Parallel Model Combination - A Primer, Robustness in Language and Speech Processing, T-61.182

Karthikesh Raju,
Student No.: 56689P,
Lab. of Computer & Information Sc.,
Helsinki University of Tech.,
karthik@james.hut.fi

June 11, 2003

1 Introduction

Modern speech recognition systems are designed to work under many situations. This acts as a boon and a bane as it is difficult to control certain acoustic environments. Difference in acoustic environments can cause a mismatch between the training and test conditions. This can affect the performance of these systems. What is required is a speech recognizer that is robust to changes in the environment that it operates in.

Speech can be characterized by a slowly changing waveform, and the task of a recognizer is to produce an estimate of the occurred word (termed the word string) associated with this waveform. Four classes of speech recognition systems have evolved over years:

1. Template Based : Recognition here is performed by matching the unknown speech utterance to any of the templates of speech patterns stored in the recognizer. These techniques are inherently incapable of modeling the wide variations encountered in speech.
2. Knowledge-Based: Speech spectrograms are analyzed and are encoded as rules. These rule based systems have met with limited success for want of quantifying expert knowledge and combining the different views from experts.
3. Stochastic: Uncertainty in speech is modeled by probabilistic models.
4. Connectionist: A simple computing unit stores knowledge about the speech signal. Now speech can be regarded as finding the right combination of these simple units. A wide range of topologies have been investigated. RNNs and MLPs along with Hidden Markov Models (HMMs) are the most popular speech recognizition systems now-a-days.

1.1 Environmental Effects on Speech

A Speech recoginizer's performance is severely affected by changes in the acoustic environment that it operates in. The following lists some of the environmental effects that severely degrade the quality of a speech recognizer:

1. Wide range of noise sources, that are uncorrelated with the speech signal. They can be short lived and non-stationary as cars passing, door slamming etc, or can be continuous as fans etc.
2. Speakers alter their speech pattern in response to stress - Lombard effect.
3. Input equipment - i.e microphone and the transmission channel affects the speech signal.

In a general framework, considering all noise sources as independent, the corrupted-speech signal, $y(t)$, can be written as:

$$y(t) = [([s(t)|^{stress}_{Lombard}]_{n_1(t)} + n_1(t)) * h_{mike}(t) + n_2(t) * h_{channel}(t)] + n_3(t) \tag{1}$$

where $n_1(t)$ is the background noise, $h_{mike}(t)$ is the impulse response of the microphone, $n_2(t)$ and $h_{channel}(t)$ are the additive noise and the impulse response, respectively, of the channel, and $n_3(t)$ is the receiver noise.

With certain assumptions, and ignoring the effect of speaker stress, combining all other noise sources, one can rewrite the above equation as:

$$y(t) = s(t) * h(t) + n(t) \tag{2}$$

$h(t)$, the convolutional noise component, contains all the correlated aspects of noise, and n(t), is the total additive noise.

2 Speech Parameterization

Hidden Markov Models (HMMs) are usually used for speech recognizition. Speech is usually converted from its raw waveform to some feature vector before HMMs are used. This involves, digitization of the signal, and then converting the digitized signal to a set of feature vectors. The feature vector contains all the useful speech information, while the data rate is reduced. Cepstral Parameters are generally used in modern speech recognition systems. The conversion is usually as follows

$$o(t) \overset{FFT}{\Longrightarrow} \mathbf{o}(f,\tau) \overset{\sum||^{\gamma}}{\Longrightarrow} \mathbf{O}(\tau) \overset{\log()}{\Longrightarrow} \mathbf{O}^l(\tau) \overset{\mathbf{C}}{\Longrightarrow} \mathbf{O}^c(\tau) \tag{3}$$

where the sample at time is $t : o(t)$, $\mathbf{o}(f,\tau)$ is the FFT value at frequency f of the vector τ, $\sum|.|^{\gamma}$ indicates taking the magnitude of the FFT values and raising them to γ before summing, C is the matrix associated with discrete Cosine Transform (DCT). The three domains indicated by the above formula are called:

1. Linear-Spectral Domain ($\mathbf{O}(\tau)$)
2. Log-Spectral Domain ($\mathbf{O}^l(\tau)$)
3. Cepstral Domain ($\mathbf{O}^c(\tau)$)

Observations in any of these domains can be used in the models. However, in the Log-Spectral domain assumptions that feature vectors are independent of one another is rather weak. This leads to full covariance matrices when using multi-variate Gaussian distributions for modeling. In the Cepstral domain this independence is well defined. These parameters are a more compact representation than the other two and hence they allow truncation of the feature vector without loss in performance. This results in a model with less parameters.

The above first order assumption that each observation vector is independent of the preceding one does not hold for large vocabulary speech recognition. Dynamic coefficients may be incorporated into the feature vector to capture some of the correlation between vectors. There are many forms of dynamic parameters, but the most popular form of them are delta parameters, $\Delta\mathbf{O}^c(\tau)$, calculated as simple differences or as a linear regression over a number of frames as:

$$\Delta\mathbf{O}^c(\tau) = \frac{\sum_{t=d}^{D} t(\mathbf{O}^c(\tau+t) - \mathbf{O}^c(\tau-t))}{2\sum_{t=d}^{D} t^2} \tag{4}$$

where $d = 1$ for regression and $d = D$ for simple differences, D being the width of the dynamic parameters.

Extending the same to another state of acceleration, delta-delta parameters, $\Delta^2\mathbf{O}^c(\tau)$, are also calculated using simple differences or the linear regression of delta parameters.

Of-course, the use of dynamic parameters violates the stationarity assumption behind the use of HMMs, as here one state may depend on the previous or following states.

3 Parallel Model Combination

Parallel model combination (PMC) is a model based compensation scheme. It approximates the "best" compensation scheme to train on noise-corrupted speech data. This scheme can compensate static and dynamic coefficients in both additive and convolutive noise environments.

The basis for any model based scheme is to choose a model for the corrupted-speech. Here, as done in most speech processing tasks, a standard Hidden Markov Model (HMM) with Gaussian emission probabilities are used. The basic assumption is a factorizable joint probability distribution for the states. Written in terms of likelihoods, this means

$$\mathcal{L}(\mathbf{O}^c(\tau)|\mathcal{M}_s, \mathcal{M}_n) = \sum_{m=1}^{M} \hat{c}_m \mathcal{N}(\mathbf{O}^c(\tau); \hat{\mu}_m^c, \hat{\sigma}_m^c) \tag{5}$$

where $\hat{c}^m, \hat{\mu}_m^c, \hat{\sigma}_m^c$ are the weights, the means and the variances. It is necessary to estimate or decide the number of mixture components M, the same way as in standard HMM training. This need not be exact here, due to the complex nature of the corrupted-speech distribution.

The usual approach to estimate the new model parameters would be to add noise to the clean training data at the waveform level and re-estimate the parameters, but this is practically infeasible due to the requirement of

1. The whole training database

2. Sufficient amount of noise samples

3. Enormous computational power to perform training, every time the noise changes

Assuming that the clean speech model contains sufficient information about the statistics of the training data, a model of the background noise can be generated using the noise samples that best approximates the background noise. These models can then be combined to accurately estimate the corrupted-speech model. This form the basis of parallel model combination.

3.1 Basic PMC

Basic PMC is illustrated in Fig. 1. Clean speech models and noise models are combined in the Linear or Log-Spectral domains using a function that approximates this combination effectively. This function is the *mismatch* function. This mismatch function can also be effectively used even when cepstral parameters are used. This requires transformation back and forth from the Log-Spectral to Cepstral domain as described in Section 2.

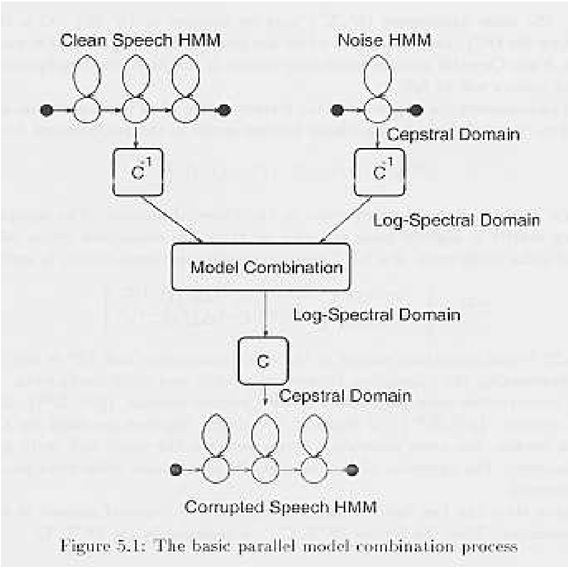

Figure 1: Basic Parallel Model Combination

3.2 Mismatch Function

A *mismatch function* describes how noise affects the clean speech parameters, under the following assumptions

1. Speech is independent of the noise
2. They are additive in the time domain and power spectrum level

3. A single Gaussian or multiple Gaussian mixture model contains sufficient information to represent the distribution of observation vectors in the Cepstral or Log-Spectral domain

4. Frame/State alignment is not altered by noise

3.2.1 Mismatch function: Static Parameters + Additive Noise

The static noise-corrupted speech "observations" in the Log-Spectral domain are given by the mismatch function

$$\begin{aligned} \mathcal{O}_i^l(\tau) &= \mathcal{F}(S_i^l(\tau), N_i^l(\tau)) \\ &= \log(g \exp(S_i^l(\tau)) + \exp(N_i^l(\tau))) \end{aligned} \tag{6}$$

where g is the compensation term for the level differences between the clean speech and the noisy speech.

3.2.2 Mismatch Function: Dynamic Parameters + Additive Noise

Dynamic parameters are required for medium to large vocabulary speech recognition. These parameters provide information about the correlation between successive frames. If statistics about the correlation between successive frames are known, then the statistics for the corrupted delta and delta-delta parameters can be estimated using the static parameter mismatch function described above. However, this requires the determination of correlation between successive frames and it necessitates the use of full or partial covariance matrices. This can be avoided by re-expressing the delta parameters in terms of parameters of known statistics (and at times if necessary independent) as:

$$\begin{aligned} \Delta\mathcal{O}_i^l(\tau) &= \mathcal{F}^{\Delta}(S_i^l(\tau-w), N_i^l(\tau-w), \Delta S_i^l(\tau), \Delta N_i^l(\tau)) \\ &= \log(\exp(\Delta S_i^l(\tau) + S_i^l(\tau-w) + g^l) + \exp(\Delta N_i^l(\tau) + N_i^l(\tau-w))) \\ &\quad - \log(\exp(S_i^l(\tau-w) + g^l) + \exp(N_i^l(\tau-w))) \end{aligned} \tag{7}$$

where $g^l = \log(g)$. This equation above involved rewriting the corrupted-speech Cepstral delta coefficients in terms of static and delta coefficients of the clean speech and interfering noise. The expression for the delta coefficient at time τ is dependent on the static coefficients at time $\tau - w$.

Similar mismatch functions can be derived for the delta-delta parameters. For more information refer to [1],[2].

3.2.3 Mismatch Function : Additive + Convolutive Noise

Assuming that a channel distorts the clean speech in addition to the noise, the corrupted speech observation is now

$$\mathcal{O}_i(\tau) = H_i S_i(\tau) + N_i(\tau) \tag{8}$$

where **H** is the channel impulse response, in other words it represents the channel difference between training and testing. It is assumed that the convolutional noise is not time varying and is independent of the incoming signal. Furthermore, ignoring the variations of the convolutional noise within a frequency bin, the mismatch function can be written as

$$\begin{aligned} \mathcal{O}_i^l(\tau) &= \mathcal{F}^H(S_i^l(\tau), N_i^l(\tau), H_i^l) \\ &= \log(\exp(S_i^l(\tau) + H_i^l) + \exp(N_i^l(\tau))) \end{aligned} \tag{9}$$

The above equation is based on the assumption that the noise model statistics are known. Since **H**, the channel statistics is not known, estimation of them are necessary. A technique to estimate the statistics of the channel is described in [1].

The additive and convolutional noise mismatch function can be related to the additive noise mismatch functions for both the static and the dynamic case. In the dynamic case, the dynamic parameters are not altered by the addition of convolutional noise, provided the noise is not time varying.

Hence, after estimating **H**, the static speech parameters may be shifted and the same estimation techniques used as in the additive noise case, can be used here.

3.2.4 Variations of the Mismatch function

There exists several variations of the mismatch function like:

1. Mismatch function in the magnitude domain
2. Matrix-Based Dynamic Coefficients
3. Continuous Time Dynamic Coefficients

In the above techniques, the mismatch function varies significantly from the simple cases described in the previous paragraphs. [1] describes and derives the mismatch function in these cases.

3.3 Training Statistical Data

Unlike usual HMM training where observations are available, here HMMs are trained using the statistics about the speech and noise. The standard HMM training algorithms are modified accordingly to allow for training of models on statistical data. ML estimation is used for the same.

1. Re-estimation of new mean, variance of the Mixture M_m of state s_j of the corrupted speech model are expressed now in terms of expected values of observations, instead of summations.

2. PDFs now describe the observations for a particular state. PDFs are a function of known statistics, represented by the clean speech and noise model.
3. The state transition matrix contains temporal information, which remains valid as long as frame/state alignment is not altered by the noise (both additive and convolutive).

Hence the parameters of the corrupted speech can be calculated by the use of the above PDF, assuming a multiple Gaussian mixture model.

Standard iterative procedures for HMM training can be used here, keeping the above alterations in mind. However, if the frame/state alignment is unaltered, a single pass, non-iterate estimate is also possible. Here all the mixtures are treated independently, as if they are in distinct states. This technique is the *non-iterative PMC*.

On the other hand, if there is an alteration in the frame/state alignment, the PDF is no-longer factorizable. In this case an iterative estimation scheme is needed, and it is referred to as *iterative PMC*.

3.4 Non-Iterative PMC

Non-iterative PMC assumes that the frame/state alignment is unaltered. This also means that it is not necessary to re-estimate the transition probabilities or mixture component weights. Hence, this is the simplest and computationally most efficient PMC method. Figure 2 shows how components combine for a particular speech state and noise state.

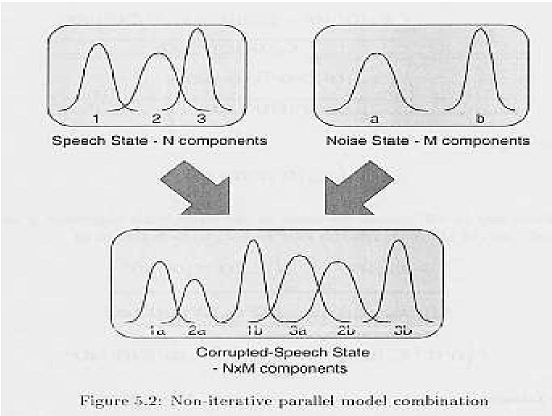

Figure 2: Non-Iterative Parallel Model Combination

The speech state has 2 components and the noise state has 3 components, and finally the corrupted-speech has six components.

Now the expressions for mean and variance for the static parameters are

$$\begin{aligned} \hat{\mu}_i^l &= \mathcal{E}\{\mathcal{F}(S_i^l, N_i^l)\} \\ \hat{\Sigma}_{ij}^l &= \mathcal{E}\{\mathcal{F}(S_i^l, N_i^l)\mathcal{F}(S_j^l, N_j^l)\} - \hat{\mu}_i^l\hat{\mu}_j^l \end{aligned} \quad (10)$$

For the dynamic parameters, the mean and the covariance are given by

$$\begin{aligned}
\hat{\mu}_i^l &= \mathcal{E}\{\mathcal{F}^\Delta(S_i^{*l}, N_i^{*l}, \Delta S_i^l, \Delta N_i^l)\} \\
\hat{\Sigma}_{ij}^l &= \mathcal{E}\{\mathcal{F}^\Delta(S_i^{*l}, N_i^{*l}, \Delta S_i^l, \Delta N_i^l)\mathcal{F}^\Delta(S_j^{*l}, N_j^{*l}, \Delta S_j^l, \Delta N_j^l)\} - \Delta\hat{\mu}_i^l \Delta\hat{\mu}_j^l
\end{aligned} \tag{11}$$

where $*$ indicates that the statistics are generated at time $\tau - w$. Delta-Delta parameters have a similar form.

Unfortunately, since these expressions are based on PDFs, no closed form expressions for either the compensated mean, or covariance for the static, delta or delta-delta parameters exists. Hence, to obtain the values for the above expressions, necessary approximations about the distribution are made. This leads to the following two approaches:

3.4.1 Log-Normal Approximation

Log-normal approximation is based on the assumption that *sum of two log-normally distributed variables are also approximately log normally distributed*. This implies that estimates of the mean and variance of the corrupted-speech model in the Linear-Spectral domain is sufficient to derive the corrupted-speech model in the Cepstral domain.

Given that, the speech and noise are independent (Assumptions from Section 3.2), then the corrupted speech parameters can be written as

$$\begin{aligned}
\hat{\mu} &= g\mu + \tilde{\mu} \\
\hat{\Sigma} &= g^2\Sigma + \tilde{\Sigma}
\end{aligned} \tag{12}$$

where the parameters (μ, Σ) are associated with the Log-normal distribution associated with the Gaussian distribution μ^l, Σ^l and the parameters $(\tilde{\mu}, \tilde{\Sigma})$ are related to the noise parameters.

For clean speech, the relation between the Linear-spectral and Log-Spectral domain parameters can be written as

$$\begin{aligned}
\mu_i &= \exp(\mu_i^l + \Sigma_{ii}^l/2) \\
\Sigma_{ij} &= \mu_i\mu_j[\exp(\Sigma_{ij}^l) - 1]
\end{aligned} \tag{13}$$

Since the corrupted speech is assumed to be log-normally distributed in the Linear-Spectral domain, the required distribution in the Linear-Spectral domain, $(\hat{\mu}^l, \hat{\Sigma}^l)$, may be obtained as

$$\begin{aligned}
\hat{\mu}_i^l &= \log(\mu_i) - 0.5\log(\frac{\hat{\Sigma}_{ii}}{\hat{\mu}_i^2} + 1) \\
\hat{\Sigma}_{ij}^l &= \log(\frac{\hat{\Sigma}_i j}{\hat{\mu}_i\hat{\mu}_j} + 1)
\end{aligned} \tag{14}$$

which is inverse of Eq(13).

Of-course the above technique for approximating the pdf has the following constraints:

1. The sum of two log-normally distributed variable are not itself log-normally distributed
2. The first two moments are only matched

Usually the log-normal approximation provides a good approximation of the corrupted means, but calculation of correlations between the terms is a complicated function of speech and noise parameters and hence the Log-Normal approximation cannot be applied to compensate the delta parameter covariance matrix, or the delta-delta parameters.

3.4.2 Log-Add Approximation

Log-Add approximation is a simple approximation based on the assumption that the variances are small, and so the static parameters can be written as

$$\hat{\mu}_i^l \approx \mathcal{F}(\mu_i^l, \tilde{\mu}_i^l) \quad = \quad \log(\exp(\mu_i^l) + \exp(\tilde{\mu}_i^l))$$

This method is analogous to a simple and efficient implementation of a hypothesized Wiener filter, with means obtained from HMMs, and thus can be re-written as

$$\hat{\mu}^c \approx \hat{\mu}^c + \mathbf{C}[\log(\exp(\mu_i^l) + \exp(\tilde{\mu}_i^l)) - \mu^l]$$

The log-add approximation can also be applied to the dynamic parameters, when the effect of variance of the data on the non-linearity of the mismatch function can be ignored.

Using Log-Add approximation has the added advantage that the covariance matrix is not used in the compensation. This frees the mapping of the covariance matrix from and to the Cepstral domain. Since this procedure involves inverse covariances, a significant amount of computational load can be reduced.

3.4.3 Alternative

The other alternative to the Log-Normal and Log-Add approximation can be the use of numerical integration techniques to approximate the PDF. This involves two-dimensional integration and four-dimensional integration to estimate the means and covariances. For static parameters and even higher numbers of dimension to integrate over for the dynamic parameters. This is computationally very expensive.

3.5 Iterative PMC

Non-iterative PMC does not provide "true" ML estimates of the corrupted-speech distributions. The implicit assumption in non-iterative PMC is that for a particular component pairing the corrupted-speech distribution is approximately Gaussian. This is usually not the case [1]. Modeling of these complex distributions are done by Iterative PMC. Usually in this method, there are no restrictions on the frame/state alignment, and this can be allowed to vary.

The usual problem with iterative PMC is to calculate the PDF which is a function of all the elements in the feature vector. A feasible approach to this is the *Data-driven PMC*, termed as DPMC. The basic principle is shown in Fig. 3. In DPMC, the calculation of statistics of

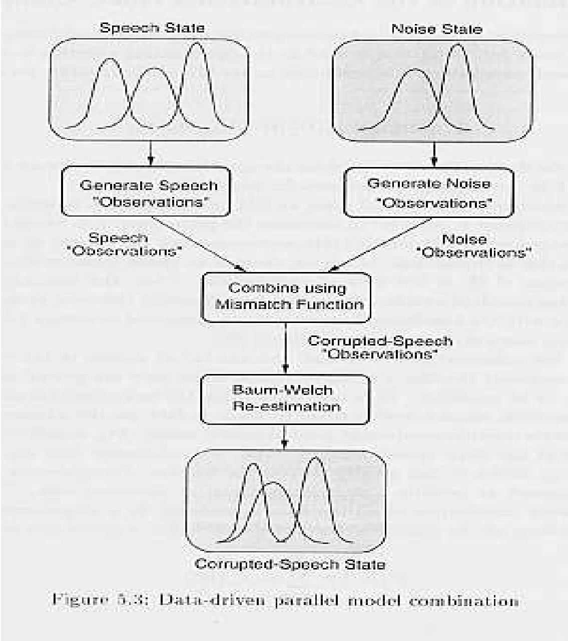

Figure 3: Data-Driven Parallel Model Combination

all n dimensions of the observation vector using the PDF is avoided by generating a speech "observation" and noise "observation" for a particular pair of speech and noise states and combining them using the appropriate mismatch function. Now, HMM training is done assuming a multiple mixture component single-emitting-state model. The initial state for the HMMs can be obtained from the non-iterative PMC described above. Thus, by generating "observations", integration of Gaussians is avoided, all that is now needed is to accumulate the appropriate statistics. The computational overhead is now dependent on the number of data points generated.

Log-Add approximation can be considered as a non-iterative, simplified and faster DPMC which updates the means. It generates a single observation for each of the speech and noise component, at the mean of their respective distributions. They are then combined according to the appropriate mismatch function to yield the corrupted-speech "observation", which is an estimate of the corrupted-speech mean.

4 Extensions

PMC usually involves three main stages as seen above: mapping HMM parameters into the Linear-Spectral Domain, constructing the corrupted-speech model and finally performing the reverse mapping. Faster techniques would mean precomputing and storing all the clean HMM statistics in both these domains. This is however, feasible only for tasks with low vocabulary and hence impracticable for large real time applications. More diagonal covariances in Cepstral domain results in full covariance matrix in the Linear-spectral domain, which increases the memory requirements for each storage. [4] introduces a scheme to speed up PMC without the increase in memory requirements. The clean speech parameters are known *apriori* and can be mapped to the Linear-Spectral domain for PMC compensation. So, a fast technique is needed to move to the Linear-Spectral domain without increase in memory requirements.

First clean HMM parameters are encoded in the cepstral domain using PCA. PCA is performed on each of the parameter sets separately, this is an off-line part and PCA provides the Eigen vectors and the prototype vectors and matrices during online model compensation. Moreover PCA can achieve good reconstruction of the vectors in the training set. Fig 4 shows the block diagram describing PCA based PMC. The authors

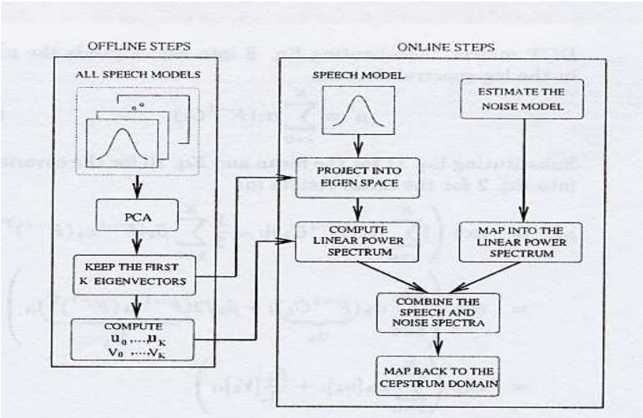

Figure 4: Data-Driven Parallel Model Combination

call this algorithm as *PCA-PMC*

[3] introduces *MAM*, which is a model-combination-based acoustic mapping scheme. The implicit idea is to simulate the distortion caused by additive noise using model combination as described in Section 3. Then in the second step acoustic mapping is performed that compensates the distortion in the signal (feature vectors).

5 Conclusions

This report aims to provide an introduction to a robust speech recognition scheme called Parallel Model Combination. PMC tries to model the statistics of the corrupted speech by generating clean speech and noise models and combining them using a mismatch function. This approach is different from other schemes like template matching or knowledge-based approaches. It addresses the fundamental issue of mismatch between training and test conditions. Two extensions (PCA-PMC and MAM) have also been briefly described.

From the results and the amount of papers/hits generated by a search engine (eg. google) one can see that PMC is gaining popularity among speech researchers.

References

[1] Mark John Francis Gales, "Model-Based Techniques for Noise Robust Speech Recognition", Dissertation, Gonville and Caius College, Univ. of Cambridge, Sept 1995.

[2] M.J.F. Gales, S.J. Young, "Robust Continuous Speech Recognition Using Parallel Model Combination", IEEE Transactions on Speech and Audio Processing Volume 4.

[3] Martin Westphal, Alex Waibel "Model-Combination-Based Acoustic Mapping", *ICASSP 2001*, Salt Lake City, May 2001

[4] Ruhi Sarikaya, John H.L. Hanses "PCA-PMC: A novel use of a priori knowledge for fast parallel model combination", *ICASSP 2000*, Istanbul, Turkey, June 2000