We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

6,900
Open access books available

186,000
International authors and editors

200M
Downloads

154
Countries delivered to

Our authors are among the
TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK
CITATION
INDEX
INDEXED

WEB OF SCIENCE™
Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Open Access Database www.i-techonline.com

14

Autocorrelation-based Methods for Noise-Robust Speech Recognition

Gholamreza Farahani, Mohammad Ahadi and
Mohammad Mehdi Homayounpour
Amirkabir University of Technology
Iran

1. Introduction

One major concern in the design of speech recognition systems is their performance in real environments. In such conditions, different sources could exist which may interfere with the speech signal. The effects of such sources could generally be classified as additive noise and channel distortion. As the names imply, noise is usually considered as additive in spectral domain while channel distortion is multiplicative and therefore appears as an additive part in logarithmic spectrum. These could both result in severe performance degradations in *automatic speech recognition* (ASR) systems. Thus, in recent years, a substantial amount of research has been devoted to improving the performance of Automatic Speech Recognition (ASR) Systems in such environments.

The main approaches taken to improve the performance of ASR systems could be roughly divided into three main categories, namely, robust speech feature extraction; speech enhancement and model-based compensation for noise.

The main goal of the robust speech feature extraction techniques is to find a set of parameters, to represent speech signal in the ASR system, that are robust against the variations in the speech signal due to noise or channel distortions. Extensive research has resulted in such well-known techniques as RASTA filtering (Hermansky & Morgan, 1994), *cepstral mean normalization* (CMN) (Kermorvant, 1999), use of dynamic spectral features (Furui, 1986), *short-time modified coherence* (SMC) (Mansour & Juang, 1989a) and also *one-sided autocorrelation LPC* (OSALPC) (Hernando & Nadeu, 1997), *differential power spectrum* (DPS) (Chen et al., 2003) and *relative autocorrelation sequence* (RAS) (Yuo & Wang, 1998, 1999).

In the case of speech enhancement, some initial information about speech and noise is needed to allow the estimation of noise and clean up of the noisy speech. Widely used methods in this category include *spectral subtraction* (SS) (Beh & Ko, 2003; Boll, 1979) and Wiener filtering (Lee et al., 1996).

In the framework of model-based compensation, statistical models such as Hidden Markov Models (HMMs) are usually considered. The compensation techniques try to remove the mismatch between the trained models and the noisy speech to improve the performance of ASR systems. Methods such as *parallel model combination* (PMC) (Gales & Young, 1995, 1996), *vector Taylor series* (VTS) (Acero et al., 2000; Kim et al., 1998; Moreno, 1996; Moreno et al.,

Source: Robust Speech Recognition and Understanding, Book edited by: Michael Grimm and Kristian Kroschel,
ISBN 987-3-90213-08-0, pp.460, I-Tech, Vienna, Austria, June 2007

1996; Shen et al., 1998) and *weighted projection measure* (WPM) (Mansour & Juang, 1989b) can be classified into this category.

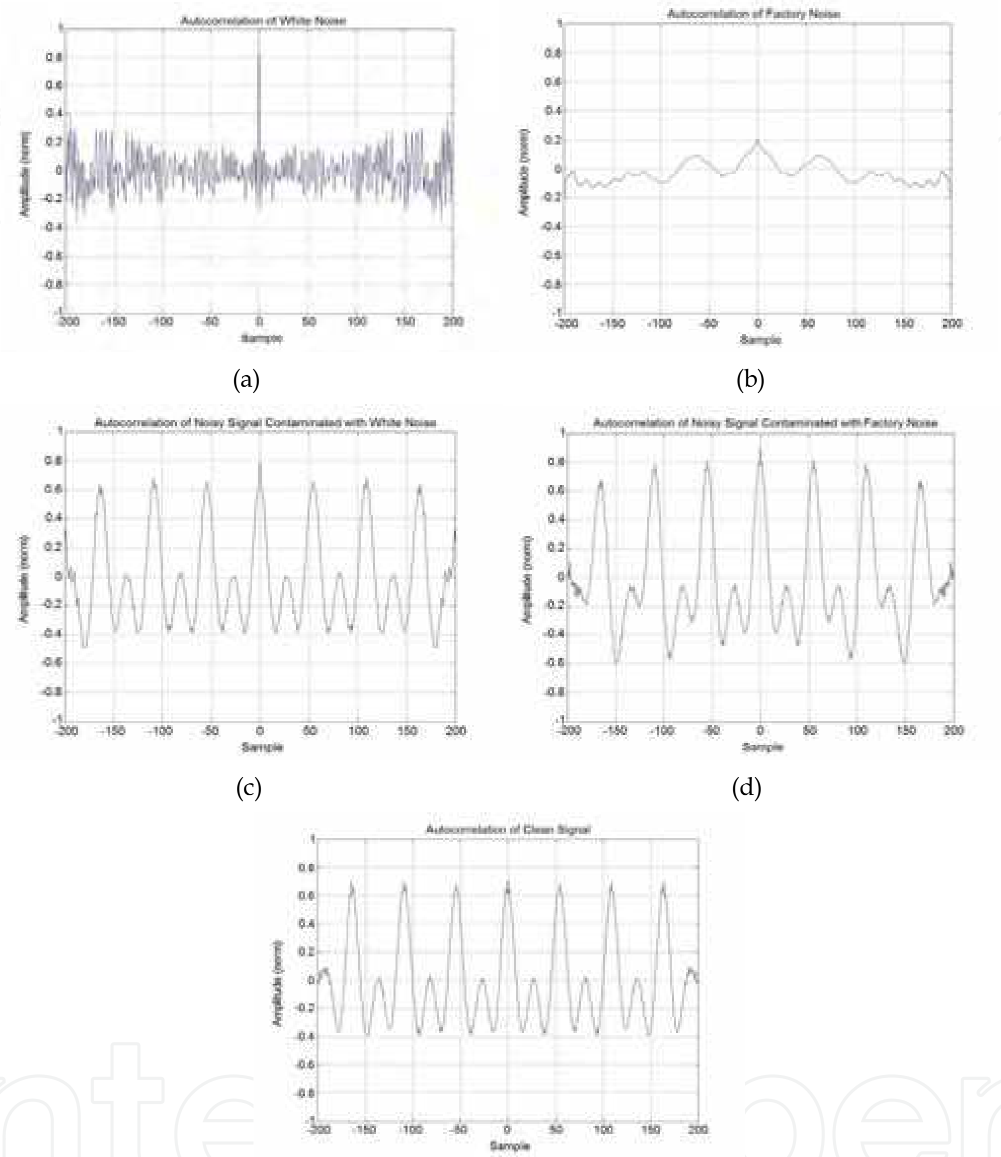
From another point of view, methods can be categorized according to the type of distortion they deal with. Methods used to suppress the effect of additive noise such as SS, lin-log RASTA, PMC, DPS, *minimum variance distortion-less response* (MVDR) (Yapanel & Dharanipragada, 2003; Yapanel and Hansen, 2003) and RAS can be placed in one category while those trying to remove channel distortion such as CMN, logarithmic-RASTA, *blind equalization* (BE) (Mauuary, 1996, 1998) and *weighted Viterbi recognition* (WVR) (Cui et al., 2003) are placed in the other category.

Although all the aforementioned efforts had a certain level of success in speech recognition tasks, it is still necessary to investigate new algorithms to further improve the performance of ASR systems. Extracting appropriate speech features is crucial in obtaining good performance in ASR systems since all of the succeeding processes in such systems are highly dependent on the quality of the extracted features. Therefore, robust feature extraction has attracted much attention in the field. Use of the autocorrelation domain in speech feature extraction has recently proved to be successful for robust speech recognition. A number of feature extraction algorithms have been devised using this domain as the initial domain of choice. These algorithms were initiated with the introduction of SMC (Mansour & Juang, 1989a) and OSALPC (Hernando & Nadeu, 1997). Recently, further improvements in this field have been reported (Yuo & Wang, 1998, 1999; Shannon and Paliwal, 2004).

Pole preserving is an important property of the autocorrelation domain, i.e. if the original signal can be modelled by an all-pole sequence which has been excited by an impulse train and a white noise, the poles of the autocorrelation sequence would be the same as the poles of the original signal (McGinn & Johnson, 1989). This means that the features extracted from the autocorrelation sequence could replace the features extracted from the original speech signal. Another property of autocorrelation sequence is that for many typical noise types, noise autocorrelation sequence is more significant in lower lags. Therefore, noise-robust spectral estimation is possible with algorithms that focus on the higher lag autocorrelation coefficients such as *autocorrelation mel-frequency cepstral coefficient* (AMFCC) method (Shannon & Paliwal, 2004). Moreover, as the autocorrelation of noise could in many cases be considered relatively constant over time, a high pass filtering of the autocorrelation sequence, as is done in RAS, could lead to substantial reduction of the noise effect.

Furthermore, it has been shown that preserving spectral peaks is very important in obtaining a robust set of features for ASR (Padmanabhan, 2000; Strope & Alwan, 1998; Sujatha et al., 2003). Methods such as *peak-to-valley ratio locking* (Zhu, 2001) and *peak isolation* (PKISO) (Strope & Alwan, 1997) have been found very useful in speech recognition error rate reduction. In DPS, as an example, differentiation in the spectral domain is used to preserve the spectral peaks while the flat parts of the spectrum, that are believed to be more vulnerable to noise, are almost removed.

Each of the above-mentioned autocorrelation-based methods has its own disadvantages. RAS, while working well in low SNRs, does not perform as well in higher SNRs and clean condition. The main reason is that while filtering the lower frequency parts of noisy autocorrelation sequence can lead to the suppression of noise in low SNR conditions (large noise energies), it in fact filters out parts of the signal autocorrelation sequence in high SNRs. However, the removal of the lower lags of the sequence leads to a good performance in high SNRs in comparison to high-pass filtering.



(a)

(b)

(c)

(d)

(e)

Figure 1. (a) Autocorrelation sequence of white noise, (b) Autocorrelation sequence of factory noise, (c) Autocorrelation sequence for one frame of speech signal contaminated with white noise at 10 dB SNR, (d) Autocorrelation sequence for one frame of speech signal contaminated with factory noise at 10 dB SNR, (e) Autocorrelation of clean speech signal (vowel).

IntechOpen

According to (Shannon & Paliwal, 2004), AMFCC works well for car and subway noises of the Aurora 2 task, but in babble and exhibition noises does not work as well. This is attributed to high similarities between the properties of the latter two and speech.

Fig. 1 displays examples of white and factory noise autocorrelation sequences (a and b), together with sequences for a speech signal (vowel) contaminated with those noises. Apparently, the lower lags of white noise are more important in comparison to those of factory noise that features a more spread out sequence.

Also we can classify autocorrelation-based robust feature extraction methods, from another point of view, into two major fields, i.e. magnitude and phase domains. Some examples of the methods that work in the magnitude domain are RAS, AMFCC, SS, RASTA filtering etc. On the other hand, an example for a phase domain method is *phase autocorrelation* (PAC) (Ikbal et al., 2003). However, recent studies on speech perception have revealed the importance of the phase of speech signal (Paliwal & Alsteris, 2003; Bozkurt et al., 2004; Bozkurt & Couvreur, 2005). The above-mentioned findings in the phase domain have persuaded further work using signal phase information in the feature vector.

In this chapter, we discuss some of the newest robust feature extraction approaches.

2. Autocorrelation-based Feature Extraction Background

While different methods have used different approaches to autocorrelation-based feature extraction, these methods could roughly be divided into two sub-categories: those that use the amplitude and those that use the phase. We will discuss these two groups of methods separately.

2.1. Autocorrelation amplitude-based approaches

2.1.1. Calculation of the autocorrelation for noisy signal

If we assume *v(m,n)* to be the additive noise, *x(m,n)* noise-free speech signal and *h(n)* impulse response of the channel, then the noisy speech signal *y(m,n)* can be written as

$$y(m,n) = x(m,n) * h(n) + v(m,n) \qquad 0 \le m \le M-1, \quad 0 \le n \le N-1, \tag{1}$$

where * denotes the convolution operation, *N* is the frame length, *n* is the discrete time index in a frame, *m* is the frame index and *M* is the number of frames. If *x(m,n)*, *v(m,n)* and *h(n)* are considered uncorrelated, the autocorrelation of the noisy speech can be expressed as

$$r_{yy}(m,k) = r_{xx}(m,k) * h(k) * h(k) + r_{vv}(m,k) \qquad 0 \le m \le M-1, \quad 0 \le k \le N-1, \tag{2}$$

where $r_{yy}(m,k)$, $r_{xx}(m,k)$ and $r_{vv}(m,k)$ are the short-time autocorrelation sequences of the noisy speech, clean speech and noise respectively and *k* is the autocorrelation sequence index within each frame. Since additive noise is assumed to be stationary, its autocorrelation sequence can be considered the same for all frames. Therefore, the frame index, *m*, can be omitted from the additive noise part in (2) leading to

$$r_{yy}(m,k) = r_{xx}(m,k) * h(k) * h(k) + r_{vv}(k) \quad 0 \le m \le M-1, \quad 0 \le k \le N-1. \tag{3}$$

The one-sided autocorrelation sequence of each frame can then be calculated using an unbiased estimator. However, alternatively, one may use a biased estimator for its

IntechOpen

calculation. The unbiased and biased estimators for the calculation of one-sided autocorrelation sequence are given in (4) and (5) respectively (Yuo & Wang, 1998, 1999).

$$r_{yy}(m,k) = \frac{1}{N-K}\sum_{i=0}^{N-1-K} y(m,i)y(m,i+k) \tag{4}$$

$$r_{yy}(m,k) = \sum_{i=0}^{N-1-K} y(m,i)y(m,i+k) \tag{5}$$

2.1.2. Filtering of one-sided autocorrelation sequence

As our target in this chapter is to remove, or reduce, the effect of additive noise from noisy speech signal, the channel effect, *h(k)*, may be removed at this point from (3). We will then have

$$r_{yy}(m,k) = r_{xx}(m,k) + r_{vv}(k) \qquad 0 \le m \le M-1\ ,\quad 0 \le k \le N-1\,. \tag{6}$$

In order to reduce the effect of channel distortion, one may add a simple approach, such as cepstral mean normalization, to the final cepstral coefficients, as we will mention later.

Considering the noise autocorrelation to be constant over the frames of concern, differentiating both sides of equation (6) with respect to *m*, will remove the noise autocorrelation and yields (Yuo & Wang, 1999)

$$\frac{\partial r_{yy}(m,k)}{\partial m} = \frac{\partial r_{xx}(m,k)}{\partial m} + \frac{\partial r_{vv}(k)}{\partial m} \cong \frac{\partial r_{xx}(m,k)}{\partial m} = \frac{\sum_{t=-L}^{L} t.r_{yy}(m+t,k)}{\sum_{t=-L}^{L} t^2}$$

$$0 \le m \le M-1,\ 0 \le k \le N-1\,. \tag{7}$$

Equation (7) is equal to a filtering process on the temporal one-sided autocorrelation trajectory by a *FIR* filter where *L* is the length of the filter. This filtering process can be written in *z* domain as

$$H(z) = \frac{\sum_{t=-L}^{L} t.z^{t}}{\sum_{t=-L}^{L} t^2} = \frac{-L.z^{-L} + (-L+1).z^{(-L+1)} + \ldots + (-2).z^{-2} + (-1).z^{-1} + z + 2.z^2 + \ldots + (L-1).z^{L-1} + L.z^{L}}{2.(1+2^2+\ldots+(L-1)^2+L^2)} \tag{8}$$

2.1.3. Lower lag elimination

As we mentioned before, the main effects of noise autocorrelation on the clean speech signal autocorrelation is on its lower lags. Therefore eliminating the lower lags of the noisy speech signal autocorrelation should lead to removal of the main noise components. The maximum autocorrelation index to be removed is usually found experimentally. The resulting sequence would be

$$\hat{r}_{yy}(m,k) = r_{yy}(m,k), \qquad D < m \le M-1, \quad 0 < k \le K-1$$

$$\hat{r}_{yy}(m,k) = 0, \qquad 0 \le m \le D, \quad 0 \le k \le K-1, \tag{9}$$

where *D* is the elimination threshold.

IntechOpen

2.2. Autocorrelation phase-based approaches

As mentioned earlier, feature extraction from magnitude spectrum will be obtained by applying DFT on the frame samples. DFT assumes each frame, *y(m,n)*, is a part of periodic signal, $\tilde{y}(m,n)$, which is defined as :

$$\tilde{y}(m,n) = \sum_{k=-\infty}^{+\infty} y(m,n+kN) \qquad 0 \le m \le M-1, \quad 0 \le n \le N-1. \tag{10}$$

The estimator for the calculation of autocorrelation sequence is then given as:

$$\tilde{r}_{yy}(m,k) = \sum_{i=0}^{N-1} \tilde{y}(m,i)\tilde{y}(m,i+k) \quad 0 \le m \le M-1, \; 0 \le k \le N-1. \tag{11}$$

Another view to equation (11) is that $\tilde{r}_{yy}(m,k)$ gives the correlation between the samples spaced at interval *k*, which is computed as dot product of two vectors in *N*-dimensional domain, i.e.

$$Y_0 = \{\tilde{y}(m,0), \tilde{y}(m,1), ..., \tilde{y}(m,N-1)\}$$

$$Y_k = \{\tilde{y}(m,k), ..., \tilde{y}(m,N-1), \tilde{y}(m,0), ..., \tilde{y}(m,k-1)\}$$

$$\tilde{r}_{yy}(m,k) = Y_0{}^T Y_k. \tag{12}$$

If we carry out these steps for clean speech, *x(n,m)*, we would have

$$X_0 = \{\tilde{x}(m,0), \tilde{x}(m,1), ..., \tilde{x}(m,N-1)\}$$

$$X_k = \{\tilde{x}(m,k), ..., \tilde{x}(m,N-1), \tilde{x}(m,0), ..., \tilde{x}(m,k-1)\}$$

$$\tilde{r}_{xx}(m,k) = X_0{}^T X_k \tag{13}$$

where $\tilde{x}(m,n)$ is the periodic signal obtained from *x(m,n).*

IntechOpen

Clearly, the autocorrelation sequences for clean and noisy signals are different. Therefore, features extracted from autocorrelation sequences would be sensitive to noise. From (12), the magnitudes of two vectors Y_0 and Y_k are the same. If we assume $|Y(m)|$ to be the magnitude of vectors and $\theta_y(m,k)$ the angle between them, then the relationship between the autocorrelation, $\tilde{r}_{yy}(m,k)$, magnitude of the vectors and the angle between them would be

$$\tilde{r}_{yy}(m,k) = |Y(m)|^2 \cos\theta_y(m,k), \qquad 0 \le m \le M-1, \quad 0 \le k \le N-1. \tag{14}$$

Now the angle $\theta_y(m,k)$ between the two vectors will be calculated as:

$$\theta_y(m,k) = \cos^{-1}\left(\frac{\tilde{r}_{yy}(m,k)}{|Y(m)|^2}\right) \qquad 0 \le m \le M-1,\ 0 \le k \le N-1. \tag{15}$$

IntechOpen

3. Autocorrelation-based robust feature extraction

In this section we will describe several autocorrelation-based approaches developed by the authors to deal with the problem of robust feature extraction. These methods use either amplitude or phase in the autocorrelation domain, as discussed above. Performance of these methods in various noisy speech recognition tasks will be discussed in a later section.

3.1. Differentiated autocorrelation sequence (DAS)

In this section first we present the calculation of differential power spectrum and then describe DAS method.

3.1.1. Calculating differential power spectrum (DPS)

Although not necessarily an autocorrelation-based approach, we discuss DPS here as it will be used in the following discussions. If the noise and clean speech signals are assumed mutually uncorrelated, by applying short-time DFT to both sides of equation (6), we can calculate the relationship between autocorrelation power spectrums of noisy speech signal, clean speech signal and noise as follows:

$$Y(\omega) = FT\{r_{yy}(m,k)\} \approx FT\{r_{xx}(m,k)\} + FT\{r_{vv}(k)\} = X(\omega) + V(\omega), \tag{16}$$

where $FT[.]$ denotes the Fourier Transform and ω indicates radian frequency. The differential power spectrum will then be defined as

$$Diff_Y(\omega) = Y'(\omega) = \frac{dY(\omega)}{d\omega}, \tag{17}$$

where $\frac{d}{d\omega}$ or prime represent differentiation with respect to ω.

Therefore, by applying differentiation to both sides of equation (16) we have:

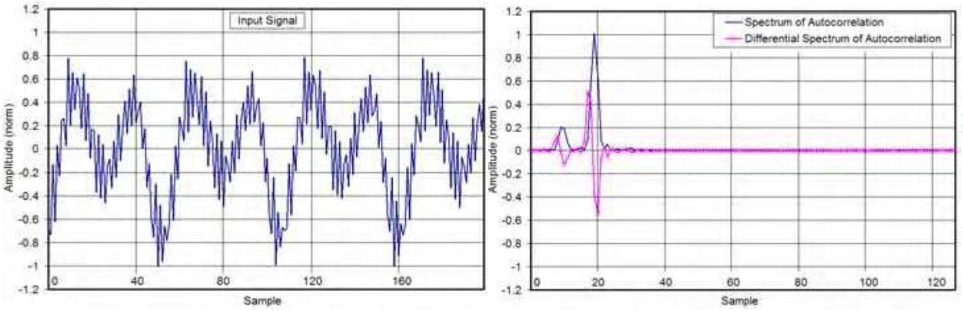
$$Diff_Y(\omega) = \frac{dY(\omega)}{d\omega} = \frac{dX(\omega)}{d\omega} + \frac{dV(\omega)}{d\omega} = Diff_X(\omega) + Diff_V(\omega), \tag{18}$$

where $Diff_X(\omega)$ and $Diff_V(\omega)$ are differential autocorrelation power spectrums of clean speech signal and noise respectively.

IntechOpen

In discrete domain, the definition of DPS can be approximated by the following equation

$$Diff_Y(k) = Diff_X(k) + Diff_V(k) \approx \sum_{l=-Q}^{P} a_l Y(k+l) \cong \sum_{l=-Q}^{P} a_l \left[X(k+l) + V(k+l)\right],\ \ 0 \le k \le K-1 \tag{19}$$



(a) (b)

Figure 2. (a) A sample speech signal, and (b) the autocorrelation spectrum magnitude and the differentiated autocorrelation spectrum magnitude of the same signal with a 512-point FFT. Only 128 points of the spectrum are shown for clarity. The sample signal is one frame of phone /iy/ (Farahani et al., 2007).

where P and Q are the orders of the difference equation, a_l are real-valued weighting coefficients and K is the length of FFT. The absolute value of the differentiation indicates the peak points of the spectrum.

3.1.2. DAS algorithm

This approach combines the advantages of RAS and DPS. In this algorithm (Farahani & Ahadi, 2005; Farahani et al., 2007), splitting the speech signal into frames and applying a pre-emphasis filter, the autocorrelation sequence of the frame signal is obtained using either an unbiased or a biased estimator, as shown in (4) and (5). A *FIR* filter is then applied to the noisy speech signal autocorrelation sequence. Hamming windowing and short-time Fourier transform constitute the next stages. Then, the differential power spectrum of the filtered signal is calculated. By differentiation of the spectrum, we preserve the peaks, except that each peak is split into two, one positive and one negative, and the flat part of the power spectrum is approximated to zero.

Fig. 2 depicts a sample speech signal, its short-time autocorrelation spectrum and the differentiated short-time autocorrelation spectrum. This sample signal is one frame of phone /iy/. In order to simplify the representation, only the significant lower-frequency parts of the spectrum have been shown and the non-significant parts omitted.

As shown in Fig. 2 and mentioned above, the flat parts of the spectrum have been transformed to zero by differentiation and each peak of it split into two positive and negative parts. Since the spectral peaks convey the most important information in speech signal, this fact that the differential power spectrum retains spectral peaks means that we will not lose the important information of the speech signal (Chen et al., 2003). Furthermore, since noise spectrum is often flat and the differentiation either reduces or omits the relatively flat parts of the spectrum, it will lead to suppression of the noise effect on the spectrum. A set of cepstral coefficients can then be derived by applying a conventional mel-frequency filter-bank to the resultant spectrum and finally passing the logarithm of bin outputs to the DCT block. Fig. 3 displays the overall front-end diagram of this method. We call these new features Differentiated Autocorrelation Sequence (DAS).

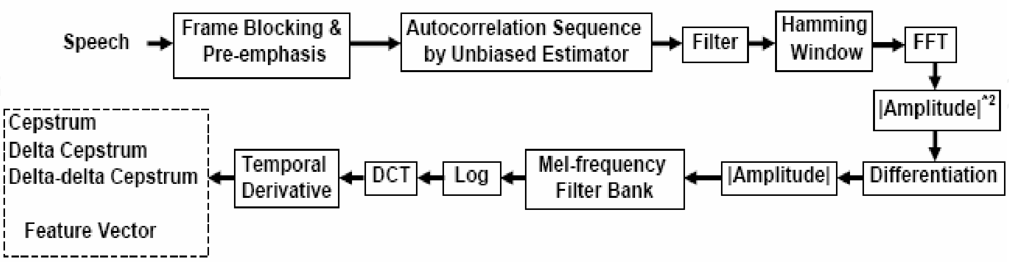


Figure 3. Block diagram of DAS front-end for robust feature extraction.

3.1.3. Experiments

If the process of feature extraction does not include the extraction of autocorrelation spectral peaks, as mentioned in (Yuo & Wang, 1999), using 5 frames of speech for filtering can lead to the best results. Therefore, in order to be able to compare our results to the best results of RAS, the same filter length (*L*=2) is used. Thus, equation (8) will be simplified as

$$H(z) = \frac{\sum_{t=-2}^{2} t.z^t}{\sum_{t=-2}^{2} t^2} = \frac{-2z^{-2} - 1z^{-1} + z + 2z^2}{10} . \qquad (20)$$

Furthermore, as discussed in (Chen et al., 2003), the best results for spectral differentiation was obtained using the following equation

$$Diff(k) = Y(k) - Y(k+1) , \qquad (21)$$

i.e. a simple difference.

The above-mentioned parameter and formula were used in the implementation of RAS, DPS, DAS, SPFH, ACP, ACPAF and APP methods on different mentioned tasks, unless otherwise specified. All model creation, training and tests in all our experiments have been carried out using the HMM toolkit (HTK, 2002).

3.1.3.1. Database

Three speech corpora have been used throughout the experiments reported in this chapter. These include an isolated-word Farsi task, a continuous Farsi task and Aurora 2, a noisy English connected digits task.

The Isolated-word Farsi task is a set of isolated-word Farsi (Persian) speech data collected from 65 male and female adult speakers uttering the names of 10 Iranian cities. The data was collected in normal office conditions with SNRs of 25dB or higher and a sampling rate of 16 kHz. Each speaker uttered 5 repetitions of words, some of which were removed from the corpus due to problems that occurred during the recordings. A total of 2665 utterances from 55 speakers were used for HMM model training. The test set contained 10 speakers (5 male

& 5 female) that were not included in the training set. The noise was then added to their speech in different SNRs. The noise data was extracted from the NATO RSG-10 corpus (SPIB, 1995). We have considered babble, car, factory and white noises and added them to the clean signal at 20, 15, 10, 5, 0 and -5 dB SNRs.

The Continuous Farsi task is a speaker-independent medium-vocabulary continuous speech Farsi (Persian) corpus. FARSDAT speech corpus was used for this set of experiments (Bijankhan et al., 1994; FARSDAT). This corpus was originally collected from 300 male and female adult speakers uttering 20 Persian sentences in two sessions. The sentences uttered by each speaker were randomly selected from a set of around 400 sentences. Some of the speakers were removed from the corpus due to their accent or problems occurred during the recordings. The data was originally collected in quiet environment with SNRs of 25dB or higher and a sampling rate of 44.1 kHz. The sampling rate was later reduced to 16 kHz. A total of 1814 utterances from 91 speakers were used for HMM model training in these experiments. The test set contained 46 speakers that were not included in the training set. A total of 889 utterances were used as the test set. The noise was then added to the speech in different SNRs. As with the isolated-word experiments, the noise data was extracted from the NATO RSG-10 corpus and included babble, car, factory and F16 noises added to the clean signal at 20, 15, 10, 5, 0 and -5 dB SNRs.

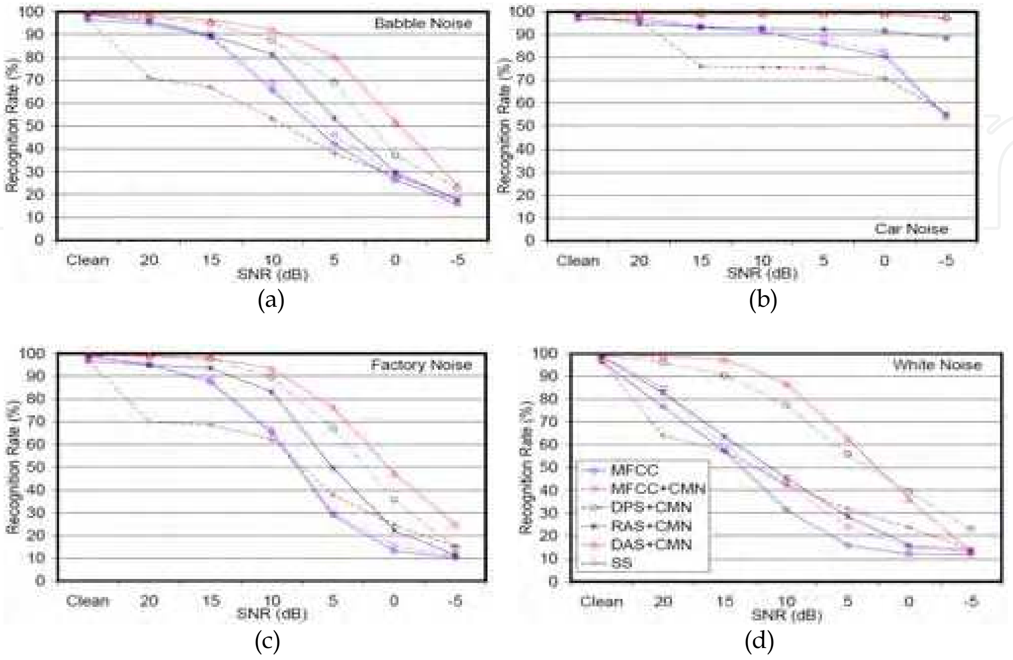
Aurora 2 (Hirsch & Pearce, 2000) is a noisy connected-digit recognition task. It includes two training modes, training on clean data only (clean-condition training) and training on clean and noisy data (multi-condition training). In clean-condition training, 8440 utterances from TIDigits speech corpus (Leonard, 1984) containing those of 55 male and 55 female adults are used. For multi-condition mode, 8440 utterances from TIDigits training part are split equally into 20 subsets with 422 utterances in each subset. Suburban train, babble, car and exhibition hall noises are added to these 20 subsets at SNRs of 20, 15, 10, 5, 0 and -5 dB.

Three test sets are defined in Aurora 2, named A, B and C. 4004 utterances from TIDigits test data are divided into four subsets with 1001 utterances in each. One noise is added to each subset at different SNRs.

In test set A, suburban train, babble, car and exhibition noises are added to the above mentioned four subsets, leading to a total of $4 \times 7 \times 1001$ utterances. Test set B is created similar to test set A, but with four different noises, namely, restaurant, street, airport and train station. Finally, test set C contains two of four subsets with speech and noise filtered using different filter characteristics in comparison to the data used in test sets A and B. The noises used in this set are suburban train and street.

3.1.3.2. Results on the isolated-word Farsi task

The experiments were carried out using MFCC (for comparison purposes), MFCC applied to the signal enhanced by spectral subtraction, RAS-MFCC, cepstral coefficients derived using DPS and DAS. In all cases, 25 msec. frames with 10 msec. of frame shifts and a pre-emphasis coefficient of 0.97 were used. Also, for each speech frame, a 24-channel mel-scale filter-bank was used. Here, each word was modelled by an 8-state left-right HMM and each state was represented by a single-Gaussian PDF. The feature vectors were composed of 12 cepstral and log-energy parameters, together with their first and second order derivatives (39 coefficients in total). Fig. 4 depicts the results of the implementation.



(a) (b) (c) (d)

Figure 4. Isolated-word recognition results for speech signal contaminated with (a) babble, (b) car, (c) factory and (d) white noises in different SNRs. The results correspond to MFCC, MFCC+CMN, DPS+CMN, RAS+CMN, DAS+CMN and SS methods for isolated-word task with 1 mixture component per state.

Also, in Tables 1 and 2, the clean recognition results and the average noisy speech recognition results are included for comparison purposes. The average values mentioned in Table 2 were calculated over the results obtained from 0 dB to 20 dB SNRs, omitting the clean and -5 dB results. This is in accordance with average result calculations in Aurora 2 task.

Note that, all the reported results are for clean-trained models and no matched condition or multi-conditioned noisy training was carried out. Furthermore, also for comparison purposes, the results of an implementation of cepstral mean normalization, as a feature post-processing enhancement method, applied after standard MFCC, DPS and DAS parameter extractions, are included. These are denoted by CMN. Note that, for better comparison, the results of an implementation of spectral subtraction as an initial enhancement method, applied before standard MFCC parameter extraction, are also included. These are denoted by SS and the algorithm was applied as explained in (Junqua & Haton, 1996). As can be seen in Fig. 4, DAS outperforms all other methods in almost all noise types and SNRs. The average results on different SNRs, as shown in Table 2, are again considerably better for DAS in comparison to the other feature extraction techniques, except for the case of car noise, where it is very close to DPS results. As an example, DAS has about 29% reduction on the average word error rate for babble noise, compared to DPS, which performs the best among the others. Similar conclusions can be made from this table for factory and white noises.

IntechOpen

Feature type	Recognition Rate (%)
MFCC	96.60
SS	96.60
MFCC+CMN	99.20
DPS+CMN	99.20
RAS+CMN	98.80
DAS+CMN	99.20

Table 1. Comparison of baseline isolated-word recognition rates for various feature types.

Feature type	Average Recognition Rate (%)			
	Babble	Car	Factory	White
MFCC (Baseline)	63.60	89.44	57.92	38.68
SS	51.60	78.88	52.72	43.88
MFCC+CMN	66.00	91.00	59.24	45.08
DPS+CMN	77.28	99.24	77.84	71.84
RAS+CMN	70.00	92.88	68.72	47.24
DAS+CMN	83.88	99.12	82.80	76.36

Table 2. Comparison of average isolated-word recognition rates for various feature types with babble, car, factory and white noises.

3.1.3.3. Results on continuous Farsi task

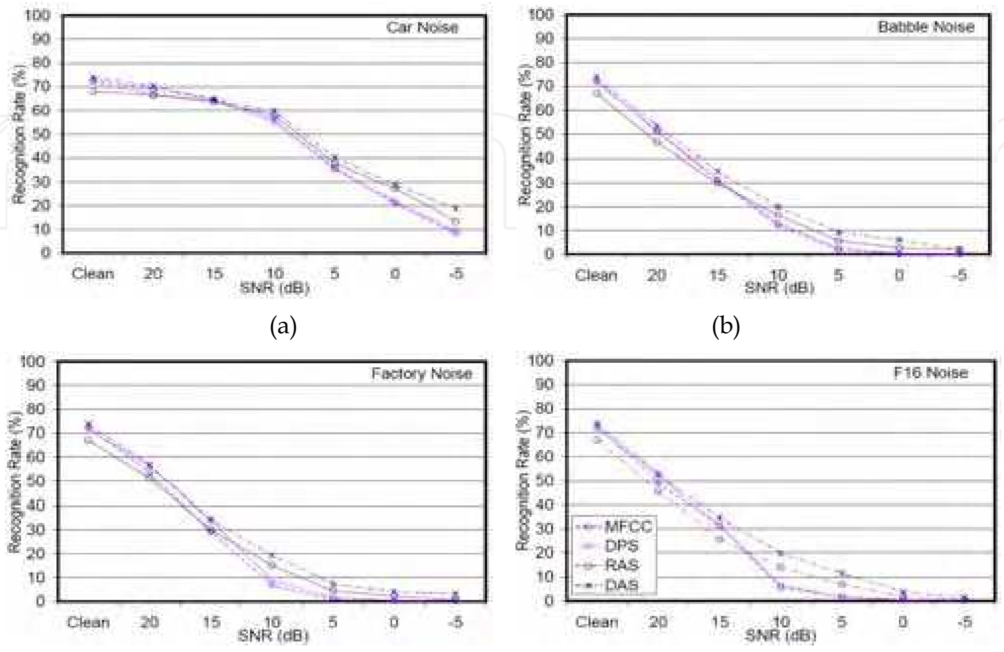
The feature parameters were extracted similar to the isolated-word recognition case. The modeling was carried out using 30 context-independent models for the basic Farsi phonemes plus silence and pause models. These, except the pause model, consisted of 3 states per model, while the pause model included one state only. The number of mixture components per state was 6 and no grammar was used during the recognition process to enable us better evaluate our acoustic models under noisy conditions. The size of the recognition lexicon was around 1200.

IntechOpen

Figures 5 and 6 display the results of our FARSI CSR experiments. These results were obtained using a set of parameters similar to the isolated-word experiments parameters.

According to Fig. 5, DAS outperforms all other front-ends such as RAS, DPS and MFCC in almost all cases. While RAS performance is acceptable in low SNRs, it performs inferior to other front-ends in SNRs over 10dB. DPS performs slightly better than MFCC except in car noise.

Fig. 6 depicts the results obtained when CMN is applied alongside the above front-ends. Here again, the DAS front-end outperforms the others in almost all cases.



(a)

(b)

(b)

(d)

Figure 5. Continuous speech recognition accuracies for speech signal contaminated with (a) babble, (b) car, (c) factory and (d) F16 noises in different SNRs. The results correspond to MFCC, DPS, RAS and DAS front-ends and 6 mixture components per state, without CMN.

Table 3 summarizes recognition rates for various front-ends with/without CMN and contaminated with babble, car, factory and f16 noises with six mixture components per state. According to this table, DAS with/without CMN outperforms the other methods in average, proving its effectiveness in noisy speech recognition.

3.1.3.4. Results on Aurora 2 task

The features in this case were computed using 25 msec. frames with 10 msec. of frame shifts. Pre-emphasis coefficient was set to 0.97. For each speech frame, a 23-channel mel-scale filter-bank was used. The feature vectors were composed of 12 cepstral and log-energy parameters, together with their first and second order derivatives (39 coefficients in total). MFCC feature extraction and all model creation, training and tests were carried out using the HMM toolkit.

Fig. 7 shows the results obtained using MFCC, DPS, RAS and DAS front-ends with CMN on models created using the clean-condition training section of Aurora 2 task. Once again, the DAS front-end leads to better recognition rates in comparison to other methods. Also, Fig. 8 depicts the recognition rates of different methods obtained using the multi-condition training set of Aurora 2 task. The multi-condition results of different front-ends show very close performances. However, DAS still performs slightly better compared to the others.

3.1.3.5. Adjusting the Parameters

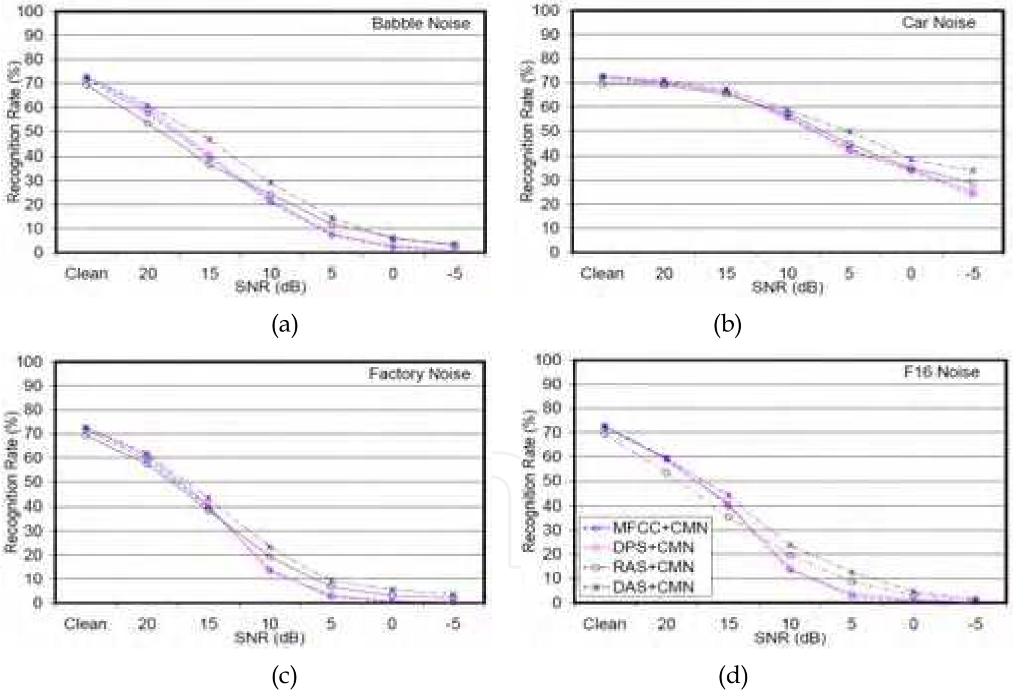
For parameter setting in DAS, the length of the FIR filter (filter type is same as (8)) and the order of differentiation was taken into consideration. A set of preliminary experiments were carried out to find the most appropriate filtering and differentiation parameters. These experiments were performed on the continuous Farsi task as explained in section 3.1.3.1. Here, the length of the filter was changed from L=1 (three frames) to L=5 (eleven frames) in steps. Also, biased and unbiased estimators were used for calculating the one-sided autocorrelation sequence. Table 4 summarize the results by displaying the average recognition accuracies on the test set with various noises (babble, car, factory and F16) and in various SNRs, with two different autocorrelation estimators. At this step, for computing the autocorrelation spectral peaks the differentiation defined in (21) was considered.

As can be seen in Table 4, the best average recognition results using DAS features were obtained using the filter length L=3. Furthermore, the unbiased estimator led to better results compared to the biased one.

In order to find the best differentiation methods, the following differentiation formulas were used.

$$Diff(k) = Y(k) - Y(k+2)\,, \tag{22}$$

$$Diff(k) = Y(k-2) + Y(k-1) - Y(k+1) - Y(k+2)\,, \tag{23}$$



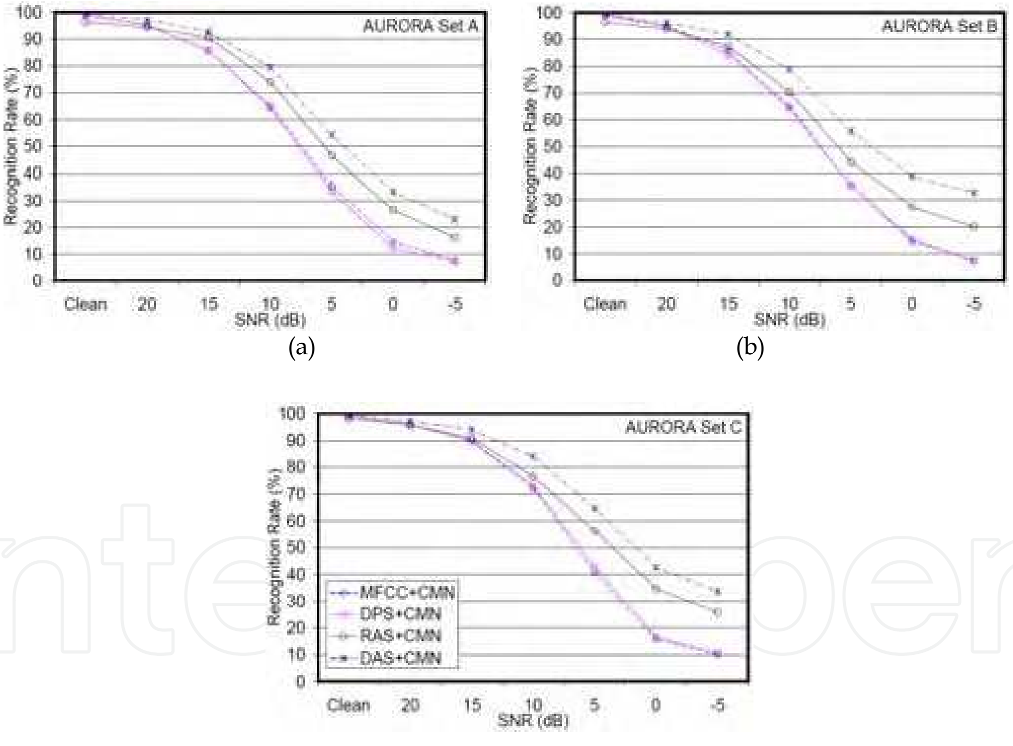
(a) (b)

(c) (d)

Figure 6. Continuous speech recognition accuracies for speech signal contaminated with (a) babble, (b) car, (c) factory and (d) f16 noises in different SNRs. The results correspond to MFCC, DPS, RAS and DAS front-ends with CMN and 6 mixture components per state.

Recognition Rate (%) - 6 mix per state				
Noise Type	Babble	Car	Factory	F16
MFCC (Baseline)	19.23	49.39	18.23	17.64
MFCC+CMN	25.39	53.69	23.22	23.24
DPS	19.90	48.06	20.32	18.42
DPS+CMN	26.43	53.62	23.80	23.54
RAS	20.32	50.67	20.70	21.84
RAS+CMN	26.44	54.21	25.05	24.03
DAS	24.72	52.78	26.22	26.42
DAS+CMN	31.31	57.09	28.67	28.84

Table 3. Comparison of average continuous speech recognition accuracies for various feature types in babble, car, factory and f16 noises with different SNRs. Recognition was carried out using 6 mixture components per state.

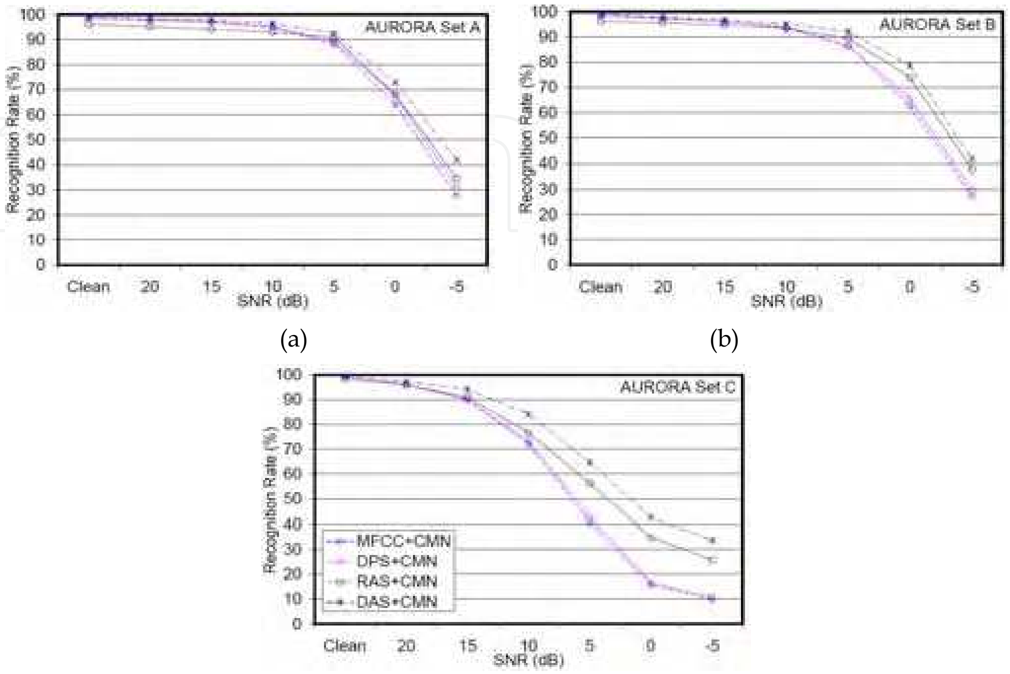


(a)

(b)

(c)

Figure 7. Average recognition accuracies for clean-condition on AURORA2.0 (a) set A, (b) set B and (c) set C. The results correspond to MFCC, DPS, RAS and DAS front-ends with CMN.



(a)

(b)

(c)

Figure 8. Average recognition accuracies for AURORA 2 multi-condition training. (a) set A, (b) set B and (c) set C. The results correspond to MFCC, DPS, RAS and DAS front-ends with CMN.

Since the filter with length L=3, using unbiased estimator, had led to better results in noisy speech recognition, this type of filter and estimator have been tested together with the difference equations given in (21), (22) and (23). The results are reported in Table 5.

According to this table, the one sample difference equation, (21), has led to better overall results, compared to (22) and (23).

Biased Estimator					Unbiased Estimator			
Filter Lenght	Babble	Car	Factory	F16	Babble	Car	Factory	F16
L=1 3 frames	22.07	49.06	22.59	21.72	24.08	51.80	23.77	23.25
L=2 5 frames	22.97	50.00	23.24	22.95	24.72	52.78	24.43	24.53
L=3 7 frames	24.69	51.34	24.41	24.42	26.00	53.24	26.49	25.38
L=4 9 frames	23.82	50.89	23.76	23.15	25.31	53.06	25.61	23.78
L=5 11 frames	23.07	49.91	23.35	22.89	25.09	52.59	25.41	23.76

Table 4. The average continuous speech recognition rates using DAS for various noise and SNRs with different filter lengths. Biased and Unbiased estimator was used for one-sided autocorrelation sequence calculation and the models featured 6 mixture components per state.

3.2. Spectral Peaks of filtered higher-lag autocorrelation sequence (SPFH)

In this method, after splitting the speech signal into frames and pre-emphasizing, the autocorrelation sequence of the frame signal was obtained using an unbiased estimator (equation (4)). The lower lags of the autocorrelation sequence were then removed according to the criterion discussed in section 3.2.2. A *FIR* high-pass filter similar to section 3.1.2 was then applied to the signal autocorrelation sequence to further suppress the effect of noise. Hamming windowing and short-time Fourier transform were the next stages. In the next step, the differential power spectrum of the filtered signal was found using (19). This has an effect similar to what discussed in 3.1.2, leading to even further suppression of the effect of noise. The steps of this algorithm are shown in Fig. 9. The resultant feature set was called *spectral peaks of filtered higher-lag autocorrelation sequence* (SPFH) (Farahani et al., 2006b).

Noise Type	Recognition Rate (%) equation (21)	Recognition Rate (%) equation (22)	Recognition Rate (%) equation (23)
Babble	26.00	24.83	25.19
Car	53.24	51.56	52.35
Factory	26.49	25.17	25.82
F16	25.38	23.71	24.25

Table 5. CSR averaged accuracies over various SNRs with different noise types, filter length L=3 and unbiased estimator for one-sided autocorrelation sequence obtained using equations (21), (22) and (23).

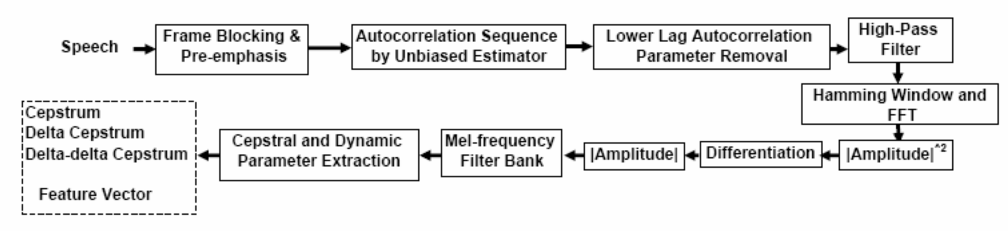

Speech
Frame Blocking & Pre-emphasis
Autocorrelation Sequence by Unbiased Estimator
Lower Lag Autocorrelation Parameter Removal
High-Pass Filter
Hamming Window and FFT
|Amplitude|^2
Differentiation
|Amplitude|
Mel-frequency Filter Bank
Cepstral and Dynamic Parameter Extraction
Cepstrum
Delta Cepstrum
Delta-delta Cepstrum
Feature Vector

Figure 9. Front-end diagram for SPFH feature extraction

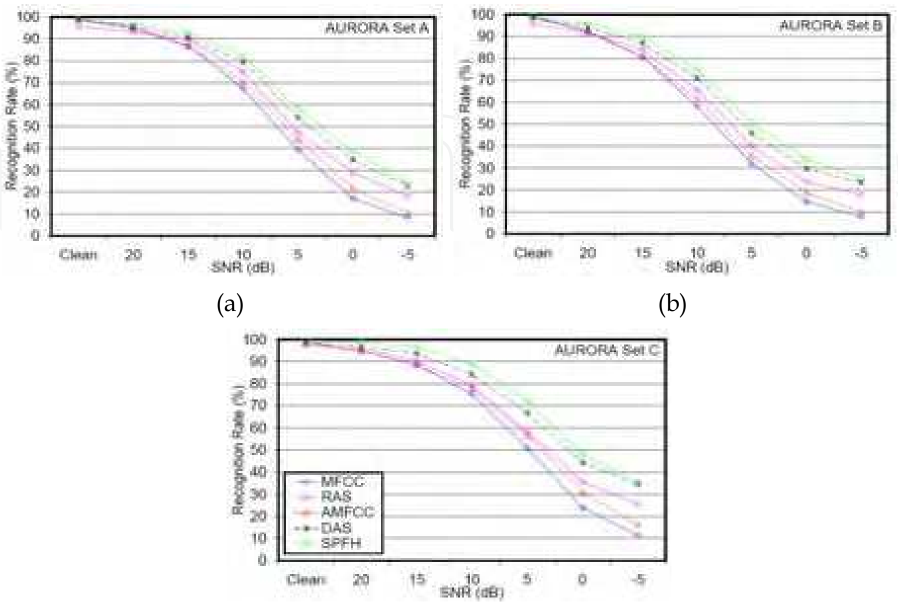
3.2.1. SPFH results on Aurora 2 task

The discussed approach was implemented on Aurora 2 task. Fig. 10 depicts the results obtained using MFCC, RAS, AMFCC, DAS and SPFH front-ends. According to this figure, SPFH has led to better recognition rates in comparison to other methods for all test sets. Also, in Table 6, average recognition rates obtained for each test set of Aurora 2 are shown.

As shown in Fig. 10, the recognition rates using MFCC are seriously degraded in lower SNRs, while, AMFCC, RAS, DAS and SPFH are more robust to different noises with SPFH outperforming all the others.

3.2.2. Adjusting the parameters

In our experiments we have used a filter length of L=2. Furthermore, the best results for spectral differentiation were obtained using a simple difference equation (21). Therefore we



(a)

(b)

(c)

Figure 10. Average recognition rates on Aurora 2 task. (a) Test set a, (b) Test set b, (c) Test set c. The results correspond to MFCC, RAS, AMFCC, DAS and SPFH methods.

Feature type	Average Recognition Rate (%)		
	Set A	Set B	Set C
MFCC	61.13	55.57	66.68
RAS	66.77	60.94	71.81
AMFCC	63.41	57.67	69.72
DAS	70.90	65.57	77.17
SPFH	73.61	68.98	80.89

Table 6. Comparison of Average recognition rates for various feature types on Aurora 2 test sets.

used (21) in our experiments. In order to find the most suitable autocorrelation lag for discarding, we have tested several different lag values. The results are reported in Fig. 11. According to this figure, the best results were obtained when lags of lower than 2.5 ms (20 samples) were discarded. Hence, this value was used as the discarding threshold in our experiments. The same value was also used with AMFCC.

3.3. Autocorrelation peaks after filtering (ACPAF) and autocorrelation peaks (ACP) methods

In this section, first we explain the extraction of the frequency locations of peaks. Then we explain ACPAF and ACP methods and finally report the results of their implementation.

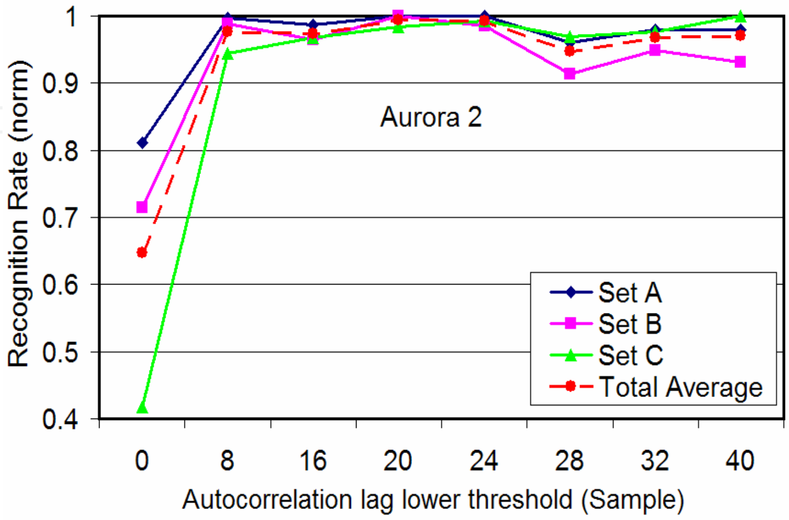


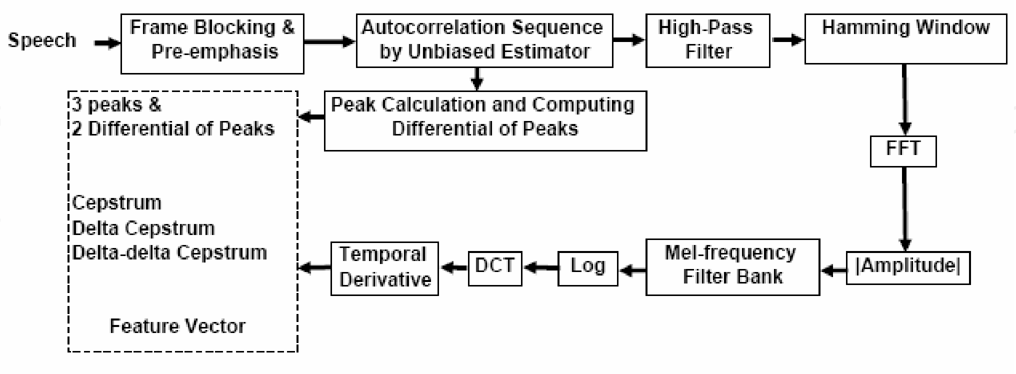
Figure 11. Average normalized recognition rates for each of the three test sets of Aurora 2 and their total average versus autocorrelation lag threshold.

3.3.1. Peak frequency location

For peak calculations, we used the *peak threading* method that is rather accurate in finding the location of peak frequencies in spectral domain (Strope & Alwan, 1998). For this, we first applied a set of triangular filters to the signal. These filters had bandwidths of 100 Hz for centre frequencies below 1 kHz and bandwidths of one tenth the centre frequency for the frequencies above 1 kHz. Then, an AGC (Automatic Gain Control) was applied to the filter outputs. In our implementation, we used a typical AGC that slowly adapts the output level, so that its value is maintained near that of the target level when the levels of input change. Therefore, the inputs below 30 dB are amplified linearly by 20dB and inputs above 30 dB are amplified increasingly less. After finding the isolated peaks, the peaks were threaded together and smoothed. Then three peak frequencies and two peak derivatives were found and added to the feature vector.

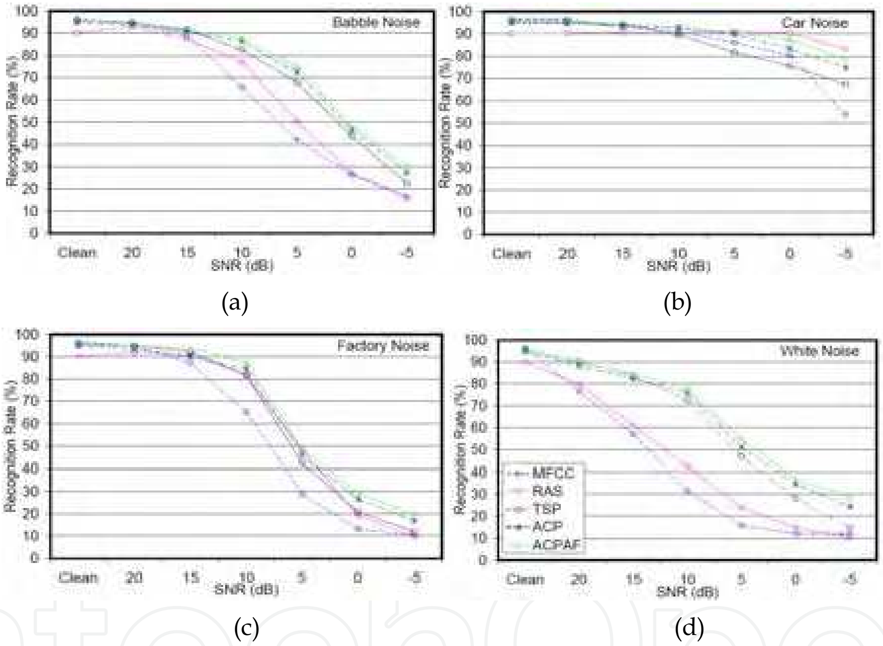
3.3.2. Algorithm implementation

Due to the importance of spectral peaks and also the effectiveness of autocorrelation function, the autocorrelation domain has also been used for extracting frequencies of the first three peaks and two derivatives of them. Fig. 12 depicts the block diagram of this feature extraction approach. Once again, feature extraction starts with frame blocking, pre-emphasis and unbiased autocorrelation calculation. Then, the first three spectral peak locations and their derivatives are calculated using the signal autocorrelation, to be later added to the feature vector. Furthermore, the front-end diagram continues with/without filtering, as pointed out in (8). Hamming windowing, FFT and the rest of blocks normally used in MFCC calculations constitute the remainder of feature extraction procedure. If the filter is used after the autocorrelation of the signal, a cleaner signal, compared to the original noisy signal, could result.



Speech
Frame Blocking & Pre-emphasis
Autocorrelation Sequence by Unbiased Estimator
High-Pass Filter
Hamming Window
Peak Calculation and Computing Differential of Peaks
3 peaks & 2 Differential of Peaks
FFT
Cepstrum
Delta Cepstrum
Delta-delta Cepstrum
Temporal Derivative
DCT
Log
Mel-frequency Filter Bank
|Amplitude|
Feature Vector

Figure 12. Front-end diagram to extract ACPAF features in autocorrelation domain.



(a)

(b)

(c)

(d)

Figure 13. Recognition Rates for different noises on isolated-word Farsi corpus. (a) babble, (b) car, (c) factory and (d) white noises in different SNRs. The results correspond to MFCC, RAS, TSP, ACP and ACPAF.

In Fig. 12, the box named "peak calculation and computing differential of peaks" displays how the peak threading method can be integrated in this front-end. Since after the application of autocorrelation function, the spectral peaks become clearer, we expect the resultant feature vectors to be more robust to noise.

As mentioned in (Strope & Alwan, 1998; Farahani et al., 2006a), the peaks of the speech spectrum are important for speech recognition. Hence, we decided to add three peak frequencies and two peak derivatives to the feature vector. The spectral peaks obtained using unfiltered signal autocorrelation, as depicted in the front-end diagram, are called ACP (autocorrelation peaks) and those obtained using filtered signal autocorrelation, ACPAF (autocorrelation peaks after filtering). For comparison purposes, we have also implemented a feature extraction procedure similar to (Strope & Alwan, 1998), except that a different AGC was used, as explained above. This will be called *threaded spectral peaks* (TSP).

3.3.3. Experiments

The speech corpus used in these experiments is the speaker-independent isolated-word Farsi (Persian) corpus. Our experiments were carried out using MFCC (for comparison purposes), RAS, GDF, TSP and our three new methods, ACP and ACPAF. The feature vectors for both proposed methods were composed of 12 cepstral and a log-energy parameter, together with their first and second derivatives and five extra components of which three are for the first three formants and the other two for the frequency peak derivatives. Therefore, our feature vectors were of size 44. For implementation of these methods, we have used a filter length of L=2. Also unbiased autocorrelation sequence is used for feature extraction.

Fig. 13 depicts the results of the implementations. Also the averages of the results are reported in Table 7. Once again, the average values mentioned in this table are calculated over the results obtained from 0 dB to 20 dB SNRs, omitting the clean and -5 dB results. As is clear, the recognition rates using MFCC features are seriously degraded by different noises, while RAS method exhibits more robustness. Adding the frequencies of peaks approved the effectiveness of autocorrelation domain for peak tracking. As the results indicate, while ACP achieves a noticeable improvement in the baseline performance in noise, the combination of ACP with FIR filter works better than ACP alone, outperforming other methods in noisy conditions. The results obtained can be listed in brief as:

1. The ACPAF outperforms other methods.
2. The improvements for car noise are very slight, as most of the feature extraction techniques perform almost similar in that case.
3. Appending frequency peaks to the feature vector can further improve the results obtained using autocorrelation based features.

3.4. Autocorrelation peaks and phase features (APP)

In this section, feature extraction in phase domain plus the extraction of extra feature parameters in autocorrelation domain will be discussed (Farahani et al., 2006c). Due to the effectiveness of autocorrelation function in preserving peaks, we will also report the results of using the autocorrelation domain for extracting the first 3 formants of the speech signal (Strope & Alwan, 1998; Farahani et al., 2006a).

In Fig. 14 we have shown the procedure followed to extract feature parameters in autocorrelation phase parameters. Most of the diagram is similar to previously discussed methods with the exception of the calculation of the phase angle, $\theta_y(m,k)$, as mentioned in (15). As it is clear from (15), these features are related only to the phase variations, in contrast to the features based on the magnitude, such as MFCC, that are related to both $|Y(m)|$ and $\theta_y(m,k)$ (Ikbal et al., 2003).

Feature type	Average Recognition Rate (%)			
	Babble	Car	Factory	White
MFCC	63.60	89.44	57.92	38.68
RAS	67.04	90.56	66.40	44.44
TSP	76.16	87.20	66.44	64.36
ACP	77.92	90.76	68.16	66.44
ACPAF	79.28	92.12	70.76	68.64

Table 7. Comparison of average recognition rates for various feature types with babble, car, factory and white noises.

Here again, the first three spectral peak locations and their derivatives are also calculated using the signal autocorrelation spectrum, as explained in 3.3.1 and later added to the feature vector in phase domain. The new coefficients were named *Autocorrelation Peaks and Phase features* (APP).

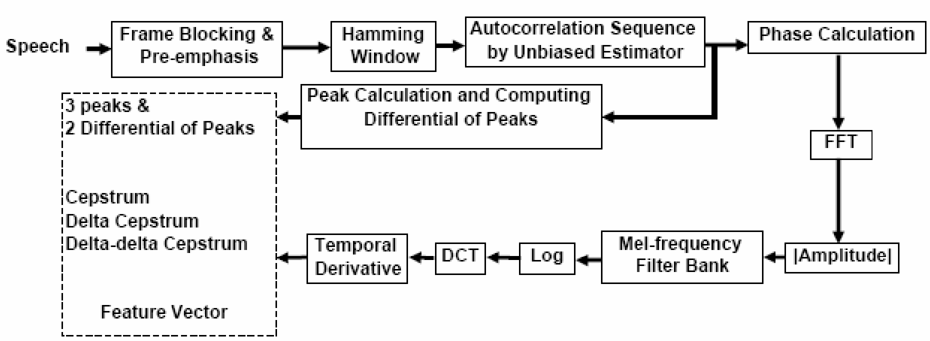
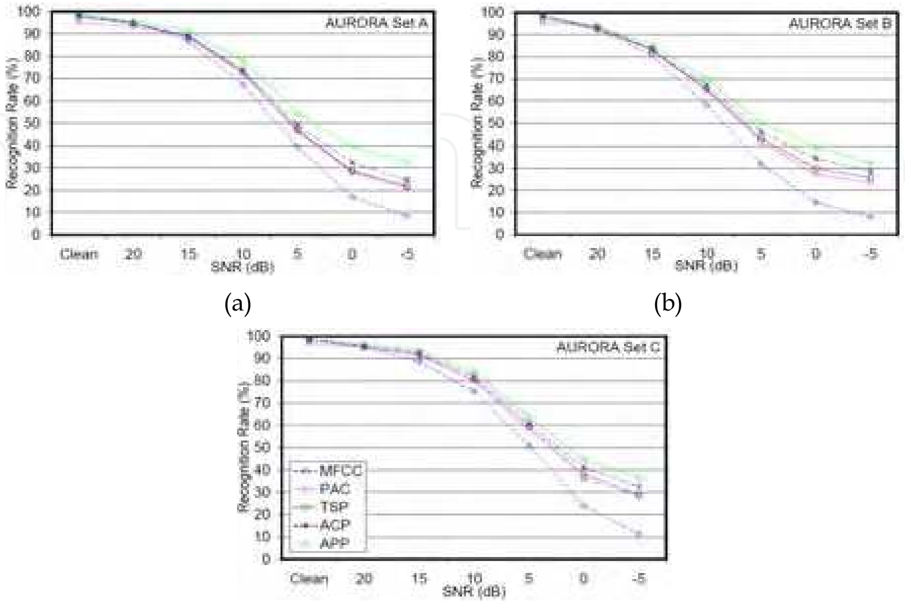

Speech
Frame Blocking & Pre-emphasis
Hamming Window
Autocorrelation Sequence by Unbiased Estimator
Phase Calculation
Peak Calculation and Computing Differential of Peaks
3 peaks & 2 Differential of Peaks
FFT
Cepstrum
Delta Cepstrum
Delta-delta Cepstrum
Temporal Derivative
DCT
Log
Mel-frequency Filter Bank
|Amplitude|
Feature Vector

Figure 14. Front-end diagram to extract features in phase domain along with autocorrelation function.

3.4.1. Experiments

This method was implemented on Aurora 2 task. The feature vectors were composed of 12 cepstral and one log-energy parameters, together with their first and second derivatives and five extra components of which three were for the first three formants and the other two for the frequency peak derivatives. Therefore, the overall feature vector size was 44. In this method, we have used a filter length of L=2.

Fig. 15 displays the results obtained using MFCC, PAC (*phase autocorrelation*) and APP. Also, for comparison purposes, we have included the results of adding spectral peaks to feature vectors calculated using magnitude spectrum and called it TSP (*threaded spectral peaks*) (Strope & Alwan, 1998) and ACP (*autocorrelation peaks*) (Farahani et al., 2006a). According to Fig. 15, APP has led to better recognition rates in comparison to most of the other methods and outperformed other methods for all test sets. This result shows that the autocorrelation domain is more appropriate for peak isolation in phase domain.



(a)

(b)

(c)

Figure 15. Average recognition rates on Aurora 2 database. (a) Test set a, (b) test set b and (c) test set c. The results correspond to MFCC, PAC, TSP, ACP, and APP methods.

In Table 8, we have summarized the average recognition rates obtained for each test set of Aurora 2. As can be seen, average recognition rates for features extracted using the autocorrelation domain with phase features (APP) tops all other results obtained.

Feature type	Average Recognition Rate (%)		
	Set A	Set B	Set C
MFCC	61.13	55.57	66.68
PAC	66.02	62.25	72.60
TSP	66.31	62.86	72.89
ACP	68.03	64.86	74.46
APP	71.83	67.69	76.53

Table 8. Comparison of Average recognition rates for various feature types on three test sets of Aurora 2 task.

4. Conclusion

In this chapter, the importance of autocorrelation domain in robust feature extraction for speech recognition was discussed. To prove the effectiveness of this domain, some recently proposed methods for robust feature extraction against additive noise were discussed. These methods resulted in cepstral feature sets derived from the autocorrelation spectral domain. The DAS algorithm used the differentiated filtered autocorrelation spectrum of the noisy signal to extract cepstral parameters. We noted that similar to RAS and DPS, DAS can better

IntechOpen

preserve speech spectral information for recognition. Experimental results were used to verify the improvements obtained using DAS feature set in comparison to MFCC, RAS and DPS. Its superior performance in comparison to both RAS and DPS is an indication of the rather independent effectiveness of the two steps in reducing the effect of noise. Its combination with CMN has also been found effective. The impact of filter length and type of differentiation on recognition results were also examined.

Also, the performance of DAS and RAS have further been improved by SPFH where the effect of noise has further been suppressed using an extra step of discarding the lower lags of the autocorrelation sequence. The experiments showed the better performance of this new approach in comparison to the previous autocorrelation-based robust speech recognition front-ends.

Techniques based on the above methods and the use of spectral peaks were also discussed in this chapter. The proposed front-end diagrams in autocorrelation domain, ACP and ACPAF, were evaluated together with several different robust feature extraction methods. The usefulness of these techniques was shown and the results indicated that the spectral peaks inherently convey robust information for speech recognition, especially in autocorrelation domain. Better parameter optimization for these two methods can be a basis for the future work as it is believed to have important influence on the system performance.

As discussed, the features extracted in magnitude domain are more sensitive to the background noise in comparison to the phase domain. Appending the frequencies of spectral peaks and their derivatives to feature parameters extracted in phase domain, similar to what was done in magnitude domain, led to even better results in comparison to the parameters extracted using the magnitude spectrum. Once again, the autocorrelation domain was used here for spectral peak extraction.

5. References

Acero, A.; Deng, L.; Kristjansson, T. & Zhang, J. (2000). HMM adaptation using vector Taylor series for noisy speech recognition. *Proc. ICSLP*, 3, 869-872.

Beh, J. & Ko, H. (2003). A novel spectral subtraction scheme for robust speech recognition: spectral subtraction using spectral harmonics of speech. *Proc. ICASSP*, I (648-651).

Bijankhan, M. et al. (1994). FARSDAT-The Speech Database of Farsi Spoken Language. *Proc. 5th Australian International Conference on Speech Science and Technology (SST'94).*

Boll, S.F. (1979). Suppression of acoustic noise in speech using spectral subtraction. *IEEE Trans. on Acoustic, Speech and Signal Processing*. 27 (2), 113-120.

Bozkurt, B. & Couvreur, L. (2005). On the use of phase information for speech recognition. *Proc. EUSIPCO*, Antalya, Turkey.

Bozkurt, B.; Doval, B.; D'Alessandro, C. & Dutoit, T. (2004). Improved differential phase spectrum processing for formant tracking. *Proc. ICSLP*, Jeju, Korea.

Chen, J.; Paliwal, K.K. & Nakamura, S. (2003). Cepstrum derived from differentiated power spectrum for robust speech recognition. *Speech Communication*. 41 (2-3), 469-484.

Cui, X.; Bernard, A. & Alwan, A. (2003). A noise robust ASR back-end technique based on weighted Viterbi recognition. *Proc. Eurospeech*, 2169-2172.

Farahani, G. & Ahadi, S.M. (2005). Robust features for noisy speech recognition based on filtering and spectral peaks in autocorrelation domain. *Proc. EUSIPCO*, Antalya, Turkey.

Farahani, G.; Ahadi, S. M. & Homayounpoor, M. M. (2006a). Use of spectral peaks in autocorrelation and group delay domains for robust speech recognition. *Proc. ICASSP*, Toulouse, France.

Farahani, G.; Ahadi, S. M. & Homayounpoor, M. M. (2006b). Robust feature extraction using spectral peaks of the filtered higher lag autocorrelation sequence of the speech signal. *Proc. ISSPIT*, Vancouver, Canada.

Farahani, G.; Ahadi, S. M. & Homayounpour, M. M. (2006c). Robust Feature Extraction based on Spectral Peaks of Group Delay and Autocorrelation Function and Phase Domain Analysis. *Proc. ICSLP*, Pittsburgh PA, USA.

Farahani, G.; Ahadi S. M. & Homayounpour, M. M. (2007). Features based on filtering and spectral peaks in autocorrelation domain for robust speech recognition. *Computer. Speech and Language*, 21, 187-205.

FARSDAT. FARSDAT Persian speech database. Available from http://www.elda.org/catalogue/en/speech/ S0112.html.

Furui, S. (1986). Speaker-independent isolated word recognition using dynamic features of speech spectrum. *IEEE Trans. on Acoustics, Speech and Signal Processing*. 34 (1), 52-59.

Gales, M.J.F. & Young, S.J. (1995). Robust speech recognition in additive and convolutional noise using parallel model combination. *Computer Speech and Language*. 9, 289-307.

Gales, M.J.F. & Young, S.J. (1996). Robust Continuous Speech Recognition Using Parallel Model Combination. *IEEE Trans. Speech Audio Processing*. 4 (5), 352-359.

Hermansky, H. & Morgan, N. (1994). RASTA processing of speech. *IEEE Trans. Speech Audio Processing*. 2 (4), 578-589.

Hernando, J. & Nadeu, C. (1997). Linear prediction of the one-sided autocorrelation sequence for noisy speech recognition. *IEEE Trans. Speech Audio Processing*. 5 (1), 80-84.

Hirsch, H.G. & Pearce, D. (2000). The Aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions. *Proc. ISCA ITRW ASR*.

HTK. (2002). The hidden Markov model toolkit. Available from http://htk.eng.cam.ac.uk.

Ikbal, S.; Misra, H. & Bourlard, H. (2003). Phase autocorrelation (PAC) derived robust speech features. *Proc. ICASSP*, 133-136, Hong Kong.

Junqua, J-C. & Haton, J-P. (1996). *Robustness in Automatic Speech Recognition: Fundamentals and Applications*, Kluwer Academic Press, Norwell.

Kermorvant, C. (1999). A comparison of noise reduction techniques for robust speech *recognition*. IDIAP-RR99-10.

Kim, D.Y.; Un, C.K. & Kim, N.S. (1998). Speech recognition in noisy environments using first-order vector Taylor series. *Speech Communication*. 24 (1), 39-49.

Lee, C.-H.; Soong, F.K. & Paliwal, K.K. (1996). *Automatic speech and speaker recognition*. Kluwer Academic Publishers.

Leonard, R. (1984). A database for speaker-independent digit recognition. *Proc. ICASSP*, 328-331.

Mansour, D. & Juang, B.-H. (1989a). The short-time modified coherence representation and noisy speech recognition. *IEEE Trans. on Acoustics and Signal Processing*. 37 (6), 795-804.

Mansour, D. & Juang, B.H. (1989b). A Family of Distortion Measures Based Upon Projection Operation for Robust Speech Recognition. *IEEE Trans. on Speech and Audio Processing*. 37 (11), 1659-1671.

Mauuary, L. (1996). Blind equalization for robust telephone based speech recognition. *In: Proceedings of the European Signal Processing Conference*.

Mauuary, L. (1998). Blind equalization in the cepstral domain for robust telephone based speech recognition. *In Proceedings of the European Signal Processing Conference*.

McGinn, D.P. & Johnson, D.H. (1989). Estimation of all-pole model parameters from noise-corrupted sequence. *IEEE Trans. on Acoustics, Speech and Signal Processing*. 37 (3), 433-436.

Moreno, P.J. (1996). *Speech recognition in noisy environments*. PhD Thesis, Carnegie-Mellon University.

Moreno, P.J.; Raj, B. & Stern, R.M. (1996). A vector Taylor series approach for environment–independent speech recognition. *Proc. ICASSP*, 733-736.

Padmanabhan, M. (2000). Spectral peak tracking and its use in speech recognition. *Proc. ICSLP.*

Paliwal, K. K. & Alsteris, L. D. (2003). Usefulness of phase spectrum in human speech perception. *Proc. Eurospeech*, Geneva, Switzerland.

Shannon, B.J. & Paliwal, K.K. (2004). MFCC Computation from Magnitude Spectrum of higher lag autocorrelation coefficients for robust speech recognition. *Proc. ICSLP*, 129-132.

Shen, J.-L.; Hung, J.-W. & Lee, L.-S. (1998). Improved robust speech recognition considering signal correlation approximated by Taylor series. *Proc. ICSLP*.

SPIB. (1995). SPIB noise data. Available from http://spib.rice.edu/spib/select_noise.html.

Strope, B. & Alwan, A. (1997). A model of dynamic auditory perception and its application to robust word recognition. *IEEE Trans. on Speech and Audio Processing*. 5 (5), 451-464.

Strope, B. & Alwan, A. (1998). Robust word recognition using threaded spectral peaks. *Proc. ICASSP*, 625-628, Washington, USA.

Sujatha, J.; Prasanna Kumar, K.R.; Ramakrishnan, K.R. & Balakrishnan, N. (2003). Spectral maxima representation for robust automatic speech recognition. *Proc. Eurospeech*, 3077-3080.

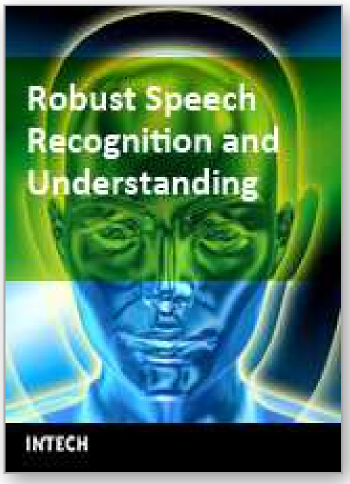
Yapanel, U.H. & Dharanipragada, S. (2003). Perceptual MVDR-based cepstral coefficients (PMCCs) for noise robust speech recognition. *Proc. ICASSP*, 644-647.

Yapanel, U.H. & Hansen, J.H.L. (2003). A New Perspective on feature extraction for robust in-vehicle speech recognition. *Proc. Eurospeech*, 1281-1284.

Yuo, K.-H. & Wang, H.-C. (1998). Robust features derived from temporal trajectory filtering for speech recognition under the corruption of additive and convolutional noises. *Proc. ICASSP*, 577-580.

Yuo, K.-H. & Wang, H.-C. (1999). Robust features for noisy speech recognition based on temporal trajectory filtering of short-time autocorrelation sequences. *Speech Communication*. 28, 13-24.

Zhu, Q.; Iseli, M.; Cui, X. & Alwan, A. (2001). Noise robust feature extraction for ASR using the AURORA2 database. *Proc. Eurospeech.*



Robust Speech Recognition and Understanding
Edited by Michael Grimm and Kristian Kroschel

ISBN 978-3-902613-08-0
Hard cover, 460 pages
Publisher I-Tech Education and Publishing
Published online 01, June, 2007
Published in print edition June, 2007

This book on Robust Speech Recognition and Understanding brings together many different aspects of the current research on automatic speech recognition and language understanding. The first four chapters address the task of voice activity detection which is considered an important issue for all speech recognition systems. The next chapters give several extensions to state-of-the-art HMM methods. Furthermore, a number of chapters particularly address the task of robust ASR under noisy conditions. Two chapters on the automatic recognition of a speaker's emotional state highlight the importance of natural speech understanding and interpretation in voice-driven systems. The last chapters of the book address the application of conversational systems on robots, as well as the autonomous acquisition of vocalization skills.

How to reference
In order to correctly reference this scholarly work, feel free to copy and paste the following:

Gholamreza Farahani, Mohammad Ahadi and Mohammad Mehdi Homayounpour (2007). Autocorrelation-based Methods for Noise-Robust Speech Recognition, Robust Speech Recognition and Understanding, Michael Grimm and Kristian Kroschel (Ed.), ISBN: 978-3-902613-08-0, InTech, Available from: http://www.intechopen.com/books/robust_speech_recognition_and_understanding/autocorrelation-based_methods_for_noise-robust_speech_recognition

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2007 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License, which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen