Joint Uncertainty Decoding for Noise Robust Speech Recognition

H. Liao and M. J. F. Gales

Cambridge University Engineering Department
Trumpington Street, Cambridge, CB2 1PZ, UK
{hl251, mjfg}@eng.cam.ac.uk

Abstract

Background noise can have a significant impact on the performance of speech recognition systems. A range of fast feature-space and model-based schemes have been investigated to increase robustness. Model-based approaches typically achieve lower error rates, but at an increased computational load compared to feature-based approaches. This makes their use in many situations impractical. The uncertainty decoding framework can be considered an elegant compromise between the two. Here, the uncertainty of features is propagated to the recogniser in a mathematically consistent fashion. The complexity of the model used to determine the uncertainty may be decoupled from the recognition model itself, allowing flexibility in the computational load. This paper describes a new approach within this framework, `Joint` uncertainty decoding. This approach is compared with the uncertainty decoding version of `SPLICE`, standard `SPLICE`, and a new form of front-end CMLLR. These are evaluated on a medium vocabulary speech recognition task with artificially added noise.

1. Introduction

It is well known that speech recognition performance degrades in the presence of environmental noise. When models trained in clean conditions are used in the real world, the mismatch between the training conditions and the test causes significant loss in recognition accuracy. Two approaches to improving noise robustness are feature-based and model-based compensation schemes. In feature-based schemes an estimate of the clean speech is made using a noise-model, or representation of the effects of the noise on the speech. `SPLICE` [1] is one recent example of this approach. Alternatively in model-based approaches, the parameters of the system are altered to reflect speech in the new acoustic environment. Examples in this class include Parallel Model Combination (PMC) [2] and Vector Taylor Series (VTS) compensation [3]. Model-based approaches often yield better performance than feature-based compensation schemes, especially in low SNR conditions, or in complex recognition tasks. However model-based schemes are usually more computationally expensive, especially if the acoustic environment is rapidly changing. Recently an elegant compromise between the two schemes, uncertainty decoding, has been proposed [4]. This approach allows the uncertainty of features to be propagated to the recogniser in a mathematically consistent fashion. The complexity of the model used to determine the uncertainty may be decoupled from the recognition model itself, allowing flexibility in the computational load associated with the scheme. This approach has been used to give a version of the `SPLICE` algorithm incorporating uncertainty [5].

In this paper an alternative implementation within the uncertainty decoding framework is presented, `Joint` uncertainty decoding. This new approach is compared to both the standard and uncertainty versions of `SPLICE`. In addition, the approach is contrasted with constrained MLLR [6] as the resultant compensation may be viewed as an extended version of a linear feature-space transformation. These schemes are evaluated on a medium vocabulary speech recognition task with artificially added noise.

2. Uncertainty Decoding Framework

The effects of environmental noise can be represented in a dynamic Bayesian network as shown in figure 1. Here, the noise

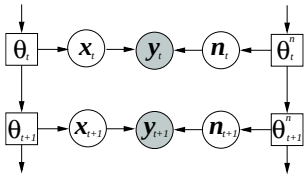

Figure 1: Uncertainty Decoding DBN

corrupted speech observation $\boldsymbol{y}_t$ at time t is assumed to be conditionally independent of all other observations given the clean speech $\boldsymbol{x}_t$ and the noise $\boldsymbol{n}_t$ at that time. The clean speech and noise are assumed to be generated by HMMs with states θ_t^n for the noise[1] and θ_t for the clean speech. Under these assumptions the likelihood of the corrupted observation may be expressed as

$$p(\boldsymbol{y}_t|\mathcal{M},\check{\mathcal{M}},\boldsymbol{\theta}_t) = \int p(\boldsymbol{y}_t|\boldsymbol{x}_t,\check{\mathcal{M}})p(\boldsymbol{x}_t|\mathcal{M},\theta_t)d\boldsymbol{x}_t \quad (1)$$

where

$$p(\boldsymbol{y}_t|\boldsymbol{x}_t,\check{\mathcal{M}}) = \int p(\boldsymbol{y}_t|\boldsymbol{x}_t,\boldsymbol{n}_t)p(\boldsymbol{n}_t|\check{\mathcal{M}},\theta_t^n)d\boldsymbol{n}_t \quad (2)$$

and $\check{\mathcal{M}}$ the front-end compensation model. The acoustic model $\mathcal{M}$ consists of Gaussian components each defined by a prior, c_m, mean, $\boldsymbol{\mu}^{(m)}$, and variance, $\boldsymbol{\Sigma}^{(m)}$. The likelihood calculation thus has two distinct parts. Only the first, $p(\boldsymbol{y}_t|\boldsymbol{x}_t,\check{\mathcal{M}})$, is a function of the noise. Equation 1 does not depend on the noise given the form of $p(\boldsymbol{y}_t|\boldsymbol{x}_t,\check{\mathcal{M}})$. Uncertainty decoding takes advantage of this factorisation by using an appropriate form of approximation for the conditional distribution of the corrupted speech given the clean speech for a particular noise environment. As the complexity of this approximation may be independent of the complexity of the actual acoustic models, there

Hank Liao is funded by Toshiba Research Europe Ltd. This work made use of equipment kindly supplied by IBM under a SUR award.

[1] A single state is assumed for the noise model in this paper.

is a large degree of flexibility in choosing the computational cost of the decoding process.

An example of using uncertainty decoding is the uncertainty version of SPLICE [5]. An N-component Gaussian Mixture Model (GMM) is used to approximate the conditional distribution. Equation 2 is re-written using Bayes' rule as

$$p(\boldsymbol{y}_t|\boldsymbol{x}_t,\check{\mathcal{M}}) \approx \frac{\sum_{n=1}^{N} p(\boldsymbol{x}_t|\boldsymbol{y}_t,\check{s}_n,\check{\mathcal{M}}) p(\boldsymbol{y}_t|\check{s}_n,\check{\mathcal{M}}) \check{c}_n}{p(\boldsymbol{x}_t|\check{\mathcal{M}})} \quad (3)$$

where the parameters associated with component $\check{s}_n$, are the prior, $\check{c}_n$, and $\check{\mu}_i^{(n)}$ and $\check{\sigma}_i^{(n)2}$, the mean and variance of dimension i of $(\boldsymbol{x}_t - \boldsymbol{y}_t)$ given the GMM component. Directly marginalising this form of conditional distribution is highly complex. Hence the GMM in the denominator is approximated by a single Gaussian component with the parameters $\bar{\boldsymbol{\mu}}_x$ and $\bar{\boldsymbol{\Sigma}}_x$. This yields the following form of the conditional corrupted speech posterior for a particular front-end component $\check{s}_n$

$$p(\boldsymbol{y}_t|\boldsymbol{x}_t,\check{\mathcal{M}},\check{s}_n) = f(\boldsymbol{y}_t,\check{s}_n)\mathcal{N}\left(\boldsymbol{A}^{(n)}\boldsymbol{y}_t + \boldsymbol{b}^{(n)};\boldsymbol{x}_t,\boldsymbol{\Sigma}_{\mathsf{b}}^{(n)}\right) \quad (4)$$

where $f(\boldsymbol{y}_t,\check{s}_n)$ is only a function of the corrupted observation and uncertainty model component [7] and the diagonal matrix $\boldsymbol{A}^{(n)}$, bias vector $\boldsymbol{b}^{(n)}$ and variance offset $\boldsymbol{\Sigma}_{\mathsf{b}}^{(n)}$ are given by

$$a_{ii}^{(n)} = \frac{\bar{\sigma}_{xi}^2}{\bar{\sigma}_{xi}^2 - \check{\sigma}_i^{(n)2}}, \quad \sigma_{\mathsf{b}i}^{(n)2} = a_{ii}^{(n)}\check{\sigma}_i^{(n)2} \quad (5)$$

$$b_i^{(n)} = a_{ii}^{(n)}\left(\check{\mu}_i^{(n)} - \frac{\check{\sigma}_i^{(n)2}}{\bar{\sigma}_{xi}^2}\bar{\mu}_{xi}\right) \quad (6)$$

for dimension i. Due to the approximation for the GMM in the denominator of equation 3, the denominator in the estimation of $a_{ii}^{(n)}$ can go negative. Flooring the denominator term avoids this. To improve the efficiency, rather than summing over all the components, only the most probable component $\check{s}_{n^*}$ is commonly used, selected by the component posterior

$$\check{s}_{n^*} = \arg\max_{\check{s}_n}\left(\frac{\check{c}_n p(\boldsymbol{y}_t|\check{s}_n,\check{\mathcal{M}})}{\sum_{i=1}^{N}\check{c}_i p(\boldsymbol{y}_t|\check{s}_i,\check{\mathcal{M}})}\right) \quad (7)$$

With this simplification, the overall number of Gaussian evaluations during decoding remains unchanged, and the term $f(\boldsymbol{y}_t,\check{s}_n)$ can be ignored since it now does not affect the recognition results. After marginalising over the components, the noise corrupted speech likelihood of equation 1 for state θ_t, is given by

$$p(\boldsymbol{y}_t|\mathcal{M},\check{\mathcal{M}},\theta_t) \propto \sum_{m\in\theta_t} c_m \mathcal{N}\left(\boldsymbol{A}^{(n^*)}\boldsymbol{y}_t + \boldsymbol{b}^{(n^*)};\boldsymbol{\mu}^{(m)},\boldsymbol{\Sigma}^{(m)} + \boldsymbol{\Sigma}_{\mathsf{b}}^{(n^*)}\right) \quad (8)$$

where the marginalisation of the two Gaussian distributions simplifies to a single Gaussian. One problem is that the cost of applying the variance bias is a function of the complexity of the acoustic model, $\mathcal{M}$, rather than the uncertainty model, $\check{\mathcal{M}}$. However for a diagonal variance bias, this cost is small.

3. Joint Uncertainty Decoding

The approach taken in this paper is to again approximate the conditional distribution in equation 2 with a GMM, but use the GMM directly. Now

$$p(\boldsymbol{y}_t|\boldsymbol{x}_t,\check{\mathcal{M}}) \approx \sum_{n=1}^{N} P(\check{s}_n|\boldsymbol{x}_t,\check{\mathcal{M}}) p(\boldsymbol{y}_t|\boldsymbol{x}_t,\check{\mathcal{M}},\check{s}_n) \quad (9)$$

With this form of the conditional, two main issues arise: the component posterior $P(\check{s}_n|\boldsymbol{x}_t,\check{\mathcal{M}})$ is a function of the clean speech, not the corrupted observation; and the form that the component compensation parameters $p(\boldsymbol{y}_t|\boldsymbol{x}_t,\check{s}_n,\check{\mathcal{M}})$ should take.

In this work a simple approximation is used for the component posterior given the "clean" speech. Here

$$P(\check{s}_n|\boldsymbol{x}_t,\check{\mathcal{M}}) \approx P(\check{s}_n|\boldsymbol{y}_t,\check{\mathcal{M}}) \quad (10)$$

where the model $\check{\mathcal{M}}$ is now matched to the test condition rather than the clean speech. This decouples the front-end distribution from being dependent on the acoustic model through the clean speech variable $\boldsymbol{x}_t$. However, the conditional distribution may change significantly over the clean speech integral. Thus using the same front-end distribution for every clean acoustic model Gaussian is not optimal.

The parameters of the conditional distribution given the front-end model component, $\check{s}_n$, are determined from the joint distribution of the clean and corrupted speech. This joint distribution is assumed to be Gaussian, hence for component $\check{s}_n$

$$\begin{bmatrix}\boldsymbol{x}_t\\ \boldsymbol{y}_t\end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix}\boldsymbol{\mu}_x^{(n)}\\ \boldsymbol{\mu}_y^{(n)}\end{bmatrix}, \begin{bmatrix}\boldsymbol{\Sigma}_x^{(n)} & \boldsymbol{\Sigma}_{xy}^{(n)}\\ \boldsymbol{\Sigma}_{yx}^{(n)} & \boldsymbol{\Sigma}_y^{(n)}\end{bmatrix}\right) \quad (11)$$

The conditional distribution will therefore also be Gaussian. When this form is used in the uncertainty decoding framework, the conditional likelihood of the corrupted speech observation has the same form as equation 4, but the parameters are now given by

$$\begin{aligned}\boldsymbol{A}^{(n)} &= \boldsymbol{\Sigma}_x^{(n)}\boldsymbol{\Sigma}_{yx}^{(n)\text{-}1}\\ \boldsymbol{b}^{(n)} &= \boldsymbol{\mu}_x^{(n)} - \boldsymbol{A}^{(n)}\boldsymbol{\mu}_y^{(n)}\\ \boldsymbol{\Sigma}_{\mathsf{b}}^{(n)} &= \boldsymbol{A}^{(n)}\boldsymbol{\Sigma}_y^{(n)}\boldsymbol{A}^{(n)\mathsf{T}} - \boldsymbol{\Sigma}_x^{(n)}\end{aligned} \quad (12)$$

and the normalisation term $f(\boldsymbol{y}_t,\check{s}_n)$ is simply $|\boldsymbol{A}^{(n)}|$. As the form of the conditional distribution is the same as that of SPLICE the final decoding likelihood, again using only the component with the largest posterior, has the same form as equation 8. However, in contrast to SPLICE where the form of the bias vector, given in equation 5, means that the variance bias term and the feature transform are diagonal, the transform and bias for Joint may be full. Though a full transformation of the features may be efficiently applied, the resultant covariance matrix for *every* component in the decoding system will be full. This will dramatically increase the computational load. The variance bias, $\boldsymbol{\Sigma}_{\mathsf{b}}^{(n)}$, may be restricted to be diagonal, or block-diagonal, by requiring that each block of the joint covariance matrix in equation 11 is diagonal, or block-diagonal.

It is interesting to compare this form of uncertainty decoding with SPLICE. In SPLICE the denominator in equation 3 is approximated by a single Gaussian component. This simplified the marginalisation, but requires the setting of a floor on the transform scaling to ensure that the variance was positive definite. In contrast the Joint distribution does not require this approximation, but assumes that the posteriors of the clean data can be approximated by the posteriors of the corrupted speech. It is also possible to relate the two forms of compensation parameters. For example the variance for the SPLICE scheme may be expressed in terms of the joint distribution parameters in equation 11 as

$$\check{\boldsymbol{\Sigma}}^{(n)} = \boldsymbol{\Sigma}_y^{(n)} + \boldsymbol{\Sigma}_x^{(n)} - \boldsymbol{\Sigma}_{xy}^{(n)} - \boldsymbol{\Sigma}_{yx}^{(n)} \quad (13)$$

4. Model-based Uncertainty Decoding

It is interesting to note that the final likelihood expression for both the SPLICE and Joint uncertainty decoding, equation 8, is similar to constrained MLLR [6]. The standard form of the CMLLR likelihood calculation is given by

$$p(\boldsymbol{y}_t|\boldsymbol{\theta}_t, \mathcal{M}, \check{\mathcal{M}}) = \sum_{m \in \theta_t} c_m |\boldsymbol{A}^{(r_m)}| \mathcal{N}\left(\boldsymbol{A}^{(r_m)}\boldsymbol{y}_t + \boldsymbol{b}^{(r_m)}; \boldsymbol{\mu}^{(m)}, \boldsymbol{\Sigma}^{(m)}\right) \quad (14)$$

where r_m indicates the transform-class that acoustic model component m is assigned to and $\check{\mathcal{M}}$ now denotes the model compensation parameters. There are some interesting differences between the uncertainty decoding, equation 8, and equation 14. First, the transform is estimated using the differences between the clean speech and noise corrupted speech, rather than maximum likelihood training. Second, there is a bias applied on the variances. This increases the compensation cost, but can yield improved recognition performance in noise, as discussed in section 6. The final difference is that the transform is determined by the component with the greatest posterior in the front-end. CMLLR is normally implemented by associating transforms with components of the system. The first two differences are fundamental to the different forms of compensation. The final difference motivates a modification to both the CMLLR scheme and the Joint uncertainty decoding scheme.

Instead of estimating joint distributions and transforms per region of the acoustic space partitioned by a front-end GMM, they could be trained for each transform class in a similar fashion to CMLLR. For example, rather than estimate $\boldsymbol{\Sigma}_{xy}$ for a component $\check{s}_n$, it is found for each transform class r_m

$$\boldsymbol{\Sigma}_{xy}^{(r_m)} = \frac{\sum_{m \in r_m} \gamma_m(t) \boldsymbol{x}_t \boldsymbol{y}_t^{\mathsf{T}}}{\sum_{m \in r_m} \gamma_m(t)} - \boldsymbol{\mu}_x^{(r_m)} \boldsymbol{\mu}_y^{(r_m)\mathsf{T}} \quad (15)$$

where $\gamma_m(t)$ is the component posterior at time instance t. The joint mean, $[\boldsymbol{\mu}_x^{(r_m)\mathsf{T}} \ \boldsymbol{\mu}_y^{(r_m)\mathsf{T}}]^{\mathsf{T}}$, and other covariance terms can be similarly obtained. It is now possible to estimate a Joint uncertainty decoding transform for each transform class. This has the advantage that the posterior approximation in equation 10 is unnecessary. Also, for standard uncertainty decoding, as the front-end component changes, a new variance bias must be applied to each acoustic model component. This is not necessary in this transform class approach. However the disadvantage, in the same fashion as CMLLR, is that at each time instance multiple transformed feature-spaces are required, each with a different normalisation term $|\boldsymbol{A}^{(r_m)}|$. This approach will be referred to as model-based Joint uncertainty decoding.

5. Front-end CMLLR

As it is useful to compare the uncertainty decoding schemes to approaches such as CMLLR, CMLLR can be modified to use a GMM front-end selection process. This is simply achieved by associating a single CMLLR transform with each front-end component $\check{s}_n$. These transforms can be estimated using a slightly modified version of the training algorithm described in [6]. For example, to accumulate the sufficient statistic $\mathbf{G}^{(in)}$ the accumulation is modified to

$$\mathbf{G}^{(in)} = \sum_{t,m} \frac{P(\check{s}_n|\boldsymbol{y}_t, \check{\mathcal{M}})}{\sigma_i^{(m)2}} \gamma_m(t) \boldsymbol{\zeta}_t \boldsymbol{\zeta}_t^{\mathsf{T}} \quad (16)$$

where $\boldsymbol{\zeta}_t$ is the extended observation vector $[1 \ \boldsymbol{y}_t^{\mathsf{T}}]^{\mathsf{T}}$. A similar expression can be obtained for $\mathbf{k}^{(in)}$. Compared to standard CMLLR this has the advantage that only a single transform is active at each time instance. This form of transform will be referred to as Front-end CMLLR (FE-CMLLR).

6. Results

This section describes preliminary results comparing the various schemes described in this paper. For this work, noise was artificially added to a medium vocabulary speech recognition task, the 1000 word Resource Management (RM) database. Operations Room noise from the NOISEX-92 database was added at the waveform level. Though this task is artificial and is expected to yield better performance than would be obtained on realistic data, it allows a comparison of the various techniques in a highly controlled fashion. RM was used as a speaker independent task which consists of 109 training speakers reading 3990 sentences of prompted script totalling 3.8 hours. All results are quoted as an average of three of the four available test sets, Feb'89, Oct'89 and Feb'91. This gives a total of 30 test speakers and 900 utterances. State-clustered triphone models were built using the standard RM recipe in the alpha version of HTK 3.3. The standard front-end, MFCC plus normalised energy with delta and delta-delta parameters, were used for all experiments. A range of noise SNRs from 32 dB to 8dB were examined, however the results are only quoted at 20dB SNR. For further details of other SNRs see [7].

The RM database was selected for evaluation, rather than, for example, the small vocabulary AURORA digit string recognition task, because uncertainty decoding is expected to be more important on more complex tasks. To verify the performance of the SPLICE implementations, both standard and with uncertainty, the code was run on AURORA giving similar performance to that in [5], where only relatively small gains from uncertainty decoding were obtained.

The GMMs for the front-end uncertainty models were trained using iterative mixture splitting on the clean speech data. The compensation parameters, either those associated with uncertainty decoding or the CMLLR transforms, were estimated using stereo data for the specific noise condition. This allows the techniques to be assessed without having to consider inaccuracies that result from the noise estimation process, or approximations in the mismatch function. In practical situations the compensation parameters can be estimated using PMC or VTS style schemes. This is discussed in more detail in [7].

6.1. Feature-based Compensation

Initially feature-based compensation was evaluated. All these schemes use a GMM in the front-end to determine the appropriate component for the compensation scheme. Only diagonal versions of the FE-CMLLR and Joint schemes were assessed.

System	With Uncertainty	# Front-end Components			
		1	4	16	256
Clean	—	33.2			
SPLICE	No	24.6	20.7	17.0	12.3
FE-CMLLR	No	16.3	15.3	12.8	13.5
SPLICE	Yes	11.4	12.4	12.2	9.9
Joint	Yes	10.7	9.2	9.8	8.2
Matched	—	7.2			

Table 1: Feature-based compensation WER (%) at 20dB SNR

Table 1 shows the performance of the various schemes against the number of components in the front-end GMM. As expected the matched scheme, generated using single-pass retraining [2], significantly out-performed the standard clean system. This matched system, the "perfect" model-based approach[2], is the baseline number for experiments. For reference, the error rate on clean uncorrupted data was 3.3%, demonstrating the considerable confusability that results from the addition of noise where the error rate was more than doubled to 7.2%. The two schemes with no uncertainty decoding, standard `SPLICE` and FE-CMLLR, both gave reasonable gains over the baseline, clean, system. However further large reductions in WER are achieved by using `SPLICE` with uncertainty or the `Joint` approach. Using a single component front-end with either scheme was better than the best non-uncertainty decoding approach. This is interesting since it illustrates the importance of incorporating the variance bias term to allow some frames to be effectively de-weighted. Comparing the `SPLICE` and `Joint` uncertainty schemes, `Joint` appears to be better with fewer components, but with 256 components the performance of the two is approximately the same. This may be explained by the very simple posterior approximation used in the Joint scheme. The overall performance of the best scheme was still about 2.0% absolute worse than the matched approach.

6.2. Model-based `Joint` Compensation

The `Joint` and CMLLR forms can also be applied in a model-based manner. For these experiments the complexity of the transforms was also varied to determine what effect the more complex `Joint` transforms will yield.

System	Transform Structure	# Transform Classes		
		1	4	16
Clean	—	33.2		
CMLLR	Diagonal	16.3	14.6	10.3
	Full	17.8	14.9	9.2
Model-Based `Joint`	Diagonal	10.7	9.8	8.4
	Full	10.4	8.0	7.4
Matched	—	7.2		

Table 2: Model-based compensation WER (%) at 20dB SNR

As expected, when using more complex, or additional transforms the performance of the system generally improves. In table 2 using 16 full CMLLR transforms yields an error rate of 9.2%, the performance of the best front-end uncertainty scheme. Note the performance of the standard model-based CMLLR was generally better than the FE-CMLLR, though at the additional computational expense of multiple input transforms. Comparing the diagonal model-based `Joint` approach with the front-end `Joint` scheme results in table 1, shows that the model-based approach is better as the number of components/transforms increases. This is not really surprising given the posterior approximation used. Interestingly, using a full model-based `Joint` approach consistently yielded the best performance over all the schemes. Unfortunately, this scheme is computationally very expensive for decoding as the variance bias is a full matrix, bearing an overall computational cost of a full covariance matrix system. The performance of the 16 transform full `Joint` system gave an error rate, 7.4%, that is approximately the same as the matched system.

[2]This matched scheme can be improved upon, for example see [2].

One approach to reducing the computational load of the full scheme would be to diagonalise the variance bias term. This still gives a full transform, $\boldsymbol{A}$, but an approximate diagonal variance bias, $\boldsymbol{\Sigma}_{\mathrm{b}}$. Unfortunately, using this simple approach produced poor performance with error rates of about 30%.

7. Conclusions

This paper has discussed the application of uncertainty decoding to noise robust speech recognition. The framework allows the uncertainty to be propagated from the front-end process into the recognition search. Two forms of uncertainty decoding were compared, the `SPLICE` formulation and a new `Joint` one. Both schemes are based on the use of a GMM in the front-end, though making very different approximations to allow for efficient operation. In addition, a model-based version of the `Joint` algorithm was briefly discussed along with a modified version of CMLLR, FE-CMLLR. The performance of the various schemes was evaluated on an artificially noise corrupted version of RM. As expected, the maximum likelihood trained FE-CMLLR transforms performed better than MMSE `SPLICE` at fewer numbers of front-end components. Uncertainty decoding was found to be far more accurate than the front-end compensation schemes `SPLICE` and FE-CMLLR. However, even with a 256-component GMM in the front-end the best system was still 2% worse than the matched system. The performance of the model-based compensation schemes, where transforms were associated with sets of recogniser components rather than front-end components, were generally better than the equivalent front-end scheme. Furthermore, using a full model-based `Joint` transform gave an error rate approximately the same as the matched scheme.

The experiments presented in this paper were artificial in two ways: corrupted speech was simulated by adding noise to clean speech and the compensation parameters were estimated on stereo data. Future work will examine real found data, such as broadcast news, and the application of schemes such as VTS to determine the compensation parameters.

8. References

[1] L. Deng, A. Acero, M. Plumpe, and X. D. Huang, "Large vocabulary speech recognition under adverse acoustic environments," in *Proc. ICSLP*, Beijing, China, Oct. 2000.

[2] M. J. F. Gales, "Model-based techniques for noise robust speech recognition," Ph.D. dissertation, Cambridge University, 1995.

[3] P. Moreno, "Speech recognition in noisy environments," Ph.D. dissertation, Carnegie Mellon University, 1996.

[4] T. Kristjansson and B. J. Frey, "Accounting for uncertainty in observations: A new paradigm for robust speech recognition," in *Proc. ICASSP*, Orlando, Florida, May 2002.

[5] J. Droppo, A. Acero, and L. Deng, "Uncertainty decoding with splice for noise robust speech recognition," in *Proc. ICASSP*, Orlando, Florida, May 2002.

[6] M. J. F. Gales, "Maximum Likelihood Linear Transformations For HMM-Based Speech Recognition," *Computer Speech and Language*, vol. 12, Jan. 1998.

[7] H. Liao and M. J. F. Gales, "Uncertainty decoding for noise robust speech recognition," University of Cambridge, Tech. Rep. CUED/F-INFENG/TR499, 2004, available from: mi.eng.cam.ac.uk/~hl251.