CAMBRIDGE UNIVERSITY
ENGINEERING DEPARTMENT

FACTOR ANALYSED HIDDEN MARKOV MODELS FOR SPEECH RECOGNITION

A-V.I. Rosti & M.J.F. Gales

CUED/F-INFENG/TR.453
April 4, 2003

Cambridge University Engineering Department
Trumpington Street
Cambridge. CB2 1PZ
England
E-mail: {avir2, mjfg}@eng.cam.ac.uk

Abstract

Recently various techniques to improve the correlation model of feature vector elements in speech recognition systems have been proposed. Such techniques include semi-tied covariance HMMs and systems based on factor analysis. All these schemes have been shown to improve the speech recognition performance without dramatically increasing the number of model parameters compared to standard diagonal covariance Gaussian mixture HMMs. This paper introduces a general form of acoustic model, the factor analysed HMM. A variety of configurations of this model and parameter sharing schemes, some of which correspond to standard systems are examined. An EM algorithm for the parameter optimisation is presented along with a number of methods to increase the efficiency of training. The performance of FAHMMs on medium to large vocabulary continuous speech recognition tasks is investigated. The experiments show that without elaborate complexity control an equivalent or better performance compared to a standard diagonal covariance Gaussian mixture HMM system can be achieved with considerably fewer parameters.

1 Introduction

Hidden Markov models (HMMs) with continuous observation densities have been widely used for speech recognition tasks. The observation densities associated with each state of the HMMs should be sufficiently general to capture the variations among individual speakers and acoustic environments. At the same time, the number of parameters describing the densities should be as low as possible to enable fast and robust parameter estimation when using a limited amount of training data. Gaussian mixture models (GMMs) are the most commonly used form of state distribution model. They are able to approximate non-Gaussian densities, including densities with multiple modes. One of the issues when using GMMs is the form of covariance matrix for each component. Using full covariance components increases the number of parameters dramatically which can result in poor parameter estimates. Hence, components with diagonal covariance matrices are commonly used in HMMs for speech recognition. Diagonal covariance GMMs can model correlations between the feature vector elements. However, it would be beneficial to have uncorrelated feature vectors for each component when diagonal covariance matrices are used.

A number of schemes to tackle this intra-frame correlation problem have been proposed. One approach to decorrelate the feature vectors is to transform each set of vectors assigned to a particular component so that the diagonal covariance matrix assumption becomes valid. This system would, however, have the same complexity as full covariance GMMs. Alternatively, a single global decorrelation transform could be used [4]. Unfortunately, it is hard to find a single transform that decorrelates speech feature vectors for all states in an HMM system. Semi-tied covariance matrices (STCs) [4] can be viewed as a halfway solution. A class of states with diagonal covariance matrices can be transformed into full covariance matrices via a class specific linear transform. Systems employing STC generally yield better performance than standard diagonal covariance HMMs, or single global transforms, without dramatically increasing the number of model parameters.

Subspace models are an alternative approach to transform schemes for spatial correlation modelling. Heteroscedastic linear discriminant analysis (HLDA) [5, 12] models the feature vectors via a linear projection matrix applied to some lower dimensional vectors superimposed with noise spanning the uninformative, "nuisance" dimensions. There is a close relationship between STC and HLDA. The parameter estimation is similar and both can be viewed as feature space transform schemes. Alternatives to systems based on LDA-like projections are schemes based on factor analysis [18, 8]. These model the covariance matrix via a linear probabilistic process applied to a simpler lower dimensional representation called factors. The factors can be viewed as state vectors and the factor analysis as a generative observation process. Each component of a standard HMM system can be replaced with a factor analysed covariance model [18]. This dramatically increases the number of model parameters due to an individual loading matrix attached to each component. The loading matrix and the underlying factors can be shared among several components as in shared factor analysis (SFA). This system is closely related to the factor analysis invariant to linear transformations of data [8] without the global linear transformation. SFA also assumes the factors being distributed according to a standard normal distribution. Alternatively the standard factor analysis can be extended by modelling the factors with GMMs as in independent factor analysis (IFA) [1]. IFA also assumes independence between the individual factors which corresponds to a multiple stream system, each stream consisting of one dimension (factor).

This paper introduces an extension to the standard factor analysis which is applicable to HMMs. The model is called factor analysed HMM (FAHMM). FAHMMs belong to a broad class of generalised linear Gaussian models [16] which extends the set of standard linear Gaussian models [17]. Generalised linear Gaussian models are state space models with linear state evolution and observation processes, and Gaussian mixture distributed noise processes. The underlying HMM generates piecewise constant state vector trajectories that are mapped into the observation space via linear probabilistic observation processes. FAHMM combines the observation process from SFA with the standard diagonal covariance Gaussian mixture HMM acting as a state evolution process. Alternatively, it can be viewed as a dynamic version of IFA with Gaussian mixture model as the observation noise. Due to the factor analysis based observation process, FAHMMs should model the intra-frame correlation better than diagonal covariance matrix HMMs, yet be more compact than full covariance matrix HMMs. In addition, FAHMMs allow a variety of configurations and subspaces to be explored.

The second section of this paper describes the theory behind FAHMMs including efficient likelihood calculation and the parameter estimation. Implementation issues arising from increased number of model parameters and resource constraints are discussed in the following section. An efficient two level training scheme is described as well. A number of experiments with different configurations in medium to large vocabulary speech recognition tasks are presented in Section 4. Conclusions and future work are also provided.

1.1 Notation

In this paper, bold capital letters are used to denote matrices, e.g. $\boldsymbol{A}$, bold letters refer to vectors, e.g. $\boldsymbol{a}$, and plain letters represent scalars, e.g. c. All vectors are column vectors unless otherwise stated. Prime is used to denote the transpose of a matrix or a vector, e.g. $\boldsymbol{A}', \boldsymbol{a}'$. The determinant of a matrix is denoted by $|\boldsymbol{A}|$. Gaussian distributed vectors, e.g. $\boldsymbol{x}$ with mean vector, $\boldsymbol{\mu}$, and covariance matrix, $\boldsymbol{\Sigma}$, are denoted by $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. The likelihood of a vector $\boldsymbol{z}$ being generated by the above Gaussian; i.e., the Gaussian evaluated at the point $\boldsymbol{z}$, is represented as $p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{z}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$. Vectors distributed according to a Gaussian mixture model are denoted by $\boldsymbol{x} \sim \sum_m c_m \mathcal{N}(\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)$. The lower case letter p is used to represent a continuous distribution, whereas a capital letter P is used to denote a probability mass function of a discrete variable. The probability that a discrete random variable, ω, equals m is denoted by $P(\omega = m)$.

2 Factor Analysed Hidden Markov Models

First, the theory behind factor analysis is revisited and a generalisation to factor analysis to employ Gaussian mixture distributions is presented. The factor analysed HMM is introduced in a generative model framework. Efficient likelihood calculation and parameter optimisation for FAHMMs are then presented. The section is concluded by relating several configurations of FAHMMs to standard systems.

2.1 Factor Analysis

Factor analysis is a statistical method for modelling the covariance structure of high dimensional data using a small number of latent (hidden) variables. It is often used to model the data instead of Gaussian distribution with full covariance matrix. Factor analysis can be described by the following generative model

$$\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}) \tag{1}$$

$$\boldsymbol{o} = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{v}, \quad \boldsymbol{v} \sim \mathcal{N}(\boldsymbol{\mu}^{(o)}, \boldsymbol{\Sigma}^{(o)}) \tag{2}$$

where $\boldsymbol{x}$ is a collection of k factors (k-dimensional state vector) and $\boldsymbol{o}$ is a p-dimensional observation vector. The covariance structure is captured by the factor loading matrix (observation matrix), $\boldsymbol{C}$, which represents the linear transform relationship between the state vector and the observation vector. The mean of the observations is determined by the error (observation noise) modelled as a single Gaussian with mean vector $\boldsymbol{\mu}^{(o)}$ and diagonal covariance matrix $\boldsymbol{\Sigma}^{(o)}$. The observation process in Equation 2 can be expressed as a conditional distribution, $p(\boldsymbol{o}|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{o}; \boldsymbol{C}\boldsymbol{x} + \boldsymbol{\mu}^{(o)}, \boldsymbol{\Sigma}^{(o)})$. Also, the observation distribution is a Gaussian with mean vector $\boldsymbol{\mu}^{(o)}$ and covariance matrix $\boldsymbol{C}\boldsymbol{C}' + \boldsymbol{\Sigma}^{(o)}$.

The number of model parameters in a factor analysis model is $pk + 2p$. It should be noted that any non-zero state space mean vector, $\boldsymbol{\mu}^{(x)}$, can be absorbed by the observation mean vector by adding $\boldsymbol{C}\boldsymbol{\mu}^{(x)}$ into $\boldsymbol{\mu}^{(o)}$. Furthermore, any non-identity state space covariance matrix, $\boldsymbol{\Sigma}^{(x)}$, can be transformed into an identity matrix using eigen decomposition, $\boldsymbol{\Sigma}^{(x)} = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}'$. $\boldsymbol{Q}$ consists of the eigenvectors of $\mathbf{\Sigma}^{(x)}$ and $\mathbf{\Lambda}$ is a diagonal matrix with the eigenvalues of $\mathbf{\Sigma}^{(x)}$ on the main diagonal. The eigen decomposition always exists and is real valued since the covariance matrix is symmetric positive semi-definite. The transformation can be subsumed into the observation matrix by multiplying $\boldsymbol{C}$ from the right by $\boldsymbol{Q}\Lambda^{1/2}$. It is also essential that the observation noise covariance matrix be diagonal. Otherwise, the sample statistics of the data can be set as the observation noise and leave the loading matrix to zero. A reduction in the number of model parameters compared to a full covariance model can be achieved by choosing the state space dimensionality according to $k < (p-1)/2$.

Factor analysis has been extended to employ Gaussian mixture distributions for the factors in IFA [1] and the observation noise in SFA [8]. As in the standard factor analysis above, there is a degeneracy present in these systems. The covariance matrix of one state space component can be subsumed into the loading matrix and one state space noise mean vector can be absorbed by the observation noise mean. Therefore, the factors in SFA can be assumed to obey standard normal distribution. The effective number of free parameters in a factor analysis model with Gaussian mixture noise models is given by $2(M^{(x)}-1)k + kp + 2M^{(o)}p$ where $M^{(x)}$ and $M^{(o)}$ represent the number of mixture components in state and observation space respectively.

2.2 Generative Model of Factor Analysed HMM

Factor analysed hidden Markov model is a dynamic state space generalisation of a multiple component factor analysis system. The k-dimensional state vectors, $\boldsymbol{x}_t$, are generated by a standard diagonal covariance Gaussian mixture HMM. The p-dimensional observation vectors, $\boldsymbol{o}_t$, are generated by a multiple noise component factor analysis observation process. A generative model for FAHMM can be described by the following two equations

$$\boldsymbol{x}_t \sim \mathcal{M}^{hmm}, \qquad \mathcal{M}^{hmm} = \{a_{ij}, c_{jn}^{(x)}, \boldsymbol{\mu}_{jn}^{(x)}, \mathbf{\Sigma}_{jn}^{(x)}\} \tag{3}$$

$$\boldsymbol{o}_t = \boldsymbol{C}_t \boldsymbol{x}_t + \boldsymbol{v}_t, \qquad \boldsymbol{v}_t \sim \sum_m c_{jm}^{(o)} \mathcal{N}(\boldsymbol{\mu}_{jm}^{(o)}, \mathbf{\Sigma}_{jm}^{(o)}) \tag{4}$$

where the observation matrices, $\boldsymbol{C}_t$, may be dependent on the HMM state or tied over multiple states. The HMM state transition probabilities from state i to state j are represented by a_{ij} and the state and observation space mixture distributions are described by the mixture weights $\{c_{jn}^{(x)}, c_{jm}^{(o)}\}$, mean vectors $\{\boldsymbol{\mu}_{jn}^{(x)}, \boldsymbol{\mu}_{jm}^{(o)}\}$ and diagonal covariance matrices $\{\mathbf{\Sigma}_{jn}^{(x)}, \mathbf{\Sigma}_{jm}^{(o)}\}$.

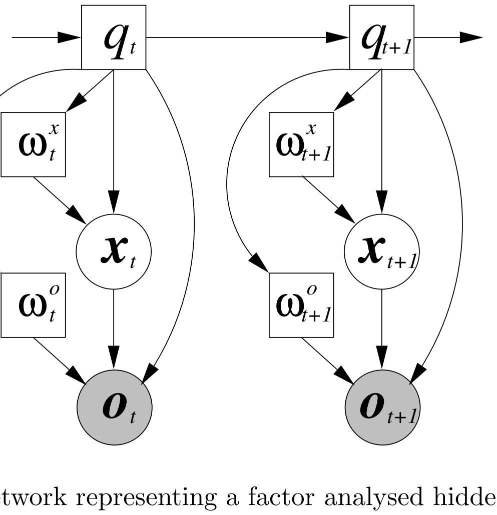

Figure 1: Bayesian network representing a factor analysed hidden Markov model.

Dynamic Bayesian networks (DBN) [6] are often presented in conjunction with generative models to illustrate the conditional independence assumptions made in a statistical model. A DBN describing a FAHMM is shown in Figure 1. The square nodes represent discrete random variables such as the HMM state $\{q_t\}$, and state and observation space mixture component indicators $\{\omega_t^x, \omega_t^o\}$. Continuous random variables such as the state vectors, $\boldsymbol{x}_t$, are represented by round nodes. Shaded nodes depict observable variables, $\boldsymbol{o}_t$, leaving all the other FAHMM variables hidden. A conditional independence assumption is made between variables that are not connected by directed arcs. The state conditional independence assumption between the output distributions of a standard HMM is also used in a FAHMM.

2.3 FAHMM Likelihood Calculation

An important aspect of any generative model is the complexity of the likelihood calculations. The generative model in Equations 3 and 4 can be represented by the following two Gaussian distributions

$$p(\boldsymbol{x}_t|q_t = j, \omega_t^x = n) = \mathcal{N}(\boldsymbol{x}_t; \boldsymbol{\mu}_{jn}^{(x)}, \boldsymbol{\Sigma}_{jn}^{(x)}) \tag{5}$$

$$p(\boldsymbol{o}_t|\boldsymbol{x}_t, q_t = j, \omega_t^o = m) = \mathcal{N}(\boldsymbol{o}_t; \boldsymbol{C}_j \boldsymbol{x}_t + \boldsymbol{\mu}_{jm}^{(o)}, \boldsymbol{\Sigma}_{jm}^{(o)}) \tag{6}$$

The distribution of an observation $\boldsymbol{o}_t$ given the state $q_t = j$, state space component $\omega_t^x = n$ and observation noise component $\omega_t^o = m$ can be obtained by integrating the state vector $\boldsymbol{x}_t$ out of the product of the above Gaussians. The resulting distribution is also a Gaussian and can be written as

$$b_{jmn}(\boldsymbol{o}_t) = p(\boldsymbol{o}_t|q_t = j, \omega_t^o = m, \omega_t^x = n) = \mathcal{N}\big(\boldsymbol{o}_t; \boldsymbol{\mu}_{jmn}, \boldsymbol{\Sigma}_{jmn}\big) \tag{7}$$

where

$$\boldsymbol{\mu}_{jmn} = \boldsymbol{C}_j \boldsymbol{\mu}_{jn}^{(x)} + \boldsymbol{\mu}_{jm}^{(o)} \tag{8}$$

$$\boldsymbol{\Sigma}_{jmn} = \boldsymbol{C}_j \boldsymbol{\Sigma}_{jn}^{(x)} \boldsymbol{C}_j' + \boldsymbol{\Sigma}_{jm}^{(o)} \tag{9}$$

The state distribution of a FAHMM state j can be viewed as an $M^{(o)}M^{(x)}$ component full covariance matrix GMM with mean vectors given by Equation 8 and covariance matrices given by Equation 9.

The likelihood calculation requires inverting $M^{(o)}M^{(x)}$ full p by p covariance matrices in Equation 9. If the amount of memory is not an issue, the inverses and the corresponding determinants can be computed prior to starting off with the training and recognition. However, this can rapidly become impractical for large system. A more memory efficient implementation requires the computation of the inverses and determinants on the fly. These can be efficiently obtained using the following equality for matrix inverses [11]

$$(\boldsymbol{C}_j \boldsymbol{\Sigma}_{jn}^{(x)} \boldsymbol{C}_j' + \boldsymbol{\Sigma}_{jm}^{(o)})^{-1} = \boldsymbol{\Sigma}_{jm}^{(o)-1} - \boldsymbol{\Sigma}_{jm}^{(o)-1} \boldsymbol{C}_j (\boldsymbol{C}_j' \boldsymbol{\Sigma}_{jm}^{(o)-1} \boldsymbol{C}_j + \boldsymbol{\Sigma}_{jn}^{(x)-1})^{-1} \boldsymbol{C}_j' \boldsymbol{\Sigma}_{jm}^{(o)-1} \tag{10}$$

where the inverses of the covariance matrices $\boldsymbol{\Sigma}_{jm}^{(o)}$ and $\boldsymbol{\Sigma}_{jn}^{(x)}$ are trivial to compute since they are diagonal. The full matrices, $\boldsymbol{C}_j' \boldsymbol{\Sigma}_{jm}^{(o)-1} \boldsymbol{C}_j + \boldsymbol{\Sigma}_{jn}^{(x)-1}$, to be inverted are only k by k matrices. This is dramatically faster than inverting full p by p matrices if $k \ll p$. The determinants needed in the likelihood calculations can be obtained using the following equality [11]

$$|\boldsymbol{C}_j \boldsymbol{\Sigma}_{jn}^{(x)} \boldsymbol{C}_j' + \boldsymbol{\Sigma}_{jm}^{(o)}| = |\boldsymbol{\Sigma}_{jm}^{(o)}||\boldsymbol{\Sigma}_{jn}^{(x)}||\boldsymbol{C}_j' \boldsymbol{\Sigma}_{jm}^{(o)-1} \boldsymbol{C}_j + \boldsymbol{\Sigma}_{jn}^{(x)-1}| \tag{11}$$

where again the determinants of the diagonal covariance matrices are trivial to compute and often the determinant of the k by k matrix is obtained as a by-product of its inverse; e.g., when using Cholesky decomposition. In a large system, a compromise has to made between precomputing of the inverse matrices and computing them on the fly. For example, caching of the inverses can be employed because some components are likely to be computed more often than others when pruning is used.

The Viterbi algorithm [19] can be used to produce the most likely state sequence the same way as with standard HMMs. The likelihood of an observation $\boldsymbol{o}_t$ given only the state $q_t = j$ can be obtained by marginalising the likelihood in Equation 7 as follows

$$b_j(\boldsymbol{o}_t) = p(\boldsymbol{o}_t|q_t = j) = \sum_{m=1}^{M^{(o)}} c_{jm}^{(o)} \sum_{n=1}^{M^{(x)}} c_{jn}^{(x)} b_{jmn}(\boldsymbol{o}_t) \tag{12}$$

Any Viterbi algorithm based decoder such as *token passing* algorithm [20] can be easily modified to support FAHMMs this way. The modifications to forward-backward algorithm are discussed in the training section below.

2.4 Optimising FAHMM Parameters

A maximum likelihood (ML) criterion is used to optimise the FAHMM parameters. It is also possible to find discriminative training scheme such as minimum classification error [18] but for this initial work only ML training is considered. In common with standard HMM training the expectation maximisation (EM) algorithm is used. The auxiliary function for FAHMMs can be written as

$$\mathcal{Q}(\mathcal{M}, \hat{\mathcal{M}}) = \sum_{\{Q_T\}} \int P(Q|\boldsymbol{O}, \mathcal{M}) p(\boldsymbol{X}|\boldsymbol{O}, Q, \mathcal{M}) \log p(\boldsymbol{O}, \boldsymbol{X}, Q|\hat{\mathcal{M}}) d\{\boldsymbol{X}_T\} \tag{13}$$

where $\{Q_T\}$ and $\{X_T\}$ represent all the possible discrete state and continuous state sequences of length T respectively. $\boldsymbol{O} = \boldsymbol{o}_1, \ldots, \boldsymbol{o}_T$ is a sequence of observation vectors and $\boldsymbol{X} = \boldsymbol{x}_1, \ldots, \boldsymbol{x}_T$ is a sequence of state vectors. $\mathcal{M}$ represents the set of current model parameters.

The sufficient statistics of the first term, $P(Q|\boldsymbol{O}, \mathcal{M})$, in the auxiliary function in Equation 13 can be obtained using the standard forward-backward algorithm with likelihoods given by Equation 12. For the state transition probability optimisation, two sets of sufficient statistics are needed, the posterior probabilities of being in state j at time t, $\gamma_j(t) = P(q_t = j|\boldsymbol{O}, \mathcal{M})$, and being in state i at time $t-1$ and in state j at time t, $\xi_{ij}(t) = P(q_{t-1} = i, q_t = j|\boldsymbol{O}, \mathcal{M})$. For the distribution parameter optimisation the component posteriors, $\gamma_{jmn}(t) = P(q_t = j, \omega_t^o = m, \omega_t^x = n|\boldsymbol{O}, \mathcal{M})$, have to be estimated. These can be obtained within the forward-backward algorithm as follows

$$\gamma_{jmn}(t) = \frac{1}{p(\boldsymbol{O})} c_{jm}^{(o)} c_{jn}^{(x)} b_{jmn}(\boldsymbol{o}_t) \sum_{i=1}^{N_s} a_{ij} \alpha_i(t-1) \beta_j(t) \tag{14}$$

where N_s is the number of HMM states in the model, $\alpha_i(t-1)$ is the standard forward variable representing the joint likelihood of being in state i at time $t-1$ and the partial observation sequence up to $t-1$, $p(q_{t-1} = i, \boldsymbol{o}_1, \ldots, \boldsymbol{o}_{t-1})$, and $\beta_j(t)$ is the standard backward variable corresponding to the posterior of the partial observation sequence from time $t+1$ to T given being in state j at time t, $p(\boldsymbol{o}_{t+1}, \ldots, \boldsymbol{o}_T|q_t = j)$.

The second term, $p(\boldsymbol{X}|\boldsymbol{O}, Q, \mathcal{M})$, in the auxiliary function in Equation 13 is the state vector distribution given the observation sequence and the discrete state sequence. Only the first and second-order statistics are required since the distributions are conditionally Gaussian given the state and the mixture components. As derived in [16] the sufficient statistics can be written as

$$\hat{\boldsymbol{x}}_{jmn}(t) = \boldsymbol{\mu}_{jn}^{(x)} + \boldsymbol{K}_{jmn}\big(\boldsymbol{o}_t - \boldsymbol{C}_j \boldsymbol{\mu}_{jn}^{(x)} - \boldsymbol{\mu}_{jm}^{(o)}\big) \tag{15}$$

$$\hat{\boldsymbol{R}}_{jmn}(t) = \boldsymbol{\Sigma}_{jn}^{(x)} - \boldsymbol{K}_{jmn} \boldsymbol{C}_j \boldsymbol{\Sigma}_{jn}^{(x)} + \hat{\boldsymbol{x}}_{jmn}(t) \hat{\boldsymbol{x}}'_{jmn}(t) \tag{16}$$

where $\boldsymbol{K}_{jmn} = \boldsymbol{\Sigma}_{jn}^{(x)} \boldsymbol{C}'_j \big(\boldsymbol{C}_j \boldsymbol{\Sigma}_{jn}^{(x)} \boldsymbol{C}'_j + \boldsymbol{\Sigma}_{jm}^{(o)}\big)^{-1}$. It should be noted that the matrix inverted in the equation for $\boldsymbol{K}_{jmn}$ is exactly the same as the inverse covariance matrix in Equation 10 and the same efficient algorithms presented in Section 2.3 apply.

Given the two sets of sufficient statistics above the model parameters can be optimised by solving a standard maximisation problem. The parameter update formulae for the underlying HMM parameters in FAHMM are very similar to standard HMM except the above state vector distribution statistics replace the observation sample moments. The state parameter update formulae can be written as

$$\hat{a}_{ij} = \frac{\sum_{t=2}^{T} \xi_{ij}(t)}{\sum_{t=2}^{T} \gamma_i(t-1)} \tag{17}$$

$$\hat{c}_{jn}^{x} = \frac{\sum_{t=1}^{T} \sum_{m=1}^{M^{(o)}} \gamma_{jmn}(t)}{\sum_{t=1}^{T} \gamma_j(t)} \tag{18}$$

$$\hat{\boldsymbol{\mu}}_{jn}^{(x)} = \frac{\sum_{t=1}^{T} \sum_{m=1}^{M^{(o)}} \gamma_{jmn}(t) \hat{\boldsymbol{x}}_{jmn}(t)}{\sum_{t=1}^{T} \sum_{m=1}^{M^{(o)}} \gamma_{jmn}(t)} \tag{19}$$

$$\hat{\boldsymbol{\Sigma}}_{jn}^{(x)} = \operatorname{diag}\Bigg(\frac{\sum_{t=1}^{T} \sum_{m=1}^{M^{(o)}} \gamma_{jmn}(t) \hat{\boldsymbol{R}}_{jmn}(t)}{\sum_{t=1}^{T} \sum_{m=1}^{M^{(o)}} \gamma_{jmn}(t)} - \hat{\boldsymbol{\mu}}_{jn}^{(x)} \hat{\boldsymbol{\mu}}_{jn}^{(x)\prime} \Bigg) \tag{20}$$

where diag(·) sets all the off-diagonal elements of the matrix argument to zeros. The cross-products of the new state space mean vectors and the first-order accumulates have been simplified in Equation 20. This can only be done if the mean vectors are updated during the same iteration, and the covariance matrices and the mean vectors are tied on the same level.

The new observation matrix, $\hat{\boldsymbol{C}}_j$, has to be optimised row by row as in SFA [8]. The scheme adopted in this paper follows closely the maximum likelihood linear regression (MLLR) transform matrix optimisation [3]. The lth row vector $\hat{\boldsymbol{c}}_{jl}$ of the new observation matrix can be written as

$$\hat{\boldsymbol{c}}_{jl} = \boldsymbol{k}_{jl}' \boldsymbol{G}_{jl}^{-1} \tag{21}$$

where the k by k matrices $\boldsymbol{G}_{jl}$ and the k dimensional column vectors $\boldsymbol{k}_{jl}$ are defined as follows

$$\boldsymbol{G}_{jl} = \sum_{m=1}^{M^{(o)}} \frac{1}{\sigma_{jml}^{(o)2}} \sum_{t=1}^{T} \sum_{n=1}^{M^{(x)}} \gamma_{jmn}(t) \hat{\boldsymbol{R}}_{jmn}(t) \tag{22}$$

$$\boldsymbol{k}_{jl} = \sum_{m=1}^{M^{(o)}} \frac{1}{\sigma_{jml}^{(o)2}} \sum_{t=1}^{T} \sum_{n=1}^{M^{(x)}} \gamma_{jmn}(t) \big(o_{tl} - \mu_{jml}^{(o)}\big) \hat{\boldsymbol{x}}_{jmn}(t) \tag{23}$$

where $\sigma_{jml}^{(o)2}$ is the lth diagonal element of the observation covariance matrix $\boldsymbol{\Sigma}_{jm}^{(o)}$, o_{tl} and $\mu_{jml}^{(o)}$ are the lth elements of the current observation and the observation noise mean vectors, respectively.

Table 1: Standard systems related to FAHMMs.

system	relation to FAHMMs
HMM	$M^{(x)} = 0$
SFA	$M^{(x)} = 1$
dynamic IFA	$M^{(o)} = 1$
STC	$k = p$ and $\boldsymbol{v}_t = \mathbf{0}$
Covariance EMLLT	$k > p$ and $\boldsymbol{v}_t = \mathbf{0}$

Given the new observation matrix, the observation noise parameters can be optimised using the following formulae

$$\hat{c}_{jm}^{(o)} = \frac{\sum_{t=1}^{T} \sum_{n=1}^{M^{(x)}} \gamma_{jmn}(t)}{\sum_{t=1}^{T} \gamma_j(t)} \tag{24}$$

$$\hat{\boldsymbol{\mu}}_{jm}^{(o)} = \frac{\sum_{t=1}^{T} \sum_{n=1}^{M^{(x)}} \gamma_{jmn}(t)\big(\boldsymbol{o}_t - \hat{\boldsymbol{C}}_j \hat{\boldsymbol{x}}_{jmn}(t)\big)}{\sum_{t=1}^{T} \sum_{n=1}^{M^{(x)}} \gamma_{jmn}(t)} \tag{25}$$

$$\begin{aligned}\hat{\boldsymbol{\Sigma}}_{jm}^{(o)} = {} & \frac{1}{\sum_{t=1}^{T} \sum_{n=1}^{M^{(x)}} \gamma_{jmn}(t)} \sum_{t=1}^{T} \sum_{n=1}^{M^{(x)}} \gamma_{jmn}(t) \mathrm{diag}\Big(\boldsymbol{o}_t \boldsymbol{o}_t' - \Big[\, \hat{\boldsymbol{C}}_j \;\; \hat{\boldsymbol{\mu}}_{jm}^{(o)} \Big] \big[\, \boldsymbol{o}_t \hat{\boldsymbol{x}}_{jmn}'(t) \;\; \boldsymbol{o}_t \,\big]' \\ & - \big[\, \boldsymbol{o}_t \hat{\boldsymbol{x}}_{jmn}'(t) \;\; \boldsymbol{o}_t \,\big] \Big[\, \hat{\boldsymbol{C}}_j \;\; \hat{\boldsymbol{\mu}}_{jm}^{(o)} \Big]' + \Big[\, \hat{\boldsymbol{C}}_j \;\; \hat{\boldsymbol{\mu}}_{jm}^{(o)} \Big] \begin{bmatrix} \hat{\boldsymbol{R}}_{jmn}(t) & \hat{\boldsymbol{x}}_{jmn}(t) \\ \hat{\boldsymbol{x}}_{jmn}'(t) & 1 \end{bmatrix} \Big[\, \hat{\boldsymbol{C}}_j \;\; \hat{\boldsymbol{\mu}}_{jm}^{(o)} \Big]' \Big)\end{aligned} \tag{26}$$

Detailed derivation of the parameter optimisation can be found in [16].

A direct implementation of the training algorithm is inefficient due to the heavy matrix computations required to obtain the state vector statistics. An efficient two level implementation of the training algorithm is presented in Section 3.4. Obviously, there is no need to compute the off-diagonal elements of the new covariance matrices in Equations 20 and 26.

2.5 Standard Systems Related to FAHMMs

A number of standard systems can easily be related to FAHMMs. Since the FAHMM training algorithm described above is based on EM algorithm, it is only applicable if there is observation noise. Some of the related systems have the observation noise set to zero which means that different optimisation methods have to be used. The related systems are presented in Table 1 and their properties are further discussed below.

- By setting the number of state space mixture components to zero, $M^{(x)} = 0$, FAHMM reduces to a standard diagonal covariance Gaussian mixture HMM. The observation noise acts as the state conditional output distribution and the observation matrix is made redundant because no state vectors will be generated.

- By setting the number of state space mixture components to one, $M^{(x)} = 1$, FAHMM corresponds to SFA [8]. Even though the state space distribution parameters are modelled explicitly, there are effectively an equal number of free parameters in this FAHMM and SFA which assumes the state distribution with zero mean and identity covariance.
- By setting the number of observation space distribution components to one, $M^{(o)} = 1$, FAHMM corresponds to a dynamic version of IFA [1]. The only difference to the standard IFA is the independent state vector element (factor) assumption which would require a multiple stream (factorial) HMM [7] with one dimensional streams in the state space. Effectively multiple streams can model a larger number of distributions but the independence assumption is relaxed in this FAHMM assuming uncorrelated factors instead of independent.
- By setting the observation noise to zero, $\boldsymbol{v}_t = \mathbf{0}$, and setting the state space dimensionality equal to the observation space dimensionality, $k = p$, FAHMM reduces to a semi-tied covariance matrix HMM. The only difference to the original STC model in [4] is that the mean vectors are also transformed in FAHMM.
- By setting the observation noise to zero, $\boldsymbol{v}_t = \mathbf{0}$, and setting the state space dimensionality greater than the observation space dimensionality, $k > p$, FAHMM becomes a covariance version of extended maximum likelihood linear transformation (EMLLT) [15] scheme. FAHMM is based on a generative model which requires every state space covariance matrix being a valid covariance matrix; i.e. positive semi-definite. EMLLT is an inverse covariance model where the parameter matrix, Λ_j, corresponding to the FAHMM state vector variances, may also have non-positive values as long as the effective covariance matrices will be positive semi-definite.

3 Implementation Issues

When factor analysed HMMs are applied for large vocabulary continuous speech recognition (LVCSR) there are a number of efficiency issues that must be addressed. As EM training is being used, an appropriate initialisation scheme is essential. Furthermore, in common with standard LVCSR systems, parameter tying may be used extensively. In addition, there is a large amount of matrix operations that need to be computed. Issues with numerical accuracy have to be considered. Finally, as there are two sets of hidden variables in FAHMMs, an efficient two level training scheme is presented.

3.1 Initialisation

One major issue with maximum likelihood training is that there are a number of local maxima. An appropriate initialisation scheme may improve the chances of finding a good solution. A sensible starting point is to use a standard HMM. A single Gaussian HMM can be converted to an equivalent FAHMM as follows

$$\boldsymbol{\mu}_j^{(x)} = \boldsymbol{\mu}_{j[1:k]} \tag{27}$$

$$\boldsymbol{\Sigma}_j^{(x)} = \frac{1}{2}\boldsymbol{\Sigma}_{j[1:k]} \tag{28}$$

$$\boldsymbol{C}_j = \boldsymbol{I} \tag{29}$$

$$\boldsymbol{\mu}_j^{(o)} = \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\mu}_{j[k+1:p]} \end{bmatrix} \tag{30}$$

$$\boldsymbol{\Sigma}_j^{(o)} = \begin{bmatrix} \frac{1}{2}\boldsymbol{\Sigma}_{j[1:k]} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_{j[k+1:p]} \end{bmatrix} \tag{31}$$

where $\boldsymbol{\mu}_{j[1:k]}$ represent the first k elements of the mean vector and $\boldsymbol{\Sigma}_{j[1:k]}$ is the upper left k by k submatrix of the covariance matrix associated with state j of the initial HMM.

The above initialisation scheme assumes that the first k feature vector elements are the most significant. In the experiments, the state space dimensionality was chosen to be $k = 13$ which corresponds to the static parameters in a standard 39-dimensional feature vector. Alternative feature selection techniques such as Fisher ratio or recognition based can also be used within this initialisation scheme.

3.2 Parameter Sharing

As discussed in Section 2.1, the order of number of free parameters per state in a FAHMM is the same as in a factor analysis model with Gaussian mixture distributions. Table 2 summarises the numbers of free parameters for HMM and FAHMM states. The dimensionality of the state space, k, and the number of observation noise components, $M^{(o)}$, have the largest influence on the complexity of FAHMMs.

Table 2: Order of number of free parameters using $M^{(x)}$ state space components, $M^{(o)}$ observation noise components and no sharing of individual FAHMM parameters.

System	Free Parameters
HMM ($M^{(x)} = 0$)	$2M^{(o)}p$
FAHMM ($M^{(x)} > 0$)	$2(M^{(x)} - 1)k + pk + 2M^{(o)}p$

When context-dependent HMM systems are trained the selection of the model set is often based on decision-tree clustering [2]. However, implementing decision-tree clustering for FAHMMs is very complicated. Since the clustering is not optimal [14], decision-tree clustered HMM models may be considered as a sufficiently good starting point for FAHMM initialisation. The initialisation of the context-dependent models can be done the same way as using standard context-independent HMMs described above.

In addition to state clustering, it is sometimes useful to share some of the individual FAHMM parameters. It is possible to tie any number of parameters between arbitrary number of models at various levels of the model. For example, the observation matrix can be shared globally or between classes of states as in semi-tied covariance HMMs [4]. A global observation noise distribution could represent a stationary noise environment corrupting all the speech data. Implementing an arbitrary tying scheme is closely related to standard HMM systems [20]. The sufficient statistics required for the tied parameter are accumulated over the entire class sharing it before updating. If the mean vectors and the covariance matrices of the state space noise are tied on a different level, all the cross terms between the first-order accumulates and the updated mean vectors have to be used in the covariance matrix update formula in Equation 20.

3.3 Numerical Accuracy

The matrix inversion described in Section 2.3 and the parameter estimation require many matrix computations. Numerical accuracy may become an issue due to the vast amount of sums of products. In the experiments it was found that double precision had to be used in all the intermediate operations. Nevertheless, single precision was used to store the accumulates and model parameters due to the memory usage.

A lot of training data is required to get reliable estimates for the covariance matrices in a large vocabulary speech recognition system. Sometimes the new variance elements may become too small which causes trouble in the likelihood calculations. To avoid problems with FAHMMs the full covariance matrices in Equation 9 must be guaranteed to be non-singular. The matrix $\boldsymbol{C}_j \boldsymbol{\Sigma}_{jn}^{(x)} \boldsymbol{C}_j'$ is at most rank k provided the state space variances are valid. Therefore, it is essential that the observation noise variances are floored properly. In the experiments it was found that the flooring scheme usually implemented in HMM systems [20] is sufficient for the observation variances in FAHMMs. With very large model sets the new estimates for the state space variances may become negative due to insufficient data for the component. In the experiments such variance elements were not updated.

3.4 Efficient Two Level Training

To increase the speed of training, a two level algorithm is adopted. The component specific first and second-order statistics form the sufficient statistics required in the parameter estimation described in Section 2.4. This can be verified by substituting the state vector statistics, $\hat{\boldsymbol{x}}_{jmn}(t)$ and $\hat{\boldsymbol{R}}_{jmn}(t)$, in Equations 15 and 16 into the update Equations 17-26. The sufficient statistics can be written as

$$\tilde{\gamma}_{jmn} = \sum_{t=1}^{T} \gamma_{jmn}(t) \tag{32}$$

$$\tilde{\boldsymbol{\mu}}_{jmn} = \sum_{t=1}^{T} \gamma_{jmn}(t)\boldsymbol{o}_t \tag{33}$$

$$\tilde{\boldsymbol{R}}_{jmn} = \sum_{t=1}^{T} \gamma_{jmn}(t)\boldsymbol{o}_t\boldsymbol{o}_t' \tag{34}$$

Given these accumulates and the current model parameters, $\mathcal{M}$, the required accumulates for the new parameters can be estimated. Since the estimated state vector statistics depend on both the data accumulates and the current model parameters an extra level of iterations can be introduced. After updating the model parameters, new state vector distribution given the old data accumulates and the new model parameters can be estimated. These within iterations are guaranteed to increase the log-likelihood of the data. Figure 2 illustrates the increase of the auxiliary function values during three full iterations, 10 within iterations each.

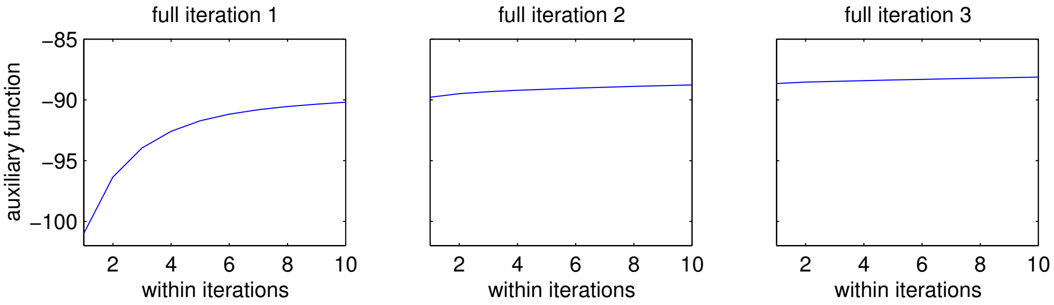

Figure 2: Auxiliary function values against within iterations during 3 full iterations.

The efficient training algorithm can be summarised as follows

1. Collect the data statistics using forward-backward algorithm;
2. Estimate the state vector distribution $p(\boldsymbol{x}_t|j, m, n, \boldsymbol{O}, \mathcal{M})$;
3. Estimate new model parameters $\hat{\mathcal{M}}$;
4. If the auxiliary function value has not converged go to step 2 and update the parameters $\hat{\mathcal{M}} \rightarrow \mathcal{M}$;
5. If the average log-likelihood of the data has not converged go to step 1 and update the parameters $\hat{\mathcal{M}} \rightarrow \mathcal{M}$.

The within iterations decrease the number of full iterations needed in training. The overall training time becomes shorter because less time has to be spent collecting the data accumulates. The average log-likelihoods of the training data against the number of full iterations are illustrated in Figure 3. Four iterations of embedded training were first applied to the baseline HMM. The FAHMM system with $k = 13$ was initialised as described in Section 3.1. Both, one level training and more efficient two level training with 10 within iterations, were used and the corresponding log-likelihoods are shown in the figure.

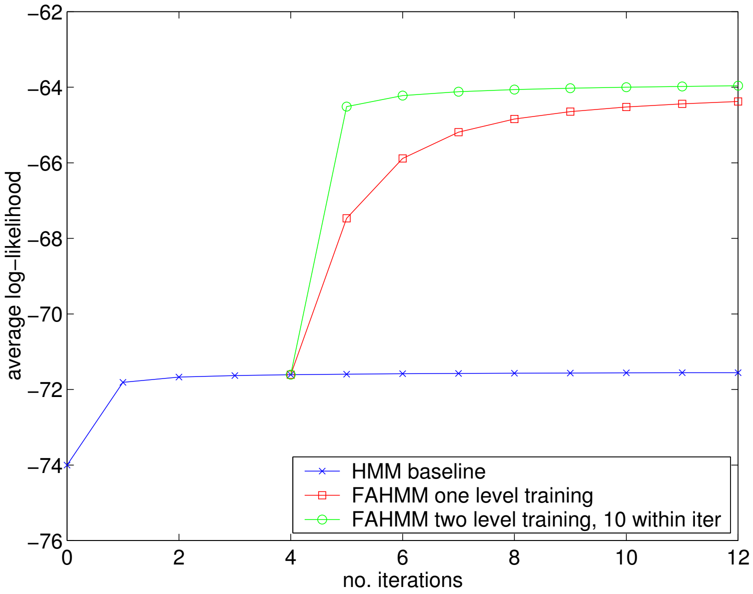

Figure 3: Log-likelihood values against full iterations for baseline HMM and an untied FAHMM with $k = 13$. One level training and more efficient two level training with 10 within iterations were used.

4 Results

The results in this section are presented to illustrate the performance of some FAHMM configurations on medium to large speech recognition tasks. Only a small number of possible configurations have been examined and the configurations have not been chosen in accordance with any optimal criterion. The aim is to show how FAHMMs perform with some possible configurations as well as compare them to standard semi-tied systems.

4.1 Resource Management

For initial experiments, a standard medium size speech recognition task, the ARPA Resource Management (RM) task, was used. Following the HTK "RM Recipe" [20], the baseline system was trained starting from a flat start single mixture monophone system. 3990 sentences {`train+dev_aug`} were used for training. After four iterations of embedded training, the monophone models were cloned to produce a single mixture cross word triphone system. These initial triphone models were trained with two iterations of embedded training after which a decision-tree clustering was applied to produce a tied state triphone system. This system was used as an initial model set for standard HMM, STC and FAHMM systems. 1200 sentences {`feb89+oct89+feb91+sep92`} with a simple word-pair grammar were used for evaluation.

The baseline HMM system was produced by standard mixing up procedure [20] using four iterations of embedded training per mixture configuration until no decrease in the word error rate

Table 3: Order of number of free parameters and word error rates in Resource Management baseline HMM, global full transform semi-tied HMM and global full observation matrix FAHMM systems.

$M^{(o)}$	1	2	3	4	5	6
HMM	78 7.79%	156 6.68%	234 5.05%	312 4.32%	390 4.09%	468 3.99%
STC	78 7.06%	156 5.30%	234 4.32%	312 3.93%	390 3.83%	468 3.85%
GFAHMM	117 6.52%	195 4.88%	273 4.28%	351 3.94%	429 3.68%	507 3.77%

was achieved. The word error rates with the order of number of free parameters per state up to 6 components are presented on the first row in Table 3. The best performance was 3.76% obtained with 10 mixture components. The order of number of free parameters in the best baseline system was 780 per state. As an additional baseline a global semi-tied HMM system was built. The single mixture baseline HMM system was converted to the STC system by adding a global full 39 by 39 identity transformation matrix. The number of free parameters increased globally by 1521 compared to the baseline HMM system. As discussed in Section 2.5, this system corresponds to a FAHMM with state space dimensionality $k = 39$ and zero observation noise. The number of mixture components was increased by the mixing up procedure. 9 full iterations of embedded training were used with 20 within iterations and 20 row by row transform iterations [4]. The results are presented on the second row in Table 3. The best semi-tied performance was 3.83% obtained with 5 mixture components. As usual, the performance when using STC is better with fewer mixture components. However, increasing the number of mixture components in a standard HMM system can be seen to model the intra-frame correlation better.

A FAHMM system with state space dimensionality $k = 39$ and a global observation matrix was built for comparison with the STC system above. The global full 39 by 39 observation matrix was initialised to an identity matrix and the variance elements of the single mixture baseline HMM system were evenly distributed among the observation and state space variances as discussed in Section 3.1. The number of state space components was fixed to one and the observation space components were increased by the mixing up procedure. The system corresponds to a global full loading matrix SFA with non-identity state space covariance matrices. The number of additional free parameters per state was 39 due to the state space covariance matrices and 1521 globally due to the loading matrix. 9 full iterations of embedded training were used along with 20 within iterations. The results are presented on the third row in Table 3. The best performance, 3.68%, was achieved with 5 mixture components. The difference in the number of free parameters between the best baseline and the best FAHMM system was 351 per state. Compared to the STC system, FAHMM has only 39 additional free parameters per state. The FAHMM system provides a relative word error rate reduction of 4% to the STC system.

These initial experiments show the relationship between FAHMMs and STC in practise. However, the training and recognition using full state space FAHMMs is a lot more complex than using global STC even though the observation matrix is shared globally. Since STC does not have observation noise, the global transform can be applied to the feature vectors in advance and full covariance matrices are not needed in the likelihood calculation. The performance of FAHMMs using lower dimensional state space is investigated in the experiments below.

4.2 Minitrain

The Minitrain 1998 Hub5 HTK system [10] was used as a larger speech recognition task. The baseline was a gender independent decision-tree clustered tied state cross word triphone Gaussian mixture HMM system. The 18 hour Minitrain set containing 398 conversation sides of Switchboard-1 corpus and defined by BBN [13] was used as the acoustic training data. The test data set was the

Table 4: Order of number of free parameters and word error rates in Minitrain FAHMM system with $k = 13$.

$M^{(o)}$ $M^{(x)}$	1	2	4
1	585 53.3%	663 51.7%	819 51.0%
2	611 53.3%	689 51.4%	845 51.3%
4	663 53.0%	741 51.0%	897 50.9%
6	715 52.8%	793 50.7%	949 51.0%
8	767 52.6%	845 51.0%	▨

subset of the 1997 Hub5 evaluation set used in [10]. The best performance, 51.0%, was achieved with 12 components which corresponds to 936 parameters per state. Mixing up was not continued further since the performance started degrading after 12 components.

FAHMM system with state space dimensionality 13 was built starting from the single component baseline system. An individual 39 by 13 observation matrix initialised as an identity matrix was attached to each state. The first 13 variance elements of the HMM models were evenly distributed among the observation and state space variances as discussed in Section 3.1. The mixing up was started from the single component baseline system increasing the number of state space components while fixing the number of observation space components. The number of observation space components of single state space component system was then increased and fixed until all the state space components were mixed up and so on. The results up to the best performance per column are presented in Table 4. As discussed in Section 2.5, the row corresponding to $M^{(x)} = 1$ is related to a SFA system and the first column corresponding to $M^{(o)} = 1$ is related to a dynamic IFA without the independent element assumption. The same performance as the best baseline HMM system was achieved using FAHMMs with 2 observation and 4 state space components. The difference in the number of free parameters per state is considerable, 195. The best FAHMM performance, 50.7%, was also achieved using fewer free parameters than the best baseline system though it is not statistically significant.

These experiments show how the FAHMM system performs in a large speech recognition task when low dimensional state space is used. As the state space dimensionality and the initialisation were selected based on intuition, the results seem promising. Choosing the state space dimensionality automatically is very challenging problem and it can be expected to improve the performance. Complexity control and more elaborate initialisation schemes will be studied in the future.

4.3 Switchboard 68 Hours

For the experiments performed in this section, a 68 hour subset of the Switchboard (Hub5) acoustic training data set was used. 862 sides of the Switchboard-1 and 92 sides of the Call Home English were used. The set is described as "h5train00sub" in [9]. As with Minitrain, the baseline was a gender independent decision-tree clustered tied state cross word triphone Gaussian mixture HMM system. The 1998 Switchboard evaluation data set was used for testing. The baseline HMM system word error rates with the order of number of free parameters are presented on the first row in Table 5. The performance of the baseline system was going up with increasing number of components until 30 components were used. However, the number of free parameters in such a system is impractically high, 2340 per state. 14 component system seems to be a reasonable compromise because the word error rate, 46.5%, seems to be a local stationary point. As an additional baseline a global semi-tied covariance HMM system was trained the same way as in the

Table 5: Order of number of free parameters and word error rates in Hub5 68 hour baseline HMM, global full transform semi-tied HMM, SFA and global observation matrix SFA systems with $k = 13$.

$M^{(o)}$	1	2	4	6	8	10	12	14	16
HMM	78 55.1%	156 52.4%	312 49.6%	468 48.5%	624 47.7%	780 47.2%	936 46.7%	1092 46.5%	1248 46.5%
STC	78 54.3%	156 50.4%	312 48.4%	468 47.3%	624 46.7%	780 46.3%	936 46.3%	1092 45.8%	1248 45.7%
SFA	585 49.1%	663 48.0%	819 47.2%	975 46.6%	1131 46.3%	1287 46.4%	1443 46.0%	1599 45.8%	1755 45.9%
GSFA	91 55.2%	169 52.1%	325 49.4%	481 48.4%	637 47.4%	793 46.9%	949 46.7%	1105 46.4%	1261 46.1%

RM experiments. The results for the STC system are presented on the second row in Table 5. The best performance, 45.7%, in the STC system was obtained using 16 components.

FAHMM system with state space dimensionality $k = 13$ was built starting from the single component baseline system. An individual 39 by 13 observation matrix initialised as an identity matrix was attached to every state. The first 13 variance elements of the HMM models were evenly distributed among the observation and state space variances as discussed in Section 3.1. The mixing up procedure for both state space and observation noise components was carried out as in the Minitrain experiments above until no further gains were achieved. Unfortunately, filling up a complete table was not feasible since the training time grows too long when increasing the effective number of full covariance matrices over 16. The most interesting results here are achieved using only one state space component which corresponds to the SFA. The results are presented on the third row in Table 5. It is worth noting that the best baseline performance is achieved using FAHMMs with considerably fewer free parameters. The 12 component baseline performance is also achieved by using FAHMMs with fewer parameters - namely 2 observation and 8 state space components which correspond to 845 free parameters per state.

To see how the tying of parameters influence the results, a FAHMM system with state space dimensionality $k = 13$ and a global observation matrix $\boldsymbol{C}$ was built starting from the single component baseline system as usual. The observation matrix was initialised as a 39 by 13 identity matrix and the variance elements of the HMM models were evenly distributed among the observation and state space variances as discussed in Section 3.1. As before, the completion of the table was not feasible due to the number of effective full covariance components in the system. The single state space component system appears to be the most interesting but no conclusions about the untested configurations can be made. The results for the single state space component system are presented on the fourth row in Table 5. The 12 observation component system achieved the same performance as the 12 component baseline system but further increasing the number of components proved to be quite interesting. The 16 observation space component system achieved the same performance as 24 component baseline system with 611 free parameters fewer. It should also be noted that the STC system outperforms these configurations of FAHMMs in this task. Further experiments with different evaluation data should be conducted.

These experiments show the current implementation of the FAHMM system has its limits when the task size is increased from the Minitrain task. The same arguments about the initialisation and chosen state space dimensionality as after the Minitrain experiments can be made. The main contribution of these experiments was to show how an equivalent performance to HMMs can be achieved using fewer model parameters in a large speech recognition task with simple configurations of FAHMMs.

5 Conclusions

This paper has introduced the factor analysed HMM which is a general form of acoustic model. It combines a standard Gaussian mixture HMM with a shared and independent factor analysis models. FAHMM provides a better model for the correlation between the feature vector elements than a standard diagonal covariance matrix HMM. It can be viewed as a compromise between diagonal and full covariance matrix systems. In addition, FAHMM can be viewed as a general state space model which allows a number of subspaces to be explored. A variety of configurations and sharing schemes, some of which correspond to standard systems, have been investigated. The estimation using EM algorithm is presented along with several schemes to improve both, time and memory efficiency. The speech recognition performance is evaluated in experiments using medium to large vocabulary continuous speech recognition tasks. The results show that equivalent or slightly better performance to standard diagonal covariance Gaussian mixture HMMs can be achieved with considerably fewer model parameters.

Due to the flexibility of FAHMMs a large number of configurations can be explored. Different techniques to optimally choose the configuration have to be investigated. Another important question is how to choose an optimal state space dimensionality. The model complexity has become a standard problem in speech recognition and machine learning over the recent years. Yet, a successful scheme for speech recognition systems has not been published. Current complexity controls are derived from Bayesian schemes based on correctly modelling some held-out data. However, it is well known that the models giving highest log-likelihood for some data do not automatically have better recognition performance on unseen data. The future work also involves the complexity control in FAHMM based systems.

6 Acknowledgements

A-V.I. Rosti is funded by an EPSRC studentship and Tampere Graduate School in Information Science and Engineering. He received additional support from the Finnish Cultural Foundation. This work made use of equipment kindly supplied by IBM under an SUR award.

References

[1] H. Attias. Independent factor analysis. *Neural Computation*, 11(4):803–851, 1999.

[2] L.R. Bahl, P.V. de Souza, P.S. Gopalkrishnan, D. Nahamoo, and M.A. Picheny. Context dependent modelling of phones in continuous speech using decision trees. In *Proceedings DARPA Speech and Natural Language Processing Workshop*, pages 264–270, 1991.

[3] M.J.F. Gales. Maximum likelihood linear transformations for HMM-based speech recognition. *Computer Speech and Language*, 12(2):75–98, 1998.

[4] M.J.F. Gales. Semi-tied covariance matrices for hidden Markov models. *IEEE Transactions on Speech and Audio Processing*, 7(3):272–281, 1999.

[5] M.J.F. Gales. Maximum likelihood multiple subspace projections for hidden Markov models. *IEEE Transactions on Speech and Audio Processing*, 10(2):37–47, 2002.

[6] Z. Ghahramani. Learning dynamic Bayesian networks. In C.L. Giles and M. Gori, editors, *Adaptive Processing of Sequences and Data Structures*, volume 1387 of *Lecture Notes in Computer Science*, pages 168–197. Springer, 1998.

[7] Z. Ghahramani and M.I. Jordan. Factorial hidden Markov models. *Machine Learning*, 29:245–273, 1997.

[8] R.A. Gopinath, B. Ramabhadran, and S. Dharanipragada. Factor analysis invariant to linear transformations of data. In *Proceedings International Conference on Speech and Language Processing*, pages 397–400, 1998.

[9] T. Hain, P.C. Woodland, G. Evermann, and D. Povey. The CU-HTK March 2000 HUB5E transcription system. In *Proceedings Speech Transcription Workshop*, 2000.

[10] T. Hain, P.C. Woodland, T.R. Niesler, and E.W.D. Whittaker. The 1998 HTK system for transcription of conversational telephone speech. In *Proceedings International Conference on Acoustics, Speech and Signal Processing*, pages 57–60, 1999.

[11] D.A. Harville. *Matrix Algebra from a Statistician's Perspective*. Springer, 1997.

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

[13] D.R.H. Miller and J.W. McDonough. BBN 1997 Acoustic Modelling. Presented at Conversational Speech Recognition Workshop DARPA Hub-5E Evaluation, May 1997.

[14] H.J. Nock, M.J.F. Gales, and S.J. Young. A comparative study of methods for phonetic decision-tree state clustering. In *Proceedings European Conference on Speech Communication and Technology*, pages 111–114, 1997.

[15] P.A. Olsen and R.A. Gopinath. Modeling inverse covariance matrices by basis expansion. In *Proceedings International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 945–948, 2002.

[16] A-V.I. Rosti and M.J.F. Gales. Generalised linear Gaussian models. Technical Report CUED/F-INFENG/TR.420, Cambridge University Engineering Department, 2001. Available via anonymous ftp from `ftp://svr-ftp.eng.cam.ac.uk/pub/reports/`.

[17] S. Roweis and Z. Ghahramani. A unifying review of linear Gaussian models. *Neural Computation*, 11(2):305–345, 1999.

[18] L. Saul and M. Rahim. Maximum likelihood and minimum classification error factor analysis for automatic speech recognition. *IEEE Transactions on Speech and Audio Processing*, 8(2):115–125, 1999.

[19] A.J. Viterbi. Error bounds for convolutional codes and an asymptotically optimal decoding algorithm. *IEEE Transactions on Information Theory*, IT-13:260–269, 1967.

[20] S. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev, and P. Woodland. *The HTK Book (for HTK Version 3.0)*. Cambridge University, 2000.