15

HMM Adaptation Using Statistical Linear Approximation for Robust Speech Recognition

Berkovitch Michael[1] and Shallom D.Ilan[1,2]
[1]*Ben Gurion University*
[2]*AudioCodes*
Israel

1. Introduction

Automatic Speech Recognition (ASR) systems, show degraded recognition performance when train and operate on mismatched environments. This mismatch can be caused due to different microphones, noise conditions, communication channels, acoustical environment etc.
This work is motivated, in part, by the Distributed Speech Recognition (DSR) architecture. The DSR uses ASR server that provides speech recognition services to different devices that may operate in different environments (i.e. mobile devices). Thus, the ASR server must implement environment compensation techniques. The traditional ASR environment compensation techniques use filtering and noise masking, spectral subtraction and multi microphones array. These techniques are usually implemented in the ASR front end and aims to provide clean speech samples to the ASR engine. State of the art ASR systems use Hidden Markov Models (HMM) to represent the stochastic nature of the speech features. These statistical models achieves high recognition rate when trained and tested at the same environmental condition. To add noise robustness for these models, methods such as Maximum Likelihood Linear Regression (MLLR) and Parallel Model Combination (PMC) had been developed. These methods perform an adaptation of the ASR engine to better fit the recognition environment. The main drawback of these methods is there computational complexity and the need of large adaptation data, which makes them not suitable for real-time application.
The environment compensation technique, presented in this chapter, is an extension of the Statistical Linear Approximation (SLA) method originally applied in the feature space to the model space. Using this environment compensation technique, new adapted HMMs set are created Using the clean speech HMMs and the noise model. The adapted HMMs are then used for the recognition of the degraded speech. The proposed robustness method is highly attractive for the Distributed Speech Recognition (DSR) architecture, since there is no impact on the Front End structure and neither on the ASR topology. Experiments, using this method, show high recognition rates in various noise conditions, close to the case of matched training (i.e. recognition and training performed in the same degraded environment).

2. Technique for Robust Speech Recognition

Techniques for Robust Speech Recognition can be divided into three categories.

- Extraction of environmental invariant (robust) features out of the input speech waveform.
- Data compensation ("cleaning") methods of either the input speech or its features .
- Model compensation methods which manipulate the acoustic models to better fit the noise environment .

These noise robustness techniques and there locations in the ASR decoder are shown in the following figure.

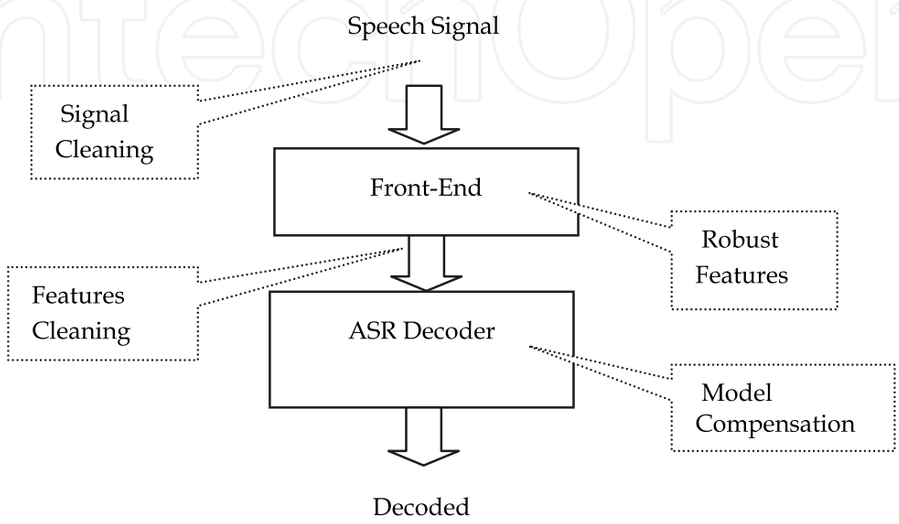

Fig. 1. Techniques for noise robustness scheme

2.1 Environmental robust features

Cepstral Mean Normalization (CMN) is a popular method for channel robust features. CMN efficiently reduce the effects of unknown linear filtering in the absence of additive noise CMN uses the fact that convolutive distortion is additive in the cepstral domain, shown in Eq.(1).

$$y_c = x_c + h_c \tag{1}$$

Here, yc, xc and hc are the corrupted, clean and channel cepstral coefficients respectively.

$$\hat{x}_c = y_c - E[y_c] \tag{2}$$

The CMN subtracts the long-term average of cepstral vectors from the incoming cepstral coefficients, resulting with estimation of the clean cepstral by Eq. (2). CMN can be seen as high-pass filtering of the cepstral coefficients, making them less sensitive to channel and speaker variation. Practically, the non-zero residuals of the mean reflect the channel distortion and speakers variability. This simple and effective procedure is applied to both the training and testing data.

2.2 Data compensation

Data Compensation refers to the process of restoring the clean speech signal or features from the degraded data. Data compensation methods were first introduced to the field of speech enhancement and then were adapted to robust ASR.

Spectral Subtraction is a popular additive noise suppression method. The basic assumption of spectral subtraction is that the effects of the additive noise can be modeled as a bias in the spectrum domain. The corrupted speech expected power spectrum can then be written as

$$|Y_i|^2 = |X_i|^2 + |N_i|^2 \tag{3}$$

The noise bias is estimated using a section of the signal that contains only background noise.

$$|\bar{N}|^2 = \frac{1}{M}\sum_{i=0}^{M-1}|Y_i|^2 \tag{4}$$

The clean speech power spectrum is then estimated using Eq.(5).

$$|\hat{X}_i|^2 = \max\left\{\left(|Y_i|^2 - \alpha|\bar{N}|^2\right), \beta|\bar{N}|^2\right\} \tag{5}$$

Where α and β are adjustment factors.
Spectral Normalization was introduced by stockham et al to compensate for the effects of linear filtering. This algorithm estimates the average power spectra of speech in the training data and then applies the linear filter to the testing speech to "best" convert its spectrum to that of the training speech.
Another well known data compensation method is the Minimum Mean Squared Error (MMSE) uses a-priory statistical model of speech features to derive with a point estimate $\hat{x}$ for the clean speech features [22]. MMSE is defined in its general form using Eq.(6)

$$\hat{x}_{mmse} = \int x \cdot p(x \mid y)dx = \int x \cdot p(x,n,h \mid y)dxdndh \tag{6}$$

Where *y, x* represents the corrupted and clean speech, *n, h* represents the additive and convolutive noise and *p(x,n,h | y)* is the joint conditional distribution. To estimate the clean speech we first need to formulate the relation between the clean and noisy speech signals. This relation is assumed in its general form as

$$y = x + f(x,n,h) \tag{7}$$

Therefore the MMSE estimation can be written as

$$\hat{x}_{mmse} = y - \int f(x,n,h) \cdot p(x,n,h \mid y)dxdndh \tag{8}$$

The joint conditional distribution is approximate using Vector Taylor Series (VTS), moreno had introduced this method to the log-spectral domain.

2.3 Model compensation

In Acoustical Model compensation we accept the presence of noise in the feature domains, and adapt the pattern recognition models to match the new acoustic environment, taking into account the noise statistics and the speech models (trained in the reference clean environment). The recognition is then performed using the models adapted to the noise conditions. Some well known model compensation techniques are the parallel Model compensation and multi-pass retraining.

Parallel Model Combination (PMC) is widely used for HMM compensation, it uses the fact that in the linear domain the corrupted speech is expressed as a summation of the additive noise and clean speech. Thus, the clean speech and additive noise cepstral model parameters (i.e. means and covariance) are transformed into the linear domain. There, they are combined and transformed back into the cepstral domain, as illustrated in Figure 2. Mathematically, the transformation of the mean vector and the covariance matrix between the cepstral-domain and the linear-spectral domain is defined in two steps, first the cepstral coefficients are transformed to the log-spectral using

$$\begin{aligned} \underline{\mu}^{\log} &= C^{-1}\underline{\mu}^{cep} \\ \Sigma^{\log} &= C^{-1}\Sigma^{cep}\left(C^{-1}\right)^{T} \end{aligned} \tag{9}$$

Then they are transformed to the linear-spectral domain using

$$\begin{aligned} \mu_i^{lin} &= \exp\left(\mu_i^{\log} + 0.5\cdot\Sigma_{ii}^{\log}\right) \quad 0 \le i < N \\ \Sigma_{ij}^{lin} &= \mu_i^{lin}\mu_j^{lin}\left(\exp\left(\Sigma_{ij}^{\log}\right) - 1\right) \quad 0 \le i,j < N \end{aligned} \tag{10}$$

In the linear domain the noise and clean speech distribution is log-normal, in this domain the corrupted speech is summation of the noise and clean speech. Although summation of log-normal distributions is not log-normal, the PMC assumes it is log-normal. The corrupted speech means and covariance are then transformed back to the Log domain using the following inverse transform

$$\begin{aligned} \mu_i^{\log} &= \log(\mu_i^{lin}) - 0.5\log\left(\frac{\Sigma_{ii}^{lin}}{\left(\mu_i^{lin}\right)^2} + 1\right) \quad 0 \le i < N \\ \Sigma_{ij}^{\log} &= \log\left(\frac{\Sigma_{ij}^{lin}}{\mu_i^{lin}\mu_j^{lin}} + 1\right) \quad 0 \le i,j < N \end{aligned} \tag{11}$$

Multi/Single Pass Retraining is an off-line model compensation method. In this method the speech models are retrained using speech database recorded in the corrupted environment. If the corrupted recognition environment is known a-priory, one can create off-line synthetic database to retrain the speech models. Since ASR maximizes their performance when trained and tested under the same environment condition, this is probably the best one can do. Unfortunately, this method is not applicable for real-time adaptation.

2.4 Model compensation motivation

Figure 3 illustrates the relations between data and model compensation robustness techniques. Using data compensation we pass in the feature space from the noisy data to the clean data. Using model compensation we pass in the model space from the clean model to

the noisy model. Therefore, using "clean model" with data compensation in the feature space and using noisy data with noise compensated models can be viewed as a symmetric process.

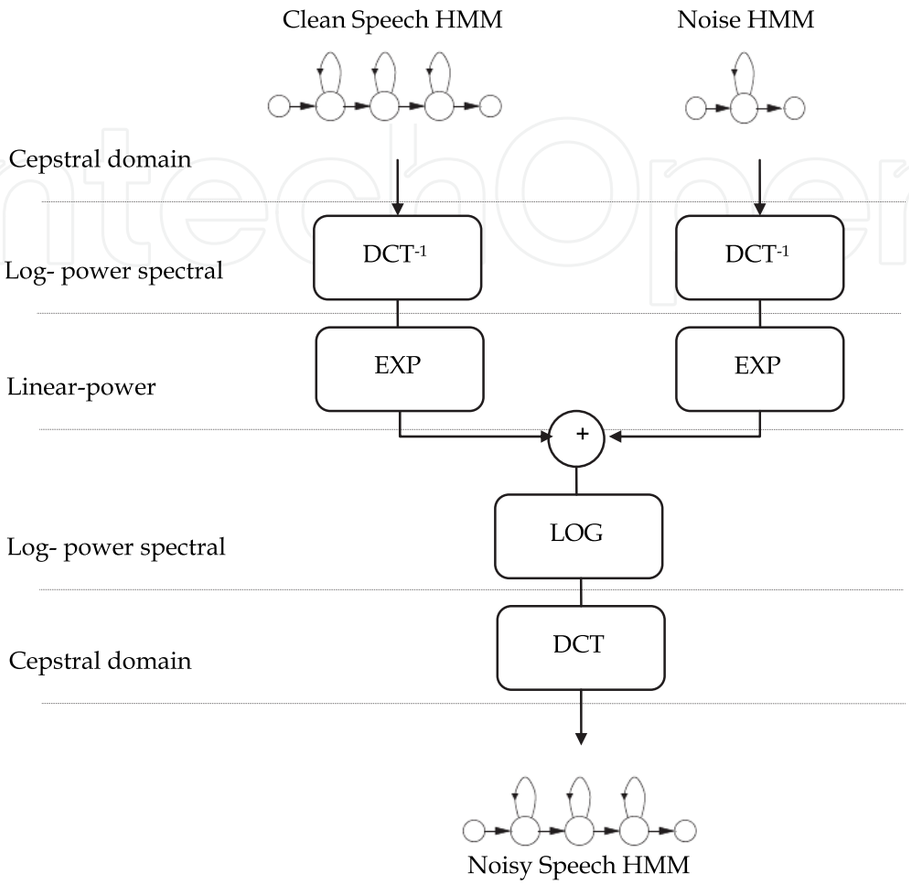

Fig. 2. Parallel Model Combination (PMC) block diagram

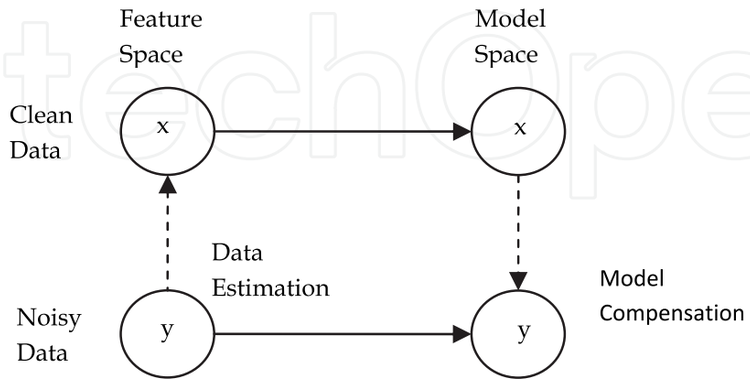

Fig. 3. Model compensation vs. feature estimation.

To illustrate these two noise robustness techniques, we will use a simple binary classifier. The objective of this classifier is to associate each input vector x to one of the two classes *A*,*B*

($A,B \in S$) each with Gaussian pdf, seen in Figure 2.10.The classification problem can be written as

$$\frac{p(x|S=A)\cdot p(S=A)}{p(x|S=B)\cdot p(S=B)} = \begin{cases} >1 & x \in A \\ <1 & x \in B \end{cases} \tag{12}$$

When no noise is added, the error probability of the classifier, i.e. selecting A when B or vice versa, is given by

$$Err(x) = p(A)\cdot \int_{x\in B} p(x|S=A)dx + p(B)\cdot \int_{x\in A} p(x|S=B)dx \tag{13}$$

In the case of noisy observation, x becomes a hidden variable, the classifier job is to overcome the noise, and derive with the correct classification. Data compensation algorithms such as MMSE is used to derive with a point estimation $\hat{x}_{mmse}$ of this hidden variable by

$$\hat{x}_{mmse} = \int_{x} x\cdot p(x|y)dx \tag{14}$$

And then uses the "clean" classifier. In this case the classifier error probability is

$$Err(\hat{x}_{mmse}) = p(A) \int_{\hat{x}_{mmse}\in B} p(\hat{x}_{mmse}|S=A)dx + p(B) \int_{\hat{x}_{mmse}\in A} p(\hat{x}_{mmse}|S=B)dx \tag{15}$$

Model compensation method are used to derive with a new probability model $p(y|s)$, the classifier error probability is then

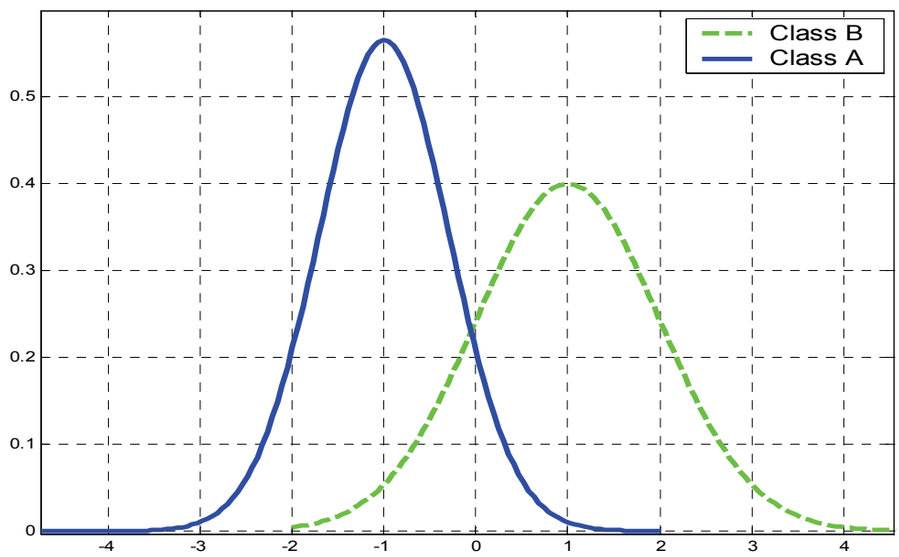

Fig. 4. The binary classification problem

$$
\begin{aligned}
Err(y) &= p(A) \int_{y \in B} p(y|S=A)dy + p(B) \int_{y \in A} p(y|S=B)dy \\
Err(y) &= p(A) \int_{y \in B} \int_{x} p(y|x)p(x|S=A)dxdy \\
&+ p(B) \int_{y \in A} \int_{x} p(y|x)p(x|S=B)dxdy
\end{aligned} \tag{16}
$$

The advantage of using model compensation rather than data compensation is reducing the computational load, since data compensation requires a sampled or frame based compensation, where model compensation requires adaptation only when the noise conditions are changed. Speech recognition in noise can be seen as a complicated version of the binary classifier. The complications arise from the stochastic representation of speech (HMM) and the non-linear effect of noise on speech features.

3. The environment model

The environment model shown in Figure 5, assumes that clean speech (x[m]) is first passes through a transfer channel (h[m]) and then degraded by a additive noise (n[m]), resulting with a corrupted speech (y) expressed by

$$y[m] = x[m] * h[m] + n[m] \tag{17}$$

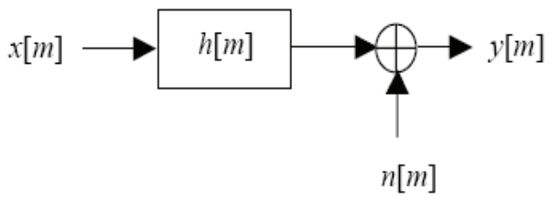

Fig. 5. The environment model.

State of the art ASR uses Mel-Frequency Cepstral Coefficients (MFCC) as their features. MFCC are obtained by passing the spectral magnitude of the noisy speech through a mel-scaled filter bank, taking its logarithm and applying the Discrete Cosine Transform (DCT). Thus, the effect of the environment in the MFCC feature spaces results with the well known environment function

$$y_c = x_c + h_c + C \cdot g(C^{-1}x_c, C^{-1}n_c, C^{-1}h_c) + err \tag{18}$$

Where y_c, x_c, n_c and h_c are the MFCC representation of degraded speech, clean speech, noise and channel respectively. C and C^{-1} are the DCT and inverse DCT matrix. The non-linear function $g(n,x,h)$ is presented by

$$g(x,n,h) = \log(1 + \exp(n - x - h)) \tag{19}$$

The term err in Eq.(18) represents a small amount of residual error due to the neglecting of the cross-correlation between the noise and clean speech.

Figure 6 and Figure 7 shows the effects of additive noise channel on speech in the MFCC domain. One can see that the additive noise changes the contour of the MFCC in non-linear manner. The lower the SNR the more noticeable is the non-linear effect. On the other hand, the channel effect in the MFCC domain can be seen as a-bias tilt, where the lower frequencies are attenuated, while higher frequencies are amplified. These plots also illustrate the de-convolutive property of the DCT transform.

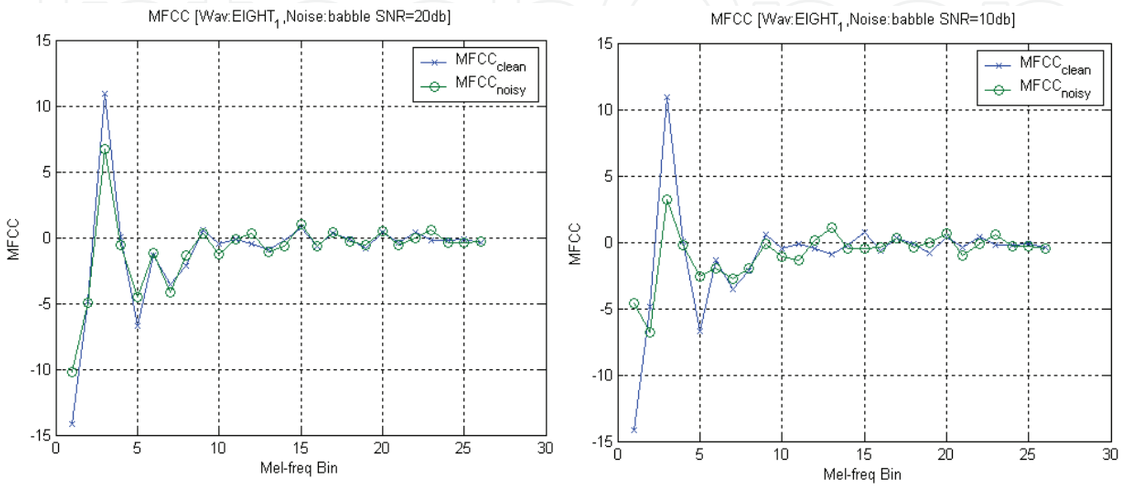

Fig. 6. Effect of additive noise on the MFCC

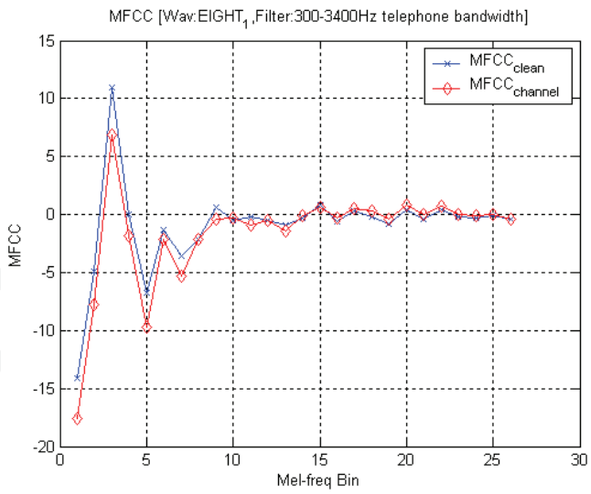

Fig. 7. Effect of linear filtering on the MFCC and Log-spectral features

It can be seen that the degraded speech MFCC is a non-linear transformation of the clean speech, noise and channel MFCC. This non-linearity makes it difficult to find a close analytical solution for the statistics of the degraded signal. The SLA method, shown in this chapter, is used to approximate this non-linearity, and by that, to derive an approximation for the statistics of the degraded speech

3.1 Effect of the environment model on MFCC distribution

When using model compensation it is important to understand the environment effect on the MFCC distribution. Clean speech, noise and channel MFCCs has Gaussian distribution but the degraded speech MFCCs distribution is no longer Gaussian. Nevertheless, the degraded speech MFCCs distribution could still be approximated using Gaussian distribution by

$$\begin{aligned}\mu_{y_c} &= E\left[x_c + f(x_c, n_c, h_c)\right] = \mu_{x_c} + E\left[f(x_c, n_c, h_c)\right] \\ &= \mu_{x_c} + \iiint_{x_c, n_c, h_c} p(x_c)p(n_c)p(h_c)f(x_c, n_c, h_c)dx_c dn_c dh_c \\ \Sigma_{y_c} &= E\left[\left(x_c + f(x_c, n_c, h_c)\right)\left(x_c + f(x_c, n_c, h_c)\right)^T\right] - \mu_{y_c}\left(\mu_{y_c}\right)^T\end{aligned} \tag{20}$$

Where

$$f(x_c, n_c, h_c) = h_c + C\log\left(1 + \exp\left(C^{-1}\left(n_c - x_c - h_c\right)\right)\right) \tag{21}$$

To evaluate the degraded speech MFCCs distribution, a Monte-Carlo simulation had been used.Large number of points, drawn from the clean speech and noise models, were combined together using Eq.(17) to produce the corrupted MFCC. Figure 8 illustrate the corrupted MFCC "true" distribution (solid line) and its Gaussian estimation (dotted line) for different values of Σx. The noise model was set to be Gaussian with μn=0 and Σn=4dB, the clean data was also model using Gaussian with fix mean μx=10 and different covariance Σx=100, 20, 10 and 5dB. The degraded speech MFCCs distribution is clearly non-Gaussian. Though for small Σx values it can be well model using Gaussian distribution. For large Σx values the resulting corrupted distribution can be model using mixture of Gaussians Fortunately, ASR contains GMM thus each Gaussian mixture has small covariance values. Typical value range for the clean speech Gaussian mixture variance is 5-20dB.

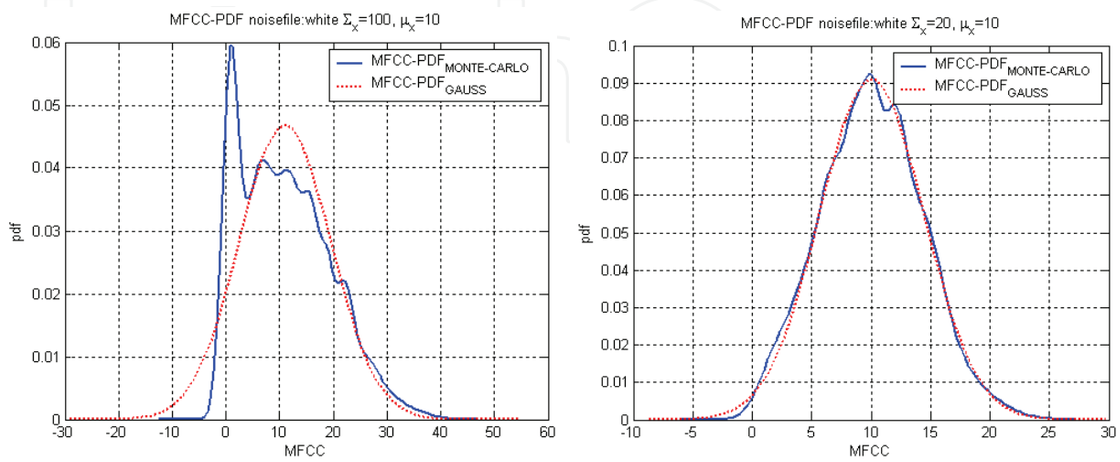

Fig. 8. Effect of the environment function on the distribution of cepstral coefficient

Figure 9 shows the effects of babble noise on the distribution of the third cepstrum coefficient (MFCC3) of the digit /eight/ for different SNRs. The noise affects both the mean and variance, resulting with mean shift and variance reduce. One can see that the lower the SNR the greater is the dissimilarity between the clean and corrupted MFCC PDFs. The effect of noise over MFCC features distribution is evident. Thus, the distributions representing clean speech features do not represent appropriately the corrupted speech features. The following paragraphs introduce the SLA-HMM method to approximate the effect of noise on the clean speech distribution and compensates for it, to achieve high noise robustness.

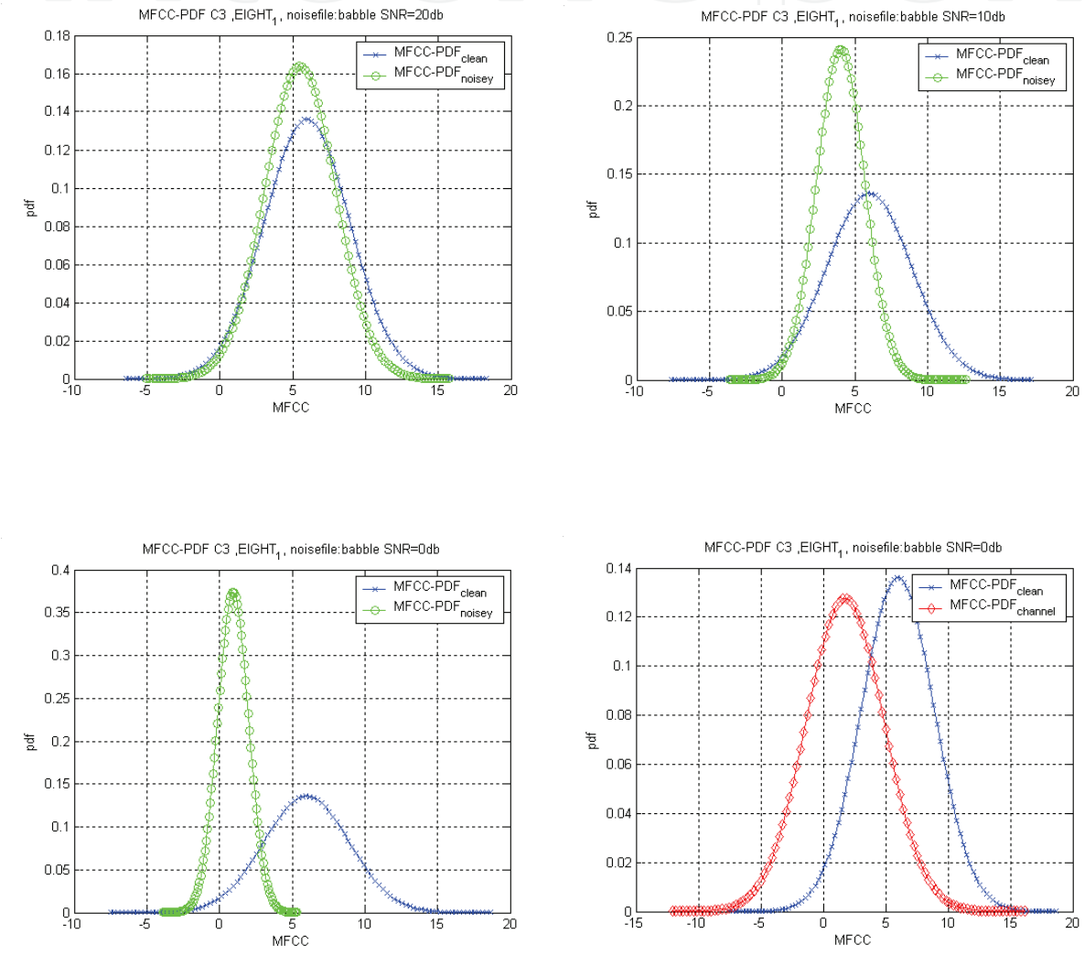

Fig. 9. Clean speech MFCC pdf vs. noisy speech MFCC for different SNRs

4. HMM adaptation for noise robustness

4.1 Statistical linear approximation

The Statistical Linear Approximation (SLA) method, used to approximates a nonlinear function with a linear combination of it variables, around a fix point. This method, assumes that the non-linear function variables are independent random variables with Gaussian

distribution. To derive with the formulation of the SLA approximation, lets define g(x,n,h) as an arbitrary non-linear function with three independent variables, e.g. the clean speech, noise and channel respectively. Define $\tilde{g}(x,n,h)$ to be a linear approximation of $g(x,n,h)$ around a fix point (x_0, n_0, h_0,) given by

$$\tilde{g}(x,n,h) = \mathrm{a}^m \cdot (x - x_0) + \mathrm{b}^m \cdot (n - n_0) + \mathrm{c}^m \cdot (h - h_0) + \mathrm{d}^m \tag{22}$$

Where $\{\mathrm{a}^m,\mathrm{b}^m,\mathrm{c}^m,\mathrm{d}^m\}$ are the linearization coefficients that need to be evaluate. Using the SLA method an optimal, in the Mean Square Error (MSE) sense, linearization coefficients can be found.

The m order Taylor series expansion of the non-linear function $g(x,n,h)$, around a fix point (x_0, n_0, h_0) is written by the following polynomial

$$\begin{aligned} P_g^m(x,n,h) &= \sum_{k=0}^{m} \frac{1}{k!}\left((x-x_0)\frac{d}{dx}+(n-n_0)\frac{d}{dn}+(h-h_0)\frac{d}{dh}\right)^k \cdot g(x_0,n_0,h_0) \\ &= \sum_{k=0}^{m}\sum_{j=0}^{k}\sum_{i=0}^{j} \zeta_{k,j,i}\cdot(x-x_0)^i (n-n_0)^{j-i}(h-h_0)^{k-j} \end{aligned} \tag{23}$$

Where $\zeta_{k,j,i}$ define as

$$\zeta_{k,j,i} = \frac{1}{i!\cdot(j-i)!\cdot(k-j)!}\cdot\frac{d^k g(x_0,n_0,h_0)}{dx^i dn^{j-i} dh^{k-j}} \tag{24}$$

The linear coefficients are then found by minimizing the MSE between the m order Taylor series expansion and the linear approximation, given the assumptions about the variables x, n, h.

$$\begin{aligned} \varepsilon_{rr} &= \arg\min_{\mathrm{a}^m,\mathrm{b}^m,\mathrm{c}^m,\mathrm{d}^m}\left(E[(P_g^m-\tilde{g})^2]\right) \\ \varepsilon_{rr} &= E[(P_g^m)^2]+E[(\tilde{g})^2]-2\cdot E[P_g^m\tilde{g}] \\ \varepsilon_{rr} &= E[(P_g^m)^2]+E[\tilde{g}^2]-2\cdot E[(\mathrm{a}^m(x-x_0)+\mathrm{b}^m(n-n_0) \\ &+\mathrm{c}^m(h-h_0)+\mathrm{d}^m)\cdot P_g^m] \end{aligned} \tag{25}$$

The error function around $(x_0, n_0, h_0) = (\mu_x, \mu_n, \mu_h)$ Expressed by

$$\begin{aligned} \varepsilon_{rr} &= E[(P_g^m)^2]+\left(\mathrm{a}^m\Sigma_x\left(\mathrm{a}^m\right)^T+\mathrm{b}^m\Sigma_n\left(\mathrm{b}^m\right)^T+\mathrm{c}^m\Sigma_h\left(\mathrm{c}^m\right)^T+\left(\mathrm{d}^m\right)^2\right) \\ &-2\left(\mathrm{a}^m E[(x-\mu_x)P_g^m]+\mathrm{b}^m E[(n-\mu_n)P_g^m]+\mathrm{c}^m E[(h-\mu_h)P_g^m]+\mathrm{d}^m E[P_g^m]\right) \end{aligned} \tag{26}$$

The linearization coefficients, which minimizes the MSE, are found by solving the following equations

$$\begin{aligned}
&\frac{d\varepsilon_{rr}}{d\mathrm{a}^m} = 2\mathrm{a}^m\Sigma_x - 2E[(x-\mu_x)\cdot P_g^m] = 0\\
&\frac{d\varepsilon_{rr}}{d\mathrm{b}^m} = 2\mathrm{b}^m\Sigma_n - 2E[(n-\mu_n)\cdot P_g^m] = 0\\
&\frac{d\varepsilon_{rr}}{d\mathrm{c}^m} = 2\mathrm{c}^m\Sigma_h - 2E[(h-\mu_h)\cdot P_g^m] = 0\\
&\frac{d\varepsilon_{rr}}{d\mathrm{d}^m} = 2\mathrm{d}^m - 2E[P_g^m] = 0
\end{aligned} \tag{27}$$

After some algebra, the following expressions to the linearization coefficients are derived

$$\begin{aligned}
\mathrm{a}^m &= \Sigma_x^{-1}\sum_{k=0}^{m}\sum_{j=0}^{k}\sum_{i=0}^{j}\zeta_{k,j,i}\cdot E[(x-\mu_x)^{i+1}(n-\mu_n)^{j-i}(h-\mu_h)^{k-j}]\\
\mathrm{b}^m &= \Sigma_n^{-1}\sum_{k=0}^{m}\sum_{j=0}^{k}\sum_{i=0}^{j}\zeta_{k,j,i}\cdot E[(x-\mu_x)^{i}(n-\mu_n)^{j-i+1}(h-\mu_h)^{k-j}]\\
\mathrm{c}^m &= \Sigma_h^{-1}\sum_{k=0}^{m}\sum_{j=0}^{k}\sum_{i=0}^{j}\zeta_{k,j,i}\cdot E[(x-\mu_x)^{i}(n-\mu_n)^{j-i}(h-\mu_h)^{k-j+1}]\\
\mathrm{d}^m &= \sum_{k=0}^{m}\sum_{j=0}^{k}\sum_{i=0}^{j}\zeta_{k,j,i}\cdot E[(x-\mu_x)^{i}(n-\mu_n)^{j-i}(h-\mu_h)^{k-j}]
\end{aligned} \tag{28}$$

This is further simplified by using the well-known property for Gaussian PDF shown in Eq.(29), where all variables assume to be independent Gaussian.

$$E[(y-\mu_y)^m] = \begin{cases} 0 & \mathrm{m\_}odd \\ 1\cdot 3\cdot\ldots\cdot(m-1)\Sigma_y^{m/2} & otherwise \end{cases} \tag{29}$$

One can see that for m=1 the SLA linear approximation is the same as the Taylor expansion. Higher order of m introduces more statistical information to the approximation, making the approximation more accurate.

4.2 Statistical linear approximation for HMM adaptation

The SLA-HMM adaptation framework, shown in Figure 10, uses the pre-trained clean speech HMMs and the noise model (both in the MFCC feature space), to update each HMM state PDF, using the SLA method. The output of this process is a set of new robust HMMs. These robust HMM have the same structure as the clean HMM, but with updated states distributions.The additive noise is modeled using single Gaussian, which is good approximation for stationary noise. For none stationary noises multi-mode Gaussian can be used. The noise model is trained during the non-voice periods, and updated to reflect the noise

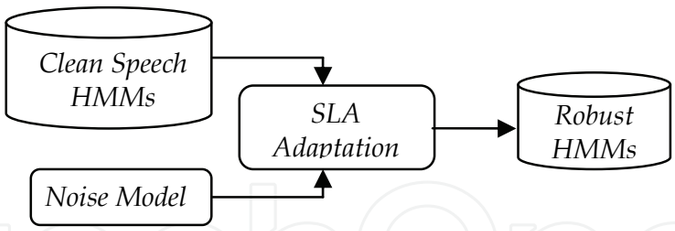

Fig. 10. SLA- HMM adaptation scheme.

To derived with the SLA approximation of the noise robust HMM, we start with an approximation of the environment function, in the MFCC domain, as given by Eq.(30)

$$\underline{y}_c = \underline{x}_c + \underline{h}_c + C \cdot g_l(C^{-1}\underline{x}_c, C^{-1}\underline{n}_c, C^{-1}\underline{h}_c)) \tag{30}$$

Using Eq.(22) the linear approximation of Eq.(30) is

$$\begin{aligned} \underline{y}_c &\approx \underline{x}_c + \underline{h}_c + \mathrm{A}^m(\underline{x}_c - \underline{\mu}_{x_c}) + \mathrm{B}^m(\underline{n}_c - \underline{\mu}_{n_c}) \\ &+ \mathrm{C}^m(\underline{h}_c - \underline{\mu}_{h_c}) + C\mathrm{d}^m \end{aligned} \tag{31}$$

Where the matrices {A^m,B^m,C^m} are given by

$$\begin{aligned} \mathrm{A}^m &= C \cdot diag(\underline{\mathrm{a}}^m) \cdot C^{-1} \\ \mathrm{B}^m &= C \cdot diag(\underline{\mathrm{b}}^m) \cdot C^{-1} \\ \mathrm{C}^m &= C \cdot diag(\underline{\mathrm{c}}^m) \cdot C^{-1} \end{aligned} \tag{32}$$

Using Eq.(32) one can write an approximation to the noise speech mean and covariance matrix as follows

$$\begin{aligned} \underline{\mu}_{y_c} &= \underline{\mu}_{x_c} + \underline{\mu}_{h_c} + \mathrm{C} \cdot \mathrm{d}^m \\ \Sigma_{y_c} &= (I + \mathrm{A}^m)\Sigma_{x_c}(I + \mathrm{A}^m)^T + \mathrm{B}^m\Sigma_{n_c}(\mathrm{B}^m)^T + \\ &(I + \mathrm{C}^m)\Sigma_{h_c}(I + \mathrm{C}^m)^T \end{aligned} \tag{33}$$

ASR also uses the MFCC first and second derivative. Therefore, their means and covariance matrices need to be approximate. The delta and delta-delta MFCC are calculated using

$$\begin{aligned} \Delta x_c(t) &= x_c(t+2) - x_c(t-2) \\ \Delta\Delta x_c(t) &= \Delta x_c(t+2) - \Delta x_c(t-2) \end{aligned} \tag{34}$$

The delta MFCC related to the MFCC by $\Delta x_c \approx \frac{dx_c}{dt}$ 0. Thus, the noisy speech delta MFCC can be written as

$$\Delta y_c \approx \frac{dy_c}{dt} = (I + A^m)\frac{dx_c}{dt} + B^m \frac{dn_c}{dt} \tag{35}$$

Here we use the assumption that h is constant through the speech utterances. The delta and delta-delta MFCC approximated means and covariance matrices are then can be written by

$$\begin{aligned} \mu_{\Delta y_c} &= (I + A^m)\mu_{\Delta x_c} + B^m \mu_{\Delta n_c} \\ \mu_{\Delta\Delta y_c} &= (I + A^m)\mu_{\Delta\Delta x_c} + B^m \mu_{\Delta\Delta n_c} \end{aligned} \tag{36}$$

$$\begin{aligned} \Sigma_{\Delta y_c} &= (I + A^m)\Sigma_{\Delta x_c}(I + A^m)^T + B^m \Sigma_{\Delta n_c}(B^m)^T \\ \Sigma_{\Delta\Delta y_c} &= (I + A^m)\Sigma_{\Delta\Delta x_c}(I + A^m)^T + B^m \Sigma_{\Delta\Delta n_c}(B^m)^T \end{aligned} \tag{37}$$

To evaluate the SLA-HMM approximation, the "true"(using Monte-Carlo simulation) and approximated HMM PDFs were compared using the Kullback-Leibler Divergence (KL). The approximated PDFs derived using the proposed SLA method. Figure 11 shows the resulting KL-measure for SLA order 1-3 as a function of μ_{x_c} where $\sigma_{x_c} = 5$, $\mu_{n_c} = 0$ and $\sigma_{n_c} = 2$.

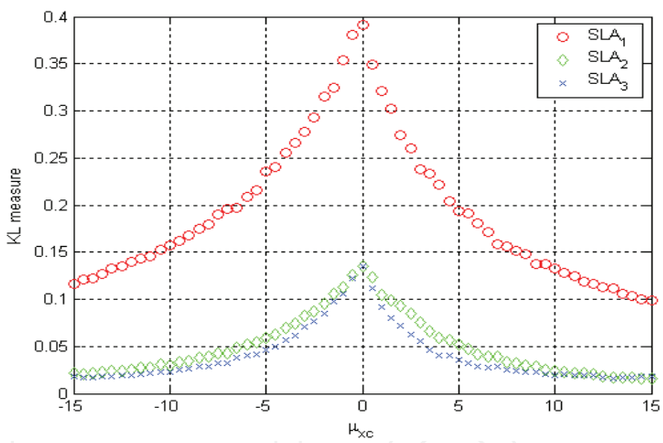

Fig. 11. Kullback-Leibler measure for different SLA order.

The triangular-like shape of the KL-measure indicates that, the larger the distance between the clean and noise MFCC means the more accurate is the approximation, i.e. for $\mu_x >> \mu_n$ the noise can be neglect, resulting with the clean speech PDF and vice versa. One can see that SLA of order 2, 3 yield with more accurate approximation then the VTS, shown by SLA1

5. Experimental results

To investigate the performance of the HMM adaptation algorithm, the well established TIDIGIT speech corpus was used. The TIDIGIT consists of 4480 utterances of isolated digits spoken by men and women for training and testing. Care was taken to balance the training material with respect to an equal number of male and female speakers and equal number of training utterances for all digits.

All speech utterances were recorded without background noise. The noisy speech data-base was created artificially by adding noise sources to the clean speech. The noise sources were taken from the NOISEX-92 database. For each noise source, the average log power of the low (0-1500Hz) and high frequencies band (1500-4000Hz) was calculated. The noise source had been divided into three test groups. Test group A contains high average log power at the low frequency band. Test group B contains high average log power at the high frequency band. Test group C contains non-stationary noises (i.e. babble, machinegun). Figure 12 depict the three noise test groups.

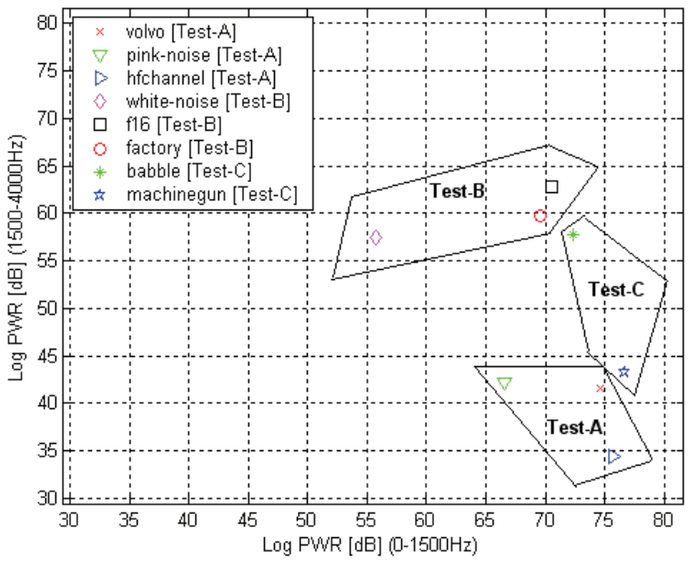

Fig. 12. Test sets noises low-band log power vs. high-band log power

HTK software tool kit had been used to perform all the recognition tests. The ASR HMM structure consists of 4 states, with 4-8 Gaussians per state depending on the available training data. These HMMs were trained using 13-dimensional MFCC feature vector and its delta and delta-delta derivative. The baseline ASR was trained using clean training data. The Baseline ASR HMMs were then retrained, using the noisy speech training data, creating the matched ASR HMMs. For HMM model adaptation algorithm evaluation, the baseline ASR and a matched trained ASR word error rate (WER), can be considered as the upper and Lower performance bounds respectively. The performance of the proposed SLA-HMM adaptation needs to be compare to the performance of matched trained recognizer. Table 1 shows the average WER results of the baseline recognizer for the different noise groups and SNRs.
For the baseline ASR, the lack in noise robustness is highly noticeable especially in the case of wide-band noise (test-b). One can see that for SNRs lower than 10 dB the ASR performance "breaks" for all noise groups.
Table 2 shows the average WER results of the matched recognizer. As expected, the matched ASR yields with high noise robustness, comparing to the base line recognizer. Nevertheless, at SNRs lower than 5dB the performance improvement starts to fail. Thus, it is expected that at low SNRs the proposed model compensation method will show the same behavior . One of the reasons for this ASR behavior is that at low SNR , ASR model topology changes may be require.

SNR [dB]	Test-A	Test-B	Test-C
Clean	0.7	0.9	0.6
20	1.0	3.4	1.4
15	2.2	11.4	4.9
10	7.3	38.0	17.1
5	19.0	67.2	34.0
0	30.5	86.8	41.8
-5	42.0	89.6	49.6
Avg	14.7	42.5	21.3

Table 1. Baseline ASR WER[%]

SNR [dB]	Test-A	Test-B	Test-C
Clean	0.6	0.9	0.8
20	0.5	0.9	1.2
15	0.7	1.8	2.6
10	1.6	5.2	5.0
5	3.6	15.0	12.4
0	8.4	42.0	36.1
-5	2.6	11.0	9.7
Avg	0.6	0.9	0.8

Table 2. Matched ASR WER[%]

5.1 Evaluation of SLA-HMM adaptation

For the evaluation of the SLA-HMM adaptation algorithm, the baseline HMMs were used to represent the clean speech models. The noise was model using a mixture of gaussian (up to four), trained by the noisy speech utterances first 20 frames, which contains noise only. To evaluate the SLA-HMM performance, the SLA-HMM ASR word error rate (WER) measurements were compared to the matched HMM ASR WER. The following figures present the average WER results attained using the SLA-HMM algorithm at different noise conditions. Each figure presents different noise group average WER results versus SNR. Results attained using high order SLA-HMM (three and above) had been omitted, as they show no or little performance improvement.

One can see that the proposed noise robustness algorithm improves the ASR performance in all of the tested noise conditions. The proposed algorithm shows high noise robustness, with performance results close to the matched trained ASR (represented by the solid line).

The experiments show that SLA-HMM of order 3 yields with the highest recognition rates, outperforming the VTS algorithm (represents by SLA-HMM of order 1). Thus, high order SLA approximation increases the algorithm accuracy, as expected.

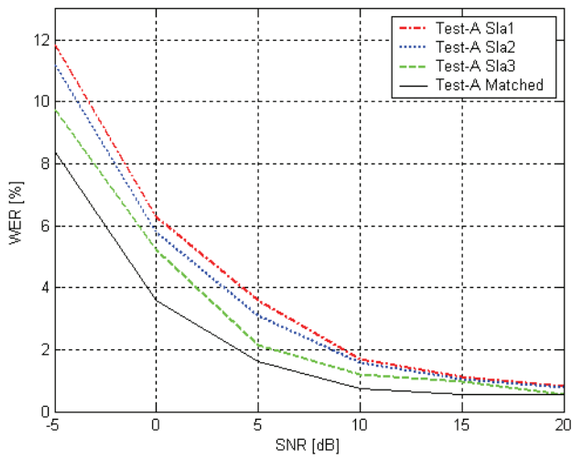

Fig. 13. SLA-HMM average WER vs. SNR using Test-A

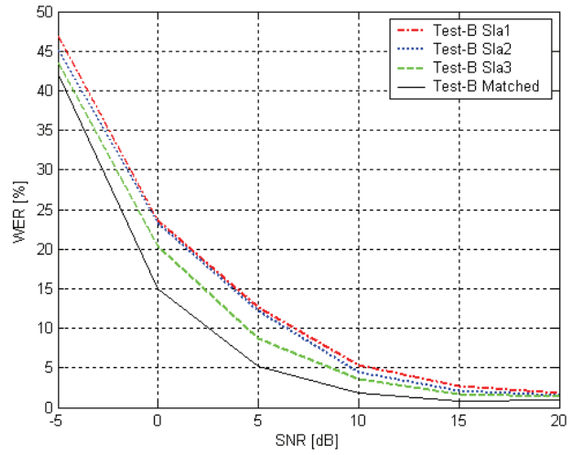

Fig. 14. SLA-HMM average WER vs. SNR using Test-B

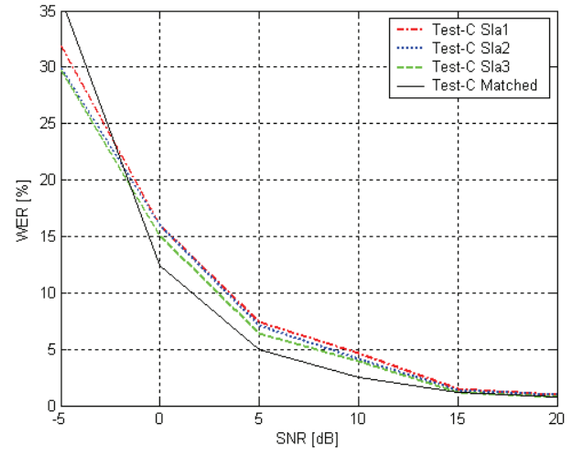

Fig. 15. SLA-HMM average WER vs. SNR using Test-C (noise model 4-GMM)

6. Conclusion

This chapter presents a robust ASR method based on model adaptation using the Statistical Linear Approximation. The proposed SLA-HMM model adaptation had achieving an average of 70% WER improvement with respect to the baseline ASR. The proposed algorithm achieves high robustness performance in variety of environmental conditions compared to the baseline recognizer. The proposed model-compensation had also shown good performance compare to the matched trained ASR

The proposed robustness algorithm has an advantage in Distributed Speech Recognition (DSR), since no changes are required to the front-end terminals and to the ASR topology, as the adaptation is done on HMMs models on the server side.

Further work will put emphasis on improving the robustness performance at very low SNR, where this algorithm had shown some decrease in performance. This performance decrease can be cope by Appling changes to the HMM topology. The proposed noise robustness algorithm had been tested using isolated word recognizer, the same algorithm can be evaluated using phone level continues speech recognizer

7. References

Acero, L., (1993). Acoustical and Environmental Robustness in Automatic Speech Recognition. Kluwer Academic Publishers

Acero, L., Kristjansson, T. & Zhang, J. (2000). Hmm Adaptation using Vector Taylor Series for Noisy Speech Recognition. *Proc ICSLP*, Vol.3, pp. 869-872

Deng, L.; Acero, A.; Jiang, L.; Droppo, J. & Hunag, X.-D. (2001). High-perfromance robustspeech recognition using stereo training data. *Proceedings of ICASSP*, Vol.4, pp. 301-304

Fujimoto M. & Ariki Y., (2004). Robust Speech Recognition in Additive and Channel Noise Environments Using GMM and EM Algorithm. *Proceedings of ICASSP*, Vol. 1, pp. 941-944

Gales, M. J. F. & Young, S. J. (1996). Robust continuous speech recognition using parallel model combination. *IEEE Trans. Speech and Audio Proc,* pp. *352-359*

Hamaguchi S.,; Kitaoka N., & Nakagawa S., (2005). Robust Speech Recognition under Noisy Environments based on Selection of Multiple Noise Suppression Methods, *IEEE-EURASIP (NSIP2005)*, pp.308–313

Kim, N. S., (1998). Statistical linear approximation for environment compensation. *IEEE Signal Processing Letters*, Vol. 5, pp. 8-10

Martin, F.,; Shikano K. & Minami Y. (1993). Recognition of noisy speech by composition of hidden Markov models, *Proceedings of. EuroSpeech,* Vol.4, pp. 1031–1034

Macho, D.,; Mauuary, L.,; Noe, B.,; Cheng, Y. M.,; Ealey, D.,; Jouvet, D.,; Kelleher, H.,; Pearce, D. & Saadoun, F. (2002). Evaluation of a Noise-Robust DSR Front-End on Aurora Databases. *Proceedings of ICSLP*, pp. 17-20

Moreno, P., (1996). Speech Recognition in Noisy Environments, Ph.D. thesis. Carnegie Mellon University

Varga, A. P.,; Steeneken, H. J. M.,; Tomlinson, M., & Jones, D., (1992). The NOISEX-92 Study on The Effect of Additive Noise on Automatic Speech Recognition. Technical Report DRA Speech Research Unit

Young, S. J.,; Evermann, G.,; Gales, M.,; Hain, T.,; Kershaw, D.,; Liu, X.,; Moore, G.,; Odell, J.,; Ollason, D.,; Povey, D.,; Valtchev, V. & Woodland, P. C., (2006). The HTK Book (for HTK Version 3.4), University of Cambridge

Speech Technologies

Edited by Prof. Ivo Ipsic

ISBN 978-953-307-996-7
Hard cover, 432 pages
Publisher InTech
Published online 23, June, 2011
Published in print edition June, 2011

This book addresses different aspects of the research field and a wide range of topics in speech signal processing, speech recognition and language processing. The chapters are divided in three different sections: Speech Signal Modeling, Speech Recognition and Applications. The chapters in the first section cover some essential topics in speech signal processing used for building speech recognition as well as for speech synthesis systems: speech feature enhancement, speech feature vector dimensionality reduction, segmentation of speech frames into phonetic segments. The chapters of the second part cover speech recognition methods and techniques used to read speech from various speech databases and broadcast news recognition for English and non-English languages. The third section of the book presents various speech technology applications used for body conducted speech recognition, hearing impairment, multimodal interfaces and facial expression recognition.

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

IntechOpen