CAMBRIDGE UNIVERSITY

ENGINEERING DEPARTMENT

NOISY CMLLR FOR NOISE-ROBUST SPEECH RECOGNITION

D.K. Kim and M.J.F. Gales

CUED/F-INFENG/TR.611

February 2009

Cambridge University Engineering Department
Trumpington Street
Cambridge. CB2 1PZ
England

E-mail: {dk369, mjfg}@eng.cam.ac.uk
http://www-svr.eng.cam.ac.uk/~mjfg

Abstract

Adaptive training is a widely used technique for building speech recognition systems on non-homogeneous training data. Recently there has been interest in applying these approaches for situations where there is significant levels of background noise. Various schemes for adaptive training are based on noise, or speaker, specific transforms of the observed noise-corrupted speech to yield estimates of the clean speech. However when there are high levels of background noise, these clean speech estimates may be poor resulting in degradations in performance. In this work, a new approach for adaptive training on noise-corrupted training data is presented. It extends a popular form of linear transform for model-based adaptation and adaptive training, constrained MLLR (CMLLR), to reflect additional uncertainty from noise-corrupted observations. This new form of transform is called noisy CMLLR (NCMLLR). NCMLLR uses a modified version of generative model between clean speech and noisy observation, similar to factor analysis (FA). However in contrast in FA here the generative model describes a transformation, rather than a covariance matrix structure. The use of NCMLLR for adaptation and adaptive training using an expectation-maximisation approach is described. Discriminative adaptive training with NCMLLR is also presented based on the minimum phone error criterion. Experiments are conducted on noise-corrupted version of Resource Management and in-car recorded digit data. In preliminary experiments this new approach achieves improvements in recognition performance over the standard approach in low signal-to-noise ratio conditions. In addition the need for adaptive training when there are a range of noise conditions in the training data is shown.

1 Introduction

Current large vocabulary speech recognition systems are normally constructed on large amounts of acoustic training data from multiple speakers with different background noise and channel conditions. The standard approach to building acoustic models is to treat all the data as a single block, multi-style training. This yields acoustic models which represent both the speech, speaker and background noise. However performance of these multi-style systems can be limited, especially if there is a wide range of noise conditions seen in the training data. An alternative approach is to use adaptive training [1, 2]. Though normally described in terms of speaker adaptive training, the same approaches may be used to handle large variabilities in the background noise, which will be the focus of this work. Noise-specific transforms are estimated during training. These transforms should compensate for the noise, allowing a "clean" canonical acoustic model to be trained on noise corrupted data. These canonical models can then be adapted to a particular test condition. This yields a purer canonical model of speech compared to multi-style training where the models incorporate all the variability of the acoustic data. Adaptive training is usually based on linear transforms, in particular constrained maximum likelihood linear regression (CMLLR) [2], as minimal changes to the standard code are required for estimating the canonical model. However, these forms of adaptive training do not deal specifically with data having high levels of background noise. CMLLR may be viewed as making an estimate of the clean speech given the noise-corrupted data and the transform, effectively it is a feature-enhancement approach. For low signal-to-noise ratio (SNR) conditions the estimate of the clean speech may be poor and should not be treated with the same level of confidence as high SNR data.

A number of noise-specific schemes that make use of feature-enhancement approaches have also been used to handle varying noise conditions in the training data. These approaches are sometimes referred to as noise adaptive training (NAT) [3]. In the same fashion as CMLLR, these techniques do not alter the level of confidence in the estimate of the clean speech to reflect the noise conditions, possibly limiting performance when trying to deal with low-SNR data. Approaches in this class include robust environment-effects suppression training (REST) [4] and NAT [3]. Discriminative versions of these adaptive training schemes have also been proposed [5, 6].

To overcome the limitation of these feature-enhancement-style approaches, model-based adaptive training schemes have been proposed for data with high levels of background noise [7, 8] using joint uncertainty decoding (JUD) [9] or vector Taylor series (VTS) [10]. These model-based approaches modify the parameters of the acoustic models, so that they are representative of the target test environment. As the model parameters are transformed, there is additional flexibility which allows the uncertainty of the clean speech estimates to be modelled. A problem with these model-based schemes is that a mismatch function representing the impact of the background noise on the clean speech must be specified for the feature-parameterisation being used. This is not simple for all feature parameterisations.

In this paper, a new approach for adaptive training on training data with a wide-range of noise conditions is presented. The scheme is described as an extension to estimating CMLLR transforms and adaptive training with CMLLR, though its structure is also related to factor-analysed-based covariance matrix schemes [11, 12]. An approximate generative model relating the "clean" speech and the observation is proposed. This model allows a noise term to be included, which is important to allow a level of confidence in the clean estimate to be described. This form of generative process may be expressed as a linear transform of the observed features and a bias on the variance (thus it may be viewed as combining CMLLR with the variance bias described in [13]) and will be referred to as noisy CMLLR (NCMLLR). The form of this transform is identical to JUD [9]. However in the NCMLLR transform the parameters are estimated in a maximum likelihood (ML) fashion rather than being based on a mismatch function and noise model parameter estimates. By directly estimating the parameters from the observation it is not necessary to specify a mismatch function allowing more complex forms of front-end processing and normalisation to be used. It also allows additional flexibility in the nature of the feature linear transform and variance bias. In addition, EM-based canonical model estimation formulae are derived using NCMLLR. Given the relationship between JUD and VTS these are closely related to those given in [8]. Finally

adaptive training with NCMLLR is extended to support discriminative training of the acoustic models based on the minimum phone error (MPE) [14] criterion.

This report is organised as follows: The next section describes adaptation and adaptive training approaches based on linear transforms. Model-based compensation approaches for robust speech recognition are reviewed in section 3. In section 4, NCMLLR is formally introduced and adaptive training with NCMLLR presented. The relationship to existing approaches is also described. Section 5 discusses practical implementation issues. Results are reported in section 6 on noise corrupted Resource Management and Toshiba in-car collected data. Lastly, conclusions are given in section 7.

2 Adaptive Model-Based Linear Transforms

Linear transform based adaptation is a widely used approach to speaker and environment adaptation when there is limited adaptation data available. The basic idea is to estimate a test-domain specific linear transform for the means and/or covariance matrices of the Gaussian components. In standard schemes, the transforms are estimated using the ML criterion given a set of hidden Markov models (HMMs). This section will review the adaptation based on the linear transforms, including maximum likelihood linear regression (MLLR) [15], covariance MLLR [16, 2], CMLLR [2] and variance bias [13]. These schemes will be given the general name of adaptive transforms, as all the transform parameters are estimated directly.

For all the transformation estimation schemes in this section and the rest of the report, the clean speech models are assumed to HMMs. The state output distributions are modelled using continuous density Gaussian mixture models with diagonal covariance matrices.

The notation used in this section differs from the standard presentation as it is linked with the next section on noise robustness. Thus the subscript $\mathtt{s}$ will be used to indicate the "clean"-speech model and $\mathtt{o}$ the corrupted speech model.

2.1 Maximum Likelihood Linear Regression

MLLR [15, 16] adaptation uses the ML criterion to estimate a linear transform to adapt the Gaussian component parameters of the HMMs. In MLLR, the mean of Gaussian component m is adapted to a particular acoustic condition by

$$\boldsymbol{\mu}_{\mathtt{o}}^{(m)} = \mathbf{A}^{(r_m)}\boldsymbol{\mu}_{\mathtt{s}}^{(m)} + \mathbf{b}^{(r_m)} = \mathbf{W}^{(r_m)}\boldsymbol{\xi}_{\mathtt{s}}^{(m)} \tag{1}$$

where $\boldsymbol{\mu}_{\mathtt{o}}^{(m)}$ is the adapted mean of component m for the target acoustic condition, $\boldsymbol{\xi}_{\mathtt{s}}^{(m)} = [1 \quad \boldsymbol{\mu}_{\mathtt{s}}^{(m)\mathsf{T}}]^{\mathsf{T}}$ is the extended mean vector, and $\mathbf{W}^{(r_m)} = [\mathbf{b}^{(r_m)} \quad \mathbf{A}^{(r_m)}]$ is the extended linear transform. The superscript r_m indicates that the transform applied to acoustic model component m is based on the regression class that component m belongs to. The total number of classes R is usually small, especially compared to the number of model components M. Components are often clustered together to form a regression class tree [17]. Using a regression tree gives an elegant means to scale the number of transforms to the available adaptation data [17, 18]. The linear transform, $\mathbf{A}^{(r_m)}$ can be diagonal, block, or a full in structure depending on the amount of supervision data available. It is worth emphasising that the transform $\mathbf{W}^{(r_m)}$ is associated with a particular test acoustic condition.

The likelihood of a particular state at time t, θ_t, given the MLLR transform can then be expressed as

$$p(\mathbf{o}_t|\mathcal{M}, \mathcal{T}, \theta_t) = \sum_{m \in \theta_t} c^{(m)} \mathcal{N}\left(\mathbf{o}_t; \mathbf{A}^{(r_m)}\boldsymbol{\mu}_{\mathtt{s}}^{(m)} + \mathbf{b}^{(r_m)}, \boldsymbol{\Sigma}_{\mathtt{s}}^{(m)}\right) \tag{2}$$

Using the ML criterion, the optimal transform can be found by maximising the likelihood of generating the adaptation data for a particular speaker. The EM algorithm [19] is applied to solve the optimisation problem. The estimation of the transform is based on a set of well-trained HMMs

$\mathcal{M}$ and the current set of transforms $\mathcal{T}$. Ignoring the constants independent of the transform, the auxiliary function for MLLR transform estimation can be written as [15]

$$\mathcal{Q}(\hat{\mathcal{T}};\mathcal{T}) = -\frac{1}{2}\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_{\mathsf{o},t}^{(m)}\left(\mathbf{o}_t - \hat{\mathbf{W}}^{(r_m)}\boldsymbol{\xi}_{\mathsf{s}}^{(m)}\right)^{\mathsf{T}}\boldsymbol{\Sigma}_{\mathsf{s}}^{(m)\text{-}1}\left(\mathbf{o}_t - \hat{\mathbf{W}}^{(r_m)}\boldsymbol{\xi}_{\mathsf{s}}^{(m)}\right) \tag{3}$$

where $\hat{\mathcal{T}}$ is the set of R MLLR transforms to be estimated $\{\hat{\mathbf{W}}^{(1)},\ldots,\hat{\mathbf{W}}^{(R)}\}$, $\gamma_{\mathsf{o},t}^{(m)}$ is the posterior occupancy of component m calculated using the forward-backward algorithm with HMMs adapted by the current transform estimate. Usually the Gaussian covariance matrices are assumed to be diagonal. This will greatly simplify the estimation of MLLR transforms. Differentiating the auxiliary function in equation 3, with respect to $\hat{\mathbf{W}}^{(r_m)}$ and equating to zero yields the ML estimate. Let $\hat{\mathbf{W}}^{(r)} = [\hat{\mathbf{w}}_1^{(r)},\cdots,\hat{\mathbf{w}}_D^{(r)}]^{\mathsf{T}}$, where $\hat{\mathbf{w}}_d^{(r)\mathsf{T}}$ is the d^{th} row of $\hat{\mathbf{W}}^{(r)}$, D is the dimensionality of the feature vector, the ML estimate of the d^{th} row is given by

$$\hat{\mathbf{w}}_d^{(r)} = \mathbf{G}_d^{(r)-1}\mathbf{k}_d^{(r)} \tag{4}$$

where the sufficient statistics for the d^{th} row are given by

$$\mathbf{G}_d^{(r)} = \sum_{m\in r}\sum_{t=1}^{T}\frac{\gamma_{\mathsf{o},t}^{(m)}}{\sigma_{\mathsf{s}d}^{(m)2}}\boldsymbol{\xi}_{\mathsf{s}}^{(m)}\boldsymbol{\xi}_{\mathsf{s}}^{(m)\mathsf{T}} \tag{5}$$

$$\mathbf{k}_d^{(r)} = \sum_{m\in r}\sum_{t=1}^{T}\frac{\gamma_{\mathsf{o},t}^{(m)}o_{td}}{\sigma_{\mathsf{s}d}^{(m)2}}\boldsymbol{\xi}_{\mathsf{s}}^{(m)} \tag{6}$$

where $m \in r$ indicates components m in regression class r, o_{td} is the d^{th} element of observation vector $\mathbf{o}_t$, $\sigma_{\mathsf{s}d}^{(m)2}$ is the d^{th} diagonal element of $\boldsymbol{\Sigma}_{\mathsf{s}}^{(m)}$. MLLR transform estimation is an iterative process. A new transform is estimated by making use of the current transform to obtain a new posterior occupance, $\gamma_{\mathsf{o},t}^{(m)}$. This process can be repeated until convergence.

The covariance matrix of each component can also be adapted using linear transforms. This is referred to as covariance MLLR. The covariance matrix may be adapted by [16, 2]

$$\boldsymbol{\Sigma}_{\mathsf{o}}^{(m)} = \mathbf{H}^{(r_m)}\boldsymbol{\Sigma}_{\mathsf{s}}^{(m)\text{-}1}\mathbf{H}^{(r_m)\mathsf{T}} \tag{7}$$

where $\mathbf{H}^{(r_m)}$ is the linear transform to adapt covariance matrices. This form has the advantage of other alternatives in that the likelihood can be efficiently calculated by transforming the mean and observations, which is much more efficient than the calculation using a full covariance matrix. The likelihood of an observation $\mathbf{o}_t$ can be expressed by

$$p(\mathbf{o}_t|\mathcal{M},\mathcal{T},\theta_t) = \sum_{m\in\theta_t}c^{(m)}|\mathbf{H}^{(r_m)}|\mathcal{N}\left(\mathbf{H}^{(r_m)\text{-}1}\mathbf{o}_t;\mathbf{H}^{(r_m)\text{-}1}\boldsymbol{\mu}_{\mathsf{s}}^{(m)},\boldsymbol{\Sigma}_{\mathsf{s}}^{(m)}\right) \tag{8}$$

A solution for estimating $\mathbf{H}^{(r_m)}$ using EM algorithm has also been derived in [2].

2.2 Constrained MLLR

A tied mean and covariance transform can be applied to both mean vectors and covariance matrices, this is referred to as a constrained linear transform [20, 2]. In this case

$$\boldsymbol{\mu}_{\mathsf{o}}^{(m)} = \mathbf{H}^{(r_m)}\boldsymbol{\mu}_{\mathsf{s}}^{(m)} + \mathbf{g}^{(r_m)} \tag{9}$$

$$\boldsymbol{\Sigma}_{\mathsf{o}}^{(m)} = \mathbf{H}^{(r_m)}\boldsymbol{\Sigma}_{\mathsf{s}}^{(m)}\mathbf{H}^{(r_m)\mathsf{T}} \tag{10}$$

where $\mathbf{H}^{(r_m)}$ is the constrained linear transform, $\mathbf{g}^{(r_m)}$ is the bias on the mean vector, and $\boldsymbol{\mu}_{\mathsf{s}}^{(m)}$ and $\boldsymbol{\Sigma}_{\mathsf{s}}^{(m)}$ are the original Gaussian parameters. This is referred to as CMLLR. In contrast to the

MLLR transform, CMLLR can be efficiently applied in the feature space. This yields a regression class-specific estimate of the clean speech. Thus

$$\mathbf{o}_t = \mathbf{H}^{(r_m)}\mathbf{s}_t + \mathbf{g}^{(r_m)} \qquad \mathbf{s}_t = \mathbf{A}^{(r_m)}\mathbf{o}_t + \mathbf{b}^{(r_m)} = \mathbf{W}^{(r_m)}\boldsymbol{\zeta}_t \tag{11}$$

where $\mathbf{A}^{(r_m)} = \mathbf{H}^{(r_m)\text{-}1}$ and $\mathbf{b}^{(r_m)} = -\mathbf{A}^{(r_m)\text{-}1}\mathbf{g}^{(r_m)}$. $\mathbf{W}^{(r_m)} = [\mathbf{b}^{(r_m)} \;\; \mathbf{A}^{(r_m)}]$ is the extended transform matrix and $\boldsymbol{\zeta}_t = [1 \;\; \mathbf{o}_t^{\mathsf{T}}]^{\mathsf{T}}$ is the extended observation vector. Then the likelihood of the observation $\mathbf{o}_t$ can be calculated by

$$p(\mathbf{o}_t|\mathcal{M},\mathcal{T},\theta_t) = \sum_{m\in\theta_t} c^{(m)}|\mathbf{A}^{(r_m)}|\mathcal{N}\left(\mathbf{A}^{(r_m)}\mathbf{o}_t + \mathbf{b}^{(r_m)}; \boldsymbol{\mu}_{\mathbf{s}}^{(m)}, \boldsymbol{\Sigma}_{\mathbf{s}}^{(m)}\right) \tag{12}$$

This form of transformation is highly efficient when the background noise conditions are rapidly changing. Rather than having to transform the model parameters, where in large vocabulary there may be hundreds of thousands of Gaussian components, only the features have be transformed at each time instance.

The CMLLR transforms are estimated by optimising the auxiliary function [2]

$$\begin{aligned}\mathcal{Q}(\hat{\mathcal{T}};\mathcal{T}) = -\frac{1}{2}\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_{\mathbf{o},t}^{(m)}\Big[&\log(|\boldsymbol{\Sigma}_{\mathbf{s}}^{(m)}|) \\ &- \log(|\hat{\mathbf{A}}^{(r_m)}|^2) + (\hat{\mathbf{W}}^{(r_m)}\boldsymbol{\zeta}_t - \boldsymbol{\mu}_{\mathbf{s}}^{(m)})^{\mathsf{T}}\boldsymbol{\Sigma}_{\mathbf{s}}^{(m)\text{-}1}(\hat{\mathbf{W}}^{(r_m)}\boldsymbol{\zeta}_t - \boldsymbol{\mu}_{\mathbf{s}}^{(m)})\Big]\end{aligned} \tag{13}$$

The ML estimate of the dth row of $\hat{\mathbf{W}}^{(r)}$, given all other rows, is expressed by

$$\hat{\mathbf{w}}_d^{(r)} = \mathbf{G}_d^{(r)\text{-}1}\left(\alpha\mathbf{p}_d^{(r)} + \mathbf{k}_d^{(r)}\right) \tag{14}$$

where $\mathbf{p}_d$ is the extended cofactor vector $[0 \; c_{d1} \; \cdots \; c_{dD}]^{\mathsf{T}}$, $c_{ij} = \text{cof}(\mathbf{A}_{ij})$ is the cofactor. A solution for α was given in [2]. The sufficient statistics for regression class r and row d are given by

$$\mathbf{G}_d^{(r)} = \sum_{m\in r}\frac{1}{\sigma_{\mathbf{s}d}^{(m)2}}\sum_{t=1}^{T}\gamma_{\mathbf{o},t}^{(m)}\boldsymbol{\zeta}_t\boldsymbol{\zeta}_t^{\mathsf{T}} \tag{15}$$

$$\mathbf{k}_d^{(r)} = \sum_{m\in r}\frac{\mu_{\mathbf{s}d}^{(m)}}{\sigma_{\mathbf{s}d}^{(m)2}}\sum_{t=1}^{T}\gamma_{\mathbf{o},t}^{(m)}\boldsymbol{\zeta}_t \tag{16}$$

where $\sigma_{\mathbf{s}d}^{(m)2}$ is the dth diagonal element of covariance matrix $\boldsymbol{\Sigma}_{\mathbf{s}}^{(m)}$ and $\mu_{\mathbf{s}d}^{(m)}$ is the dth element of $\boldsymbol{\mu}_{\mathbf{s}}^{(m)}$. Equation 14 is used to estimate each row of $\hat{\mathbf{W}}^{(r)}$ while keeping the others fixed. The estimate is guaranteed to improve the likelihood and provides a stable, row-by-row, iterative process [2].

2.3 Variance Bias

In the previous sections, the adaptation to a particular condition is assumed to be a fixed linear transform of the original speech model. Alternatively the observation vector $\mathbf{o}_t$ may be assumed to be the output from a generative model of the original speech vector $\mathbf{s}_t$ and an acoustic condition term, denoted by $\mathbf{b}_t$, in the form

$$\mathbf{o}_t = f(\mathbf{s}_t, \mathbf{b}_t) \tag{17}$$

where f is a mismatch function that describes the impact of bias on the clean speech. The additive bias model [21, 13] is a special case of the mismatch function described in the next section such that $\mathbf{o}_t$ is given by

$$\mathbf{o}_t = \mathbf{s}_t + \mathbf{b}_t \tag{18}$$

The bias transform can also be made regression class specific. Thus for component m of regression class r_m

$$\mathbf{b}_t|m, \mathcal{M}_\mathtt{b} \sim \mathcal{N}(\boldsymbol{\mu}_\mathtt{b}^{(r_m)}, \boldsymbol{\Sigma}_\mathtt{b}^{(r_m)}) \tag{19}$$

where $\mathcal{M}_\mathtt{b} = \left\{\{\boldsymbol{\mu}_\mathtt{b}^{(1)}, \boldsymbol{\Sigma}_\mathtt{b}^{(1)}\}, \ldots, \{\boldsymbol{\mu}_\mathtt{b}^{(R)}, \boldsymbol{\Sigma}_\mathtt{b}^{(R)}\}\right\}$ represents a set of R regression-class specific parameters for the bias $\mathbf{b}_t$. $\mathbf{s}_t$ is modelled by the set of HMMs $\mathcal{M}$. If $\mathbf{s}_t$ and $\mathbf{b}_t$ are independent, the means and covariance of the observed speech for each component m are derived by adding the mean $\boldsymbol{\mu}_\mathtt{b}^{(r_m)}$ and the variance $\boldsymbol{\Sigma}_\mathtt{b}^{(r_m)}$ to the means and covariances of the original speech model

$$\boldsymbol{\mu}_\mathtt{o}^{(m)} = \boldsymbol{\mu}_\mathtt{s}^{(m)} + \boldsymbol{\mu}_\mathtt{b}^{(r_m)} \tag{20}$$

$$\boldsymbol{\Sigma}_\mathtt{o}^{(m)} = \boldsymbol{\Sigma}_\mathtt{s}^{(m)} + \boldsymbol{\Sigma}_\mathtt{b}^{(r_m)} \tag{21}$$

Then the likelihood of the observation $\mathbf{o}_t$ can be calculated by

$$p(\mathbf{o}_t|\mathcal{M}, \mathcal{M}_\mathtt{b}, \theta_t) = \sum_{m \in \theta_t} c^{(m)} \mathcal{N}\left(\mathbf{o}_t; \boldsymbol{\mu}_\mathtt{s}^{(m)} + \boldsymbol{\mu}_\mathtt{b}^{(r_m)}, \boldsymbol{\Sigma}_\mathtt{s}^{(m)} + \boldsymbol{\Sigma}_\mathtt{b}^{(r_m)}\right) \tag{22}$$

If the covariance bias, $\boldsymbol{\Sigma}_\mathtt{b}^{(r_m)}$, is full then the cost of computing the likelihood in equation 22 will be the same as a full covariance matrix system. In practice the covariance bias is normally restricted to be diagonal.

In the same fashion as the other transformation parameters, the bias model parameters $\mathcal{M}_\mathtt{b}$ need to be found given the observed speech $\mathbf{O}$ and the original speech model $\mathcal{M}$. Again the ML criterion and the EM algorithm are used to find the bias parameter estimates. Let $\mathcal{M}_\mathtt{b}$ denote the current bias parameters and $\hat{\mathcal{M}}_\mathtt{b}$ the estimated bias parameters and the complete data is $\mathbf{Z} = (\mathbf{O}, \mathbf{B}, \boldsymbol{\Theta})$, where $\mathbf{B}$ and $\boldsymbol{\Theta}$ represent the bias sequence and possible hidden state sequence for $\mathbf{O}$, respectively. The auxiliary function can then expressed as

$$\mathcal{Q}(\hat{\mathcal{M}}_\mathtt{b}; \mathcal{M}_\mathtt{b}) = \mathbb{E}\left[\log p(\mathbf{Z}|\mathcal{M}, \hat{\mathcal{M}}_\mathtt{b})|\mathcal{O}, \mathcal{M}, \mathcal{M}_\mathtt{b}\right] \tag{23}$$

where $\mathbb{E}$ denotes the conditional expectation over all possible hidden sequences $(\mathbf{S}, \boldsymbol{\Theta})$, given the observation sequence $\mathbf{O}$ computed with model set $\mathcal{M}$ and bias parameter $\mathcal{M}_\mathtt{b}$. Following [21, 13], the auxiliary function may be expressed as

$$\begin{aligned}\mathcal{Q}(\hat{\mathcal{M}}_\mathtt{b}; \mathcal{M}_\mathtt{b}) = &-\frac{1}{2}\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_{\mathtt{o},t}^{(m)} \times \\ &\mathbb{E}\left[\log|\hat{\boldsymbol{\Sigma}}_\mathtt{b}^{(r_m)}| + (\mathbf{b}_t - \hat{\boldsymbol{\mu}}_\mathtt{b}^{(r_m)})^\mathsf{T}\hat{\boldsymbol{\Sigma}}_\mathtt{b}^{(r_m)\text{-}1}(\mathbf{b}_t - \hat{\boldsymbol{\mu}}_\mathtt{b}^{(r_m)})|\mathbf{o}_t, m, \mathcal{M}, \mathcal{M}_\mathtt{b}\right]\end{aligned} \tag{24}$$

where $\gamma_{\mathtt{o},t}^{(m)}$ is the posterior probability of component m given the observation sequence $\mathbf{O}$, bias parameter $\mathcal{M}_\mathtt{b}$, and model set $\mathcal{M}$. Differentiating equation 24 with respect to $\hat{\boldsymbol{\mu}}_\mathtt{b}^{(r_m)}$ and $\hat{\boldsymbol{\Sigma}}_\mathtt{b}^{(r_m)}$, and equating to zero yields

$$\hat{\boldsymbol{\mu}}_\mathtt{b}^{(r)} = \frac{\sum_{m \in r}\sum_{t=1}^{T}\gamma_{\mathtt{o},t}^{(m)}\mathbb{E}\left[\mathbf{b}_t|\mathbf{o}_t, m, \mathcal{M}, \mathcal{M}_\mathtt{b}^{(r_m)}\right]}{\sum_{m \in r}\sum_{t=1}^{T}\gamma_{\mathtt{o},t}^{(m)}} \tag{25}$$

$$\hat{\boldsymbol{\Sigma}}_\mathtt{b}^{(r)} = \frac{\sum_{m \in r}\sum_{t=1}^{T}\gamma_{\mathtt{o},t}^{(m)}\mathbb{E}\left[\mathbf{b}_t\mathbf{b}_t^\mathsf{T}|\mathbf{o}_t, m, \mathcal{M}, \mathcal{M}_\mathtt{b}^{(r_m)}\right]}{\sum_{m \in r}\sum_{t=1}^{T}\gamma_{\mathtt{o},t}^{(m)}} - \hat{\boldsymbol{\mu}}_\mathtt{b}^{(r_m)}\hat{\boldsymbol{\mu}}_\mathtt{b}^{(r_m)\mathsf{T}} \tag{26}$$

The conditional expectation of the bias vector and its outer product in equations 25 and 26 can be calculated as

$$\mathbb{E}\left[\mathbf{b}_t|\mathbf{o}_t, m, \mathcal{M}, \mathcal{M}_\mathtt{b}^{(r_m)}\right] = (\boldsymbol{\Sigma}_\mathtt{s}^{(m)\text{-}1} + \boldsymbol{\Sigma}_\mathtt{b}^{(r_m)\text{-}1})^{\text{-}1}(\boldsymbol{\Sigma}_\mathtt{s}^{(m)\text{-}1}(\mathbf{o}_t - \boldsymbol{\mu}_\mathtt{b}^{(r_m)}) + \boldsymbol{\Sigma}_\mathtt{b}^{(r_m)\text{-}1}\boldsymbol{\mu}_\mathtt{s}^{(m)}) \tag{27}$$

$$\begin{aligned}\mathbb{E}\left[\mathbf{b}_t\mathbf{b}_t^\mathsf{T}|\mathbf{o}_t, m, \mathcal{M}, \mathcal{M}_\mathtt{b}^{(r_m)}\right] = &(\boldsymbol{\Sigma}_\mathtt{s}^{(m)\text{-}1} + \boldsymbol{\Sigma}_\mathtt{b}^{(r_m)\text{-}1})^{\text{-}1} \\ &+ \mathbb{E}\left[\mathbf{b}_t|\mathbf{o}_t, m, \mathcal{M}, \mathcal{M}_\mathtt{b}^{(r_m)}\right]\mathbb{E}\left[\mathbf{b}_t^\mathsf{T}|\mathbf{o}_t, m, \mathcal{M}, \mathcal{M}_\mathtt{b}^{(r_m)}\right]\end{aligned} \tag{28}$$

Thus the bias parameter estimation consists of iteratively applying the expectation step in equations 27 and 28, and the maximisation step in equations 25 and 26 starting from the initial bias values. Each iteration is guaranteed not to decrease the likelihood of the observation data [21, 13].

3 Predictive Model-Based Noise Compensation

The previous section has described the use of general linear transformations of both the mean and covariance matrix parameters to represent the differences between the "clean" speech acoustic models and speech in the target acoustic conditions. This section describes the use of model-based compensation schemes, where the model parameters are modified using an estimate of the noise model in the target environment and a mismatch function that describes the impact of that noise on the model parameters. A review of three model-based compensation scheme is given: parallel model combination (PMC) [22]; VTS [10]; and JUD [23]. These schemes will be described under the general heading of predictive transforms. Though these predictive schemes are not investigated in the results in this paper, they motivate limitations in the standard linear transforms described in the previous section.

The production of the underlying speech signal is influenced by the acoustic environment such as additive background noise and channel distortions [24]. The standard model of the corrupted speech in time-domain is given by

$$o(\tau) = h(\tau) * s(\tau) + n(\tau) \tag{29}$$

where $o(\tau)$ is the noise-corrupted speech, $h(\tau)$ represents the channel or convolutional noise, $s(\tau)$ the clean speech and $n(\tau)$ the additive noise. In the cepstral domain this relationship is given by

$$\mathbf{o}_t = \mathbf{s}_t + \mathbf{h} + \mathbf{C}\log(\mathbf{1} + \exp(\mathbf{C}^{-1}(\mathbf{n}_t - \mathbf{s}_t - \mathbf{h})) \tag{30}$$

where matrices $\mathbf{C}$ and $\mathbf{C}^{-1}$ are the discrete cosine transform (DCT) matrix and its inverse. The log() and exp() functions indicate element-wise operations that yield a vector of the same dimensionality as the input vector. In the above expression, as is commonly done, the convolutional noise term, $\mathbf{h}$, has been assumed to be constant, independent of time.

Equation 30 only describes the impact of the noise on the *static* model parameters. In practice improved performance is obtained by compensating all the model parameters [22], including the delta and delta-delta parameters. A commonly used approach to handle this is the continuous-time approximation [25, 26]. The following section only consider the static parameter compensation[1].

For all these predictive approaches it is necessary to estimate the parameters of the noise model, normally $\{\boldsymbol{\mu}_{\mathbf{n}}, \boldsymbol{\Sigma}_{\mathbf{n}}, \boldsymbol{\mu}_h\}$. These are usually estimated using the ML criterion [27, 28]. The advantage of using these predictive schemes over more general adaptive approaches such as MLLR and CMLLR is that the number of parameters to estimate for the target acoustic condition is normally very small, and the non-linear impact of the noise of the clean speech models is better represented. However the disadvantage is that the mismatch function needs to be specified. Approximations in the definition of the mismatch function can impact performance.

3.1 Parallel Model Combination

PMC [26, 22] is a term that describes a set of model-based compensation approaches. These all aim to approximate the following expectations where the corrupted speech is related to the clean speech and the noise models using equation 30.

$$\boldsymbol{\mu}_{\mathbf{o}}^{(m)} \approx \mathbb{E}\left\{\mathbf{o}_t|m\right\} \tag{31}$$

$$\boldsymbol{\Sigma}_{\mathbf{o}}^{(m)} \approx \mathbb{E}\left\{\mathbf{o}_t\mathbf{o}_t^{\mathsf{T}}|m\right\} - \boldsymbol{\mu}_{\mathbf{o}}^{(m)}\boldsymbol{\mu}_{\mathbf{o}}^{(m)\mathsf{T}} \tag{32}$$

Obtaining expressions for this corrupted speech output distribution given the clean acoustic model and a noise model is not simple because the corrupted speech is a non-linear function of the noise,

[1] The notation has been kept the same as the previous section to help understanding.

channel and clean speech. To address this problem various forms of approximation have been proposed.

The simplest forms of approximation are the log-normal and log-add approximations. The log-normal approximation is based on the assumption that the sum of two log-normally distributed variables is itself approximately log-normally distributed. Log-add is a simple approximation based on the assumption that the variances are small, so the corrupted mean of the static dimension can be written as

$$\boldsymbol{\mu}_{\mathtt{o}}^{(m)} = \mathbf{C} \log \left(\mathbf{C}^{-1} \exp(\boldsymbol{\mu}_{\mathtt{s}}^{(m)} + \boldsymbol{\mu}_{\mathtt{h}}) + \mathbf{C}^{-1} \exp(\boldsymbol{\mu}_{\mathtt{n}}) \right) \tag{33}$$

The covariance matrix is assumed to be unaltered, $\boldsymbol{\Sigma}_{\mathtt{o}}^{(m)} = \boldsymbol{\Sigma}_{\mathtt{s}}^{(m)}$.

More complex, and accurate, versions have also been derived including data-driven PMC (DPMC) and iterative PMC (IPMC) [26]. Though more accurate, these approaches are computationally expensive.

3.2 Vector Taylor Series

An alternative approximation to those referred to as PMC is the Vector Taylor Series (VTS) approximation. A truncated VTS is used to approximate the non-linearities in equation 30. The first-order VTS approximation of the static corrupted speech may be expressed as

$$\mathbf{o}_{\mathtt{vts}t} = \mathbf{o}\Big|_{\mu_0{}^{(m)}} + \mathbf{J}_{\mathtt{s}}^{(m)} \left(\mathbf{s}_t - \boldsymbol{\mu}_{\mathtt{s}}^{(m)} \right) + \mathbf{J}_{\mathtt{n}}^{(m)} \left(\mathbf{n}_t - \boldsymbol{\mu}_{\mathtt{n}} \right) + \mathbf{J}_{\mathtt{h}}^{(m)} \left(\mathbf{h} - \boldsymbol{\mu}_{\mathtt{h}} \right) \tag{34}$$

where $|_{\mu_0{}^{(m)}}$ is the Taylor series expansion point indicating the function is evaluated at, the clean speech component mean $\boldsymbol{\mu}_{\mathtt{s}}^{(m)}$, and current estimates of the additive noise mean $\boldsymbol{\mu}_{\mathtt{n}}$ and channel noise $\boldsymbol{\mu}_{\mathtt{h}}$. The Jacobian matrices are given by

$$\mathbf{J}_{\mathtt{s}}^{(m)} = \mathbf{I} - \mathbf{C}\mathbf{F}\mathbf{C}^{-1} \tag{35}$$

$$\mathbf{J}_{\mathtt{h}}^{(m)} = \frac{\partial \mathbf{o}_t}{\partial \mathbf{h}_t}\Big|_{\mu_0{}^{(m)}} = \frac{\partial \mathbf{o}_t}{\partial \mathbf{s}_t}\Big|_{\mu_0{}^{(m)}}, \qquad \mathbf{J}_{\mathtt{n}}^{(m)} = \frac{\partial \mathbf{o}_t}{\partial \mathbf{n}_t}\Big|_{\mu_0{}^{(m)}} = \mathbf{C}\mathbf{F}\mathbf{C}^{-1} \tag{36}$$

where the elements of the diagonal matrix $\mathbf{F}$ are

$$f_{ii} = \frac{\exp([\mathbf{C}^{-1}]_i(\boldsymbol{\mu}_{\mathtt{n}} - \boldsymbol{\mu}_{\mathtt{s}}^{(m)} - \boldsymbol{\mu}_{\mathtt{h}}))}{1 + \exp([\mathbf{C}^{-1}]_i(\boldsymbol{\mu}_{\mathtt{n}} - \boldsymbol{\mu}_{\mathtt{s}}^{(m)} - \boldsymbol{\mu}_{\mathtt{h}}))} \tag{37}$$

where $[\mathbf{C}^{-1}]_i$ is a row vector that is the ith row of the inverse DCT matrix, $\mathbf{C}^{-1}$. The terms f_{ii} vary from 0 to 1 depending on the ratio of the speech to the noise. If the noise level $\boldsymbol{\mu}_{\mathtt{n}}$ is greater than the speech $\boldsymbol{\mu}_{\mathtt{s}}^{(m)}$ in the log-spectral domain, then $f_{ii} \rightarrow 1$ and $\mathbf{J}_{\mathtt{s}}^{(m)}$ tends to zero; otherwise if little noise is present, $f_{ii} \rightarrow 0$ and $\mathbf{J}_{\mathtt{s}}^{(m)}$ also tends to identity. The term $\mathbf{J}_{\mathtt{s}}^{(m)}$ behaves in the opposite manner to $\mathbf{J}_{\mathtt{n}}^{(m)}$.

In the same fashion as PMC, the corrupted speech mean for each component m may be approximated by the expected value of equation 34 [29]

$$\boldsymbol{\mu}_{\mathtt{o}}^{(m)} \approx \mathbb{E}\left\{ \mathbf{o}_{\mathtt{vts}t} | m \right\} \tag{38}$$

$$= \boldsymbol{\mu}_{\mathtt{s}}^{(m)} + \boldsymbol{\mu}_{\mathtt{h}} + \mathbf{C} \log(\mathbf{1} + \exp(\mathbf{C}^{-1}(\boldsymbol{\mu}_{\mathtt{n}} - \boldsymbol{\mu}_{\mathtt{s}}^{(m)} - \boldsymbol{\mu}_{\mathtt{h}}))) \tag{39}$$

assuming the clean speech, additive noise and channel noise are independent of each other. This yields the same estimate of the mean as the PMC log-add approximation. The covariance is given by

$$\boldsymbol{\Sigma}_{\mathtt{o}}^{(m)} \approx \mathbb{E}\left\{ \mathbf{o}_{\mathtt{vts}t} \mathbf{o}_{\mathtt{vts}t}^{\mathsf{T}} | m \right\} - \boldsymbol{\mu}_{\mathtt{o}}^{(m)} \boldsymbol{\mu}_{\mathtt{o}}^{(m)\mathsf{T}} \tag{40}$$

$$\approx \mathbf{J}_{\mathtt{s}}^{(m)} \boldsymbol{\Sigma}_{\mathtt{s}}^{(m)} \mathbf{J}_{\mathtt{s}}^{(m)\mathsf{T}} + \mathbf{J}_{\mathtt{h}}^{(m)} \boldsymbol{\Sigma}_{\mathtt{h}} \mathbf{J}_{\mathtt{h}}^{(m)} + \mathbf{J}_{\mathtt{n}}^{(m)} \boldsymbol{\Sigma}_{\mathtt{n}} \mathbf{J}_{\mathtt{n}}^{(m)} \tag{41}$$

where $\mathbf{\Sigma}_{\mathtt{n}}$ and $\mathbf{\Sigma}_{\mathtt{h}}$ denote the variance of the static additive noise and the variance of the static channel, respectively. Since the Jacobian matrices are full, the corrupted speech covariance matrix will also be full and hence is normally diagonalised for standard decoders. Also, it is often assumed that the channel noise does not vary, that is $\mathbf{\Sigma}_{\mathtt{h}} = 0$. Hence the static corrupted speech variance may be given by

$$\mathbf{\Sigma}_{\mathtt{o}}^{(m)} \approx \text{diag}\left(\mathbf{J}_{\mathtt{s}}^{(m)}\mathbf{\Sigma}_{\mathtt{s}}^{(m)}\mathbf{J}_{\mathtt{s}}^{(m)} + \mathbf{J}_{\mathtt{n}}^{(m)}\mathbf{\Sigma}_{\mathtt{n}}\mathbf{J}_{\mathtt{n}}^{(m)}\right) \tag{42}$$

3.3 Joint Uncertainty Decoding

Recently there has been interest in uncertainty decoding (UD) framework for speech recognition that incorporates the notion of uncertainty introduced by environmental noise [30, 31, 32, 33, 23]. These approaches have the property that the uncertainty varies with the noise level and is propagated to the recogniser as an additional variance bias to the model variance. In observation uncertainty forms [30], the uncertainty is represented by the variance of the residual occurred during enhancement. Hence the clean speech posterior is passed to the decoder instead of using a point estimate of the features. Alternatively, in UD schemes [34, 23], the conditional distribution of the corrupted speech, given the clean speech is propagated into the recognition stage. Two approaches for UD are front-end and model-based schemes. In front-end UD [34, 23], by using approximation of the conditional distribution the uncertainty information passed into the decoding stage depends only on the observed features which is independent from the back-end acoustic model. `SPLICE` with uncertainty [34] and the front-end JUD method [23] are the two specific forms of front-end UD. However, there are issues with using these forms of uncertainty decoding in low SNR conditions [9].

In the model-based UD [23], the conditional distributions are computed over the regression classes and thus the uncertainty information depends on the class of the acoustic model component. JUD [23] has shown to be an effective model-based approaches which are characterised by a feature transform and an uncertainty bias on the model variances. The transform and uncertainty bias for JUD are derived from a model of the joint distribution of the clean and the noisy speech for the class r_m. The joint distribution is assumed to be Gaussian such that

$$\begin{bmatrix} \mathbf{s}_t \\ \mathbf{o}_t \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu}_{\mathtt{s}}^{(r_m)} \\ \boldsymbol{\mu}_{\mathtt{o}}^{(r_m)} \end{bmatrix}, \begin{bmatrix} \mathbf{\Sigma}_{\mathtt{s}}^{(r_m)} & \mathbf{\Sigma}_{\mathtt{so}}^{(r_m)} \\ \mathbf{\Sigma}_{\mathtt{os}}^{(r_m)} & \mathbf{\Sigma}_{\mathtt{o}}^{(r_m)} \end{bmatrix}\right) \tag{43}$$

and thus the conditional distribution will also be Gaussian. Using the Gaussian form for the corrupted speech conditional distribution, the corrupted speech observation likelihood for state θ_t may be approximated by

$$p(\mathbf{o}_t|\mathcal{M}, \mathcal{T}, \theta_t) = \int_{\mathcal{R}^D} p(\mathbf{o}_t|\mathbf{s}_t, \mathcal{T}) p(\mathbf{s}_t|\theta_t) d\mathbf{s}_t \tag{44}$$

$$\approx \sum_{m \in \theta_t} c^{(m)} |\mathbf{A}^{(r_m)}| \mathcal{N}\left(\mathbf{A}^{(r_m)}\mathbf{o}_t + \mathbf{b}^{(r_m)}; \boldsymbol{\mu}_{\mathtt{s}}^{(m)}, \mathbf{\Sigma}_{\mathtt{s}}^{(m)} + \mathbf{\Sigma}_{\mathtt{b}}^{(r_m)}\right) \tag{45}$$

and the transform parameters

$$\begin{aligned} \mathbf{A}^{(r_m)} &= \mathbf{\Sigma}_{\mathtt{s}}^{(r_m)} \mathbf{\Sigma}_{\mathtt{os}}^{(r_m)\text{-}1}, \qquad \mathbf{b}^{(r_m)} = \boldsymbol{\mu}_{\mathtt{s}}^{(r_m)} - \mathbf{A}^{(r_m)} \boldsymbol{\mu}_{\mathtt{o}}^{(r_m)} \\ \mathbf{\Sigma}_{\mathtt{b}}^{(r_m)} &= \mathbf{A}^{(r_m)} \mathbf{\Sigma}_{\mathtt{o}}^{(r_m)} \mathbf{A}^{(r_m)\mathsf{T}} - \mathbf{\Sigma}_{\mathtt{s}}^{(r_m)} \end{aligned} \tag{46}$$

and $\mathcal{T} = \left\{\{\mathbf{A}^{(1)}, \mathbf{b}^{(1)}, \mathbf{\Sigma}_{\mathtt{b}}^{(1)}\}, \ldots, \{\mathbf{A}^{(R)}, \mathbf{b}^{(R)}, \mathbf{\Sigma}_{\mathtt{b}}^{(R)}\}\right\}$, r_m denotes the regression class r that component m belongs to, and R the total number of regression classes. Increasing the number of classes R to equal the number of model components M, and using a diagonal acoustic model variances, is equivalent to VTS model compensation of each individual acoustic model component [35]. However by having R far smaller than the number of Gaussian components in the system, JUD can be computationally far less expensive than VTS.

Comparing this form of resultant likelihood calculation to the form of CMLLR in equation 12, shows that JUD is similar to CMLLR, both may be viewed as estimating the "clean" speech, but JUD introduces a variance bias term to indicate that uncertainty in the estimate of the clean speech will increase as the SNR decreases. Contrasting the JUD likelihood in equation 45 with the bias form in equation 22 shows that, though both allow the variance to increase, the estimate of the clean speech in the bias form is far simpler than that in JUD. A clear direction of interest is to combine CMLLR with variance biases, to yield a transform similar in structure to JUD. Though in theory this could be implemented using the hierarchical transform structure in HTK V3.4 [36], an integrated transform estimation scheme would be preferable.

4 Adaptive Training

Adaptive training [1] is a powerful technique for building speech recognition systems on non-homogeneous training data. The basic concept of adaptive training is to use one or more transforms for each speaker and acoustic environment when training the acoustic models. These transforms should remove the unwanted variability (the speaker changes and environment changes) allowing a "neutral", canonical, model to be estimated. Thus during training two sets of models are generated: a canonical model set for the desired true variability of the speech data; and a set of transforms to represent the unwanted variability. The canonical model represents the speech variability of the training data, which is independent of speaker and acoustic conditions. The form of the canonical model used here is the standard HMMs denoted by $\mathcal{M}$. A set of transforms $\mathcal{T}$ represent the unwanted non-speech variabilities such as different speaker and/or acoustic environment. Usually a linear transform of the acoustic models is used to represent the acoustic condition of a particular homogeneous block and adapt the canonical model to that particular acoustic condition. The canonical model is estimated given the set of transforms accounting for non-speech variabilities. Hence, in recognition, the canonical model must be adapted by an appropriate transform to represent both speech and specific non-speech variabilities of a particular test acoustic condition. Due to the modelling of desired speech variability, the canonical model is more amenable to being adapted to a new test acoustic condition than a multi-style trained system.

4.1 ML Adaptive Training

Assume that the training data is written as $\mathcal{O} = \{\mathbf{O}^{(1)}, \ldots, \mathbf{O}^{(H)}\}$ and $\mathcal{H}_{\mathtt{ref}} = \{\mathcal{H}_{\mathtt{ref}}^{(1)}, \ldots, \mathcal{H}_{\mathtt{ref}}^{(H)}\}$, where $\mathbf{O}^{(h)}$ is the observation sequence of a homogeneous block associated with a particular acoustic condition h, $\mathcal{H}_{\mathtt{ref}}^{(h)}$ is the corresponding transcription sequence. Given the canonical model, the homogeneous blocks are assumed to be independent of each. Thus the log-likelihood of the heterogeneous training data may be expressed as

$$\log(p(\mathcal{O}|\mathcal{M}, \mathcal{T}, \mathcal{H}_{\mathtt{ref}})) = \sum_{h=1}^{H} \log\left(p(\mathbf{O}^{(h)}|\mathcal{M}, \mathcal{T}^{(h)}, \mathcal{H}_{\mathtt{ref}}^{(h)})\right) \tag{47}$$

where the set of transforms is denoted by $\mathcal{T} = \left\{\mathcal{T}^{(1)}, \ldots, \mathcal{T}^{(H)}\right\}$. In ML adaptive training estimation of the canonical model and transform set estimation are interleaved. The estimate of the canonical model parameters maximise equation 47

$$\hat{\mathcal{M}} = \arg\max_{\mathcal{M}} \left\{\log(p(\mathcal{O}|\mathcal{M}, \mathcal{T}, \mathcal{H}_{\mathtt{ref}}^{(h)}))\right\} \tag{48}$$

where $\mathcal{T}$ is the current set of ML transforms given the canonical model estimate. The transform estimate for homogeneous block h is given by

$$\hat{\mathcal{T}}^{(h)} = \arg\max_{\mathcal{T}} \left\{\log(p(\mathbf{O}^{(h)}|\hat{\mathcal{M}}, \mathcal{T}, \mathcal{H}_{\mathtt{ref}}^{(h)}))\right\} \tag{49}$$

Linear transforms such as MLLR and CMLLR have been successfully applied in adaptive training [1, 2]. The transform estimation formulae for adaptive training are the same as for adaptation described in section 2. In this section, the estimation of the canonical model given the set of transforms is given for adaptive training with CMLLR [2] is described. The auxiliary function used to estimate the acoustic models can be written as [2]

$$\begin{aligned}\mathcal{Q}(\hat{\mathcal{M}};\mathcal{M},\hat{\mathcal{T}}) = & \ \mathbb{E}\left[\log p(\mathcal{O},\boldsymbol{\Theta}|\hat{\mathcal{M}},\hat{\mathcal{T}})|\mathcal{O},\mathcal{M},\hat{\mathcal{T}}\right] \\ = & -\frac{1}{2}\sum_{m=1}^{M}\sum_{h=1}^{H}\sum_{t=1}^{T^{(h)}}\gamma_{\mathtt{o},t}^{(mh)}\Big[\log(|\hat{\boldsymbol{\Sigma}}_{\mathtt{s}}^{(m)}|) \\ & -\log(|\hat{\mathbf{A}}^{(r_m h)}|^2)+(\hat{\mathbf{s}}_t^{(r_m h)}-\hat{\boldsymbol{\mu}}_{\mathtt{s}}^{(m)})^{\mathsf{T}}\hat{\boldsymbol{\Sigma}}_{\mathtt{s}}^{(m)\text{-}1}(\hat{\mathbf{s}}_t^{(r_m h)}-\hat{\boldsymbol{\mu}}_{\mathtt{s}}^{(m)})\Big]\end{aligned} \tag{50}$$

where $\hat{\mathbf{s}}_t^{(r_m h)} = \hat{\mathbf{W}}^{(r_m h)}\boldsymbol{\zeta}_t$ is the transformed observation and $\gamma_{\mathtt{o},t}^{(mh)}$, is the posterior probability of an observation being generated by component m, homogeneous block h, with transcription $\mathcal{H}_{\mathtt{ref}}^{(h)}$. The resultant estimate for mean and covariance for each component m are given by

$$\hat{\boldsymbol{\mu}}_{\mathtt{s}}^{(m)} = \frac{\sum_{h=1}^{H}\sum_{t=1}^{T^{(h)}}\gamma_{\mathtt{o},t}^{(mh)}\hat{\mathbf{s}}_t^{(r_m h)}}{\sum_{h=1}^{H}\sum_{t=1}^{T^{(h)}}\gamma_{\mathtt{o},t}^{(mh)}} \tag{51}$$

$$\hat{\boldsymbol{\Sigma}}_{\mathtt{s}}^{(m)} = \mathrm{diag}\left(\frac{\sum_{h=1}^{H}\sum_{t=1}^{T^{(h)}}\gamma_{\mathtt{o},t}^{(mh)}\left(\hat{\mathbf{s}}_t^{(r_m h)}-\hat{\boldsymbol{\mu}}_{\mathtt{s}}^{(m)}\right)\left(\hat{\mathbf{s}}_t^{(r_m h)}-\hat{\boldsymbol{\mu}}_{\mathtt{s}}^{(m)}\right)^{\mathsf{T}}}{\sum_{h=1}^{H}\sum_{t=1}^{T^{(h)}}\gamma_{\mathtt{o},t}^{(mh)}}\right) \tag{52}$$

ML adaptive training has also been used with both VTS [8] and JUD [7] model-based compensation. In [7] a second-order gradient descent-based schemes was described to fund the canonical model parameters. Alternatively an expectation-maximisation (EM) approach was used in [8] based on the approach used in [21]. The approach described in [8] is closely related to the estimation of the canonical model in this paper described in section 5.

4.2 Discriminative Adaptive Training

The previous section has described adaptive training with the ML criterion. However state-of-the-art speech recognition systems are normally trained using discriminative criteria. Commonly used discriminative training criteria include: minimum classification error (MCE) [37]; maximum mutual information (MMI) [38]; and MPE [14]. This section will review discriminative training and discriminative adaptive training (DAT) based on the MPE criterion [39]. MPE training is an example of minimum Bayes' risk training. It can be expressed as

$$\mathcal{F}_{\mathtt{MPE}}(\mathcal{M}) = \sum_{h=1}^{H}\sum_{\mathcal{H}}P(\mathcal{H}|\mathbf{O}^{(h)},\mathcal{M})\mathcal{L}(\mathcal{H},\mathcal{H}_{\mathtt{ref}}^{(h)}) \tag{53}$$

where $\mathbf{O}^{(h)}$ is the observation sequence of the training utterance h, $\mathcal{H}$ denotes a possible hypothesis of the training data and $\mathcal{L}(\mathcal{H},\mathcal{H}_{\mathtt{ref}}^{(h)})$ is the error (loss) measured in phones of the hypothesis $\mathcal{H}$ given the reference $\mathcal{H}_{\mathtt{ref}}^{(h)}$. Whereas the ML criterion can be optimised by the EM algorithm, a weak-sense auxiliary function [39] is often used to optimise the MPE criterion.

Adaptive training within the framework of discriminative training has been used to improve the recognition performance. An ideal DAT approach is to discriminatively update both the canonical model and the transform parameters during training. However this approach is problematic when unsupervised adaptation is required to be used, as there is no correct transcription available. Though it is possible to generate hypotheses, and treat them as if they were correct as in ML training, discriminative criteria are more sensitive to errors in these hypotheses than ML training [40]. To avoid this problem, a simplified DAT scheme is often adopted in which only

the canonical model is discriminatively updated given a set of ML estimated transforms [40]. The basic training procedures are as follows:

1. perform standard ML adaptive training resulting in a canonical model estimate $\hat{\mathcal{M}}_{\texttt{ML}}$ and a set of ML transform estimates $\hat{\mathcal{T}}_{\texttt{ML}} = \{\hat{\mathcal{T}}_{\texttt{ML}}^{(1)}, \cdots, \hat{\mathcal{T}}_{\texttt{ML}}^{(H)}\}$.
2. given the ML transform estimates, $\hat{\mathcal{T}}_{\texttt{ML}}$, the canonical model is discriminatively trained.
3. repeat step (2) to yield, the final discriminative canonical model, $\hat{\mathcal{M}}_{\texttt{MPE}}$.

In adaptation, given the discriminatively trained model $\hat{\mathcal{M}}_{\texttt{MPE}}$, the ML criterion is used to estimate transform parameters for each homogeneous block of the test data. DAT with the linear transforms such as MLLR and CMLLR has been investigated [40]. However discriminative versions of the model-based compensation schemes have not previously been investigated.

5 Noisy CMLLR

Noisy CMLLR (NCMLLR) is a new form of linear transform. It may be viewed from two very different perspectives. The first, and the one adopted in this work, is to view it as an extension to the standard CMLLR transform described in section 2. The second is realted to forms of shared factor analysis model [12, 41] but modified so that the shared loading matrices and noise variances are used as transforms rather than as a method for covariance matrix modelling. The form of the NCMLLR transformation is exactly the same as that of JUD, described in section 3, but the parameters of the transform are estimated using ML, rather than based on a mismatch function and noise model estimates. It can thus be used in situations where deriving accurate mismatch functions is not possible..

In addition to transform estimation, adaptive training using NCMLLR is also described. Given the relationship of the form of NCMLLR to JUD, The canonical model estimation for adaptive training with NCMLLR in section 5.3 can be directly applied to JUD, rather than using the second-order optimisation scheme described in [7]. Note, since JUD and VTS are equivalent when the number of regression classes in JUD is the same as the number of Gaussian components, the adaptive training scheme described here is the same as that described in [8] under these conditions as well.

5.1 Generative Model

This section presents a probabilistic model for the corrupted speech in the feature space. First, assume that the corrupted observation $\mathbf{o}_t$ can be written as a generative model of the clean speech vectors $\mathbf{s}_t$ for each regression class r_m in the form

$$\mathbf{o}_t = \mathbf{H}^{(r_m)}\mathbf{s}_t + \mathbf{g}^{(r_m)} + \mathbf{n}_t \tag{54}$$

where $\mathbf{H}^{(r_m)}$ is a linear transform and $\mathbf{g}^{(r_m)}$ is a bias on the clean speech, r_m denotes the regression class r that component m belongs to, and $\mathbf{n}_t$ is a zero-mean Gaussian additive noise with covariance matrix $\mathbf{\Psi}^{(r_m)}$ such that $\mathbf{n}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{\Psi}^{(r_m)})$. Furthermore, the clean speech $\mathbf{s}_t$ is assumed to be generated by state θ_t of an HMM. The acoustic model $\mathcal{M}$ consists of Gaussian components each defined by a prior, $c^{(m)}$, mean, $\boldsymbol{\mu}_{\mathbf{s}}^{(m)}$, and diagonal covariance matrix, $\mathbf{\Sigma}_{\mathbf{s}}^{(m)}$, so that the likelihood can be written as

$$p(\mathbf{s}_t|\mathcal{M}, \mathcal{T}, \theta_t) = \sum_{m \in \theta_t} c^{(m)} \mathcal{N}(\mathbf{s}_t; \boldsymbol{\mu}_{\mathbf{s}}^{(m)}, \mathbf{\Sigma}_{\mathbf{s}}^{(m)}) \tag{55}$$

The corrupted speech observation $\mathbf{o}_t$ at time t is assumed to be conditional independent of all other observations given the clean speech and the noise at that time. Then the corrupted speech likelihood for a state θ_t is

$$p(\mathbf{o}_t|\mathcal{M}, \mathcal{T}, \theta_t) = \sum_{m \in \theta_t} c^{(m)} \mathcal{N}\left(\mathbf{o}_t; \mathbf{H}^{(r_m)}\boldsymbol{\mu}_{\mathbf{s}}^{(m)} + \mathbf{g}^{(r_m)}, \mathbf{H}^{(r_m)}\mathbf{\Sigma}_{\mathbf{s}}^{(m)}\mathbf{H}^{(r_m)\mathsf{T}} + \mathbf{\Psi}^{(r_m)}\right) \tag{56}$$

This can be rewritten as a transformation of the observations

$$p(\mathbf{o}_t|\mathcal{M},\mathcal{T},\theta_t) = \sum_{m\in\theta_t} c^{(m)}|\mathbf{A}^{(r_m)}|\mathcal{N}\left(\mathbf{A}^{(r_m)}\mathbf{o}_t+\mathbf{b}^{(r_m)};\boldsymbol{\mu}_{\mathsf{s}}^{(m)},\boldsymbol{\Sigma}_{\mathsf{s}}^{(m)}+\boldsymbol{\Sigma}_{\mathsf{b}}^{(r_m)}\right) \tag{57}$$

where $\mathbf{A}^{(r_m)} = \mathbf{H}^{(r_m)\text{-}1}$, $\mathbf{b}^{(r_m)} = -\mathbf{H}^{(r_m)\text{-}1}\mathbf{g}^{(r_m)}$ and $\boldsymbol{\Sigma}_{\mathsf{b}}^{(r_m)} = \mathbf{A}^{(r_m)}\boldsymbol{\Psi}^{(r_m)}\mathbf{A}^{(r_m)\mathsf{T}}$. This form of transformation will be referred to as noisy CMLLR (NCMLLR). NCMLLR has the same form as the JUD transform [9]. It is also similar to CMLLR [2] where from equation 12

$$p(\mathbf{o}_t|\mathcal{M},\mathcal{T},\theta_t) = \sum_{m\in\theta_t} c^{(m)}|\mathbf{A}^{(r_m)}|\mathcal{N}\left(\mathbf{A}^{(r_m)}\mathbf{o}_t+\mathbf{b}^{(r_m)};\boldsymbol{\mu}_{\mathsf{s}}^{(m)},\boldsymbol{\Sigma}_{\mathsf{s}}^{(m)}\right) \tag{58}$$

and the observation in feature space, and clean speech estimate, are given by

$$\mathbf{o}_t = \mathbf{H}^{(r_m)}\mathbf{s}_t + \mathbf{g}^{(r_m)}; \quad \mathbf{s}_t = \mathbf{A}^{(r_m)}\mathbf{o}_t + \mathbf{b}^{(r_m)} \tag{59}$$

Both CMLLR and NCMLLR have an affine transform of the features, but in equation 57 there is an additional variance bias, $\boldsymbol{\Sigma}_{\mathsf{b}}^{(r_m)}$, for modelling the changes in the variance of the corrupted speech due to noise $\mathbf{n}_t$, hence the name. Whereas JUD transforms are only dependent on the estimated noise model, NCMLLR specifically requires the noise-corrupted adaptation data to estimate the transform parameters. With NCMLLR the amount of training data required will depend on the number and complexity of transforms used.

NCMLLR is also related to schemes based on factor analysis (FA) [11], which have the same general form as the generative model in equation 54. FA is a standard statistical method for modelling the covariance matrices of high dimensional data using a small number of latent variables. In FA the latent variables of dimension $P < D$, D is a dimension of the corrupted speech $\mathbf{o}_t$, are defined to be independent and Gaussian with unit variance such that $\mathbf{s}_t \sim \mathcal{N}(\mathbf{0},\mathbf{I}_P)$, where $\mathbf{I}_P$ is the $P \times P$ identity matrix. Shared FA and FA invariant to linear transformations (FACILT) [12] are generalised form of standard FA for modelling covariance matrices. One version of these approaches would yield the following covariance matrix

$$\boldsymbol{\Sigma}_{\mathsf{o}}^{(m)} = \mathbf{H}^{(r_m)}\boldsymbol{\Sigma}_{\mathsf{s}}^{(m)}\mathbf{H}^{(r_m)\mathsf{T}} + \boldsymbol{\Psi}^{(r_m)}$$

This form of covariance matrix is identical to the form in equation 56. FA may also be extended to employ an underlying mixture of Gaussians HMM as latent variables. This is called factor analysed HMM (FAHMM) [41]. As NCMLLR and FA have the same generative model, the same EM framework for parameter estimation can be used for both [11, 12].

Though NCMLLR and FA-based schemes are related to one another, the fundamental starting point for NCMLLR is to use the shared "loading" matrices and noise variances as a transformation from the clean speech to corrupted speech distributions. Whereas for FACILT and FAHMMs they are used for covariance matrix modelling. Another important difference is that NCMLLR is constrained to have the same dimensionality space for both the observations, $\mathbf{o}_t$, and the clean speech, $\mathbf{s}_t$. Though a restriction this allows certain efficiencies in the likelihood calculation. In FA and related schemes the calculation of the log-likelihood is more expensive than the diagonal covariance matrix case, as the covariance matrix in equation 60 will be full[2]. In contrast for NCMLLR if the covariance matrix for the clean speech is diagonal, then by restricting $\boldsymbol{\Sigma}_{\mathsf{b}}^{(r_m)} = \mathbf{A}^{(r_m)}\boldsymbol{\Psi}^{(r_m)}\mathbf{A}^{(r_m)\mathsf{T}}$ to be diagonal the likelihood calculation in equation 57 will only have the cost of a diagonal covariance matrix calculation. Furthermore as NCMLLR is a transform, the clean speech model must be trained in an adaptive training fashion. Thus the summations for both the estimation of the transforms and the canonical will differ from the forms used for FACILT for example. The full derivations for MCMLLR are therefore given here, so that it is clear where the summation differences come from, and the subtleties of the differences between NCMLLR and related FA-based schemes is clear.

[2]Depending on P, the matrix inversion lemmas, also known as the Sherman-Morrisson-Woodbury formula, can be used to make this more efficient than the standard full-covariance matrix calculation.

5.2 Transform Estimation

It is required to determine the NCMLLR transform parameter $\mathcal{T}^{(r_m)} = \{\mathbf{H}^{(r_m)}, \mathbf{g}^{(r_m)}, \mathbf{\Psi}^{(r_m)}\}$ (based on equation 56), or equivalently $\mathcal{T}^{(r_m)} = \{\mathbf{A}^{(r_m)}, \mathbf{b}^{(r_m)}, \mathbf{\Sigma}_{\mathsf{b}}^{(r_m)}\}$ (based on equation 57), given the observation sequence $\mathbf{O}$, the acoustic model parameters $\mathcal{M}$ and a hypothesis $\mathcal{H}$. The NCMLLR transform parameters are found by maximising the likelihood of the noisy observation as follows:

$$\hat{\mathcal{T}} = \arg\max_{\mathcal{T}} \{\log(p(\mathbf{O}|\mathcal{M}, \mathcal{T}, \mathcal{H}))\} \tag{60}$$

Directly finding the NCMLLR parameters that optimises equation 60 is difficult because of the hidden state sequence and unobserved "clean" speech vectors. Hence an iterative EM approach, similar to those described in [21, 11, 12], is used. Since the clean speech vectors are also considered to be hidden, the complete data is $\mathbf{Z} = \{\mathbf{O}, \mathbf{S}, \mathbf{\Theta}\}$ where $\mathbf{S}$ and $\mathbf{\Theta}$ represent the clean speech sequence and possible hidden state sequences for $\mathbf{O}$ given the transcription, respectively. Let $\mathcal{T}$ be the current NCMLLR parameters and $\hat{\mathcal{T}}$ the NCMLLR parameters to be estimated, the auxiliary function for transform parameters can then expressed as

$$\mathcal{Q}(\hat{\mathcal{T}}; \mathcal{T}) = \mathbb{E}\left[\log p(\mathbf{Z}|\mathcal{M}, \hat{\mathcal{T}})|\mathbf{O}, \mathcal{M}, \mathcal{T}\right] \tag{61}$$

where $\mathbb{E}$ denotes the conditional expectation over all possible hidden sequences $(\mathbf{S}, \mathbf{\Theta})$, given the observation sequence $\mathbf{O}$ computed with parameter set $\mathcal{M}$ and transform set $\mathcal{T}$. Following Appendix A, the auxiliary function may be expressed as

$$\begin{aligned}\mathcal{Q}(\hat{\mathcal{T}}; \mathcal{T}) = -\frac{1}{2}\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_{\mathsf{o},t}^{(m)}\ \mathbb{E}\Big[&\log|\hat{\mathbf{\Psi}}^{(r_m)}| + \\ &(\mathbf{o}_t - \hat{\mathbf{H}}^{(r_m)}\mathbf{s}_t - \hat{\mathbf{g}}^{(r_m)})^{\mathsf{T}}\hat{\mathbf{\Psi}}^{(r_m)\text{-}1}(\mathbf{o}_t - \hat{\mathbf{H}}^{(r_m)}\mathbf{s}_t - \hat{\mathbf{g}}^{(r_m)})|\mathbf{o}_t, m, \mathcal{M}, \mathcal{T}^{(r_m)}\Big]\end{aligned} \tag{62}$$

where $\gamma_{\mathsf{o},t}^{(m)}$ is the posterior probability of component m given the observation sequence $\mathbf{O}$, NCMLLR parameter set $\mathcal{T}$, and model set $\mathcal{M}$. To simplify the estimation of the NCMLLR parameters, the auxiliary function can be rewritten as

$$\begin{aligned}\mathcal{Q}(\hat{\mathcal{T}}; \mathcal{T}) = -\frac{1}{2}\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_{\mathsf{o},t}^{(m)}\times \\ \mathbb{E}\left[\log|\hat{\mathbf{\Psi}}^{(r_m)}| + (\mathbf{o}_t - \hat{\mathbf{V}}^{(r_m)}\boldsymbol{\zeta}_t)^{\mathsf{T}}\hat{\mathbf{\Psi}}^{(r_m)\text{-}1}(\mathbf{o}_t - \hat{\mathbf{V}}^{(r_m)}\boldsymbol{\zeta}_t)|\mathbf{o}_t, m, \mathcal{M}, \mathcal{T}^{(r_m)}\right]\end{aligned} \tag{63}$$

where $\hat{\mathbf{V}}^{(r_m)} = [\hat{\mathbf{g}}^{(r_m)} \quad \hat{\mathbf{H}}^{(r_m)}]$ is the extended transformation matrix and $\boldsymbol{\zeta}_t = [1 \quad \mathbf{s}_t^{\mathsf{T}}]^{\mathsf{T}}$is the extended clean speech vector.

It is hard to directly optimise equation 63 for all the transform parameters. Instead an iterative approach is used where initially the variance bias $\hat{\mathbf{\Sigma}}_{\mathsf{b}}^{(r_m)}$ (or equivalently $\hat{\mathbf{\Psi}}^{(r_m)}$) is estimated, then the feature transformation $\{\mathbf{A}^{(r_m)}, \mathbf{b}^{(r_m)}\}$ (or equivalently $\{\mathbf{H}^{(r_m)}, \mathbf{g}^{(r_m)}\}$) is found. The process is then repeated using the latest values for each of the parameters. Thus the NCMLLR parameter estimation is itself an iterative process, interleaving estimates of $\hat{\mathbf{\Sigma}}_{\mathsf{b}}^{(r_m)}$, and $\hat{\mathbf{A}}^{(r_m)}$ and $\hat{\mathbf{b}}^{(r_m)}$.

5.2.1 Variance Bias Estimation

Differentiating equation 63 with respect to $\hat{\mathbf{\Psi}}^{(r)}$, and equating to zero yields

$$\hat{\mathbf{\Psi}}^{(r)} = \frac{\sum_{m\in r}\sum_{t=1}^{T}\gamma_{\mathsf{o},t}^{(m)}\mathbb{E}\left[(\mathbf{o}_t - \hat{\mathbf{V}}^{(r_m)}\boldsymbol{\zeta}_t)(\mathbf{o}_t - \hat{\mathbf{V}}^{(r_m)}\boldsymbol{\zeta}_t)^{\mathsf{T}}|\mathbf{o}_t, m, \mathcal{M}, \mathcal{T}^{(r_m)}\right]}{\sum_{m\in r}\sum_{t=1}^{T}\gamma_{\mathsf{o},t}^{(m)}} \tag{64}$$

From equation 64 and using the equality $\boldsymbol{\Sigma}_{\mathsf{b}}^{(r)} = \mathbf{A}^{(r)}\boldsymbol{\Psi}^{(r)}\mathbf{A}^{(r)\mathsf{T}}$, the covariance bias can be estimated as

$$\hat{\boldsymbol{\Sigma}}_{\mathsf{b}}^{(r)} = \frac{\sum_{m\in r}\sum_{t=1}^{T}\gamma_{\mathsf{o},t}^{(m)}\mathbb{E}\left[(\hat{\mathbf{s}}_t^{(r_m)} - \mathbf{s}_t)(\hat{\mathbf{s}}_t^{(r_m)} - \mathbf{s}_t)^{\mathsf{T}}|\mathbf{o}_t, m, \mathcal{M}, \mathcal{T}^{(r_m)}\right]}{\sum_{m\in r}\sum_{t=1}^{T}\gamma_{\mathsf{o},t}^{(m)}} \tag{65}$$

where

$$\hat{\mathbf{s}}_t^{(r_m)} = \hat{\mathbf{A}}^{(r_m)}\mathbf{o}_t + \hat{\mathbf{b}}^{(r_m)} \tag{66}$$

In the presentation above, $\boldsymbol{\Sigma}_{\mathsf{b}}^{(r_m)}$ is specified as being full in equation 65. In this case the likelihood calculation in equation 57 has the same cost as using a full-covariance matrix. This is the same problem encountered when using full transforms with JUD. However in contrast to JUD it is possible to use full feature transforms $\mathbf{A}^{(r_m)}$ with diagonal $\boldsymbol{\Sigma}_{\mathsf{b}}^{(r_m)}$. Simply diagonalising equation 65 still yields an ML-solution for the variance bias but now for a diagonal form. Thus for all the experiments reported in the next section, the variance bias term is estimated using

$$\hat{\boldsymbol{\Sigma}}_{\mathsf{b}}^{(r)} = \text{diag}\left(\frac{\sum_{m\in r}\sum_{t=1}^{T}\gamma_{\mathsf{o},t}^{(m)}\mathbb{E}\left[(\hat{\mathbf{s}}_t^{(r_m)} - \mathbf{s}_t)(\hat{\mathbf{s}}_t^{(r_m)} - \mathbf{s}_t)^{\mathsf{T}}|\mathbf{o}_t, m, \mathcal{M}, \mathcal{T}^{(r_m)}\right]}{\sum_{m\in r}\sum_{t=1}^{T}\gamma_{\mathsf{o},t}^{(m)}}\right) \tag{67}$$

This is an important flexibility in this form of model compared to JUD and highlights one of the major differences to previous work on FA-based models. Note it is possible to map full JUD transforms to have diagonal variance biases using the predictive linear transform approach described in [42] and the NCMLLR estimation formulae described in this section.

5.2.2 Feature Transformation

Differentiating equation 63 with respect to $\hat{\mathbf{V}}^{(r)}$, and equating to zero, gives

$$\hat{\mathbf{V}}^{(r)} = \left(\sum_{m\in r}\sum_{t=1}^{T}\gamma_{\mathsf{o},t}^{(m)}\ \mathbf{o}_t\ \mathbb{E}\left[\boldsymbol{\zeta}_t^{\mathsf{T}}|\mathbf{o}_t, m, \mathcal{M}, \mathcal{T}^{(r_m)}\right]\right) \times \left(\sum_{m\in r}\sum_{t=1}^{T}\gamma_{\mathsf{o},t}^{(m)}\mathbb{E}\left[\boldsymbol{\zeta}_t\boldsymbol{\zeta}_t^{\mathsf{T}}|\mathbf{o}_t, m, \mathcal{M}, \mathcal{T}^{(r_m)}\right]\right)^{-1} \tag{68}$$

Having found $\hat{\mathbf{V}}^{(r)} = [\hat{\mathbf{g}}^{(r)}\ \ \hat{\mathbf{H}}^{(r)}]$, the estimation of the feature-transforms are obtained using

$$\hat{\mathbf{A}}^{(r)} = \hat{\mathbf{H}}^{(r)\text{-}1} \tag{69}$$

$$\hat{\mathbf{b}}^{(r)} = -\hat{\mathbf{H}}^{(r)\text{-}1}\hat{\mathbf{g}}^{(r)} \tag{70}$$

The update formulae, equations 65 and 68, can be expressed in terms of the conditional expectation of the extended clean speech vector and its outer product as follows (see Appendix B):

$$\mathbb{E}\left[\boldsymbol{\zeta}_t|\mathbf{o}_t, m, \mathcal{M}, \mathcal{T}^{(r_m)}\right] = \left[1\ \ \mathbb{E}[\mathbf{s}_t^{\mathsf{T}}|\mathbf{o}_t, m, \mathcal{M}, \mathcal{T}^{(r_m)}]\right]^{\mathsf{T}} = [1\ \ \tilde{\mathbf{s}}_t^{(m)\mathsf{T}}]^{\mathsf{T}} \tag{71}$$

$$\mathbb{E}\left[\boldsymbol{\zeta}_t\boldsymbol{\zeta}_t^{\mathsf{T}}|\mathbf{o}_t, m, \mathcal{M}, \mathcal{T}^{(r_m)}\right] = \begin{bmatrix} 1 & \tilde{\mathbf{s}}_t^{(m)\mathsf{T}} \\ \tilde{\mathbf{s}}_t^{(m)} & \tilde{\boldsymbol{\Sigma}}_{\mathsf{s}}^{(m)} + \tilde{\mathbf{s}}_t^{(m)}\tilde{\mathbf{s}}_t^{(m)\mathsf{T}} \end{bmatrix} \tag{72}$$

where

$$\tilde{\mathbf{s}}_t^{(m)} = \tilde{\mathbf{A}}^{(m)}\mathbf{o}_t + \tilde{\mathbf{b}}^{(m)} \tag{73}$$

$$\tilde{\boldsymbol{\Sigma}}_{\mathsf{s}}^{(m)} = \left(\boldsymbol{\Sigma}_{\mathsf{s}}^{(m)\text{-}1} + \boldsymbol{\Sigma}_{\mathsf{b}}^{(r_m)\text{-}1}\right)^{-1} \tag{74}$$

and

$$\tilde{\mathbf{A}}^{(m)} = \left(\boldsymbol{\Sigma}_{\mathtt{s}}^{(m)\text{-}1} + \boldsymbol{\Sigma}_{\mathtt{b}}^{(r_m)\text{-}1}\right)^{\text{-}1} \boldsymbol{\Sigma}_{\mathtt{b}}^{(r_m)\text{-}1} \mathbf{A}^{(r_m)} \tag{75}$$

$$\tilde{\mathbf{b}}^{(m)} = \left(\boldsymbol{\Sigma}_{\mathtt{s}}^{(m)\text{-}1} + \boldsymbol{\Sigma}_{\mathtt{b}}^{(r_m)\text{-}1}\right)^{\text{-}1} \left(\boldsymbol{\Sigma}_{\mathtt{s}}^{(m)\text{-}1} \boldsymbol{\mu}_{\mathtt{s}}^{(m)} + \boldsymbol{\Sigma}_{\mathtt{b}}^{(r_m)\text{-}1} \mathbf{b}^{(r_m)}\right) \tag{76}$$

5.3 Adaptive Training

Adaptive training with NCMLLR follows the general adaptive training framework. The canonical acoustic model parameters $\mathcal{M}$ and set of NCMLLR parameters $\mathcal{T}$ are estimated such that they maximise the log-likelihood of the heterogenous training data comprised of H homogeneous block $\mathcal{O} = \{\mathbf{O}^{(1)}, \ldots, \mathbf{O}^{(h)}\}$. The log-likelihood may be written as

$$\log(p(\mathcal{O}|\mathcal{M}, \mathcal{T}, \mathcal{H}_{\mathtt{ref}})) = \sum_{h=1}^{H} \log(p(\mathbf{O}^{(h)}|\mathcal{M}, \mathcal{T}^{(h)}, \mathcal{H}_{\mathtt{ref}}^{(h)})) \tag{77}$$

where h indexes a homogeneous block of training data $\mathbf{O}^{(h)}$ and H is the total number of blocks. Thus H sets of NCMLLR parameters are required and the entire set of parameters denoted by $\mathcal{T} = \{\mathcal{T}^{(1)}, \ldots, \mathcal{T}^{(H)}\}$. $\mathcal{H}_{\mathtt{ref}} = \{\mathcal{H}_{\mathtt{ref}}^{(1)}, \ldots, \mathcal{H}_{\mathtt{ref}}^{(H)}\}$ is the corresponding transcription sequence. The set of transforms and the canonical model parameters are iteratively estimated to yield the final canonical model.

The complete data for a homogeneous block h are $\mathbf{Z}^{(h)} = \{\mathbf{O}^{(h)}, \mathbf{S}^{(h)}, \boldsymbol{\Theta}^{(h)}\}$ where $\mathbf{S}^{(h)}$ and $\boldsymbol{\Theta}^{(h)}$ represent the clean speech data sequence and possible hidden state sequences for $\mathbf{O}^{(h)}$ given the transcription, respectively. Assume that $\mathcal{M}$ and $\mathcal{T}$ are the current model and NCMLLR parameters and $\hat{\mathcal{M}}$ and $\hat{\mathcal{T}}$ the estimated model and NCMLLR parameters, respectively. The auxiliary function can then be expressed as, letting $\mathbf{Z} = \{\mathbf{Z}^{(1)}, \ldots, \mathbf{Z}^{(H)}\}$,

$$\mathcal{Q}(\hat{\mathcal{M}}, \hat{\mathcal{T}}; \mathcal{M}, \mathcal{T}) = \mathbb{E}\left[\log p(\mathbf{Z}|\hat{\mathcal{M}}, \hat{\mathcal{T}})|\mathcal{O}, \mathcal{M}, \mathcal{T}\right] \tag{78}$$

where $\mathbb{E}$ denotes the conditional expectation over all possible hidden sequences $(\mathbf{S}, \boldsymbol{\Theta})$, given the observation sequence $\mathcal{O}$ computed with parameter set $\mathcal{M}$ and transform set $\mathcal{T}$. This auxiliary function may be expressed as

$$\begin{aligned}\mathcal{Q}(\hat{\mathcal{M}}; \mathcal{M}, \mathcal{T}) = &-\frac{1}{2}\sum_{h=1}^{H}\sum_{t=1}^{T^{(h)}}\sum_{m=1}^{M}\gamma_{\mathtt{o},t}^{(mh)} \times \\ &\mathbb{E}\Big[\log|\hat{\boldsymbol{\Psi}}^{(r_m h)}| + (\mathbf{o}_t - \hat{\mathbf{H}}^{(r_m h)}\mathbf{s}_t - \hat{\mathbf{g}}^{(r_m h)})^{\mathsf{T}}\hat{\boldsymbol{\Psi}}^{(r_m h)\text{-}1}(\mathbf{o}_t - \hat{\mathbf{H}}^{(r_m h)}\mathbf{s}_t - \hat{\mathbf{g}}^{(r_m h)}) \\ &+ \log|\hat{\boldsymbol{\Sigma}}^{(m)}| + (\mathbf{s}_t - \hat{\boldsymbol{\mu}}_{\mathtt{s}}^{(m)})^{\mathsf{T}}\hat{\boldsymbol{\Sigma}}_{\mathtt{s}}^{(m)\text{-}1}(\mathbf{s}_t - \hat{\boldsymbol{\mu}}_{\mathtt{s}}^{(m)})|\mathbf{o}_t, m, \mathcal{M}, \mathcal{T}^{(r_m h)}\Big]\end{aligned} \tag{79}$$

where $\gamma_{\mathtt{o},t}^{(mh)}$ is the posterior probability of component m given the observation sequence $\mathbf{O}^{(h)}$, NCMLLR parameter set $\mathcal{T}^{(r_m h)}$, and model set $\mathcal{M}$ for homogenous block h. This auxiliary function will be iteratively optimised by first updating the NCMLLR transform parameters then the canonical model parameters. First, given the current acoustic models $\mathcal{M}$ a new set of transform $\hat{\mathcal{T}}$ is estimated, Subsequently, the canonical model parameters are updated to $\hat{\mathcal{M}}$ given this new set of transforms. Multiple iterations of this interleaved training may be performed to optimise the auxiliary function. The NCMLLR parameter estimation is described in section 5.2. It is worth emphasising that the transform estimation is based on each homogenous data block rather than the whole training dataset.

After the NCMLLR transform parameters have been estimated, the canonical model parameters must be retrained. The auxiliary function can be rewritten, ignoring terms independent of

the canonical model parameters, as

$$\mathcal{Q}(\hat{\mathcal{M}};\mathcal{M},\hat{\mathcal{T}}) = -\frac{1}{2}\sum_{h=1}^{H}\sum_{t=1}^{T^{(h)}}\sum_{m=1}^{M}\gamma_{\mathtt{o},t}^{(mh)}\times \\ \mathbb{E}\left[\log|\hat{\boldsymbol{\Sigma}}_{\mathtt{s}}^{(m)}| + (\mathbf{s}_t-\hat{\boldsymbol{\mu}}_{\mathtt{s}}^{(m)})^{\mathsf{T}}\hat{\boldsymbol{\Sigma}}_{\mathtt{s}}^{(m)\text{-}1}(\mathbf{s}_t-\hat{\boldsymbol{\mu}}_{\mathtt{s}}^{(m)})|\mathbf{o}_t,m,\mathcal{M},\hat{\mathcal{T}}^{(r_m h)}\right] \tag{80}$$

Optimising the auxiliary function, equation 80, with respect to $\hat{\boldsymbol{\mu}}_{\mathtt{s}}^{(m)}$ and $\hat{\boldsymbol{\Sigma}}_{\mathtt{s}}^{(m)}$ leads to the update formulae as follows:

$$\hat{\boldsymbol{\mu}}_{\mathtt{s}}^{(m)} = \frac{\sum_{h=1}^{H}\sum_{t=1}^{T^{(h)}}\gamma_{\mathtt{o},t}^{(mh)}\mathbb{E}\left[\mathbf{s}_t|\mathbf{o}_t,m,\mathcal{M},\hat{\mathcal{T}}^{(r_m h)}\right]}{\sum_{h=1}^{H}\sum_{t=1}^{T}\gamma_{\mathtt{o},t}^{(mh)}} \tag{81}$$

$$\hat{\boldsymbol{\Sigma}}_{\mathtt{s}}^{(m)} = \mathrm{diag}\left(\frac{\sum_{h=1}^{H}\sum_{t=1}^{T^{(h)}}\gamma_{\mathtt{o},t}^{(mh)}\mathbb{E}\left[\mathbf{s}_t\mathbf{s}_t^{\mathsf{T}}|\mathbf{o}_t,m,\mathcal{M},\hat{\mathcal{T}}^{(r_m h)}\right]}{\sum_{h=1}^{H}\sum_{t=1}^{T}\gamma_{\mathtt{o},t}^{(m)}} - \hat{\boldsymbol{\mu}}_{\mathtt{s}}^{(m)}\hat{\boldsymbol{\mu}}_{\mathtt{s}}^{(m)\mathsf{T}}\right) \tag{82}$$

These equations are similar to those given in [8], however the conditional expectations differ. The conditional expectations in equations 81 and 82 are given by

$$\mathbb{E}\left[\mathbf{s}_t|\mathbf{o}_t,m,\mathcal{M},\hat{\mathcal{T}}^{(r_m h)}\right] = \tilde{\mathbf{s}}_t^{(mh)} \tag{83}$$

$$\mathbb{E}\left[\mathbf{s}_t\mathbf{s}_t^{\mathsf{T}}|\mathbf{o}_t,m,\mathcal{M},\hat{\mathcal{T}}^{(r_m h)}\right] = \tilde{\boldsymbol{\Sigma}}_{\mathtt{s}}^{(mh)} + \tilde{\mathbf{s}}_t^{(mh)}\tilde{\mathbf{s}}_t^{(mh)\mathsf{T}} \tag{84}$$

which have been defined for homogeneous block h in equations 73 - 76. Note that here the updated transform parameters, $\hat{\mathcal{T}}^{(r_m h)}$ are used.

5.4 Discriminative Adaptive Training

The adaptive training with NCMLLR can be extended to support discriminative training of the acoustic models based on disriminative criteria. DAT with NCMLLR follows the simplified DAT strategy in which only the canonical model is discriminatively updated given a set of ML estimated NCMLLR transforms. Assume that $\hat{\mathcal{M}}$ and $\mathcal{M}$ are the model to be estimated and the current model, respectively. To estimate the canonical model parameters, the weak-sense auxiliary function can be written as [39]

$$\mathcal{Q}_{\mathtt{MPE}}(\hat{\mathcal{M}},\mathcal{M}) = \mathcal{Q}_{\mathtt{n}}(\hat{\mathcal{M}};\mathcal{M}) - \mathcal{Q}_{\mathtt{d}}(\hat{\mathcal{M}};\mathcal{M}) + \mathcal{S}(\hat{\mathcal{M}};\mathcal{M}) + \log p(\hat{\mathcal{M}}|\boldsymbol{\Phi}) \tag{85}$$

where the numerator and denominator parts, $\mathcal{Q}_{\mathtt{n}}(\hat{\mathcal{M}};\mathcal{M})$ and $\mathcal{Q}_{\mathtt{d}}(\hat{\mathcal{M}};\mathcal{M})$ have the same form as the standard ML auxiliary function for adaptive training with NCMLLR in equation 80. It is also necessary to specify appropriate forms for $\mathcal{S}(\hat{\mathcal{M}};\mathcal{M})$ and $\log p(\hat{\mathcal{M}}|\boldsymbol{\Phi})$.

The remaining terms that need to be specified for the auxiliary function in equation 85 are the smoothing function, $\mathcal{S}(\hat{\mathcal{M}};\mathcal{M})$, and the prior term $\log p(\hat{\mathcal{M}}|\boldsymbol{\Phi})$. An appropriate form of smoothing function for NCMLLR has the same form as the standard discriminative training schemes [43, 39]

$$\mathcal{S}(\hat{\mathcal{M}};\mathcal{M}) = -\frac{1}{2}\sum_{h,m}D_m\Big\{\log|\hat{\boldsymbol{\Sigma}}_{\mathtt{s}}^{(m)}| \\ + \mathtt{Tr}\left[\boldsymbol{\Sigma}_{\mathtt{s}}^{(m)}\hat{\boldsymbol{\Sigma}}_{\mathtt{s}}^{(m)\text{-}1}\right] + (\boldsymbol{\mu}_{\mathtt{s}}^{(m)}-\hat{\boldsymbol{\mu}}_{\mathtt{s}}^{(m)})^{\mathsf{T}}\hat{\boldsymbol{\Sigma}}_{\mathtt{s}}^{(m)\text{-}1}(\boldsymbol{\mu}_{\mathtt{s}}^{(m)}-\hat{\boldsymbol{\mu}}_{\mathtt{s}}^{(m)})^{\text{-}1}\Big\} \tag{86}$$

where $\boldsymbol{\mu}_{\mathtt{s}}^{(m)}$ and $\boldsymbol{\Sigma}_{\mathtt{s}}^{(m)}$ are the mean vector and covariance matrix of Gaussian component m from the current model set $\mathcal{M}$. The constant D_m is a critical value in MPE training as it needs to

make the weak-sense auxiliary function convex and yield a rapid and stable update. This is set on a per-component basis as described in [39]

$$D_m = \max\left\{ E\sum_{h=1}^{H}\sum_{t=1}^{T}\gamma_{\mathtt{do},t}^{(mh)}, 2D_{\mathtt{min}}) \right\}$$

where $D_{\mathtt{min}}$ is the minimum value to ensure that the covariance matrix of component m is semi-positive definite. E is an empirically set constant and $\gamma_{\mathtt{do},t}^{(mh)}$ are the "occupancies" computed over the model corresponding to all alternative word sequence.

The prior distribution for I-smoothing, $p(\hat{\mathcal{M}}|\boldsymbol{\Phi})$, again has a similar form as the standard auxiliary function. One commonly used distribution for $p(\hat{\mathcal{M}}|\boldsymbol{\Phi})$ is the Normal-Wishart distribution which was also used for maximum a posteriori (MAP) estimation in [44]. The logarithm of the distribution, ignoring the constants independent of the parameters, is expressed as

$$\begin{aligned}\log p(\hat{\mathcal{M}}|\boldsymbol{\Phi}) = -\frac{\tau_{\mathtt{p}}}{2}\sum_{m}\Big\{ & \log|\hat{\boldsymbol{\Sigma}}_{\mathtt{s}}^{(m)}| \\ & + \mathtt{Tr}\left[\boldsymbol{\Sigma}_{\mathtt{p}}^{(m)}\hat{\boldsymbol{\Sigma}}_{\mathtt{s}}^{(m)\text{-}1}\right] + (\hat{\boldsymbol{\mu}}_{\mathtt{s}}^{(m)} - \boldsymbol{\mu}_{\mathtt{p}}^{(m)})^{\mathsf{T}}\boldsymbol{\Sigma}_{\mathtt{s}}^{(m)\text{-}1}(\hat{\boldsymbol{\mu}}_{\mathtt{s}}^{(m)} - \boldsymbol{\mu}_{\mathtt{p}}^{(m)})\Big\}\end{aligned} \tag{87}$$

where $\boldsymbol{\Phi} = \{\tau_{\mathtt{p}}, \boldsymbol{\mu}_{\mathtt{p}}^{(m)}, \boldsymbol{\Sigma}_{\mathtt{p}}^{(m)}\}$ is the set of hyper-parameters of the I-smoothing distribution. $\tau_{\mathtt{p}}$ is the specified parameter which controls the impact of the prior and $\boldsymbol{\mu}_{\mathtt{p}}^{(m)}$ and $\boldsymbol{\Sigma}_{\mathtt{p}}^{(m)}$ are the prior hyper-parameters of the distribution for component m.

By substituting the above expressions and differentiating the weak-sense auxiliary function with respect to the model parameters and setting it to zero, the canonical model estimate for NCMLLR can be derived

$$\hat{\boldsymbol{\mu}}_{\mathtt{s}}^{(m)} = \frac{\sum_{h,t}(\gamma_{\mathtt{no},t}^{(mh)} - \gamma_{\mathtt{do},t}^{(mh)})\mathbb{E}\left[\mathbf{s}_t|\mathbf{o}_t, m\right] + D_m\boldsymbol{\mu}_{\mathtt{s}}^{(m)} + \tau_{\mathtt{p}}\boldsymbol{\mu}_{\mathtt{p}}^{(m)}}{\sum_{h,t}(\gamma_{\mathtt{no},t}^{(mh)} - \gamma_{\mathtt{do},t}^{(mh)}) + D_m + \tau_{\mathtt{p}}} \tag{88}$$

$$\hat{\boldsymbol{\Sigma}}_{\mathtt{s}}^{(m)} = \mathrm{diag}\left(\frac{\sum_{h,t}(\gamma_{\mathtt{no},t}^{(mh)} - \gamma_{\mathtt{do},t}^{(mh)})\mathbb{E}\left[\mathbf{s}_t\mathbf{s}_t^{\mathsf{T}}|\mathbf{o}_t, m\right] + D_m\mathbf{L}_{\mathtt{s}}^{(m)} + \tau_{\mathtt{p}}\mathbf{L}_{\mathtt{p}}^{(m)}}{\sum_{h,t}(\gamma_{\mathtt{no},t}^{(mh)} - \gamma_{\mathtt{do},t}^{(mh)}) + D_m + \tau_{\mathtt{p}}} - \hat{\boldsymbol{\mu}}_{\mathtt{s}}^{(m)}\hat{\boldsymbol{\mu}}_{\mathtt{s}}^{(m)\mathsf{T}}\right) \tag{89}$$

where

$$\mathbf{L}_{\mathtt{s}}^{(m)} = \boldsymbol{\Sigma}_{\mathtt{s}}^{(m)} + \boldsymbol{\mu}_{\mathtt{s}}^{(m)}\boldsymbol{\mu}_{\mathtt{s}}^{(m)\mathsf{T}} \tag{90}$$

$$\mathbf{L}_{\mathtt{p}}^{(m)} = \boldsymbol{\Sigma}_{\mathtt{p}}^{(m)} + \boldsymbol{\mu}_{\mathtt{p}}^{(m)}\boldsymbol{\mu}_{\mathtt{p}}^{(m)\mathsf{T}} \tag{91}$$

$\gamma_{\mathtt{no},t}^{(mh)}$ represents that occupancies are calculated by the forward-backward algorithm with the composite HMM model corresponding to the correct transcriptions on the target environment adaptation data, while $\gamma_{\mathtt{do},t}^{(mh)}$ means that occupancies are computed over the model corresponding to all alternative word sequence. $\mathbb{E}\left[\mathbf{s}_t|\mathbf{o}_t, m\right]$ and $\mathbb{E}\left[\mathbf{s}_t\mathbf{s}_t^{\mathsf{T}}|\mathbf{o}_t, m\right]$ are defined in equations 83 - 84[3].

6 Implementation Issues

There are a number of issues that must be considered when using NCMLLR either as a transform or in adaptive training. When using full transformation matrix, $\mathbf{A}^{(r_m)}$, it is necessary to store full outer-product observation statistics for each component. Equation 68 requires the term $\mathbf{o}_t\mathbb{E}\Big[\boldsymbol{\zeta}_t^{\mathsf{T}}|\mathbf{o}_t, m\Big]$ to be stored. From equation 73 this needs functions of $\mathbf{o}_t\mathbf{o}_t^{\mathsf{T}}$ to be accumulated for each component. For LVCSR systems this can be highly memory intensive. Terms like this

[3]The dependence on the current model parameters $\mathcal{M}$ and the transform parameter $\mathcal{T}$ in expectation are assumed for simplicity.

are not required for CMLLR estimation as observation outer-products can be accumulated at the base-class level. As all components tend not to be observed when estimating a particular transform, this can still be practical even for large vocabulary systems by only generating accumulation space on demand. For NCMLLR, this is not an issue when using diagonal transforms. Note unlike CMLLR estimation, the transform update formulae are applicable with both diagonal and full covariance canonical models, though only diagonal covariance matrix systems are used.

Scheme	Type	Parameters	# Free Parameters
JUD/VTS	Predictive	$\boldsymbol{\mu}_{\mathtt{n}}$, $\boldsymbol{\Sigma}_{\mathtt{n}}$, $\boldsymbol{\mu}_{\mathtt{h}}$	$5D_{\mathtt{s}}$
CMLLR	Adaptive	$\mathbf{A}$, $\mathbf{b}$	$2RD$
NCMLLR	(diagonal)	$\mathbf{A}$, $\mathbf{b}$, $\boldsymbol{\Sigma}_{\mathtt{b}}$	$3RD$
CMLLR	Adaptive	$\mathbf{A}$, $\mathbf{b}$	$R(D^2+D)$
NCMLLR	(full)	$\mathbf{A}$, $\mathbf{b}$, $\boldsymbol{\Sigma}_{\mathtt{b}}$	$R(D^2+2D)$

Table 1: Number of free parameters to estimate for diagonal forms of various model-based compensation schemes.D is the dimensionality of the full feature vector, D_s the number of static parameters (in the experiments $D = 3D_s$).

Table 1 compares the number of free parameters that need to be estimated for diagonal forms of various model-based compensation. The predictive schemes such as VTS and JUD only require the noise model parameters to be estimated. Hence they have a lower number of free parameters and require less adaptation data than the adaptive forms. The number of free parameters for the adaptive schemes depends on the number of regression classes R. This means that the number of parameter can be adjusted according to the amount of adaptation data. NCMLLR requires more parameters to be estimated than CMLLR due to the variance bias.

Another problem, also observed with JUD compensation [9], is that the magnitude of the transform matrix, and hence the variance bias, can become very large in low SNR regions. This is because the corrupted speech distribution is dominated by the noise in the region of low speech energy, the cross-covariance $\boldsymbol{\Sigma}_{\mathtt{so}}$ will be approximately zero. That is, the clean speech and the corrupted speech will be uncorrelated since the clean speech and noise are independent in low SNR regions. In these cases the cross-covariance term $\boldsymbol{\Sigma}_{\mathtt{so}}^{(m)}$ for component m can be expressed as

$$\boldsymbol{\Sigma}_{\mathtt{so}}^{(m)} = \mathbb{E}\left[(\mathbf{s}_t - \boldsymbol{\mu}_{\mathtt{s}}^{(m)})(\mathbf{o}_t - \boldsymbol{\mu}_{\mathtt{o}}^{(m)})^{\mathsf{T}}\right] = \boldsymbol{\Sigma}_{\mathtt{s}}^{(m)}\mathbf{H}^{\mathsf{T}} \approx \mathbf{0} \tag{92}$$

In the NCMLLR scheme, the transform matrix $\mathbf{H}$ will tend to zero in low SNR, hence $\mathbf{A} = \mathbf{H}^{-1}$ goes to infinity along with the variance bias $\boldsymbol{\Sigma}_{\mathtt{b}}$. To prevent this problem it is sensible to limit the possible values for the compensation parameters. The compensation parameters can then be restricted by enforcing a maximum on the variance bias for dimension i used in equations 73 - 76 as follows

$$\sigma_{\mathtt{b}i}^{(m)2} \leq \rho\sigma_{\mathtt{s}i}^{(m)2} \tag{93}$$

where ρ is an empirically determined constant. Performance was found to be relatively insensitive to ρ over a range of values for each of the tasks examined. The value of ρ was approximately tuned and fixed for all experiments.

7 Experiments and Results

The use of NCMLLR transforms in ML adaptation and adaptive training was evaluated on two tasks, noise corrupted Resource Management (RM) and in-car data collected by Toshiba Research Europe Ltd (TREL)[4]. Additionally recogniotion results for MPE-trained systems are given on

[4]The authors thank TREL for making this data available.

Toshiba data, as there is insufficient data in the RM database for discriminative training. Experiments were conducted using HTK V3.4 [36] with additional routines to support adaptation and adaptive training with NCMLLR.

There are a large number of possible noise robustness schemes, both enhancement-based and model-based, that NCMLLR could be compared to. In this work CMLLR for adaptation, adaptive training and discriminative adaptive training was selected. Though there are known limitations of linear transform-based schemes for noise-robustness, it allows a contrast with a standard scheme, which forms part of the motivation for the NCMLLR transform. Furthermore in these experiments adaptation was run at the speaker level. Thus there is no issue with making robust transform parameter estimates. In situations like adaptive approaches will tend to outperform predictive approaches.

7.1 Resource Management

Initial experiments were conducted on a medium vocabulary speech recognition task, the 991 word RM database. A 39 dimensional feature vector was used consisting of MFCCs, including the 0th cepstra, and associated 1st- and 2nd-order delta coefficients. Cross-word, state-clustered triphone acoustic models with 6 components per state, giving 9492 recognition components, were built using the RM recipe distributed with HTK. The RM word pair grammar was used. Operations Room noise from the NOISEX-92 was artificially added at the waveform level database to give 20 and 14 dB SNR test-sets. All results are averaged across the FEB89, OCT89 and FEB91 test sets. The ML multi-style model was built from data with Operations Room noise added at the speaker level at SNRs of 8, 14, 20, 26 or 32 dB. Thus the mismatch investigated here is primarily the range of SNR levels in training and test. The same data was used for the adaptively trained systems.

For all adaptation experiments the unadapted recognition output from the multi-style trained system (the first line in tables 2 and 3) was used for the hypotheses. 16 regression classes were used for diagonal linear transforms, while a single regression class for the full transform[5]. The variance bias $\boldsymbol{\Sigma}_{\mathbf{b}}$ was restricted to be diagonal. Though this configuration the NCMLLR transform had more parameters than the CMLLR transform, see table 1, using additional base-classes yielded minimal gains for CMLLR (this was also true on the Toshiba task). For NCMLLR transform estimation, EM was intialised with: $\mathbf{A} = \mathbf{I}$; $\mathbf{b} = \mathbf{0}$; and large diagonal biases were used as the initial values.

System	Adapt	WER (%)	
		20dB	14dB
Multi-style	—	6.96	15.44
	CMLLR	5.76	13.01
	NCMLLR	6.37	12.10
Adaptive Training	CMLLR	5.27	11.98
	NCMLLR	4.97	9.87

Table 2: Performance of ML multi-style and adaptive training with 16-diagonal transforms CMLLR and NCMLLR on the RM data.

Table 2 shows the performance of NCMLLR adaptation compared to the equivalent CMLLR set-up for multi-style trained systems using 16 base-classes and diagonal feature transformations. As expected both forms of adaptation improved the performance of the multi-style system. Interestingly, CMLLR out-performed NCMLLR at the higher SNR condition (20dB) whereas NCMLLR yielded better performance at lower SNR (14dB). This is expected as the variance bias will become more important as the SNR decreases. Also the bias term added to the variance, by definition, will be positive definite. With multi-style trained systems at low test condition SNRs this constraint

[5]Using multiple full-transforms did not yield performance gains.

means that this bias is likely to be small, yielding little possible gains over CMLLR. Note for all conditions NCMLLR yielded higher likelihoods than CMLLR. Adaptive training using both NCMLLR and CMLLR showed gains over the multi-style trained systems. Interestingly, adaptive training with NCMLLR out-performed CMLLR for both noise conditions. Overall the best performance for the low SNR condition, was using NCMLLR with NCMLLR adaptive training.

System	Adapt	WER (%)	
		20dB	14dB
Multi-style	—	6.96	15.44
	CMLLR	5.64	13.35
	NCMLLR	6.02	11.57
Adaptive Training	CMLLR	4.33	10.52
	NCMLLR	4.52	9.05

Table 3: Performance of ML multi-style and adaptive training with 1-full transform CMLLR and NCMLLR compensation on RM data.

It is also possible to use full feature transforms, though still diagonal variance biases, for NCMLLR. Table 3 shows the performance of a single full transform of NCMLLR compared to CMLLR. The same general trends as the diagonal system are observed. For the multi-style trained system, at the higher SNR condition (20dB) CMLLR slightly outperforms NCMLLR and for the lower SNR condition (14dB) the opposite is true. In contrast to the diagonal configuration in table 2, adaptive training with NCMLLR did not outperform CMLLR for the higher SNR condition, though the difference in performance is smaller than when using the multi-style trained system. For the lower SNR condition, again the best performing set-up was NCMLLR with adaptive training. This highlights the advantage of using NCMLLR when the SNR conditions are low.

7.2 Toshiba Task

The Toshiba task is a small/medium sized task with noisy speech collected in cars driving at various conditions. The WSJ SI284 training data were used to train a ML multi-style system model. Noise-corrupted multi-condition data were generated by adding car noise at the speaker level at average SNRs of 15, 18, 25 and 35 dB. The car noise added to the training data was not the same as that observed in the test data. For more information about the data and training configurations see [35]. Similar features and model topology to the RM system were used, except normalised log energy instead of 0th cepstra (this would complicate the specification of a mismatch function for predictive schemes). Again cross-word decision-tree state-clustered triphones were used, with about 6400 distinct states were generated with 16 Gaussian components per state.

In addition to ML systems, MPE systems were build: an MPE multi-style system trained; and a MPE based DAT systems. Both CMLLR and NCMLLR MPE adaptively trained systems were built. For these systems, the ML adaptive training systems were trained first and the HMM canonical model parameters were discriminatively estimated given ML estimated transforms for CMLLR and NCMLLR respectively. As this is a larger task, only diagonal versions of NCMLLR and CMLLR adaptation and adaptive training were implemented. The initial hypotheses for adaptation was generated using the ML multi-style system for the ML experiments and MPE multi-style system for the MPE ones. Again 16 base-classes for both CMLLR and NCMLLR were used.

For this work in-car collected test sets consisting of phone numbers were used. The performance was evaluated on three different conditions: engine on (ENON) , city driving (CITY) and highway driving (HWY). The average SNRs for each of the tests are 35dB, 25dB and 18 dB, respectively. Note, though only digits were used in the evaluation of the system, by using a general state-clustered triphone system allows the same acoustic model-set to be used for a range of tasks, for

example the city-names task described in [35].

System	Adapt	WER (%)		
		ENON	CITY	HWY
Multi-style	—	1.11	3.57	6.67
	CMLLR	0.31	1.08	2.36
	NCMLLR	0.51	1.16	2.09
Adaptive Training	CMLLR	0.30	1.00	2.05
	NCMLLR	0.28	0.95	1.76

Table 4: Performance of ML multi-style and adaptive training with 16-diagonal transforms CMLLR and NCMLLR compensation on Toshiba in-car phone-number task.

Table 4 shows the results on this task for ML multi-style and adaptive training models with 16-diagonal transforms. Similar results to those obtained on the RM database. For ML multi-style with the highway conditions, NCMLLR was better than CMLLR. However at the the higher SNR conditions, ENON and CITY conditions, CMLLR was better. ML adaptive training with NCMLLR outperformed CMLLR for all conditions. These results again illustrate the effectiveness of NCMLLR especially for data with lower average SNRs.

System	Adapt	WER (%)		
		ENON	CITY	HWY
Multi-style	—	0.79	2.66	4.93
	CMLLR	0.26	1.10	1.93
	NCMLLR	0.59	1.23	1.93
Adaptive Training	CMLLR	0.21	0.94	1.48
	NCMLLR	0.24	0.86	1.20

Table 5: Performance of MPE multi-style and adaptive training with 16-diagonal transforms CMLLR and NCMLLR compensation on Toshiba in-car phone-number task.

Table 5 shows the recognition results for the MPE trained systems. As expected the MPE multi-style system performed better than the standard ML system for all conditions. Furthermore adapting the multi-style trained system again produced reductions in WER. Interestingly for the multi-style MPE trained systems CMLLR always performed better, or the same as NCMLLR. This is true even for the lower SNR conditions. For disciminatively trained multi-style systems, it is not clear how much gain can be obtained with the variance bias. Also comparing the ML and MPE trained multi-style systems with adaptation, there is almost no gain from MPE training over the ML results for NCMLLR (4% relative on HWY), and smaller gains for CMLLR (18% relative on HWY), compared to the unadapted gains from MPE training (27% relative on HWY).

Discriminative adaptively trained systems were then evaluated. The results are also shown in table 5. There are a couple of interesting trends. First, other than at the highest SNR condition, ENON, NCMLLR adaptation and adaptive training outperformed CMLLR. For the HWY condition, adaptive training with NCMLLR is 19% relative lower error rate than using CMLLR. The performance of the two at the highest SNR are very similar. Second, the gains of using adaptive training are larger with MPE training than for ML. For example using NCMLLR adaptive training, a 15% relative reduction in error rate on the HWY condition was obtained using ML training, but a 38% relative reduction using MPE training. For MPE training on data with highly variable noise conditions, adaptive training appears to be important to obtain the best performance.

8 Conclusions

This report has described a new version of linear transformation motivated from model-based noise robustness schemes. The approach, noisy CMLLR (NCMLLR) combines attributes of both CMLLR and variance bias transforms. This yields a transformation that has the same form as an efficient model-based scheme, joint uncertainty decoding (JUD), but estimated using maximum likelihood. The use of this transform for both adaptation and adaptive training are described, along with its relationship to existing linear transforms and factor-analysed-based covariance modelling schemes. In addition discriminative adaptive training update formulae are given.

The performance of NCMLLR was compared to CMLLR on two different tasks. The first, a noise corrupted version of resource management where noise from the NOISEX database have been artificially added, was medium vocabulary task. The second task used test data collected by Toshiba Research Europe Ltd, and comprises a continuous digit recognition task under a range of driving conditions. On both test sets, the use of NCMLLR at low SNR conditions outperformed CMLLR when using ML-training for both multi-style and adaptive training. For discriminatively trained systems, the use of adaptive training was found to be important to make the most use of the discriminative criteria. Again at low SNRs the best performance was obtained using NCMLLR adaptaive training.

The results presented in this paper indicate the possible advantages of using NCMLLR over CMLLR when good recognition performance in low-SNR conditions is needed. Further evaluation on larger, more complex, test sets is required. In particular data collected in low-SNR conditions. The current schemes are more computationally expensive than using CMLLR. Approximations and improvements to the transform and parameter estimation to allow full transforms to be used for large adaptive training tasks are required. Finally it would be interesting to compare the EM-based adaptive training approach described in this paper to the 2nd-order Newton-based approach described in [7]. This can be done for both JUD transforms and NCMLLR.

A EM Estimation for NCMLLR

The auxiliary function in equation 61 may be extended to

$$\begin{aligned}\mathcal{Q}(\hat{\mathcal{T}};\mathcal{T}) = &\ \mathbb{E}\left[\log p(\mathbf{Z}|\mathcal{M},\hat{\mathcal{T}})|\mathbf{O},\mathcal{M},\mathcal{T}\right] \\ = &\sum_{\boldsymbol{\Theta}} P(\boldsymbol{\Theta}|\mathbf{O},\mathcal{M},\mathcal{T})\mathbb{E}\left[\log p(\mathbf{O},\mathbf{S},\boldsymbol{\Theta}\mid\mathcal{M},\hat{\mathcal{T}})\mid\mathbf{O},\boldsymbol{\Theta},\mathcal{M},\mathcal{T}\right]\end{aligned} \tag{94}$$

The term in the expectation can be rewritten as

$$\log p(\mathbf{O},\mathbf{S},\boldsymbol{\Theta}|\mathcal{M},\hat{\mathcal{T}}) = \log p(\mathbf{O}|\mathbf{S},\boldsymbol{\Theta},\mathcal{M},\hat{\mathcal{T}}) + \log p(\mathbf{S}|\boldsymbol{\Theta},\mathcal{M},\hat{\mathcal{T}}) + \log p(\boldsymbol{\Theta}|\mathcal{M},\hat{\mathcal{T}}) \tag{95}$$

We are interested in the first and second terms on the right side of this equation. Assuming statistically independent observations, these equations are extended to

$$\log p(\mathbf{O}|\mathbf{S},\boldsymbol{\Theta},\mathcal{M},\hat{\mathcal{T}}) + \log p(\mathbf{S}|\boldsymbol{\Theta},\mathcal{M},\hat{\mathcal{T}}) = \sum_{t=1}^{T}\left\{\log p(\mathbf{o}_t|\mathbf{s}_t,\boldsymbol{\theta}_t,\mathcal{M},\hat{\mathcal{T}}) + \log p(\mathbf{s}_t|\boldsymbol{\theta}_t,\mathcal{M},\hat{\mathcal{T}})\right\} \tag{96}$$

The auxiliary function can be rewritten as

$$\mathcal{Q}(\hat{\mathcal{T}};\mathcal{T}) = \sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_{\mathsf{o},t}^{(m)}\mathbb{E}\left[\log p(\mathbf{o}_t|\mathbf{s}_t,m,\mathcal{M},\hat{\mathcal{T}}) + \log p(\mathbf{s}_t|m,\mathcal{M})|\mathbf{o}_t,m,\mathcal{M},\mathcal{T}\right] \tag{97}$$

where $\gamma_{\mathsf{o},t}^{(m)}$ is the posterior probability of component m given the observation sequence $\mathbf{O}$, NCMLLR parameter set $\mathcal{T}$, and model set $\mathcal{M}$. From equations 54 and 55, we obtain

$$p(\mathbf{o}_t|\mathbf{s}_t,m,\mathcal{M},\hat{\mathcal{T}}) = \mathcal{N}\left(\mathbf{o}_t;\hat{\mathbf{H}}^{(r_m)}\mathbf{s}_t + \hat{\mathbf{g}}^{(r_m)},\hat{\boldsymbol{\Psi}}^{(r_m)}\right) \tag{98}$$

$$p(\mathbf{s}_t|m,\mathcal{M}) = \mathcal{N}\left(\mathbf{s}_t;\boldsymbol{\mu}_{\mathbf{s}}^{(m)},\boldsymbol{\Sigma}_{\mathbf{s}}^{(m)}\right) \tag{99}$$

Then the auxiliary function may be extended to

$$\begin{aligned}\mathcal{Q}(\hat{\mathcal{T}};\mathcal{T}) = -\frac{1}{2}\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_{\mathsf{o},t}^{(m)}\mathbb{E}\Big[&\log|\hat{\boldsymbol{\Psi}}^{(r_m)}| \\ &+\left(\mathbf{o}_t - \hat{\mathbf{H}}^{(r_m)}\mathbf{s}_t - \hat{\mathbf{g}}^{(r_m)}\right)^{\mathsf{T}}\hat{\boldsymbol{\Psi}}^{(r_m)\text{-}1}\left(\mathbf{o}_t - \hat{\mathbf{H}}^{(r_m)}\mathbf{s}_t - \hat{\mathbf{g}}^{(r_m)}\right) \\ &+\log|\boldsymbol{\Sigma}_{\mathbf{s}}^{(m)}| + \left(\mathbf{s}_t - \boldsymbol{\mu}_{\mathbf{s}}^{(m)}\right)^{\mathsf{T}}\boldsymbol{\Sigma}_{\mathbf{s}}^{(m)\text{-}1}\left(\mathbf{s}_t - \boldsymbol{\mu}_{\mathbf{s}}^{(m)}\right)|\mathbf{o}_t,m,\mathcal{M},\mathcal{T}^{(r_m)}\Big]\end{aligned} \tag{100}$$

B Expectation Step

The sufficient statistics for the M-Step are computed in the E-Step using during a single pass through the training set. The following expectations are required:

$$\mathbb{E}\left[\mathbf{s}_t|\mathbf{o}_t,m,\mathcal{M},\mathcal{T}^{(r_m)}\right] = \int \mathbf{s}_t p(\mathbf{s}_t|\mathbf{o}_t,m,\mathcal{M},\mathcal{T}^{(r_m)})d\mathbf{s}_t \tag{101}$$

$$\mathbb{E}\left[\mathbf{s}_t\mathbf{s}_t^{\mathsf{T}}|\mathbf{o}_t,m,\mathcal{M},\mathcal{T}^{(r_m)}\right] = \int \mathbf{s}_t\mathbf{s}_t^{\mathsf{T}} p(\mathbf{s}_t|\mathbf{o}_t,m,\mathcal{M},\mathcal{T}^{(r_m)})d\mathbf{s}_t \tag{102}$$

Using the Bayes rule the posterior probability can be computed as follows:

$$p(\mathbf{s}_t|\mathbf{o}_t,m,\mathcal{M},\mathcal{T}^{(r_m)}) = \frac{p(\mathbf{o}_t|\mathbf{s}_t,m,\mathcal{M},\mathcal{T}^{(r_m)})p(\mathbf{s}_t|m,\mathcal{M})}{\int p(\mathbf{o}_t|\mathbf{s}_t,m,\mathcal{M},\mathcal{T}^{(r_m)})p(\mathbf{s}_t|m,\mathcal{M})d\mathbf{s}_t} \tag{103}$$

Then, with a normalisation term C the numerator may be expanded to

$$p(\mathbf{o}_t|\mathbf{s}_t, m, \mathcal{M}, \mathcal{T}^{(r_m)})p(\mathbf{s}_t|m, \mathcal{M}, \mathcal{T}^{(r_m)}) \sim C \cdot \mathcal{N}(\mathbf{s}_t; \tilde{\mathbf{s}}_t^{(m)}, \tilde{\mathbf{\Sigma}}_{\mathsf{s}}^{(m)}) \tag{104}$$

where

$$\begin{aligned}
\tilde{\mathbf{\Sigma}}_{\mathsf{s}}^{(m)} &= (\mathbf{\Sigma}_{\mathsf{s}}^{(m)\text{-}1} + \mathbf{H}^{(r_m)\mathsf{T}}\mathbf{\Psi}^{(r_m)\text{-}1}\mathbf{H}^{(r_m)})^{\text{-}1} \\
&= (\mathbf{\Sigma}_{\mathsf{s}}^{(m)\text{-}1} + \mathbf{\Sigma}_{\mathsf{b}}^{(r_m)\text{-}1})^{\text{-}1} && (105) \\
\tilde{\mathbf{s}}_{\mathsf{t}}^{(m)} &= \tilde{\mathbf{\Sigma}}_{\mathsf{s}}^{(m)}(\mathbf{\Sigma}_{\mathsf{s}}^{(m)\text{-}1}\boldsymbol{\mu}_{\mathsf{s}}^{(m)} + \mathbf{H}^{(r_m)\mathsf{T}}\mathbf{\Psi}^{(r_m)\text{-}1}(\mathbf{o}_t - \mathbf{g}^{(r_m)})) \\
&= \tilde{\mathbf{\Sigma}}_{\mathsf{s}}^{(m)}\mathbf{H}^{(r_m)\mathsf{T}}\mathbf{\Psi}^{(r_m)\text{-}1}\mathbf{o}_t + \tilde{\mathbf{\Sigma}}_{\mathsf{s}}^{(m)}(\mathbf{\Sigma}_{\mathsf{s}}^{(m)\text{-}1}\boldsymbol{\mu}_{\mathsf{s}}^{(m)} - \mathbf{H}^{(r_m)\mathsf{T}}\mathbf{\Psi}^{(r_m)\text{-}1}\mathbf{g}^{(r_m)}) && (106) \\
&= \tilde{\mathbf{A}}^{(m)}\mathbf{o}_t + \tilde{\mathbf{b}}^{(m)}
\end{aligned}$$

and

$$\begin{aligned}
\tilde{\mathbf{A}}^{(m)} &= \tilde{\mathbf{\Sigma}}_{\mathsf{s}}^{(m)}\mathbf{H}^{(r_m)\mathsf{T}}\mathbf{\Psi}^{(r_m)\text{-}1} \\
&= (\mathbf{\Sigma}_{\mathsf{s}}^{(m)\text{-}1} + \mathbf{\Sigma}_{\mathsf{b}}^{(r_m)\text{-}1})^{\text{-}1}\mathbf{\Sigma}_{\mathsf{b}}^{(r_m)\text{-}1}\mathbf{A}^{(r_m)} && (107) \\
\tilde{\mathbf{b}}^{(m)} &= \tilde{\mathbf{\Sigma}}_{\mathsf{s}}^{(m)}(\mathbf{\Sigma}_{\mathsf{s}}^{(m)\text{-}1}\boldsymbol{\mu}_{\mathsf{s}}^{(m)} - \mathbf{H}^{(r_m)\mathsf{T}}\mathbf{\Psi}^{(r_m)\text{-}1}\mathbf{g}^{(r_m)}) \\
&= (\mathbf{\Sigma}_{\mathsf{s}}^{(m)\text{-}1} + \mathbf{\Sigma}_{\mathsf{b}}^{(r_m)\text{-}1})^{\text{-}1}(\mathbf{\Sigma}_{\mathsf{s}}^{(m)\text{-}1}\boldsymbol{\mu}_{\mathsf{s}}^{(m)} + \mathbf{\Sigma}_{\mathsf{b}}^{(r_m)\text{-}1}\mathbf{b}^{(r_m)}) && (108)
\end{aligned}$$

Then,

$$\mathbb{E}\left[\mathbf{s}_t|\mathbf{o}_t, m, \mathcal{M}, \mathcal{T}^{(r_m h)}\right] = \tilde{\mathbf{s}}_t^{(m)} \tag{109}$$

$$\mathbb{E}\left[\mathbf{s}_t\mathbf{s}_t^{\mathsf{T}}|\mathbf{o}_t, m, \mathcal{M}, \mathcal{T}^{(r_m)}\right] = \tilde{\mathbf{\Sigma}}_{\mathsf{s}}^{(m)} + \tilde{\mathbf{s}}_t^{(m)}\tilde{\mathbf{s}}_t^{(m)\mathsf{T}} \ . \tag{110}$$

$\mathbb{E}\left[\boldsymbol{\zeta}_t|\mathbf{o}_t, m, \mathcal{M}, \mathcal{T}^{(r_m)}\right]$ and $\mathbb{E}\left[\boldsymbol{\zeta}_t\boldsymbol{\zeta}_t^{\mathsf{T}}|\mathbf{o}_t, m, \mathcal{M}, \mathcal{T}^{(r_m)}\right]$ can be easily calculated by equations 109 and 110 as follows:

$$\mathbb{E}\left[\boldsymbol{\zeta}_t|\mathbf{o}_t, m, \mathcal{M}, \mathcal{T}^{(r_m)}\right] = [1 \ \ \mathbb{E}[\mathbf{s}_t^{\mathsf{T}}|\mathbf{o}_t, m, \mathcal{M}, \mathcal{T}^{(r_m)}]]^{\mathsf{T}} = [1 \ \ \tilde{\mathbf{s}}_t^{(m)\mathsf{T}}]^{\mathsf{T}} \tag{111}$$

$$\mathbb{E}\left[\boldsymbol{\zeta}_t\boldsymbol{\zeta}_t^{\mathsf{T}}|\mathbf{o}_t, m, \mathcal{M}, \mathcal{T}^{(r_m)}\right] = \begin{bmatrix} 1 & \tilde{\mathbf{s}}_t^{(m)\mathsf{T}} \\ \tilde{\mathbf{s}}_t^{(m)} & \tilde{\mathbf{\Sigma}}_{\mathsf{s}}^{(m)} + \tilde{\mathbf{s}}_t^{(m)}\tilde{\mathbf{s}}_t^{(m)\mathsf{T}} \end{bmatrix} \tag{112}$$

References

[1] T. Anastasakos, J. McDonough, R. Schwartz, and J. Makhoul, "A compact model for speaker-adaptive training," in *Proc. ICSLP*, 1996.

[2] M.J.F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer Speech and Language*, vol. 12, Jan. 1998.

[3] 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, pp. 806–809.

[4] W.-T. Hong and S.-H. Chen, "A robust training algorithm for adverse speech recognition," *Speech Communication*, vol. 30, pp. 273–293, 2000.

[5] W.-T. Hong, "A discriminative and robust training algorithm for noisy speech recognition," in *Proc. ICASSP*, 2003.

[6] B.-O. Kang, H.-Y. Jung, and Y.-K. Lee, "Discriminative noise adaptive training approach for an environment migration," in *Proc. Interspeech*, 2007.

[7] H. Liao and M.J.F. Gales, "Adaptive training with joint uncertainty decoding for robust recognition of noisy data," in *Proc. ICASSP*, 2007.

[8] Y. Hu and Q. Huo, "Irrelevant variability normalization based HMM training using VTS approximation of an explicit model of environmental distortions," in *Proc. Interspeech*, 2007.

[9] H. Liao and M.J.F. Gales, "Issues with uncertainty decoding for noise robust speech recognition," in *Speech Communication*, April 2008.

[10] P.J. Moreno, *Speech Recognition in Noisy Environments*, Ph.D. thesis, Carnegie Mellon University, 1996.

[11] D.B. Rubin and D.T. Thayer, "EM algorithms for ML factor analysis," *Psychometrica*, vol. 47(1), pp. 69–76, 1982.

[12] R.A. Gopinath, B. Ramabhadran, and S. Dharanipragada, "Factor analysis invariant to linear transformations of data," in *Proc. ICSLP*, 1998, pp. 397–400.

[13] A. Sankar and C.-H. Lee, "A maximum-likelihood approach to stochastic matching for robust speech recognition," *IEEE Trans. on Speech and Audio Processing*, vol. 4, pp. 190–202, May 1996.

[14] D. Povey and P.C. Woodland, "Improved discriminative training techniques for large vocabulary continuous speech recognition," in *Proc. ICASSP*, 2001.

[15] C.J. Leggetter and P.C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density HMMs," *Computer Speech and Language*, vol. 9, pp. 171–186, 1995.

[16] M.J.F. Gales and P.C. Woodland, "Mean and variance adaptation within the MLLR framework," *Computer Speech and Language*, vol. 10, pp. 249–264, 1996.

[17] C.J. Leggetter and P.C. Woodland, "Flexible speaker adaptation for large vocabulary speech recognition," in *Proceedings Eurospeech*, 1995, pp. 1155–1158.

[18] M.J.F. Gales, "The generation and use of regression class trees for MLLR adaptation," Tech. Rep. CUED/F-INFENG/TR263, University of Cambridge, 1996, Available from http://mi.eng.cam.ac.uk/reports/index-speech.html.

[19] A.P. Dempster, N.M. Laird, and D.B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society*, vol. 39, pp. 1–39, 1977.

[20] V.V. Digalakis, D. Rtischev, and L.G. Neumeyer, "Speaker adaptation using constrained estimation of Gaussian mixtures," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 2, pp. 357–366, 1995.

[21] R.C. Rose, E.M. Hofstetter, and D.A. Reynolds, "Integrated models of signal and background with application to speaker identification in noise," *IEEE Trans. on Speech and Audio Processing*, vol. 2, pp. 245–257, 1994.

[22] M.J.F. Gales and S.J. Young, "Robust continuous speech recognition using parallel model combination," *IEEE Trans. on Speech and Audio Processing*, 1996.

[23] H. Liao and M.J.F. Gales, "Joint uncertainty decoding for noise robust speech recognition," in *Proc. Interspeech*, 2005.

[24] A. Acero, *Acoustical and Environmental Robustness in Automatic Speech Recognition*, Ph.D. thesis, Carnegie Mellon University, 1990.

[25] R.A. Gopinath, M.J.F. Gales, P.S. Gopalakrishnan, S. Balakrishnan-Aiyer, and M.A. Picheny, "Robust speech recognition in noise - performance of the IBM continuous speech recognizer on the ARPA noise spoke task," in *Proc. ARPA Workshop on Spoken Language System Technology*, Austin, Texas, 1999.

[26] M.J.F. Gales, *Model-Based Techniques for Noise Robust Speech Recognition*, Ph.D. thesis, Cambridge University, 1995.

[27] H. Liao and M.J.F. Gales, "Joint uncertainty decoding for robust large vocabulary speech recognition," Tech. Rep. CUED/F-INFENG/TR552, University of Cambridge, 2006, Available from http://mi.eng.cam.ac.uk/reports/index-speech.html.

[28] J. Li, L. Deng, Y. Gong, and A. Acero, "HMM adaptation with joint compensation of additive and convolutive distortions via vector Taylor series," in *Proc. ASRU*, Kyoto, Japan, 2007.

[29] A. Acero, L. Deng, T.T. Kristjansson, and J. Zhang, "HMM adaptation using vector Taylor series for noisy speech recognition," in *Proc. ICSLP*, Beijing, China, Oct. 2000.

[30] J.A. Arrowood and M.A. Clements, "Using observation uncertainty in HMM decoding," in *Proc. ICSLP*, Denver, Colorado, Sept. 2002.

[31] L. Deng, J. Droppo, and A. Acero, "Dynamic compensation of HMM variances using the feature enhancement uncertainty computed from a parametric model of speech distortion," *IEEE Trans. on Speech and Audio Processing*, vol. 12, no. 3, May 2005.

[32] Q. Huo and C.-H. Lee, "A Bayesian predictive classification approach to robust speech recognition," *IEEE Trans. on Speech and Audio Processing*, vol. 8, no. 2, 2000.

[33] T.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.

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

[35] H. Liao, *Uncertainty Decoding For Noise Robust Speech Recognition*, Ph.D. thesis, Cambridge University, 2007.

[36] S. Young, G. Evermann, M.J.F. Gales, T. Hain, D. Kershaw, X.A. Liu, G. Moore, J.J. Odell, D. Ollason, D. Povey, V. Valtchev, and P.C. Woodland, *The HTK Book (for HTK Version 3.4)*, University of Cambridge, December 2006.

[37] B.-H. Juang, W. Hou, and C.-H. Lee, "Minimum classification error rate methods for speech recognition," *IEEE Trans. on Speech and Audio Processing*, vol. 5, pp. 257–265, 1997.

[38] L.R. Bahl, P.F. Brown, P.V. De Souza, and R.L. Mercer, "Maximum mutual information estimation of hidden Markov model parameters for speech recognition," in *Proc. ICASSP*, 1986, vol. 1, pp. 49–52.

[39] D. Povey, *Discriminative Training for Large Vocabulary Speech Recognition*, Ph.D. thesis, University of Cambridge, 2003.

[40] L. Wang, *Discriminative linear transforms for adaptation and adaptive training*, Ph.D. thesis, Cambridge University, 2006.

[41] A-V.I. Rosti and M.J.F. Gales, "Factor analysed hidden Markov models for speech recognition," *Computer Speech and Language*, vol. 18(3), pp. 181–200, 2004.

[42] M.J.F. Gales and R.C. van Dalen, "Predictive linear transforms for noise robust speech recognition," in *Proc. ASRU*, 2007, pp. 59–64.

[43] R. Schluter and W. Macherey, "Comparison of discriminative training criteria," in *Proc. ICASSP*, 1998.

[44] J.L. Gauvain and C.-H. Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Trans. on Speech and Audio Processing*, vol. 2, pp. 291–298, 1994.