE Transactionson Speech and Audio Processing 7, 126–137.

Zhu, Q., Alwan, A., 2000a. On the use of variable frame rate analysis in speech recognition. Proceedings of the International Conference on Acoustics, Speech and Signal Processing 3, 1783–1786.

Zhu, Q., Alwan, A., 2000b. Amplitude demodulation of speech spectra and its application to noise robust speech recognition. Proceedings of International Conference on Spoken Language Processing 1, 341–344.

Zhu, Q., Alwan, A., 2002. The effect of additive noise on speech amplitude spectra: a quantitative approach. The IEEE Signal Processing Letters 9 (9), 275–277.

Zhu, Q., Iseli, M., Cui, X., Alwan, A., 2001. Noise robust front-end feature extraction for automatic speech recognition. Proceedings Eurospeech 2001 1, 185–188.