Discriminative Semi-parametric Trajectory Model for Speech Recognition

K. C. Sim and M. J. F. Gales

Cambridge University Engineering Department,
Trumpington Street, Cambridge,
CB2 1PZ United Kingdom

Abstract

Hidden Markov Models (HMMs) are the most commonly used acoustic model for speech recognition. In HMMs, the probability of successive observations is assumed independent given the state sequence. This is known as the *conditional independence* assumption. Consequently, the *temporal* (inter-frame) correlations are poorly modelled. This limitation may be reduced by incorporating some form of trajectory modelling. In this paper, a general perspective on trajectory modelling is provided, where time varying model parameters are used for the Gaussian components. A discriminative semi-parametric trajectory model is then described where the Gaussian mean vector and covariance matrix parameters vary with time. The time variation is modelled as a semi-parametric function of the observation sequence via a set of centroids in the acoustic space. The model parameters are estimated discriminatively using the Minimum Phone Error (MPE) criterion. The performance of these models is investigated and benchmarked against a state-of-the-art CUHTK Mandarin evaluation systems.

Key words: speech recognition, trajectory model, discriminative training, minimum phone error

1 Introduction

Hidden Markov Models (HMMs) [18] are widely used as the acoustic model in speech recognition. A series of assumptions underlie the use of HMMs to model the speech data, some of which are poor. In particular, the "conditional independence assumption" implies that the observation output probability is

Email address: [kcs23,mjfg]@eng.cam.ac.uk (K. C. Sim and M. J. F. Gales).

Preprint submitted to Elsevier Science *23 March 2007*

conditionally independent of all other observations given the current state. This yields a constant trajectory within an HMM state. Existing ways to overcome this limitation include the use of switching linear dynamical system [19], stochastic segment model [13, 12], polynomial segment model, buried Markov model [2] and trajectory HMM [21, 22]. In general, all these models are collectively known as trajectory models. To date, maximum likelihood training of these models has had very little success in large vocabulary continuous speech recognition. In this paper, a discriminative semi-parametric trajectory model will be presented. This model represents the Gaussian mean vectors and covariance matrices as time varying parameters. These time dependent parameters are modelled as a function of the location of the current observation (and the neighbouring observations) in the acoustic space, which is represented by a series of centroids. Model parameters are discriminatively estimated using the Minimum Phone Error (MPE) [17] criterion.

One form of temporally varying mean vector is obtained by applying a time dependent bias to the static Gaussian mean. This time dependent bias is a weighted contribution from the bias vectors associated with each centroid (to be estimated discriminatively). The contribution weights are calculated as the posteriors of the centroids given the observation (and neighbouring observations). The resulting model yields an fMPE model [16, 15]. This was originally presented as a feature transformation, but may also be described in the semi-parametric trajectory framework described here. The variance of each dimension may also be scaled by a positive time dependent factor to yield a temporally varying covariance matrix. This model will be referenced to as pMPE [20]. Similar to fMPE, the time dependent scale factor is a weighted contribution from the centroid specific scales where the weights are given by the posteriors of the observations given the centroids. Both of these models and their combination may be described as a semi-parametric trajectory model.

This paper is organised as follows. Section 2 introduces several forms of trajectory models applied to speech recognition and establishes a general formulation of time varying model parameters for trajectory models. This formulation is then used to introduce a *semi-parametric* trajectory model in Section 3. Next, Section 4 derives the parameter estimation formulae of this form of model using the Minimum Phone Error (MPE) criterion. Section 4.4 discusses the implementation issues. In Section 6, experimental results are given based on a large vocabulary conversational telephone speech recognition task. Finally, conclusions are given in Section 7.

2 Trajectory Models

There are a number of modelling approaches that have attempted to overcome the HMM conditional independence assumption. These include the use of switching linear dynamical systems [19], stochastic segment models [13, 12], polynomial segment models, buried Markov models [2] and trajectory HMM [21, 22]. All these models have a common aim of relaxing the "conditional independence assumption" by allowing the state output distribution to vary with time. This time variation is achieved by adding dependency on the observation sequence, $\mathcal{O}_1^T$, either directly or indirectly using latent variables. The model parameters for the state output probability are now viewed as time dependent such that

$$p(\mathcal{O}_1^T | Q_1^T, \boldsymbol{\theta}_t) = \prod_{t=1}^{T} p(\boldsymbol{o}_t | \boldsymbol{\theta}_t) \tag{1}$$

where the time dependent parameter set, $\boldsymbol{\theta}_t$, is expressed as a function of the observation sequence, $\mathcal{O}_1^T$, state sequence, Q_1^T, and the time, t, *i.e.*

$$\boldsymbol{\theta}_t = f\left(\mathcal{O}_1^T, Q_1^T, t; \boldsymbol{\theta}\right) \tag{2}$$

The form of function, $f(.)$, with parameters, $\boldsymbol{\theta}$, defines the type of model used. As with standard HMMs, it is convenient to represent the output density function as a Gaussian Mixture Model (GMM) [8] as it can be used to model any arbitrary non-Gaussian distribution and its model parameters may be estimated efficiently. In this case, the GMM parameters are time dependent:

$$p(\boldsymbol{o}_t | \boldsymbol{\theta}_t) = \sum_{m=1}^{M} c_{smt} \mathcal{N}\left(\boldsymbol{o}_t; \boldsymbol{\mu}_{smt}, \boldsymbol{\Sigma}_{smt}\right) \tag{3}$$

where $\boldsymbol{\theta}_t = \{c_{smt}, \boldsymbol{\mu}_{smt}, \boldsymbol{\Sigma}_{smt}\}$. These time varying Gaussian parameters may be expressed as a general function of the form given in equation (2). Therefore,

$$\{c_{smt}, \boldsymbol{\mu}_{smt}, \boldsymbol{\Sigma}_{smt}\} = f\left(\mathcal{O}_1^T, Q_1^T, t; \boldsymbol{\theta}_c, \boldsymbol{\theta}_\mu, \boldsymbol{\theta}_\Sigma\right) \tag{4}$$

where $\boldsymbol{\theta}_c$, $\boldsymbol{\theta}_\mu$ and $\boldsymbol{\theta}_\Sigma$ denote the model parameters for the component weight, mean and covariance matrix respectively.

In the following sections, several trajectory and segmental models will be described within the time varying parameter formulation given by equations (1) and (2).

2.1 Explicit Temporal Correlation Modelling

One of the earliest work on explicit time correlation modelling was carried out by Wellekens [23] where correlations between adjacent frames are explicitly modelled. This yields a time varying mean of the following form:

$$\boldsymbol{\mu}_{st} = \boldsymbol{\mu}_s + \boldsymbol{\Sigma}_{su}\boldsymbol{\Sigma}_{uu}^{-1}(\boldsymbol{o}_{t-1} - \boldsymbol{\mu}_u) \tag{5}$$

where s and u denote the current and previous states. $\boldsymbol{\mu}_s$ and $\boldsymbol{\Sigma}_{ss}$ are the mean vector and covariance matrix respectively of state s. $\boldsymbol{\Sigma}_{su}$ is the cross covariance matrix between state s and u. This is a special form of equation (4) where $\boldsymbol{\theta}_\mu = \{\boldsymbol{\mu}_s, \boldsymbol{\mu}_u, \boldsymbol{\Sigma}_{ss}, \boldsymbol{\Sigma}_{uu}, \boldsymbol{\Sigma}_{su}\}$.

Vector linear prediction (VLP) is also a trajectory model [24], where the state output probability of $\boldsymbol{o}_t$ is conditionally independent of other parameters given the current state, q_t, and observation dependencies, $\mathcal{H}_t$. The resulting mean vector becomes time dependent of the form

$$\boldsymbol{\mu}_{st} = \boldsymbol{\mu}_s^{(0)} + \sum_{p=1}^{P} \boldsymbol{A}_s^{(p)} \left(\boldsymbol{o}_{t+\tau_p} - \boldsymbol{\mu}_s^{(\tau_p)}\right) \tag{6}$$

where P is the number of predictors. The mean vector given the HMM state is dependent on the observation history, $\mathcal{H}_t = \{\boldsymbol{o}_{t+\tau_p} : 1 \le p \le P, 1 \le t + \tau_p \le T\}$. This form of model is again a specific form of a time varying parameter formulation given by equation (4) with $\boldsymbol{\theta}_\mu = \{\boldsymbol{\mu}_s^{(0)}, \boldsymbol{A}_s^{(p)}, \boldsymbol{\mu}_s^{(\tau_p)}\}$.

2.2 Implicit Temporal Correlation Modelling

Another type of trajectory modelling approach is to model the temporal correlations implicitly via some form of latent structures. An example of this approach is the Buried Markov Model (BMM) [2]. This model defines an additional *latent* variable, *buried* under the hidden states of HMMs. This latent variable defines the class of dependencies of emitting an observation, $\boldsymbol{o}_t$, at time t. In [2], a Gaussian-mixture BMM was described where the state output density function is modelled by a mixture of Gaussian of the form

$$\begin{aligned} p(\boldsymbol{o}_t|\boldsymbol{h}_t, q_t = s, \boldsymbol{\theta}) &= \sum_{m=1}^{M}\sum_{v=1}^{V} P(m|s,v)P(v|\boldsymbol{h}_t)\mathcal{N}\left(\boldsymbol{o}_t; \boldsymbol{\mu}_{smvt}, \boldsymbol{\Sigma}_{smv}\right) \\ &= \sum_{m=1}^{M}\sum_{v=1}^{V} c_{smvt}\mathcal{N}\left(\boldsymbol{o}_t; \boldsymbol{\mu}_{smvt}, \boldsymbol{\Sigma}_{smv}\right) \end{aligned} \tag{7}$$

where $\boldsymbol{h}_t$ is a column vector defining the entire collection of dependencies variables any element of $\boldsymbol{o}_t$ might use and v denotes the class of h_t. M and V are the number of components and classes respectively. $P(m|s, v)$, the prior of component m, given the state s and class v, is a discrete probability table and $P(v|\boldsymbol{h}_t)$ is the probability of class v given the continuous vector $\boldsymbol{h}_t$. This formulation yields a time varying Gaussian mean vector and component weight, given by

$$\begin{aligned} \boldsymbol{\mu}_{smvt} &= \boldsymbol{A}_{smv}\boldsymbol{h}_t + \boldsymbol{b}_{smv} && (8) \\ c_{smvt} &= P(m|s, v)p(v|\boldsymbol{h}_t) && (9) \end{aligned}$$

where $\boldsymbol{A}_{smv}$, $\boldsymbol{b}_{smv}$ and $P(m|s, v)$ are model parameters that can be estimated efficiently using the EM approach [2]. These expressions are dependent on time via the vector of dependency variables, $\boldsymbol{h}_t$. The term $\boldsymbol{A}_{smv}\boldsymbol{h}_t$ may be viewed as a time varying bias applied to $\boldsymbol{b}_{smv}$, the mean of component m, given the state s and class v.

The Switching Linear Dynamical System (SLDS) [19] also belongs to the trajectory model family. The generative model of an SLDS is given by the following state-space formulation

$$\begin{aligned} \boldsymbol{x}_t &= \boldsymbol{A}_s\boldsymbol{x}_{t-1} + \boldsymbol{w}_s \\ \boldsymbol{o}_t &= \boldsymbol{C}_s\boldsymbol{x}_t + \boldsymbol{v}_s \end{aligned} \qquad \text{where} \quad \begin{cases} \boldsymbol{w}_s \sim \mathcal{N}\left(\boldsymbol{w}_s; \boldsymbol{\mu}_s^{(x)}, \boldsymbol{\Sigma}_s^{(x)}\right) \\ \boldsymbol{v}_s \sim \mathcal{N}\left(\boldsymbol{v}_s; \boldsymbol{\mu}_s^{(o)}, \boldsymbol{\Sigma}_s^{(o)}\right) \end{cases} \tag{10}$$

where s denotes the discrete state generated by the underlying Markov chain within the HMM. $\boldsymbol{A}_s$ and $\boldsymbol{C}_s$ are state dependent linear transformation matrices while $\boldsymbol{w}_s$ and $\boldsymbol{v}_s$ are random vectors whose mean vectors and covariance matrices are also dependent on s. Therefore, the mean of observation given by this model is of the following time varying form

$$\boldsymbol{\mu}_{st} = \boldsymbol{C}_s\left(\boldsymbol{A}_s\boldsymbol{x}_{t-1} + \boldsymbol{\mu}_s^{(x)}\right) + \boldsymbol{\mu}_s^{(o)} \tag{11}$$

The time varying property arises due to the dependency on the continuous latent state variable, $\boldsymbol{x}_{t-1}$. The trajectory is modelled implicitly by the state evolution process. Note, the latent state variable, $\boldsymbol{x}_t$, can be expressed as a function of $\boldsymbol{A}_s$ and $\boldsymbol{x}_{t_0}$, where t_0 is the time of entering state s, by applying the first expression in equation (10) recursively. Thus, the trajectory within a state depends on the initial latent state when entering that state. This initial latent state is a function of the previous states visited and the durations spent in those states. The trajectory is therefore an implicit function of the historical state sequence.

Up to this point, a generic formulation of time varying parameters has been used to describe various existing trajectory models. In the following section, a semi-parametric trajectory model is introduced using the same formulation.

3 Semi-parametric Trajectory Model

The form of semi-parametric trajectory model considered in this paper can be formulated by expressing the mean vector and precision matrix as follows:

$$\boldsymbol{\mu}_{smt} = \boldsymbol{A}_t \boldsymbol{\mu}_{sm} + \boldsymbol{b}_t \tag{12}$$
$$\boldsymbol{P}_{smt} = \boldsymbol{Z}_t \boldsymbol{P}_{sm} \boldsymbol{Z}_t' \tag{13}$$

where $\boldsymbol{A}_t$ and $\boldsymbol{Z}_t$ are the time dependent linear transformations for the mean vector, $\boldsymbol{\mu}_{sm}$, and precision matrix, $\boldsymbol{P}_{sm}$, respectively. Precision matrix is defined as the inverse of the covariance matrix ($\boldsymbol{P}_{sm} = \boldsymbol{\Sigma}_{sm}^{-1}$). $\boldsymbol{b}_t$ denotes a time dependent bias vector for the mean. The form of time dependent linear transformations and bias considered in this work will be described later in Section 3.1. When the linear transformations are set as identity matrices ($\boldsymbol{A}_t = \boldsymbol{Z}_t = \boldsymbol{I}$) and the bias vector is set as a zero vector ($\boldsymbol{b}_t = \boldsymbol{0}$), the above expressions degenerate to the mean and precision matrix of a standard HMM system. The form of trajectory model described by equations 12 and 13 can be viewed as applying a time varying affine transformation to the component mean vectors and precision matrices in the system. An important question is how to determine the appropriate form of time varying transformations. In this section a semi-parametric representation will be described.

It is worth pointing out that using a full transformation matrices for equations (12) and (13) is impractical in many situations due to the high computational cost in applying the transformation at each time to each Gaussian component in the system (typical LVCSR systems comprise more than 100,000 Gaussian components). This problem may be alleviated by using diagonal transforms. Transformations can then be applied independently per dimension. Equations (12) and (13) may then be expressed as scaling and shifting of the mean and diagonal precision matrix elements for each dimension:

$$\mu_{smtj} = a_{tjj} \mu_{smj} + b_{tj} \tag{14}$$
$$p_{smtj} = z_{tjj}^2 p_{smj} \tag{15}$$

where μ_{smtj} and b_{tj} are the jth element of $\boldsymbol{\mu}_{smt}$ and $\boldsymbol{b}_t$ respectively. p_{smtj}, and z_{tjj} denote the jth diagonal element of $\boldsymbol{P}_{smt}$, and $\boldsymbol{Z}_t$ respectively and $\boldsymbol{A}_t$ is assumed to be a diagonal matrix. μ_{smj} and p_{smj} denote the jth element of the *time independent* mean and precision matrix respectively for component m in state s. In this work, only diagonal covariance matrix systems are considered, with the additional constraint that $\boldsymbol{A}_t$ is an identity matrix. In Section 4, the semi-parametric trajectory model parameters estimation will be presented based on the use of an identity matrix for the mean transformation, $\boldsymbol{A}_t$ and a diagonal transform for the precision matrix, $\boldsymbol{Z}_t$. The latter transformation will be referred to as the pMPE model.

3.1 A Semi-parametric Representation

Modelling of the time variation in the linear transformation is an important aspect for these trajectory models. A *semi-parametric* representation will be considered here. First, a series of centroids is defined to represent the regions of interest in the acoustic feature space. Associated with the ith centroid, the following parameters are defined:

- $\boldsymbol{A}^{(\mathtt{i})}$: a linear transformation matrix for the mean vector
- $\boldsymbol{Z}^{(\mathtt{i})}$: a linear transformation matrix for the precision matrix
- $\boldsymbol{b}^{(\mathtt{i})}$: a bias vector for the mean vector

The corresponding time varying affine transformations discussed above will be modelled as a *weighted contribution* from all the centroids:

$$\boldsymbol{A}_t = \boldsymbol{I} + \sum_{i=1}^{n} h_i(t)\boldsymbol{A}^{(\mathtt{i})} \tag{16}$$

$$\boldsymbol{b}_t = \sum_{i=1}^{n} h_i(t)\boldsymbol{b}^{(\mathtt{i})} \tag{17}$$

$$\boldsymbol{Z}_t = \boldsymbol{I} + \sum_{i=1}^{n} h_i(t)\boldsymbol{Z}^{(\mathtt{i})} \tag{18}$$

where $h_i(t)$ denotes the contribution weights from the i centroid at time t and n is the total number of centroids. The resulting time varying mean bias has a similar form to that of a Buried Markov Model ($\boldsymbol{A}_{smv}\boldsymbol{h}_t$), as shown in equation (8). To be more precise, $h_i(t)$ is equivalent to the ith element of $\boldsymbol{h}_t$ and $\boldsymbol{b}^{(\mathtt{i})}$ is the ith column of $\boldsymbol{A}_{smv}$. The two methods differ by the way $\boldsymbol{h}_t$ is defined. Nonetheless, $\boldsymbol{h}_t$ is used to capture the temporal variation in the model parameters in both cases.

Each centroid is modelled using a Gaussian component. Let g_i denote the ith centroid represented by the Gaussian component $\mathcal{N}\left(\boldsymbol{o}_t; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i\right)$ such that the likelihood of g_i given a d-dimensional observation, $\boldsymbol{o}_t$, is given by

$$p(\boldsymbol{o}_t|g_i) = \frac{1}{\sqrt{(2\pi)^d|\boldsymbol{\Sigma}_i|}} \exp\left\{-\frac{1}{2}\left(\boldsymbol{o}_t - \boldsymbol{\mu}_i\right)' \boldsymbol{\Sigma}_i^{-1}\left(\boldsymbol{o}_t - \boldsymbol{\mu}_i\right)\right\} \tag{19}$$

The weights, $h_i(t)$ is then computed as the posterior probability of g_i given $\boldsymbol{o}_t$,

$$h_i(t) = P(g_i|\boldsymbol{o}_t) = \frac{p(\boldsymbol{o}_t|g_i)P(g_i)}{\sum_{j=1}^{n} p(\boldsymbol{o}_t|g_j)P(g_j)} \tag{20}$$

where $P(g_i)$, the prior probability of g_i, is assumed to be uniformly distributed in this work. Consider a two-dimensional example in Figure 1. The centroids may be considered as a Vector-Quantisation (VQ) codebook representing the acoustic space. The posterior probabilities, $P(g_i|\boldsymbol{o}_t)$, would then be the probabilistic quantisation of $\boldsymbol{o}_t$. Thus, the interpolation formulae given in equations (16), (17) and (18) can be viewed as the weighted contribution from the transformations associated with each centroid given the position of the observation in the acoustic space. This formulation is analogous to the way the output probabilities are computed for semi-continuous HMMs, which leads to the interpretation of the above trajectory model as a semi-parametric model.

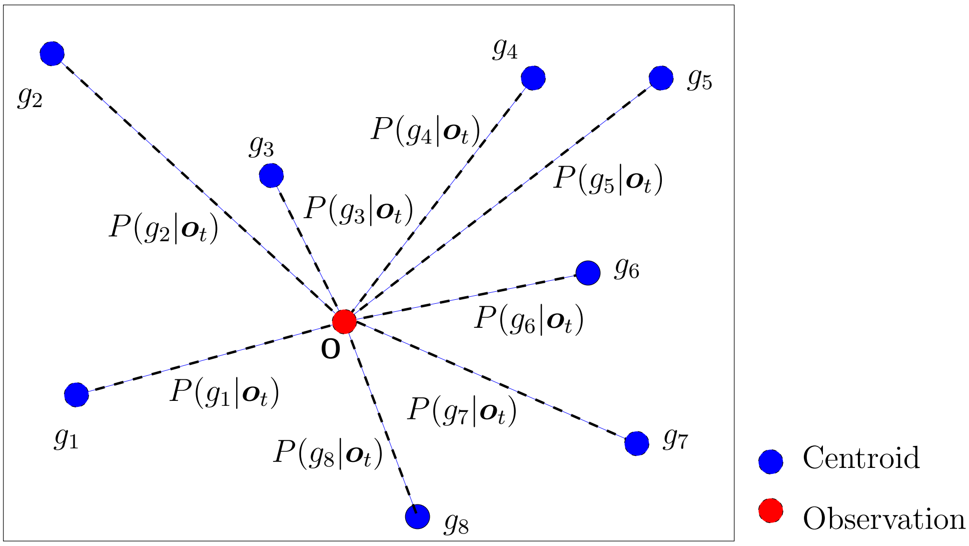
g_2
g_4
g_5
$P(g_4|\boldsymbol{o}_t)$
g_3
$P(g_5|\boldsymbol{o}_t)$
$P(g_3|\boldsymbol{o}_t)$
$P(g_2|\boldsymbol{o}_t)$
g_6
$P(g_6|\boldsymbol{o}_t)$
O
$P(g_1|\boldsymbol{o}_t)$
$P(g_7|\boldsymbol{o}_t)$
g_1
g_7
$P(g_8|\boldsymbol{o}_t)$
g_8
● Centroid
● Observation

Fig. 1. Obtaining interpolation weights from the posterior of a set of centroids given the observation sequence

Figure 2 depicts the visualisation of the semi-parametric trajectory model using a two-dimensional example. The interpolation weights are computed as a probabilistic VQ feature at each time t (see Figure 1) which tracks the observation as a smoothed trajectory. Interpolation using these time-dependent weights yields a trajectory of the Gaussian parameters, $\boldsymbol{\mu}_{smt}$ and $\boldsymbol{\Sigma}_{smt}$, conditioned upon the observation sequence, as given by equations (12) and (13) respectively.

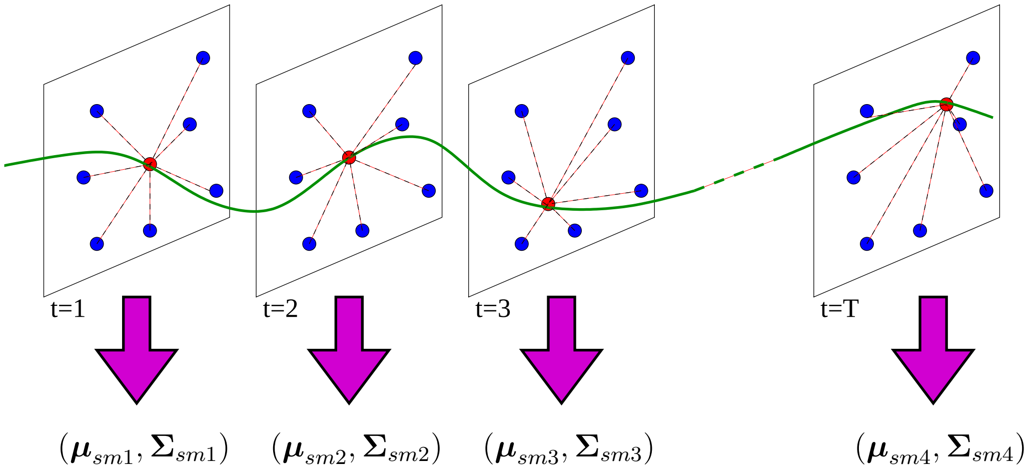
t=1
t=2
t=3
t=T
$(\boldsymbol{\mu}_{sm1}, \boldsymbol{\Sigma}_{sm1})$
$(\boldsymbol{\mu}_{sm2}, \boldsymbol{\Sigma}_{sm2})$
$(\boldsymbol{\mu}_{sm3}, \boldsymbol{\Sigma}_{sm3})$
$(\boldsymbol{\mu}_{sm4}, \boldsymbol{\Sigma}_{sm4})$

Fig. 2. A semi-parametric representation of the Gaussian parameters

3.2 Context Expansion for Semi-parametric Trajectory Model

In the semi-parametric trajectory model formulation, context expansion can be viewed as increasing the modelling power of the trajectory. All the discussions so far have been considering only the observation vector at the current time, t. It is possible to extend the dependency to a window of observations around t to allow for context expansion. Equations (16), (17) and (18) may be expressed in a more generic form as follows:

$$\boldsymbol{A}_t = \boldsymbol{I} + \sum_{\tau=-C}^{C} w(\tau) \sum_{i=1}^{n} h_i(t+\tau) \boldsymbol{A}_{\tau}^{(\mathtt{i})} \tag{21}$$

$$\boldsymbol{b}_t = \sum_{\tau=-C}^{C} w(\tau) \sum_{i=1}^{n} h_i(t+\tau) \boldsymbol{b}_{\tau}^{(\mathtt{i})} \tag{22}$$

$$\boldsymbol{Z}_t = \boldsymbol{I} + \sum_{\tau=-C}^{C} w(\tau) \sum_{i=1}^{n} h_i(t+\tau) \boldsymbol{Z}_{\tau}^{(\mathtt{i})} \tag{23}$$

where $w(\tau)$ is the window function of length $2C+1$, i.e. considering C frames on either side of the current frame. C can be viewed as the context of the trajectory. The window function used in this work follows the same as that introduced in [16], where

$$w(\tau) = \begin{cases} 1 & \tau = 0 \\ \frac{1}{2} & \tau = \pm 1, \pm 2 \\ & \vdots \\ \frac{1}{N} & \tau = \pm \frac{N(N-1)}{2}, \pm \left(\frac{N(N-1)}{2} + 1 \right), \ldots, C \end{cases} \tag{24}$$

and

$$C = \left(\frac{N(N-1)}{2} + N - 1\right) \tag{25}$$

When a window function that spans a large number of frames is used, it is necessary to tie the dynamic parameters to prevent over-training issue. This work adopts the same window length and tying scheme introduced in [16]. In that paper, a window length of 19 frames ($N = 4$, $C = 9$) was used. Without parameter tying, the number of dynamic parameters will be 19 times more than those without context expansion. To reduce the total number of free parameters, the dynamic parameters are tied across frames {1,2}, {3,4,5} and {6,7,8,9} to the left and right of the current frame, according to the partitions shown in equation (24). From the definitions of $w(\tau)$ in equation (24), this is equivalent to taking the average posteriors within the partitions so that the true expansion in terms of the dynamic parameters is only ±3 (7 times more than that without context expansion).

Though, it appears as if context expansion is essential to modelling trajectory since it takes into consideration neighbouring observation, this is not the case. The key element of this semi-parametric trajectory model lies in the fact that the position of the acoustic vector at each time is tracked in a *semi-parametric* way by using a set of centroids representing the acoustic space. Thus trajectory information is maintained by the observation vector itself. Context expansion simply extends the modelling power of the trajectory model by also considering the position of the neighbouring observation vectors. This allows the short term movement of the observation vectors to be captured, at the expense of increased model parameters.

4 Parameter Estimation

The parameterisation of the semi-parametric trajectory model can be broadly divided into those associated with the standard HMMs ($\boldsymbol{\theta}^{\text{h}}$) and those associated with the centroids ($\boldsymbol{\theta}^{\text{c}}$). In the rest of this discussion, $\boldsymbol{\theta}^{\text{h}}$ and $\boldsymbol{\theta}^{\text{c}}$ will be referred to as the *static* and *dynamic* parameters respectively to emphasise that the latter capture the temporally varying attributes of the trajectory model. This section derives the estimation formulae for these parameters using the MPE criterion [17]. The MPE objective function is a measure of the expected phone accuracy of recognising the training data given the HMM model. This is given by

$$\mathcal{R}^{\text{mpe}}(\boldsymbol{\theta}) = \sum_{u=1}^{U} \sum_{w \in \mathcal{W}_u} P(w|\mathcal{O}_1^T, \boldsymbol{\theta}) \text{PhoneAcc}(w, \hat{w}) \tag{26}$$

where $\boldsymbol{\theta}$ encompasses both $\boldsymbol{\theta}^{\mathrm{h}}$ and $\boldsymbol{\theta}^{\mathrm{c}}$. $\mathrm{PhoneAcc}(w, \hat{w})$ denotes the measure of phone accuracies of hypothesis w given the reference $\hat{w}$ and $\mathcal{W}_u$ is the set of competing hypotheses for sentence u. U is the total number of sentences in the training set. It is difficult to directly maximise this objective function directly. Instead, a weak-sense auxiliary function [17] is used. The weak-sense auxiliary function to be optimised is given by

$$\mathcal{Q}^{\mathrm{mpe}}(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}) = \sum_{t=1}^{T}\sum_{s=1}^{S}\sum_{m=1}^{M} \gamma_{sm}^{\mathrm{mpe}}(t) \log p(\boldsymbol{o}_t|\boldsymbol{\theta}) \tag{27}$$

where the log likelihood of component m in state s is given by,

$$\log p(\boldsymbol{o}_t|\boldsymbol{\theta}) = K_{sm} - \frac{1}{2}\sum_{j=1}^{d}\left\{\log(\sigma_{smtj}^2) + \frac{\left(o_{tj} - \mu_{smtj}\right)^2}{\sigma_{smtj}^2}\right\} \tag{28}$$

K_{sm} subsumes all terms that are independent of the model parameters. T is the total number of training speech frames, M is the number of Gaussian components per state and S is the total number of states in the system. $\gamma_{sm}^{\mathrm{mpe}}(t)$ is a quantity computed for MPE training [14], which can be regarded as the 'MPE posterior' of component m in state s at time t. This quantity is computed as the difference between the numerator and denominator posteriors of component m in state s at time t ($\gamma_{sm}^{\mathrm{n}}(t)$ and $\gamma_{sm}^{\mathrm{d}}(t)$ respectively). Typically, these posteriors are also smoothed by using the D-smoothing and I-smoothing techniques to obtain improved performance.

Maximising the above weak-sense auxiliary function with respect to all the model parameters ($\boldsymbol{\theta}^{\mathrm{h}}$ and $\boldsymbol{\theta}^{\mathrm{c}}$) is not trivial. Hence, these two sets of model parameters will be updated separately, each time keeping the other parameter set constant.

4.1 Static Parameters Estimation

First, consider the update of the static parameters given that the dynamic parameters are held constant. The weak-sense auxiliary function in equation (27) may be rewritten in terms of the trajectory model parameters as

$$\mathcal{Q}^{\mathrm{mpe}}(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}) = K - \frac{1}{2}\sum_{s=1}^{S}\sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{j=1}^{d} \gamma_{sm}^{\mathrm{mpe}}(t)\left\{\log(\sigma_{smtj}^2) + \frac{\left(o_{tj} - \mu_{smtj}\right)^2}{\sigma_{smtj}^2}\right\} \tag{29}$$

where K subsumes all the constant terms. The new parameters are found such that the differential of the auxiliary function with respect to the parameters at the new estimates equals to zero. Thus,

$$
\begin{aligned}
\frac{\partial \mathcal{Q}^{\mathrm{mpe}}(\boldsymbol{\theta},\hat{\boldsymbol{\theta}})}{\partial \mu_{smj}} &= \sum_{t=1}^{T}\left(\frac{\partial \mathcal{Q}^{\mathrm{mpe}}(\boldsymbol{\theta},\hat{\boldsymbol{\theta}})}{\partial \mu_{smtj}}\frac{\partial \mu_{smtj}}{\partial \mu_{smj}}\right) \\
&= \sum_{t=1}^{T}\gamma_{sm}^{\mathrm{mpe}}(t)\frac{(o_{tj}-\mu_{smtj})}{\sigma_{smtj}^{2}} = 0 \qquad (30)\\
\frac{\partial \mathcal{Q}^{\mathrm{mpe}}(\boldsymbol{\theta},\hat{\boldsymbol{\theta}})}{\partial \sigma_{smj}^{2}} &= \sum_{t=1}^{T}\left(\frac{\partial \mathcal{Q}^{\mathrm{mpe}}(\boldsymbol{\theta},\hat{\boldsymbol{\theta}})}{\partial \sigma_{smtj}^{2}}\frac{\partial \sigma_{smtj}^{2}}{\partial \sigma_{smj}^{2}}\right) \\
&= \frac{1}{2}\sum_{t=1}^{T}\gamma_{sm}^{\mathrm{mpe}}(t)\left\{\frac{\sigma_{smtj}^{2}-(o_{tj}-\mu_{smtj})^{2}}{(\sigma_{smtj}^{2})^{2}}\right\}z_{tjj}^{2} = 0 \qquad (31)
\end{aligned}
$$

Solving the above equations yields the update formulae for the jth element of the mean and variance as

$$
\mu_{smj} = \frac{x_{smj}^{\mathtt{mpe}}}{\tilde{\beta}_{smj}^{\mathtt{mpe}}} \qquad \text{and} \qquad \sigma_{smj}^{2} = \frac{w_{smj}^{\mathtt{mpe}}}{\beta_{sm}^{\mathtt{mpe}}} \qquad (32)
$$

where the sufficient statistics are given by

$$
\begin{aligned}
\beta_{sm}^{\mathtt{mpe}} &= \sum_{t=1}^{T}\gamma_{sm}^{\mathrm{mpe}}(t) \qquad (33)\\
\tilde{\beta}_{smj}^{\mathtt{mpe}} &= \sum_{t=1}^{T}\gamma_{sm}^{\mathrm{mpe}}(t)z_{tjj}^{2} \qquad (34)\\
x_{smj}^{\mathtt{mpe}} &= \sum_{t=1}^{T}\gamma_{sm}^{\mathrm{mpe}}(t)z_{tjj}^{2}\left(o_{tj}-b_{tj}\right) \qquad (35)\\
w_{smj}^{\mathtt{mpe}} &= \sum_{t=1}^{T}\gamma_{sm}^{\mathrm{mpe}}(t)z_{tjj}^{2}\left(o_{tj}-b_{tj}-\mu_{smj}\right)^{2} \qquad (36)
\end{aligned}
$$

Note that $\beta_{sm}^{\mathtt{mpe}}$ is already accumulated in the standard HMM parameters update for MPE training. $x_{smj}^{\mathtt{mpe}}$ and $w_{smj}^{\mathtt{mpe}}$ are the jth element of the mean and covariance matrix statistics given by equation (44), with the exception that the component posterior is scaled by z_{tjj}^{2} and the observation is shifted by b_{tj} for each dimension j. The additional statistics required is the d-dimensional $\tilde{\beta}_{smj}^{\mathtt{mpe}}$.

4.2 Dynamic Parameters Estimation

Having estimated the static parameters, the dynamic parameters may be estimated by keeping the static parameters constant. Here, the update of the centroid specific bias, $b_{j}^{(i)}$, and scaling factor, $z_{j}^{(i)}$, for the jth element of the mean vector and precision matrix will be described. Due to the large number of posteriors (ranging from thousands to hundreds of thousands), it is not feasible to accumulate the full second order statistics. Thus, a simple gradient optimisation approach, similar to that proposed in [16], will be used for fMPE. This approach may also be used to estimate the pMPE parameters. For both cases, an important value is the gradient of the weak-sense auxiliary function with respect to the dynamic parameters, $b_j^{(i)}$ and $z_j^{(i)}$ for all i. These gradients are given by

$$\frac{d\mathcal{Q}^{\text{mpe}}}{db_j^{(i)}} = \sum_{t=1}^{T}\sum_{s=1}^{S}\sum_{m=1}^{M}\frac{d\mathcal{Q}_{smt}^{\text{mpe}}}{db_j^{(i)}} \qquad \text{and} \qquad \frac{d\mathcal{Q}^{\text{mpe}}}{dz_j^{(i)}} = \sum_{t=1}^{T}\sum_{s=1}^{S}\sum_{m=1}^{M}\frac{d\mathcal{Q}_{smt}^{\text{mpe}}}{dz_j^{(i)}} \tag{37}$$

respectively, where $\mathcal{Q}_{smt}^{\text{mpe}}$ is defined such that $\mathcal{Q}^{\text{mpe}} = \sum_{s,m,t}\mathcal{Q}_{smt}^{\text{mpe}}$ and

$$\frac{d\mathcal{Q}_{smt}^{\text{mpe}}}{db_j^{(i)}} = \frac{\partial\mathcal{Q}_{smt}^{\text{mpe}}}{\partial b_j^{(i)}} + \frac{\partial\mathcal{Q}_{smt}^{\text{mpe}}}{\partial\mu_{smj}}\frac{\partial\mu_{smj}}{\partial b_j^{(i)}} + \frac{\partial\mathcal{Q}_{smt}^{\text{mpe}}}{\partial\sigma_{smj}^2}\frac{\partial\sigma_{smj}^2}{\partial b_j^{(i)}} \tag{38}$$

$$\frac{d\mathcal{Q}_{smt}^{\text{mpe}}}{dz_j^{(i)}} = \frac{\partial\mathcal{Q}_{smt}^{\text{mpe}}}{\partial z_j^{(i)}} + \frac{\partial\mathcal{Q}_{smt}^{\text{mpe}}}{\partial\mu_{smj}}\frac{\partial\mu_{smj}}{\partial z_j^{(i)}} + \frac{\partial\mathcal{Q}_{smt}^{\text{mpe}}}{\partial\sigma_{smj}^2}\frac{\partial\sigma_{smj}^2}{\partial z_j^{(i)}} \tag{39}$$

Equations (38) and (39) represent the *complete* differential of $\mathcal{Q}_{smt}^{\text{mpe}}$ with respect to $b_j^{(i)}$ and $z_j^{(i)}$ respectively [1] . In addition to finding the direction that maximises $\mathcal{Q}_{smt}^{\text{mpe}}$, the last two terms in the right hand side of equations (38) (referred to as the *indirect* differentials in [16]) and (39) also take into account the fact that the global shifting and scaling of the mean should be reflected by updating the static parameters. A proof of this is given in Appendix A. If only the partial differentials, rather than the complete differentials were used, the gains from dynamic parameter update tend to disappear when the static parameters are updated [16].

The partial differentials in the above equations are given by

$$\frac{\partial\mathcal{Q}_{smt}^{\text{mpe}}}{\partial b_j^{(i)}} = \frac{h_i(t)\gamma_{sm}^{\text{mpe}}(t)(o_{tj} - \mu_{smtj})}{\sigma_{smj}^2} \tag{40}$$

$$\frac{\partial\mathcal{Q}_{smt}^{\text{mpe}}}{\partial z_j^{(i)}} = h_i(t)\gamma_{sm}^{\text{mpe}}(t)(\sigma_{smtj}^2 - (o_{tj} - \mu_{smtj})^2) \tag{41}$$

$$\frac{\partial\mathcal{Q}_{smt}^{\text{mpe}}}{\partial\mu_{smj}} = \frac{\left(x_{smj}^{\mathtt{n}} - x_{smj}^{\mathtt{d}} - \mu_{smtj}\right)}{\sigma_{smj}^2} \tag{42}$$

$$\frac{\partial\mathcal{Q}_{smt}^{\text{mpe}}}{\partial\sigma_{smj}^2} = \frac{(w_{smj}^{\mathtt{n}} - w_{smj}^{\mathtt{d}}) - \sigma_{smj}^2\beta_{sm}^{\mathtt{mpe}}}{2(\sigma_{smj}^2)^2} \tag{43}$$

[1] The partial differential terms relating the bias, $b_j^{(i)}$ and variance scaling, $z_j^{(i)}$, are assumed to be small.

where x^{n}_{smj} and w^{n}_{smj} are the jth element of the MPE sufficient numerator statistics $\boldsymbol{x}^{\mathrm{n}}_{sm}$ and $\boldsymbol{W}^{\mathrm{n}}_{sm}$ respectively. These sufficient statistics are given by

$$\boldsymbol{x}^{\mathrm{n}}_{sm} = \sum_{t=1}^{T} \gamma^{\mathrm{ml}}_{sm}(t)\boldsymbol{o}_t \quad \text{and} \quad \boldsymbol{W}^{\mathrm{n}}_{sm} = \sum_{t=1}^{T} \gamma^{\mathrm{ml}}_{sm}(t)\left(\boldsymbol{o}_t - \boldsymbol{\mu}_{sm}\right)\left(\boldsymbol{o}_t - \boldsymbol{\mu}_{sm}\right)' \tag{44}$$

The denominator statistics, x^{d}_{smj} and w^{d}_{smj}, are defined in a similar fashion.

The forms of the remaining differentials $\frac{\partial \mu_{smj}}{\partial b^{(i)}_j}$, $\frac{\partial \sigma^2_{smj}}{\partial b^{(i)}_j}$, $\frac{\partial \mu_{smj}}{\partial z^{(i)}_j}$ and $\frac{\partial \sigma^2_{smj}}{\partial z^{(i)}_j}$ depend on the update methods for the static parameters, μ_{smj} and σ^2_{smj}. Ideally, MPE updates of all the parameters, including the static parameters, is preferred. However, the use of the D-smoothing and the I-smoothing with dynamic ML (or dynamic MMI) priors in standard MPE training [17] complicates the calculation of the *indirect* differentials. The next section describes a simpler form of update that yields efficient robust parameter estimation.

4.3 Interleaved Dynamic-Static Parameters Estimation

Simultaneous updates of both the static and dynamic parameters does not yield a closed form solution. A standard approach to this problem is to adopt an interleaved procedure where the static and the dynamic parameters are alternately updated. This allows the use of the gradients defined in the Sections 4.1 and 4.2. However it is still necessary to obtain the partial differentials of, for example, $\frac{\partial \mu_{smj}}{\partial b^{(i)}_j}$. To simplify this, and avoiding the issues of D-smoothing and I-smoothing, ML updates of the static parameters are considered when estimating the dynamic parameters. This makes the partial differentials simple to specify. After the dynamic parameters have been estimated, discriminative (MPE) training of the static parameters is then performed. This approach is similar to that proposed by Povey et al. in [16] and will be described in more detail below.

The interleaved parameter estimation procedure is summarised as follows:

```
1. Start from an ML trained model
2. Estimate dynamic parameters using MPE criterion
3. Estimate static parameters using ML criterion
4. When sufficient iterations performed, go to step 6
5. Go to step 2
6. Estimate static parameters using MPE criterion
```

Figure 3: *The interleaved dynamic static parameter estimation procedure*

It may seem strange to interleave updates with two different objective functions. However, provided that the appropriate static parameter update formulae are used in the complete differentials, the resulting dynamic parameters will capture the temporally varying aspect of the parameters. These complete differentials are crucial to prevent oscillation when interleaving between two different criteria [16].

The ML estimates of the static parameters are found by keeping the dynamic parameters constant, as described in Section 4.1, but using ML, rather than MPE, posteriors. The dynamic model parameters can then be estimated using the gradient in equations (38) and (39). As the static parameters are found using the ML criterion in the subsequent training iteration, the partial differential of the mean and variance with respect to the dynamic parameters are evaluated by differentiating equations in (32) with respect to $b_j^{(i)}$ and $z_j^{(i)}$, which yields

$$\frac{\partial \mu_{smj}}{\partial b_j^{(i)}} = -\frac{h_i(t)\gamma_{sm}^{\mathrm{ml}}(t)}{\tilde{\beta}_{smj}^{\mathtt{ml}}} \tag{45}$$

$$\frac{\partial \sigma_{smj}^2}{\partial b_j^{(i)}} = -\frac{2h_i(t)z_{tjj}\gamma_{sm}^{\mathrm{ml}}(t)(o_{tj}-\mu_{smtj})}{\beta_{sm}^{\mathtt{ml}}} \tag{46}$$

$$\frac{\partial \mu_{smj}}{\partial z_j^{(i)}} = \frac{2z_{tjj}\gamma_{sm}^{\mathrm{ml}}(t)(o_{tj}-\mu_{smtj})}{\tilde{\beta}_{smj}^{\mathtt{ml}}}\left(h_i(t) - \frac{z_{tjj}^2\gamma_{sm}^{\mathrm{ml}}(t)}{\tilde{\beta}_{smj}^{\mathtt{ml}}}\right) \tag{47}$$

$$\frac{\partial \sigma_{smj}^2}{\partial z_j^{(i)}} = \frac{2h_i(t)z_{tjj}\gamma_{sm}^{\mathrm{ml}}(t)(o_{tj}-\mu_{smtj})^2}{\beta_{sm}^{\mathtt{ml}}} \tag{48}$$

When $z_{tjj} = 1$, equations (45) and (46) become those of the standard fMPE presented in [16]. Once the gradient information is computed, the dynamic parameters are updated as follows:

$$\hat{b}_j^{(i)} = b_j^{(i)} + \eta_j^{(i)}\frac{d\mathcal{Q}^{\mathrm{mpe}}}{db_j^{(i)}} \qquad \text{and} \qquad \hat{z}_j^{(i)} = z_j^{(i)} + \nu_j^{(i)}\frac{d\mathcal{Q}^{\mathrm{mpe}}}{dz_j^{(i)}} \tag{49}$$

where $\hat{b}_j^{(i)}$ and $\hat{z}_j^{(i)}$ denote the updated parameters for $b_j^{(i)}$ and $z_j^{(i)}$ respectively. $\eta_j^{(i)}$ and $\nu_j^{(i)}$ are the element specific learning rate for $b_j^{(i)}$ and $z_j^{(i)}$ which are defined as

$$\eta_j^{(i)} = \frac{\alpha\bar{\sigma}_j}{\phi_{ij}^{(b)} + \rho_{ij}^{(b)}} \qquad \text{and} \qquad \nu_j^{(i)} = \frac{\alpha}{\phi_{ij}^{(z)} + \rho_{ij}^{(z)}} \tag{50}$$

respectively. α is a scalar parameter for adjusting the learning rate and $\bar{\sigma}_j$ is the average standard deviation of the Gaussian components in the system. $\phi_{ij}^{(b)}$

and $\rho_{ij}^{(b)}$ are the sum of the positive and negative contributions to the gradient of $\mathcal{Q}_{smt}^{\text{mpe}}$ with respect to $b_j^{(i)}$ at each time, t, as presented in [16]. A similar approach may be used for $\phi_{ij}^{(z)}$ and $\rho_{ij}^{(z)}$, which determine the learning rate for the precision scaling. Hence,

$$\phi_{ij}^{(b)} = \sum_{t=1}^{T} \max\left\{\sum_{s=1}^{S}\sum_{m=1}^{M} \frac{d\mathcal{Q}^{\mathtt{mpe}}}{db_j^{(i)}}, 0\right\} \quad \text{and} \quad \rho_{ij}^{(b)} = \sum_{t=1}^{T} \max\left\{-\sum_{s=1}^{S}\sum_{m=1}^{M} \frac{d\mathcal{Q}^{\mathtt{mpe}}}{db_j^{(i)}}, 0\right\}$$
$$\phi_{ij}^{(z)} = \sum_{t=1}^{T} \max\left\{\sum_{s=1}^{S}\sum_{m=1}^{M} \frac{d\mathcal{Q}^{\mathtt{mpe}}}{dz_j^{(i)}}, 0\right\} \quad \text{and} \quad \rho_{ij}^{(z)} = \sum_{t=1}^{T} \max\left\{-\sum_{s=1}^{S}\sum_{m=1}^{M} \frac{d\mathcal{Q}^{\mathtt{mpe}}}{dz_j^{(i)}}, 0\right\}$$

After updating the dynamic parameters (and the static parameters using ML), the static parameters may be updated using MPE training. This is achieved using the original update equations in (32).

It is possible to stop the training process after performing dynamic parameters, without any additional MPE training of the static parameters. In this paper to denote the difference, where only the dynamic parameters are updated this will be denoted as, for example, pMPE. Where additional MPE training of the static parameters has been performed this will be referred to as pMPE+MPE.

4.4 Implementation Issues

First, the likelihood computation of fMPE and pMPE models will be examined. As fMPE may be implemented as a feature transformation, the additional cost for the likelihood calculation of fMPE model is negligible compared to the standard HMM system [2]. For pMPE there is a slight increase in this cost. The likelihood of the model parameters, $\boldsymbol{\theta} = \{\mu_{smj}, \sigma_{smj}^2\}$, given the observation vector, $\boldsymbol{o}_t$, is given by

$$\log p(\boldsymbol{o}_t|\boldsymbol{\theta}) = K + \frac{1}{2}\sum_{j=1}^{d}\left\{\log z_{tjj} - \log \sigma_{smj}^2 - \frac{z_{tjj}(o_{tj} - \mu_{smj})^2}{\sigma_{smj}^2}\right\} \tag{51}$$

This requires an extra d multiplications and 1 addition compared to the standard model. It also requires z_{tjj} and $\sum_{j=1}^{d} \log z_{tjj}$ to be cached for each frame.

[2] The additional cost is due to the computation of the posterior probabilities for the centroids, which can be achieved efficiently by using some kind of Gaussian selection techniques [16]

pMPE parameter estimation was found to be less robust than fMPE and was more likely to be overtrained. This is not surprising as second order statistics are used. To handle this problem, the values of α for precision scaling estimation were typically set to be less than those for MPE ($\alpha < 1.0$). Furthermore, in some cases the temporally varying scale, z_{tjj}^2 tended a value close to zero. This was felt to be due to the *wrap-around* from squaring z_{tjj}. For example, if z_{tjj} is close to zero, its value may oscillate and change sign over time. However, squaring z_{tjj} ignores the sign and may result in an undesirable wrap-around effect to the trajectory of the precision scale factor. To prevent this, a minimum value is applied to z_{tjj}, similar to the concept of variance flooring:

$$\tilde{z}_{tjj} = \max\{z_{tjj}, z_{\mathtt{min}}\} \tag{52}$$

where $\tilde{z}_{tjj}$ is the floored scale factor and $z_{\mathtt{min}}$ is the scale floor. In this work, $z_{\mathtt{min}}$ has been set to 0.1.

5 Relationship to Linear Adaptation and fMPE

The general form of semi-parametric trajectory modelling given in equations (12) and (13) resembles that of linear transformation based speaker and environment adaptation, for example Maximum Likelihood Linear Regression (MLLR) adaptation formulae for mean vector [10] and covariance matrix [6] respectively. The semi-parametric formulation of equations (12) and (13) may be viewed as a time-varying linear adaptation of model parameters, depending on the position and movement of the observation vector in the acoustic space.

Instead of applying time varying transforms to the Gaussian parameters, they may also be applied to the feature vectors.

$$\hat{\boldsymbol{o}}_t = \boldsymbol{C}_t \boldsymbol{o}_t + \boldsymbol{d}_t \tag{53}$$

where $\boldsymbol{o}_t$ and $\hat{\boldsymbol{o}}_t$ are the original and transformed observation vectors. This is equivalent to setting the case where the linear transformation matrices for the mean vector and covariance matrices are the same. This is analogous to viewing Constrained MLLR [4] as a restrictive form of MLLR mean and variance adaptations.

It is interesting to note that equation (53) is identical to the fMPE model [16] when $\boldsymbol{C}_t = \boldsymbol{I}$. In this case, a time varying bias, $\boldsymbol{d}_t$, is applied to the features. This is equivalent to subtracting the same bias from the mean vectors ($\boldsymbol{d}_t =$

$-\boldsymbol{b}_t$). In [16], the time varying feature offset is given by

$$\boldsymbol{d}_t = \boldsymbol{M}\boldsymbol{h}_t \tag{54}$$

where $\boldsymbol{M}$ is a projection matrix from the high dimensional vector of posteriors ($\boldsymbol{h}_t = [h_1(t) \;\; h_2(t) \;\; \ldots \;\; h_n(t)]'$) to standard feature size. Comparing this to equation (17), it is clear that the columns of $\boldsymbol{M}$ are given by $-\boldsymbol{b}^{(\mathtt{i})}$.

6 Experimental Results

The experimental results presented in this section are based on a Conversational Telephone Speech Mandarin (CTS-M) task. All the systems in these experiments used 12 Perceptual Linear Prediction (PLP) coefficients [7] with the C0 energy term and the first three derivatives. Heteroscedastic Linear Discriminant Analysis (HLDA) [9] was then applied to project the feature down to 39 dimensions. Pitch and its first two derivatives were appended to this feature vector to yield a 42-dimensional feature-space. Finally, Gaussianisation [5] was applied to normalise the features per conversation side. For more details of the system configuration, see [5].

The acoustic models were trained using 72 hours of ldc04 and swm03 data provided by LDC. Decision tree state-clustered triphones were used. All the systems used in this work had approximately 4000 distinct states. System evaluation was performed on two test sets: dev04 (2 hours) and eval04 (1 hour).

For the semi-parametric trajectory models, 4000 centroids were used to obtain the time varying mean offset and precision scaling. These centroids were obtained from the baseline system with 4000 distinct states and one Gaussian component per states. This is considerably smaller than that used in fMPE [16] (approximately 100,000 centroids). Preliminary experiments showed that using more centroids (obtained, for example, from the Gaussian components of a 16-component per state ML system) leads to over-training problem due to limited available training data. For each observation, most of the posterior values were very small (close to zero). Gaussian clustering and minimum posterior threshold were used as a simple Gaussian selection process to improve the computational efficiency. It was empirically found that Gaussian components with high posterior values are almost always found wihtin the 5 nearest group. Furthermore, it was found that centroids with posterior value below 0.1 do not contribute very much to the trajectory modelling. These values were used in all the experiments in this paper. Without context expansion, there were on average about two non-zero posteriors per frame. The number of nonzero posteriors increases to 14 per frame when a ±3 context expansion was used.

The first set of experiments were conducted on a simple single-component-per-state systems. This allows semi-parametric trajectory modelling to be examined without the effect of *implicit* trajectory modelling due to the use of multiple components. The baseline system was obtained by performing eight MPE iterations, starting from an ML trained system. From the same ML system, the fMPE and pMPE systems were trained with 4 iterations using the *interleaved* update (see Section 4.3). These systems were furthered refined with 8 MPE iterations to yield the `fMPE+MPE` and `pMPE+MPE` systems.

Figure 4 shows the improvement in the MPE criterion with increasing training iterations for various single component systems. The top and bottom graphs correspond to systems without and with ±3 context expansions respectively. The initial dotted lines indicate fMPE/pMPE training and the final solid lines denote the 8 iterations of standard MPE training. The MPE criterion for the baseline MPE system increases from 0.49 to 0.59. Without context expansion, the fMPE and pMPE models yielded a much lower absolute MPE criterion gain of 0.023 and 0.011 respectively after 4 iterations. However, with additional eight MPE training iterations, the criterion obtained were slightly better than the baseline (approximately 0.60).

The modelling power of the systems greatly improved when a ±3 context expansion was used. This is clearly reflected from the criterion gain shown in the figure. From the same figure, the variation is the MPE criterion for the pMPE system with ±3 context expansion follows closely to that of the MPE system. Furthermore, the gain from fMPE with context expansion is clearly better than the baseline. The final systems, fMPE+MPE and pMPE+MPE, were about 0.05 and 0.03 better in terms of MPE criterion.

System	`dev04`		`eval04`	
	0	±3	0	±3
MPE	44.4		42.2	
fMPE+MPE	42.1	40.1	39.4	37.3
pMPE+MPE	43.3	41.3	40.4	38.6
fMPE+pMPE+MPE	41.6	38.9	39.2	36.6

Table 1
CER performance of 1-component fMPE and pMPE systems with 0 and ±3 context expansion on `dev04` and `eval04` for `CTS-M` task

The Character Error Rate (CER) performance of the above systems on `dev04` and `eval04` is summarised in Table 1. The baseline single component MPE alone system gave 44.4% and 42.2% CER on `dev04` and `eval04` respectively.

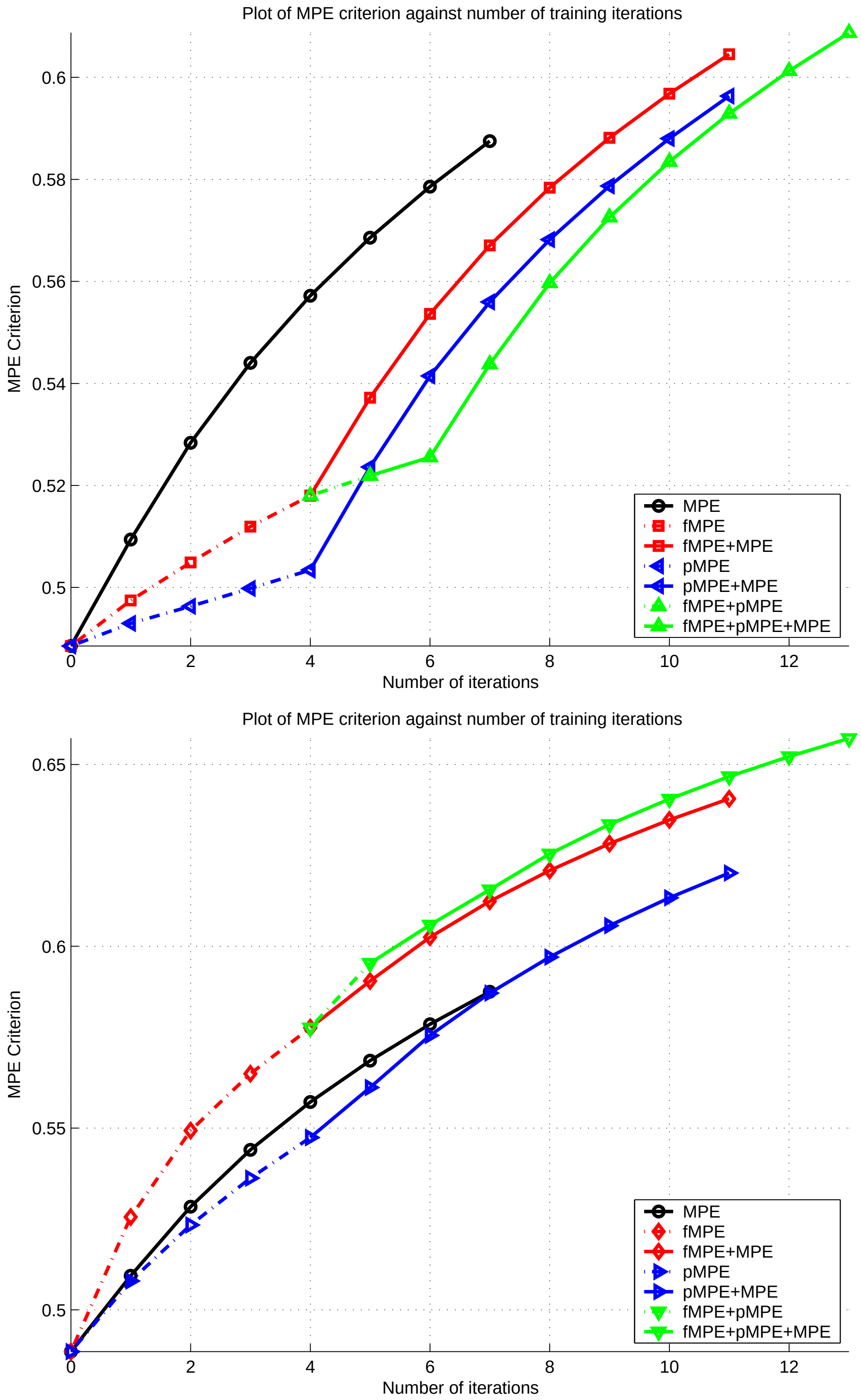

Fig. 4. Change in MPE criterion with increasing training iterations for single component systems without context expansion (top) and with ±3 context expansion (bottom)

The fMPE+MPE system improved the baseline by 2.3–2.8% without context expansion. In the same configuration, pMPE+MPE system, gave smaller gains, absolute improvements of 1.1–1.8%. In combination, fMPE+pMPE+MPE, additional gains of 0.2–0.5% over the fMPE+MPE system were obtained.

If context expansion of ±3 was used both fMPE and pMPE showed gains over the no context expansion cases. For the pMPE+MPE system absolute improvements of 3.1–3.6% were obtained over the baseline MPE system. However, again the gains from pMPE+MPE were smaller than those using fMPE+MPE. Combining the two approaches together gave total gains of about 5.5% absolute over the baseline MPE system.

Several important conjectures can be made based on these results. First, when a simple acoustic model was used, the gains obtained from the fMPE and pMPE techniques became significantly larger. The loss in the modelling power of the static parameters has been compensated by the dynamic parameters. Moreover, the pMPE+MPE system combined well with context expansion. This provides a clear indication that the pMPE+MPE with ±3 context expansion suffered from an over-fitting problem. In addition, promising gains were also obtained by combining the fMPE and pMPE techniques to yield the fMPE+pMPE+MPE system.

As previously mentioned, the fMPE and pMPE techniques may be viewed as a semi-parametric trajectory model. From this perspective, the single component fMPE and pMPE systems also provided an interesting account for the trajectory modelling aspects of the system. Because the systems under consideration have only one Gaussian component per state, the observations associated with each state in a standard HMM are independent and identically distributed (*i.i.d.*) with a normal distribution. Thus, the trajectory within each HMM state is *piece-wise constant*. By incorporating the fMPE and pMPE techniques to the single component systems, promising improvements as shown in Table 1 were obtained.

Figures 5 and 6 show the trajectory of the fMPE+MPE and pMPE+MPE models. Note that the standard MPE system models the observation sequence with a *piece-wise linear* trajectory. Both the fMPE+MPE and pMPE+MPE systems were capable of model more flexible trajectories. Because the model parameters were estimated using discriminative training, the resulting trajectories may not follow that of the observation. The actual time varying mean offset and precision scale are depicted in the bottom graphs of Figures 5 and 6 respectively. The mean offset has a range between -0.2 to 0.3 while the precision scale falls between 0.7 to 1.6.

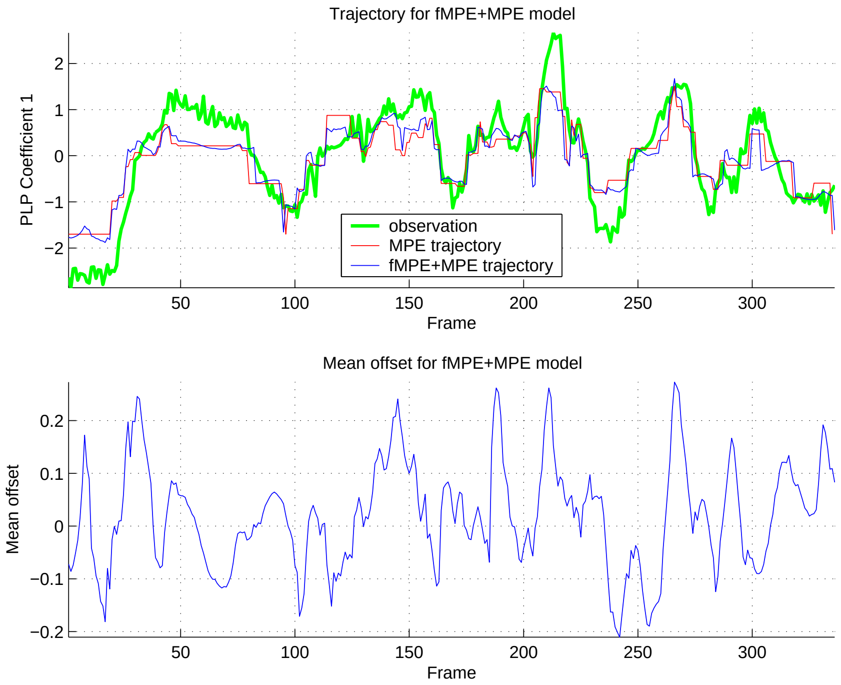

Fig. 5. Top: fMPE+MPE trajectory compared with the MPE trajectory and the corresponding observation for the first dimension of the feature; Bottom: Time varying mean offset for fMPE

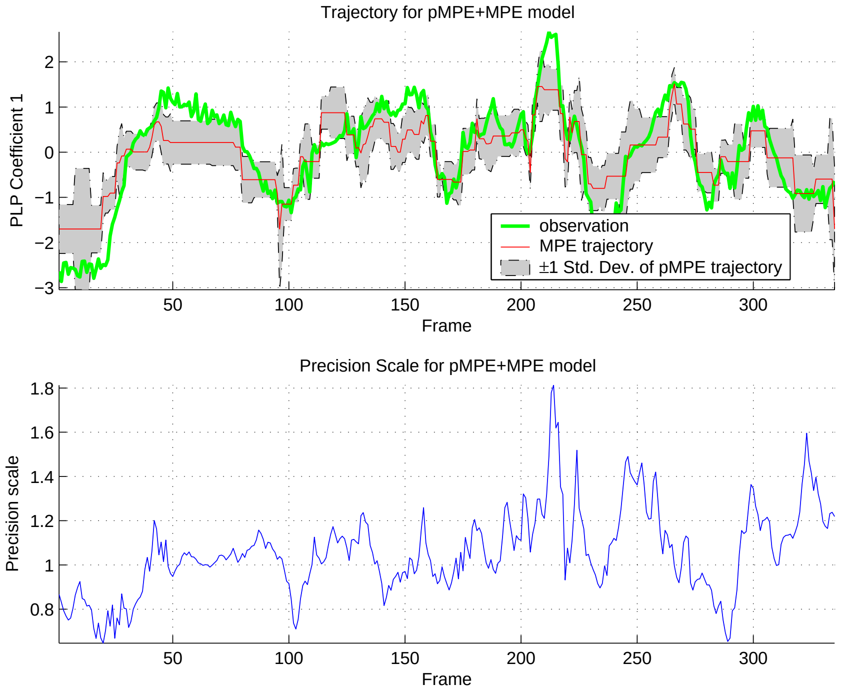

Fig. 6. Top: MPE trajectory and the corresponding observation for the first dimension of the feature. Shaded area represents the uncertainty of the pMPE trajectory within ±1 standard deviation; Bottom: Time varying precision scale for pMPE

6.2 Multiple Component Systems

As with the single component systems, the same set of experiments were also conducted on 16-component systems to examine the gains from fMPE and pMPE on more complex systems. Table 2 summarises the CER results. The CER of the baseline MPE system was 36.0% and 33.9% on `dev04` and `eval04` respectively. As expected, the gains from the fMPE+MPE and pMPE+MPE systems were found to be smaller compared to the single component systems. The former yielded gains of 0.2–0.4% without context expansion and 1.4–1.6% with ±3 context expansion. There is a large improvement to the CER performance when context information is considered. Unfortunately, apart from the 0.6% absolute improvement on `dev04`, the gains from the pMPE+MPE system almost disappeared compared to the gains obtained for the single component systems. When combining fMPE and pMPE, the 0.2% absolute gain was obtained on both test sets when without having context expansion. However, a degradation of 0.3–0.6% in performance was observed when ±3 context expansion was used. Clearly, the pMPE parameters cannot be reliably estimated when used with systems with high complexity. In the next section, an alternative approach to combining fMPE and pMPE is pursued.

System	`dev04`		`eval04`	
	0	±3	0	±3
MPE	36.0		33.9	
fMPE+MPE	35.6	34.4	33.7	32.5
pMPE+MPE	35.9	35.4	33.7	33.8
fMPE+pMPE+MPE	35.3	34.7	33.5	33.1

Table 2
CER performance of 16-component fMPE and pMPE systems with 0 and ±3 context expansion on `dev04` and `eval04` for `CTS-M` task

6.3 Systems Combination

From the above results, it was found that directly combining fMPE and pMPE did not yield good performance for 16-component systems. Another method of combining these techniques is using Confusion Network Combination (CNC) [11]. CNC is performed by first generating a set of hypotheses (in lattices format) for each individual system. These lattices are converted to sausage nets of word alternatives with confidence scores assigned to each word (known as confusion networks). Confusion networks from multiple systems are combined and rescored to obtain the word sequence with the highest confidence score.

System		dev04	eval04
S1	MPE	35.0	33.4
S2	fMPE+MPE	33.9	32.2
S3	fMPE+pMPE+MPE	34.0	32.6
S1+S2	CNC	34.1	32.2
S2+S3		33.3	31.6

Table 3
CER performance of confusion network decoding and combination of 16-component fMPE and pMPE systems with ±3 context expansion on dev04 and eval04 for CTS-M task

Table 3 shows the confusion network (CN) decoding [3] results on dev04 and eval04. Similar to the Viterbi decoding results, the fMPE+MPE system (S2) was found to be about 1.1–1.2% better than the MPE alone system (S1) while the fMPE+pMPE+MPE system (S3) was 0.1–0.4% worse than S2. In addition, the effect of 2-way system combination using CNC was examined. Due to the large performance gap between S1 and S2, the combination performance was at most the same as the best individual system. However, despite the poorer performance of S3 compared to S2, a further absolute improvement of 0.6–0.8% was obtained when these two systems are combined. This indicates that the errors made by the two system are considerably different. The resulting trajectory modelled by fMPE and pMPE are also different (see Figures 5 and 6). Therefore, shifting the mean vector and scaling the variance temporally model different aspects of the trajectory which are complimentary.

7 Conclusions

This paper has introduced a discriminative semi-parametric trajectory model. In this model, the state output probability density function is represented by a Gaussian Mixture Model (GMM) where the Gaussian mean vector and the diagonal covariance matrix varies with time. The time dependency is modelled as a smoothed function of the observation sequence using a basis superposition formulation. Each basis is associated with a centroid representing a position (or movement) in the acoustic space. The corresponding basis coefficients are derived from the posterior of its centroid given the current and possibly the surrounding observations. Hence, the basis coefficients are time dependent which results in temporally varying model parameters. It was also shown that this semi-parametric trajectory model is the same as the fMPE technique if only the mean vectors are being modelled. In addition, a novel approach of pMPE was also introduced where the precision matrix elements are modelled as temporally varying parameters. Both fMPE and pMPE were found to give gains over the MPE alone system on a conversational telephone speech Mandarin task. It was also found that combining fMPE and pMPE could be beneficial in some cases.

Acknowledgements

This work was supported by DARPA grant MDA972-02-1-0013. The thesis does not necessarily reflect the position or the policy of the US Government and no official endorsement should be inferred.

A Appendix

This section provides the proof that using the complete differentials in equation (37) to update the dynamic parameters, as described in Section 4.2, does not result in global shifting and scaling of the static parameters. This is achieved by showing that the complete differentials are zero when only one centroid is used. This is because with only one centroid, there is only one contributing factor and the mean bias and variance scale factors will be the same for all time frames. Therefore, to prevent global shifting or scaling, the complete differentials should be zero yield no update in the dynamic parameters.

A.1 Proof of Non-global Shifting for fMPE Update

First, consider the complete differential for fMPE (first part of equation (37)). Using equations (38), (40), (42), (43), (45) and (46), the complete differential for fMPE is given by

$$\begin{aligned}\sum_{s,m,t} \frac{dQ^{\text{mpe}}_{smt}}{db^{(i)}_j} =& \sum_{s,m,t} \left(\frac{\partial Q^{\text{mpe}}_{smt}}{\partial b^{(i)}_j} + \frac{\partial Q^{\text{mpe}}_{smt}}{\partial \mu_{smj}} \frac{\partial \mu_{smj}}{\partial b^{(i)}_j} + \frac{\partial Q^{\text{mpe}}_{smt}}{\partial \sigma^2_{smj}} \frac{\partial \sigma^2_{smj}}{\partial b^{(i)}_j} \right) \\ =& \sum_{s,m,t} \left(\frac{\gamma^{\text{mpe}}_{sm}(t)(o_{tj} - \mu_{smtj})}{\sigma^2_{smj}} - \frac{\left(x^{\mathtt{n}}_{smj} - x^{\mathtt{d}}_{smj} - \mu_{smtj}\right)}{\sigma^2_{smj}} \frac{\gamma^{\text{ml}}_{sm}(t)}{\tilde{\beta}^{\mathtt{ml}}_{smj}} \right) \\ =& \sum_{s,m} \left(\frac{x^{\mathtt{n}}_{smj} - x^{\mathtt{d}}_{smj} - \mu_{smtj}}{\sigma^2_{smj}} - \frac{x^{\mathtt{n}}_{smj} - x^{\mathtt{d}}_{smj} - \mu_{smtj}}{\sigma^2_{smj}} \right) = 0 \quad \text{(A.1)}\end{aligned}$$

using the fact that

$$h_i(t) = 1, \qquad \sum_{t-1}^{T} \gamma_{sm}^{\mathrm{ml}}(t) = \tilde{\beta}_{smj}^{\mathtt{ml}}, \qquad \sum_{t-1}^{T} \gamma_{sm}^{\mathrm{mpe}}(t) o_{tj} = x_{smj}^{\mathtt{n}} - x_{smj}^{\mathtt{d}} \tag{A.2}$$

and

$$\begin{aligned}\sum_{s.m.t} \frac{\partial \mathcal{Q}_{smt}^{\mathrm{mpe}}}{\partial \sigma_{smj}^2} \frac{\partial \sigma_{smj}^2}{\partial b_j^{(i)}} &= -\sum_{s,m,t} \left(\frac{(w_{smj}^{\mathtt{n}} - w_{smj}^{\mathtt{d}}) - \sigma_{smj}^2 \beta_{sm}^{\mathtt{mpe}}}{(\sigma_{smj}^2)^2} \right) \left(\frac{z_{tjj} \gamma_{sm}^{\mathrm{ml}}(t)(o_{tj} - \mu_{smtj})}{\beta_{sm}^{\mathtt{ml}}} \right) \\ &= \sum_{s,m} \left(\frac{(w_{smj}^{\mathtt{n}} - w_{smj}^{\mathtt{d}}) - \sigma_{smj}^2 \beta_{sm}^{\mathtt{mpe}}}{(\sigma_{smj}^2)^2} \right) \times 0 \end{aligned} \tag{A.3}$$

The final term simplifies to zero by using the static mean update with the mean shifts and variance scale factors initialised to zeros and ones respectively ($b_{tj} = 0$ and $z_{tjj} = 1$), *i.e.*

$$\mu_{smtj} = \mu_{smj} = \frac{\sum_{t=1}^{T} \gamma_{sm}^{\mathrm{ml}}(t) o_{tj}}{\sum_{t=1}^{T} \gamma_{sm}^{\mathrm{ml}}(t)} = \frac{\sum_{t=1}^{T} \gamma_{sm}^{\mathrm{ml}}(t) o_{tj}}{\beta_{sm}^{\mathtt{ml}}} \tag{A.4}$$

The ML statistics used in the above equation correspond to those collected in the *previous* iteration to obtain the current estimate of μ_{smj}. Since the complete differential for fMPE equals to zero when there is only one centroid, therefore the fMPE update does not yield a global shifting of the static mean vectors.

A.2 Proof of Non-global Scaling for pMPE Update

Similarly, consider the complete differential for pMPE (second part of equation (37)). Using equations (39), (41), (42), (43), (47) and (48), the complete differential for pMPE is given by

$$\begin{aligned}\sum_{s,m,t} \frac{d\mathcal{Q}_{smt}^{\mathrm{mpe}}}{dz_j^{(i)}} &= \sum_{s,m,t} \left(\frac{\partial \mathcal{Q}_{smt}^{\mathrm{mpe}}}{\partial z_j^{(i)}} + \frac{\partial \mathcal{Q}_{smt}^{\mathrm{mpe}}}{\partial \mu_{smj}} \frac{\partial \mu_{smj}}{\partial z_j^{(i)}} + \frac{\partial \mathcal{Q}_{smt}^{\mathrm{mpe}}}{\partial \sigma_{smj}^2} \frac{\partial \sigma_{smj}^2}{\partial z_j^{(i)}} \right) \\ &= \sum_{s,m,t} \frac{\gamma_{sm}^{\mathrm{mpe}}(t)(\sigma_{smj}^2 - (o_{tj} - \mu_{smtj})^2)}{\sigma_{smtj}^2} \\ &\quad + \sum_{s,m,t} \left(\frac{(w_{smj}^{\mathtt{n}} - w_{smj}^{\mathtt{d}}) - \sigma_{smj}^2 \beta_{sm}^{\mathtt{mpe}}}{(\sigma_{smj}^2)^2} \right) \left(\frac{z_{tjj} \gamma_{sm}^{\mathrm{ml}}(t)(o_{tj} - \mu_{smtj})^2}{\beta_{sm}^{\mathtt{ml}}} \right) \\ &= \sum_{s,m} \left(\beta_{sm}^{\mathtt{mpe}} - \frac{(w_{smj}^{\mathtt{n}} - w_{smj}^{\mathtt{d}})}{\sigma_{smj}^2} + \frac{(w_{smj}^{\mathtt{n}} - w_{smj}^{\mathtt{d}})}{\sigma_{smj}^2} - \beta_{sm}^{\mathtt{mpe}} \right) = 0 \end{aligned} \tag{A.5}$$

using the fact that $h_i(t) = 1$ and that the variance scale factors are initialised to unity ($z_{tjj} = 1$), then

$$\sigma^2_{smtj} = \sigma^2_{smj} = \frac{\sum_{t=1}^{T} \gamma^{\text{ml}}_{sm}(t)(o_{tj} - \mu_{smtj})^2}{\beta^{\mathtt{ml}}_{sm}} \tag{A.6}$$

$$(w^{\mathtt{n}}_{smj} - w^{\mathtt{d}}_{smj}) = \sum_{t=1}^{T} \gamma^{\text{mpe}}_{sm}(t)(o_{tj} - \mu_{smtj})^2 \tag{A.7}$$

where the ML statistics in the above equation are again obtained the previous iteration. Therefore, the complete differential for pMPE update does not result in a global scaling of the variances.

References

[1] L. E. Baum and J. A. Eagon. An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology. *Bull. Amer. Math. Soc.*, 73:360–363, 1967.

[2] J. A. Bilmes. Buried Markov models for speech recognition. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, pages 713–716, 1999.

[3] G. Evermann and P. C. Woodland. Posterior probability decoding, confidence estimation and system combination. In *Proc. Speech Transcription Workshop*, 2000.

[4] M. J. F. Gales. Maximum likelihood linear transformations for HMM-based speech recognition. Technical Report CUED/F-INFENG/TR291, Cambridge University, 1997. (via anonymous) `ftp://svr-www.eng.cam.ac.uk`.

[5] M. J. F. Gales, B. Jia, X. Liu, K. C. Sim, P. C. Woodland, and K. Yu. Development of the CUHTK 2004 RT04f mandarin conversational telephone speech transcription system. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, pages 861–864, March 2005.

[6] M. J. F. Gales and P. C. Woodland. Mean and variance adaptation within the MLLR framework. *Computer Speech and Languages*, 10:249–264, 1996.

[7] H. Hermansky. Perceptual Linear Predictive (PLP) analysis of speech. *Journal of the Acoustical Society of America*, 87(4):1738–1752, 1990.

[8] B. J. Huang, S. E. Levinson, and M. M. Sondhi. Maximum likelihood estimation for multivariate mixture observations of Markov chains. *IEEE Trans. Information Theory*, IT-32:307–309, March 1986.

[9] N. Kumar. *Investigation of Silicon-Auditory Models and Generalization of Linear Discriminant Analysis for Improved Speech Recognition.* PhD thesis, Johns Hopkins University, 1997.

[10] C. J. Legetter and P. C. Woodland. Maximum likelihood linear regression

speaker adaptation of contiuous density HMMs. *Computer Speech and Languages*, 1997.

[11] L. Mangu, E. Brill, and A. Stolcke. Finding consensus among words: Lattice-based word error minimization. In *Proc. Eur. Conf. Speech Commun. Technol.*, 1999.

[12] M. Ostendorf, V. Digalakis, and O. Kimball. From HMM's to segment models: A unified view of stochastic modeling for speech recognition. *IEEE Transactions on Speech and Audio Processing*, 4(5):360–378, 1996.

[13] M. Ostendorf and S. Roukos. A stochastic segment model for phoneme-based continuous speech recognition. *IEEE Trans on Acoustics, Speech and Signal Processing*, 37(12):1857–1869, 1989.

[14] D. Povey. *Discriminative Training for Large Vocabulary Speech Recognition.* PhD thesis, Cambridge University, 2003.

[15] D. Povey. Improvements to fMPE for discriminative training of features. In *Proc. Interspeech*, September 2005.

[16] D. Povey, B. Kingsbury, L. Mangu, G. Saon, H. Soltau, and G. Zweig. fMPE: Discriminatively trained features for speech recognition. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2005.

[17] D. Povey and P. C. Woodland. Minimum Phone Error and I-smoothing for improved discriminative training. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2002.

[18] L. A. Rabiner. A tutorial on hidden Markov models and selective applications in speech recognition. In *Proc. of the IEEE*, volume 77, pages 257–286, February 1989.

[19] A-V. I. Rosti and M. J. F. Gales. Switching linear dynamical systems for speech recognition. Technical Report CUED/F-INFENG/TR461, Cambridge University, 2003. (via anonymous) `ftp://svr-www.eng.cam.ac.uk`.

[20] K. C. Sim and M. J. F. Gales. Temporally varying model parameters for large vocabulary continuous speech recognition. In *Proc. Interspeech*, September 2005.

[21] K. Tokuda, H. Zen, and T. Kitamura. Trajectory modeling based on HMMs with the explicit relationship between static and dynamic features. In *Proc. Eur. Conf. Speech Commun. Technol.*, pages 865–868, 2003.

[22] K. Tokuda, H. Zen, and T. Kitamura. Reformulating the HMM as a trajectory model. In *Proc. of Beyond HMM – Workshop on statistical modeling approach for speech recognition*, 2004.

[23] C. J. Wellekens. Explicit time correlation in hidden Markov models for speech recognition. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, pages 384–386, 1987.

[24] P. C. Woodland. Hidden Markov models using vector linear prediction anddiscriminative output distributions. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, volume 1, pages 509–512, 1992.