Robust Sequential Data Modeling Using an Outlier Tolerant Hidden Markov Model

Sotirios Chatzis, *Member, IEEE*, Dimitrios Kosmopoulos, *Member, IEEE,* and Theodora Varvarigou, *Member, IEEE*

Abstract

Hidden Markov (chain) models using finite Gaussian mixture models as their hidden state distributions have been successfully applied in sequential data modeling and classification applications. Nevertheless, Gaussian mixture models are well-known to be highly intolerant to the presence of untypical data within the fitting data sets used for their estimation. Finite Student's-t mixture models have recently emerged as a heavier-tailed, robust alternative to Gaussian mixture models, overcoming these hurdles. To exploit these merits of Student's-t mixture models in the context of a sequential data modeling setting, we introduce in this paper a novel hidden Markov model where the hidden state distributions are considered to be finite mixtures of multivariate Student's-t densities. We derive an algorithm for the model parameters estimation under a maximum likelihood framework, assuming full, diagonal, and factor analyzed covariance matrices. The advantages of the proposed model compared to conventional approaches are experimentally demonstrated through a series of sequential data modeling applications.

I. Introduction

The hidden Markov model (HMM) is increasingly being adopted in applications since it provides a convenient way of modeling observations appearing in a sequential manner and tending to cluster or to alternate between different possible components (subpopulations) [1]. The observation emission densities associated with each hidden state of a CHMM must be capable of approximating arbitrarily complex probability density functions. Finite Gaussian mixture models (GMMs) are the most common

selection of emission distribution models in the CHMM literature, yielding the so-called Gaussian HMMs (GHMMs) [2]. The vast popularity of GHMMs stems from the well-known capability of GMMs to successfully approximate unknown random distributions, including distributions with multiple modes, while also providing a simple and computationally efficient maximum-likelihood (ML) model fitting framework, by means of the expectation-maximization (EM) algorithm [3]. Nevertheless, GMMs do also suffer from a significant drawback concerning their parameters estimation procedure, which is well-known that can be adversely affected by the presence of outliers in the data sets used for the model fitting. Hence, when outliers are present in the available fitting data sets (as it is usually the case in real-world applications), GMMs tend to require excessively high numbers of mixture components to capture the long tails of the approximated distributions (corresponding to the existing outliers), so as to retain their pattern recognition effectiveness. As a consequence of the induced model size increase, the computational efficiency of the trained models deteriorates significantly, while high requirements are also imposed in the size of the available training data sets, so as to guarantee the dependability of the model fitting procedure.

As a solution for the amelioration of these drawbacks in the static setting, the finite Student's-t mixture model (SMM) has been proposed in [4] as a highly tolerant to outliers alternative to GMMs. Finite mixtures of the longer-tailed multivariate Student's-t distribution provide a much more robust approach to the fitting of GMMs, as training observations that are atypical of a mixture component density are given reduced weight in the calculation of its parameters, under a model-inherent, soundly-founded statistical procedure. The significant tolerance of SMMs to untypical training data has been experimentally depicted in many recent articles (see e.g. [4], [5], [6], [7]), where it has been shown that SMMs can model sufficiently well the hidden patterns of the data under examination, even under the presence of significant proportions of outliers, cases where the GMMs either fail or demand a considerably large model to be fitted, thus resulting in a severe deterioration in terms of computational efficiency, as already explained.

It is a natural consequence of the outlier intolerance of GMMs that CHMMs using GMMs as their emission densities (GHMMs) do also suffer from the same outlier intolerance related issues. In modern

CHMM literature, various efforts have been made towards the attenuation of these shortcomings of GHMMs. For example, in [8], a selective training method is proposed; in [9], a top-down selective attention filter is applied on maximum-likelihood GHMM training, for robust speech recognition; in [10], a maximum confidence GHMM for robust two-dimensional pattern classification is proposed; in [11], a large margin HMM (LM-HMM) is proposed. However, the proposed have many shortcomings, among which we might mention the heuristic nature of their majority, as well as the application-specific nature of many of them. To address these issues, in this paper, we exploit the outlier tolerance merits of SMMs in the context of sequential data modeling techniques using continuous hidden Markov chain models, by proposing a novel CHMM where the hidden state observation emission distributions are modeled using finite mixtures of multivariate Student's-t densities. This way, we formulate a novel outlier-tolerant HMM, providing an effective, non-heuristic, application-independent means for robust sequential data modeling.

The remainder of this paper is organized as follows. In Section II, the proposed Student's-t HMM (SHMM) is formulated. In Section III, a multiple token treatment of the SHMM under the ML framework is conducted, using the EM algorithm. In Section IV, an effective covariance modeling technique for the SHMM based on factor analysis is proposed, to allow for the reduction of the SHMM parameters when dealing with high-dimensional data, without undermining the model's pattern recognition capacity, as is the case with the commonly applied, diagonal covariance modeling selections; further, a multiple token treatment of the resulting, factor analyzed SHMM, is performed. In Section V, the experimental evaluation of the proposed, SHMM and factor analyzed SHMM models, is conducted, considering a series of sequential data modeling applications from diverse domains. Finally, in the concluding section, the results of this paper are summarized and discussed.

II. Model Formulation

A. Student's-t mixture models (SMMs)

The adoption of the multivariate Student's-t distribution provides a way to broaden the Gaussian distribution for potential outliers. The probability density function (pdf) of a Student's-t distribution with

mean vector $\boldsymbol{\mu}$, covariance matrix $\boldsymbol{\Sigma}$, and $\nu > 0$ degrees of freedom is [12]

$$t(\boldsymbol{y}_t; \boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu) = \frac{\Gamma\left(\frac{\nu+p}{2}\right) |\boldsymbol{\Sigma}|^{-1/2} (\pi\nu)^{-p/2}}{\Gamma(\nu/2)\{1 + d(\boldsymbol{y}_t, \boldsymbol{\mu}; \boldsymbol{\Sigma})/\nu\}^{(\nu+p)/2}} \tag{1}$$

where p is the dimensionality of the observations $\boldsymbol{y}_t$, $d(\boldsymbol{y}_t, \boldsymbol{\mu}; \boldsymbol{\Sigma})$ is the squared Mahalanobis distance between $\boldsymbol{y}_t, \boldsymbol{\mu}$ with covariance matrix $\boldsymbol{\Sigma}$, and $\Gamma(s)$ is the Gamma function, $\Gamma(s) = \int_0^\infty e^{-t} z^{s-1} dz$. It can be shown (see, e.g., [12]) that, in essence, the Student's-t distribution corresponds to a Gaussian scale model [13] where the precision scalar is a Gamma distributed latent variable, depending on the degrees of freedom of the Student's-t density. That is, given

$$\boldsymbol{y}_t \sim t(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu) \tag{2}$$

it equivalently holds that [12]

$$\boldsymbol{y}_t | u_t \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}/u_t) \tag{3}$$

where the precision scalar, u_t, is distributed as

$$u_t \sim \mathcal{G}\left(\frac{\nu}{2}, \frac{\nu}{2}\right) \tag{4}$$

$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ stands for a normal distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, and $\mathcal{G}(\alpha, \beta)$ is the Gamma distribution. On the basis of these results, a finite mixture of Student's-t densities, with weights (prior probabilities) $c_1, ..., c_J$, is defined as [4]

$$p(\boldsymbol{y}_t; \boldsymbol{\Theta}) = \sum_{j=1}^{J} c_j t(\boldsymbol{y}_t; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j, \nu_j) \tag{5}$$

where, $\boldsymbol{\Theta} = \{c_j, \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j, \nu_j\}_{j=1}^J$. Here, we are using $p(.)$ as a generic notation for a probability function.

A graphical illustration of the univariate Student's-t distribution, with $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$ fixed, and for various values of the degrees of freedom ν, is provided in Fig. 1. As we observe, as $\nu \to \infty$, the Student's-t distribution tends to a Gaussian one with the same $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$. On the contrary, as ν tends to zero, the

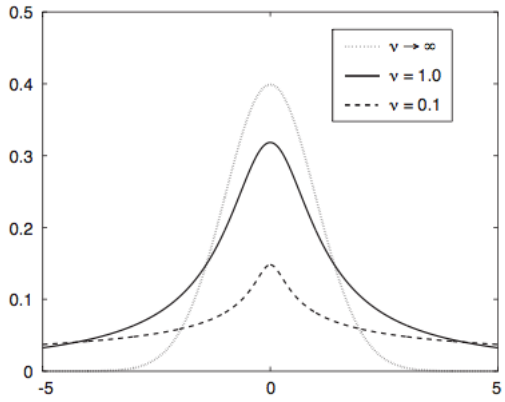


Figure 1. Univariate Student's-t distribution $t(\boldsymbol{y}_t; \boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu)$, with $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$ fixed, for various values of ν [6].

tails of the distribution become longer, thus allowing for a better handling of potential outliers, without affecting the mean or the covariance of the distribution. Thus, exploiting the harder tails of the Student's-t distribution, SMMs and SMM-based probabilistic generative models are able to handle in a considerably enhanced manner outliers residing within the fitting data sets, requiring a significantly reduced number of mixture component densities, and hence, providing better computational efficiency, better algorithm stability, improved model parameter estimates, and reduced requirements in training data availability, comparing to GMMs and GMM-based models [4], [14], [5], [6], [15].

B. The Student's-t hidden Markov model (SHMM)

Definition 1: Let us consider a finite state-space hidden Markov chain model. The considered model is a *Student's-t hidden Markov model (SHMM)* if and only if the observation emission distributions of its hidden states are considered to be finite mixtures of Student's-t densities.

Let us consider a Student's-t hidden Markov model comprising I states. Let $\{\boldsymbol{y}_t\}_{t=1}^{T}$ denote a sequence of observed data points modeled using the considered SHMM. Let us also assume for convenience, and without any loss of generality, that all the hidden state densities of the considered SHMM are approximated by Student's-t mixture models with the *same number* of component distributions, J. Then, from the conditional independence property of the hidden Markov chain [1], [2] it directly follows that the observations emitted from the same hidden state of the SHMM are independent, identically distributed

(i.i.d), in the form given by (5); that is the probability density of the observation $\boldsymbol{y}_t$ given that it is emitted from the ith model state is

$$p(\boldsymbol{y}_t; \boldsymbol{\Theta}_i) = \sum_{j=1}^{J} c_{ij} t(\boldsymbol{y}_t; \boldsymbol{\mu}_{ij}, \boldsymbol{\Sigma}_{ij}, \nu_{ij}) \tag{6}$$

where c_{ij}, $\boldsymbol{\mu}_{ij}$, $\boldsymbol{\Sigma}_{ij}$ and ν_{ij} are the mixing proportion, mean, covariance matrix and the degrees of freedom of the jth component density of the hidden distribution of the ith state of the model ($i = 1, ..., I$, $j = 1, ..., J$), respectively, and $\boldsymbol{\Theta}_i = \{c_{ij}, \nu_{ij}, \boldsymbol{\mu}_{ij}, \boldsymbol{\Sigma}_{ij}\}_{j=1}^{J}$.

III. Multiple Token Treatment of the SHMM

Training of the SHMM using multiple training sequences (tokens) can be easily conducted by means of the EM algorithm. Let us consider M independent sequences of fitting data. We assume for convenience, that all the sequences have the same length T, i.e. comprising T data points, without any loss of generality. Let the mth sequence be $\boldsymbol{y}_m = \{\boldsymbol{y}_{mt}\}_{t=1}^{T},\ m = 1, ..., M$, where $\boldsymbol{y}_{mt}$ stands for the tth data point of the mth fitting sequence. Then, from (6), we have

$$p(\boldsymbol{y}_{mt}; \boldsymbol{\Theta}_i) = \sum_{j=1}^{J} c_{ij} t(\boldsymbol{y}_{mt}; \boldsymbol{\mu}_{ij}, \boldsymbol{\Sigma}_{ij}, \nu_{ij}) \tag{7}$$

or, equivalently, using the properties of the Student's-t distribution

$$p(\boldsymbol{y}_{mt} | \{u_{ijmt}\}_{j=1}^{J}; \boldsymbol{\Theta}_i) = \sum_{j=1}^{J} c_{ij} \mathcal{N}(\boldsymbol{y}_{mt}; \boldsymbol{\mu}_{ij}, \boldsymbol{\Sigma}_{ij}/u_{ijmt}) \tag{8}$$

where u_{ijmt} is the precision scalar corresponding to the observation $\boldsymbol{y}_{mt}$ given it is generated from the jth component density of the ith hidden state distribution

$$u_{ijmt} \sim \mathcal{G}\left(\frac{\nu_{ij}}{2}, \frac{\nu_{ij}}{2}\right) \tag{9}$$

Let us denote as $\boldsymbol{s}_{mt}$ the state indicator vectors of the observed data, with $\boldsymbol{s}_{mt} = (s_{imt})_{i=1}^{I}$, and

$$s_{imt} \triangleq \begin{cases} 1, \text{ if } \boldsymbol{y}_{mt} \text{ is emitted from the } i\text{th model state } (i = 1, .., I) \\ \\ 0, \text{ otherwise} \end{cases}$$

Let us also denote as $\boldsymbol{z}_{mt}^{i}$ the state-conditional mixture component indicator vectors of the observed data, such that $\boldsymbol{z}_{mt}^{i} = (z_{jmt}^{i})_{j=1}^{J}$, and, given that $\boldsymbol{y}_{mt}$ is emitted from the ith state ($s_{imt} = 1$), it holds

$$z_{jmt}^{i} \triangleq \begin{cases} 1, \text{ if } \boldsymbol{y}_{mt} \text{ is generated from the } j\text{th component density of the state } (j = 1, .., J) \\ \\ 0, \text{ otherwise} \end{cases}$$

The EM algorithm constitutes the optimization of the conditional expectation of the complete data log-likelihood of the treated model, given the fitting data, $\boldsymbol{y}$, defined as

$$Q(\boldsymbol{\Psi}; \hat{\boldsymbol{\Psi}}) \triangleq E_{\hat{\boldsymbol{\Psi}}}(\log L_c(\boldsymbol{\Psi}) | \boldsymbol{y}) \tag{10}$$

where $\hat{\boldsymbol{\Psi}}$ denotes the obtained estimator of the model parameters vector $\boldsymbol{\Psi} = \{\boldsymbol{\Theta}_i, \pi_i, \pi_{hi}\}_{h,i=1}^{I}$, π_i are the initial state probabilities, and π_{hi} are the state transition probabilities of the Markov chain. For a continuous hidden Markov model, the expression of the complete data log-likelihood reads [4]

$$\begin{aligned} \log L_c(\boldsymbol{\Psi}) = & \sum_{m=1}^{M} \sum_{h=1}^{I} \left[s_{hm1} \log \pi_h + \sum_{i=1}^{I} \sum_{t=1}^{T-1} s_{hmt} s_{im,t+1} \log \pi_{hi} \right] \\ & + \sum_{i=1}^{I} \sum_{m=1}^{M} \sum_{t=1}^{T} s_{imt} \log p(\boldsymbol{y}_{mt}^{comp}; \boldsymbol{\Theta}_i) \end{aligned} \tag{11}$$

where $\boldsymbol{y}_{mt}^{comp}$ stands for the complete data corresponding to the tth observation of the mth sequence, $\boldsymbol{y}_{mt}$, and $\log p(\boldsymbol{y}_{mt}^{comp}; \boldsymbol{\Theta}_i)$ is the complete data log-likelihood of the emission distribution of the ith hidden state corresponding to $\boldsymbol{y}_{mt}$.

A graphical illustration of the considered SHMM can be found in Fig. 2. To provide a proper selection of the complete data $\boldsymbol{y}_{mt}^{comp}$ for the SHMM, we have to take into account that a closed form solution for log-likelihood optimization of a Student's-t distribution in the form (7) does not exist [12], [4].

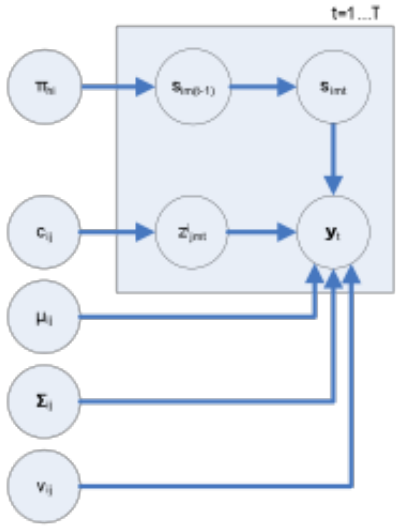


Figure 2. Directed graph of the SHMM for a sequence of T observation points, $\{\boldsymbol{y}_t\}_{t=1}^{T}$.

However, exploiting the alternative expression (8)-(9) of a Student's-t distribution as a Gaussian distribution with scaled precision, where the scalar is a Gamma distributed latent variable, a tractable optimization framework can be obtained. Under these considerations, we let the complete data corresponding to the mth sequence, $\boldsymbol{y}_m^{comp}$, be

1) the observable data $\boldsymbol{y}_{mt},\ t = 1, ..., T,\ m = 1, ..., M$
2) their state indicator vectors, $\boldsymbol{s}_{mt}$
3) their state-conditional mixture component indicator vectors, $\boldsymbol{z}_{mt}^i$
4) their corresponding precision scalars, u_{ijmt}.

Then, we have

$$p(\boldsymbol{y}_{mt}^{comp}; \boldsymbol{\Theta}_i) = \prod_{j=1}^{J} [c_{ij} p(\boldsymbol{y}_{mt}|u_{ijmt}; \boldsymbol{\Theta}_i)\ p(u_{ijmt}; \boldsymbol{\Theta}_i)]^{z_{jmt}^i}$$

which yields (ignoring constant terms)

$$\begin{aligned} \log p(\boldsymbol{y}_{mt}^{comp}; \boldsymbol{\Theta}_i) = \sum_{j=1}^{J} z_{jmt}^i \Bigg\{ -\log\Gamma\left(\frac{\nu_{ij}}{2}\right) + \frac{\nu_{ij}}{2}\Bigg[\log\left(\frac{\nu_{ij}}{2}\right) \\ + \log u_{ijmt} - u_{ijmt}\Bigg] - \frac{u_{ijmt}}{2} d(\boldsymbol{y}_{mt}, \boldsymbol{\mu}_{ij}; \boldsymbol{\Sigma}_{ij}) - \frac{1}{2}\log|\boldsymbol{\Sigma}_{ij}| + \log c_{ij} \Bigg\} \end{aligned} \tag{12}$$

The E-step on the (k+1)th iteration of the EM algorithm requires the calculation of the quantity

$Q(\mathbf{\Psi};\mathbf{\Psi}^{(k)})$, where $\mathbf{\Psi}^{(k)}$ denotes the *current* estimator (obtained by the kth iteration of the EM algorithm) of $\mathbf{\Psi}$. Using (11) and (12), we have

$$\begin{aligned} Q(\mathbf{\Psi};\mathbf{\Psi}^{(k)}) = &\sum_{m=1}^{M}\sum_{h=1}^{I}\left[\gamma_{hm1}^{(k)}\log\pi_h + \sum_{i=1}^{I}\sum_{t=1}^{T-1}\gamma_{himt}^{(k)}\log\pi_{hi}\right] \\ &+ \sum_{i=1}^{I}\sum_{m=1}^{M}\sum_{t=1}^{T}\gamma_{imt}^{(k)}E_{\mathbf{\Psi}^{(k)}}\left(\log p(\boldsymbol{y}_{mt}^{comp};\mathbf{\Theta}_i)|\boldsymbol{y}\right) \end{aligned} \tag{13}$$

where $\gamma_{imt}^{(k)}$ denote the estimators of the state emission posterior probabilities in kth iteration, defined as

$$\gamma_{imt} \triangleq p(s_{imt}=1|\boldsymbol{y}) = p(s_{imt}=1|\boldsymbol{y}_m) \tag{14}$$

$(t=1,...,T)$ and $\gamma_{himt}^{(k)}$ denote the estimators of the state transition posterior probabilities in kth iteration, defined as

$$\gamma_{himt} \triangleq p(s_{im,t+1}=1, s_{hmt}=1|\boldsymbol{y}) \tag{15}$$

$(t=1,...,T-1)$ for $m=1,...,M,\ h,i=1,...,I$. Therefore, the E-step of the algorithm comprises computation of the estimates $\gamma_{imt}^{(k)}$ and $\gamma_{himt}^{(k)}$, and of the expectation $E_{\mathbf{\Psi}^{(k)}}\left(\log p(\boldsymbol{y}_{mt}^{comp};\mathbf{\Theta}_i)|\boldsymbol{y}\right)$. Let us begin with the updates $\gamma_{imt}^{(k)}$ and $\gamma_{himt}^{(k)}$. These quantities can be obtained utilizing the forward-backward algorithm. It holds [2], [1]

$$\gamma_{himt}^{(k)} = \frac{a_{hmt}^{(k)}\pi_{hi}^{(k)}p(\boldsymbol{y}_{m,t+1};\mathbf{\Theta}_i^{(k)})b_{im,t+1}^{(k)}}{\sum_{\upsilon=1}^{I}\sum_{\phi=1}^{I}a_{\upsilon mt}^{(k)}\pi_{\upsilon\phi}^{(k)}p(\boldsymbol{y}_{m,t+1};\mathbf{\Theta}_\phi^{(k)})b_{\phi m,t+1}^{(k)}} \tag{16}$$

and

$$\gamma_{imt}^{(k)} = \frac{a_{imt}^{(k)}b_{imt}^{(k)}}{\sum_{h=1}^{I}a_{hmt}^{(k)}b_{hmt}^{(k)}} \tag{17}$$

where

$$a_{im1}^{(k)} = \pi_i^{(k)}p(\boldsymbol{y}_{m1};\mathbf{\Theta}_i^{(k)}) \tag{18}$$

$$a_{im,t+1}^{(k)} = p(\boldsymbol{y}_{m,t+1};\mathbf{\Theta}_i^{(k)})\sum_{h=1}^{I}a_{hmt}^{(k)}\pi_{hi}^{(k)}\ (t=1,..,T-1) \tag{19}$$

$$b_{hmT}^{(k)} = 1 \tag{20}$$

$$b_{hmt}^{(k)} = \sum_{i=1}^{I} \pi_{hi}^{(k)} p(\boldsymbol{y}_{m,t+1}; \boldsymbol{\Theta}_i^{(k)}) b_{im,t+1} \ (t = T-1, ..., 1) \tag{21}$$

and the expression of $p(\boldsymbol{y}_{mt}; \boldsymbol{\Theta}_i)$ is given by (7). Concerning the expectation $E_{\boldsymbol{\Psi}^{(k)}} (\log p(\boldsymbol{y}_{mt}^{comp}; \boldsymbol{\Theta}_i) | \boldsymbol{y})$, from (12), it can be shown (see Appendix A) that its estimation reduces to the computation of the quantities

$$\begin{aligned} \xi_{ijmt}^{(k)} &\triangleq E_{\boldsymbol{\Psi}^{(k)}}(z_{jmt}^{i} | \boldsymbol{y}_{mt}, s_{imt} = 1) \\ &= \frac{c_{ij}^{(k)} t(\boldsymbol{y}_{mt}; \boldsymbol{\mu}_{ij}^{(k)}, \boldsymbol{\Sigma}_{ij}^{(k)}, \nu_{ij}^{(k)})}{\sum_{h=1}^{J} c_{ih}^{(k)} t(\boldsymbol{y}_{mt}; \boldsymbol{\mu}_{ih}^{(k)}, \boldsymbol{\Sigma}_{ih}^{(k)}, \nu_{ih}^{(k)})} \end{aligned} \tag{22}$$

which stand for the conditional posterior probabilities that $\boldsymbol{y}_{mt}$ is generated from the jth component distribution of the ith state of the SHMM, given that it is emitted from the ith state of the model, and

$$\begin{aligned} u_{ijmt}^{(k)} &\triangleq E_{\boldsymbol{\Psi}^{(k)}} (u_{ijmt} | \boldsymbol{y}_{mt}) \\ &= \frac{\nu_{ij}^{(k)} + p}{\nu_{ij}^{(k)} + d(\boldsymbol{y}_{mt}, \boldsymbol{\mu}_{ij}^{(k)}; \boldsymbol{\Sigma}_{ij}^{(k)})} \end{aligned} \tag{23}$$

which stand for the posterior expected values of the precision scalars u_{ijmt}.

Further, the M-step of the algorithm is effected by performing the computations (see Appendix A)

$$\pi_i^{(k+1)} = \frac{1}{M} \sum_{m=1}^{M} \gamma_{im1}^{(k)} \tag{24}$$

$$\pi_{hi}^{(k+1)} = \frac{\sum_{m=1}^{M} \sum_{t=1}^{T-1} \gamma_{himt}^{(k)}}{\sum_{m=1}^{M} \sum_{t=1}^{T-1} \gamma_{hmt}^{(k)}} \tag{25}$$

$$c_{ij}^{(k+1)} = \sum_{m=1}^{M} \sum_{t=1}^{T} r_{ijmt}^{(k)} / \sum_{m=1}^{M} \sum_{t=1}^{T} \gamma_{imt}^{(k)} \tag{26}$$

$$\boldsymbol{\mu}_{ij}^{(k+1)} = \frac{\sum_{m=1}^{M} \sum_{t=1}^{T} r_{ijmt}^{(k)} u_{ijmt}^{(k)} \boldsymbol{y}_{mt}}{\sum_{m=1}^{M} \sum_{t=1}^{T} r_{ijmt}^{(k)} u_{ijmt}^{(k)}} \tag{27}$$

Algorithm 1 EM Algorithm for the SHMM

$k := 0$

1) Conduct the forward-backward algorithm to obtain the quantities $a_{imt}^{(k)}$ and $b_{imt}^{(k)}$.
2) Effect the E-step by computing the $\gamma_{himt}^{(k)}$, $\gamma_{imt}^{(k)}$, $\xi_{ijmt}^{(k)}$, $r_{ijmt}^{(k)}$, and $u_{ijmt}^{(k)}$, using (16), (17), (22), (31), and (23), respectively.
3) Effect the M-step by computing the $\pi_i^{(k+1)}$, $\pi_{hi}^{(k+1)}$, $c_{ij}^{(k+1)}$, $\boldsymbol{\mu}_{ij}^{(k+1)}$, $\boldsymbol{\Sigma}_{ij}^{(k+1)}$, and $\nu_{ij}^{(k+1)}$, using (24)-(29), respectively.
4) If the EM algorithm converges, **exit**; otherwise increase the iteration counter ($k := k + 1$) and goto 1.

$$\begin{aligned}\boldsymbol{\Sigma}_{ij}^{(k+1)} = &\sum_{m=1}^{M}\sum_{t=1}^{T} r_{ijmt}^{(k)} u_{ijmt}^{(k)} (\boldsymbol{y}_{mt} - \boldsymbol{\mu}_{ij}^{(k+1)})(\boldsymbol{y}_{mt} - \boldsymbol{\mu}_{ij}^{(k+1)})^T \\ &\times \left[\sum_{m=1}^{M}\sum_{t=1}^{T} r_{ijmt}^{(k)}\right]^{-1}\end{aligned} \tag{28}$$

and solving the equation

$$\begin{aligned}&1 - \psi\left(\frac{\nu_{ij}}{2}\right) + \log\left(\frac{\nu_{ij}}{2}\right) + \psi\left(\frac{\nu_{ij}^{(k)} + p}{2}\right) - \log\left(\frac{\nu_{ij}^{(k)} + p}{2}\right) \\ &+ \frac{1}{\sum_{m=1}^{M}\sum_{t=1}^{T} r_{ijmt}^{(k)}} \sum_{m=1}^{M}\sum_{t=1}^{T} r_{ijmt}^{(k)} \left(\log u_{ijmt}^{(k)} - u_{ijmt}^{(k)}\right) = 0\end{aligned} \tag{29}$$

to obtain the estimates of ν_{ij}, where, $\psi(s)$ is the digamma function and $r_{ijmt}^{(k)}$ is the joint posterior probability that $\boldsymbol{y}_{mt}$ is generated from the ith state of the model and particularly from its jth component distribution

$$r_{ijmt} \triangleq p(s_{imt} = 1, z_{jmt}^i = 1 | \boldsymbol{y}) = \gamma_{imt}\xi_{ijmt} \tag{30}$$

$$r_{ijmt}^{(k)} = \gamma_{imt}^{(k)}\xi_{ijmt}^{(k)} \tag{31}$$

An outline of the EM algorithm for the SHMM is given in Alg. 1.

Let us, now, regard the computational requirements of the EM algorithm for the SHMM. As outlined in Alg. 1, comparing to the GHMM, each iteration of the EM algorithm for the SHMM involves computation of two extra quantities: the posterior expectations of the precision scalars, $u_{ijmt}^{(k)}$, and the estimates of the degrees of freedom, $\nu_{ij}^{(k)}$. Let us begin with the precision scalars. As it is obvious from (23), the major computational requirements of their calculation are imposed by the Mahalanobis

distances $d(\boldsymbol{y}_{mt}, \boldsymbol{\mu}_{ij}^{(k)}; \boldsymbol{\Sigma}_{ij}^{(k)})$. Nevertheless, these quantities are also involved in computation of the pdf of the Student's-t densities as well as of the Gaussian ones, that is, they are computed both for the GHMM and the SHMM methods. Thereby, if properly coded, computation of the precision scalars, $u_{ijmt}^{(k)}$, incurs only a negligible computational overhead for the EM algorithm. Further, concerning the degrees of freedom estimates, $\nu_{ij}^{(k)}$, from (29), we observe that their computation involves resolution of an open-form formula, e.g., by means of the Newton's method. Nevertheless, the left hand part of (29) does not involve computationally expensive calculations. Moreover, given that, typically, $0 < \nu < 100$ [4], convergence of Newton's method is expected to be extremely fast. Thus, calculation of $\nu_{ij}^{(k)}$ should induce a rather low computational overhead, clearly not affecting the efficiency of the SHMM training algorithm. Our claims on the computational efficiency of the SHMM shall be experimentally validated in Section V.

Finally, apart from model training, two equally significant problems in HMM literature concern likelihood calculation and corresponding emitting states sequence selection for an observed sequence with respect to a trained model. Let us consider an SHMM, trained using the EM algorithm, as described above, with parameters set $\hat{\boldsymbol{\Psi}}$, and an observed sequence $\boldsymbol{y} = \{\boldsymbol{y}_t\}_{t=1}^{T}$. Then, it is straightforward to observe that both likelihood calculation and corresponding emitting states sequence selection can be conducted the same way as in the case of GHMMs, using the forward algorithm and the Viterbi algorithm, respectively. This is a direct consequence of the fact that the forward algorithm and the Viterbi algorithm are not contingent on the form of the continuous emission densities of the Markov chain states [1], and, hence, they can be applied in the SHMM without changes, and for the same computational costs.

IV. Enhancing the Efficacy of the SHMM by Application of A Covariance Modeling Technique

A significant practical concern when using CHMMs is the parameters number of the model to be fitted. Using full covariance matrices, this number might be increased dramatically, especially in cases of high-dimensional data. As a consequence, model parameters estimation becomes undependable, as the training algorithm gets prone to poor local optima and overfitting, whereas the stability of the model training

procedure undergoes a severe undermining, since, under such conditions of "sparse" training data, the EM algorithm tends to obtain spurious clusters, thus resulting in singular or near singular covariance matrices of the mixture component densities.

To resolve these issues, many researchers have considered parameter tying, e.g. sharing the covariance matrices across different states of the hidden Markov chain (see e.g. [16]). Still, this method has many drawbacks, the most significant being the considerable complication of the fitting procedure and the artistry required to design the HMM. Using diagonal covariance matrices is another common solution; diagonal covariance GMMs have the potential to approximate covariances between the elements of multivariate observations [17]. However, it would be beneficial to have uncorrelated observed vectors for each component density when diagonal covariance matrices are used [18]. Principal component analysis (PCA) [19] is a popular alternative, resolving the drawbacks of diagonal covariance models. Nevertheless, PCA is well-known to be heavily affected by outliers in the data, it does not define a proper density model outside the subspace of principal components, while component-wise variations outside this subspace are modeled uniformly, even when the data does not warrant such an assumption.

To address these issues, in this paper we consider application of factor analysis [20] as a means of model parameters reduction for the SHMM, in cases of high-dimensional data. Factor analysis is a linear scheme modeling the covariances between the elements of multivariate observations by dividing them into two parts, an unobserved systematic part, taken as a linear combination of a relatively small number of unobserved (latent) variables called factors, and an unobserved error part, whose elements are considered as uncorrelated. Thus, by postulating a finite mixture of such linear submodels (factor analyzers), a computationally efficient approximation of the distribution of the observed data can be obtained. Indeed, the mixture of factor analyzers (MFA) model has been shown to retain the pattern recognition effectiveness of full covariance GMMs, while preserving the computational efficiency merits of diagonal GMMs [20].

A. Finite mixtures of Student's-t factor analyzers

Let us consider a set of independent and identically distributed (i.i.d.), p-dimensional data, $\{\boldsymbol{y}_t\}_{t=1}^T$. Let us also consider the statistical representation of this set by a J-component mixture of factor analyzers, with weights c_j (summing to one). The latent vectors whose elements are the factors corresponding to the observed data (factor vectors) are assumed to belong to a q-dimensional space, $q < p$. Then, the considered representation is given by the expression

$$\boldsymbol{y}_t = \boldsymbol{\mu}_j + \boldsymbol{B}_j \boldsymbol{x}_{jt} + \boldsymbol{e}_{jt} \ \text{ with probability } c_j \tag{32}$$

where, $\boldsymbol{\mu}_j$ is the mean of the jth component factor analyzer, $\boldsymbol{B}_j$ is the factor loadings matrix of the jth analyzer, $\boldsymbol{x}_{jt}$ is the q-dimensional factor vector of the tth observation, $\boldsymbol{y}_t$, with respect to the jth analyzer, and $\boldsymbol{e}_{jt}$ is the error of the factor analysis model corresponding to $\boldsymbol{y}_t$ with respect to the jth analyzer.

Conventionally, the factor vectors, $\boldsymbol{x}_{jt}$, and the error vectors, $\boldsymbol{e}_{jt}$, in an MFA model (32) are assumed to follow Gaussian distributions. Alternatively, to exploit the robust (in terms of outliers) statistical modeling capabilities that the Student's-t distribution offers, as we described in Section I, a Student's-t factor analysis model has been proposed in [14], [5]. Under this regard, we have

$$\boldsymbol{x}_{jt} \sim t(\boldsymbol{0}, \boldsymbol{I}_q, \nu_j) \tag{33}$$

$$\boldsymbol{e}_{jt} \sim t(\boldsymbol{0}, \boldsymbol{D}_j, \nu_j) \tag{34}$$

where $\boldsymbol{D}_j = \text{diag}(d_{j1}, ..., d_{jp})$. From (32)-(34), it follows [14]

$$p(\boldsymbol{y}_t | \{\boldsymbol{x}_{jt}\}_{j=1}^J; \boldsymbol{\Theta}) = \sum_{j=1}^J c_j t(\boldsymbol{y}_t; \boldsymbol{\mu}_j + \boldsymbol{B}_j \boldsymbol{x}_{jt}, \boldsymbol{D}_j, \nu_j) \tag{35}$$

and, unconditionally

$$p(\boldsymbol{y}_t; \boldsymbol{\Theta}) = \sum_{j=1}^J c_j t(\boldsymbol{y}_t; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j, \nu_j) \tag{36}$$

where

$$\boldsymbol{\Sigma}_j = \boldsymbol{B}_j \boldsymbol{B}_j^T + \boldsymbol{D}_j \tag{37}$$

and $\boldsymbol{\Theta} = \{c_j, \boldsymbol{\mu}_j, \boldsymbol{B}_j, \boldsymbol{D}_j, \nu_j\}_{j=1}^J$.

B. Latent subspace covariance modeling for the SHMM using factor analysis

Definition 2: A factor analyzed SHMM is a CHMM the hidden state observation emission densities of which are modeled by finite mixtures of Student's-t factor analyzers, thus yielding an SHMM with factor analyzed covariance matrices.

To train the factor analyzed SHMM we employ a variant of the EM algorithm, namely the alternating expectation-conditional maximization (AECM) algorithm [21]. Each iteration of the AECM algorithm is split into two cycles; in each cycle, a different complete data configuration is considered. This selection allows for the derivation of a more convenient and computationally efficient form of the model training algorithm comparing to EM, where, in each iteration, in the first cycle, the equivalent SHMM is obtained by properly marginalizing over the factors, while, in the second cycle, the factor analysis-related quantities are updated.

In detail, to provide a multiple token ML treatment of the factor analyzed SHMM, we adopt the same setting as in Section III. A graphical illustration of the considered factor analyzed SHMM can be found in Fig. 3. From (36), it is apparent that, given a set of observed sequences $\boldsymbol{y} = \{\boldsymbol{y}_m\}_{m=1}^M$, $\boldsymbol{y}_m = \{\boldsymbol{y}_{mt}\}_{t=1}^T$, expression (7) of the hidden state observation emission probabilities of the SHMM does also hold for the factor analyzed SHMM, with

$$\boldsymbol{\Sigma}_{ij} = \boldsymbol{B}_{ij} \boldsymbol{B}_{ij}^T + \boldsymbol{D}_{ij} \tag{38}$$

Furthermore, conditional on the factor vectors, $\boldsymbol{x}_{ijmt}$, we have

$$p(\boldsymbol{y}_{mt} | \{\boldsymbol{x}_{ijmt}\}_{j=1}^J; \boldsymbol{\Theta}_i) = \sum_{j=1}^J c_{ij} t(\boldsymbol{y}_{mt}; \boldsymbol{\mu}_{ij} + \boldsymbol{B}_{ij} \boldsymbol{x}_{ijmt}, \boldsymbol{D}_{ij}, \nu_{ij}) \tag{39}$$

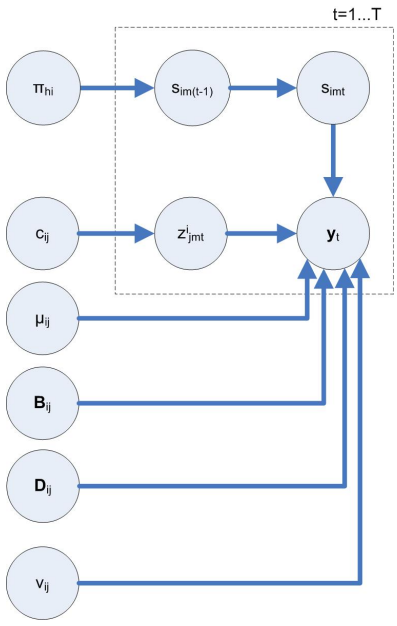


Figure 3. Directed graph of the factor analyzed SHMM for a sequence of T observation points, $\{\boldsymbol{y}_t\}_{t=1}^{T}$.

or, alternatively

$$p(\boldsymbol{y}_{mt}|\{\boldsymbol{x}_{ijmt}, u_{ijmt}\}_{j=1}^{J}; \boldsymbol{\Theta}_i) = \sum_{j=1}^{J} c_{ij}\mathcal{N}(\boldsymbol{y}_{mt}; \boldsymbol{\mu}_{ij} + \boldsymbol{B}_{ij}\boldsymbol{x}_{ijmt}, \boldsymbol{D}_{ij}/u_{ijmt}) \tag{40}$$

where the distribution of u_{ijmt} is given by (9), and $\boldsymbol{x}_{ijmt}$ is the factor vector corresponding to the observation $\boldsymbol{y}_{mt}$ with respect to the jth analyzer of the ith state of the model, with

$$\boldsymbol{x}_{ijmt} \sim t(\boldsymbol{0}, \boldsymbol{I}_q, \nu_{ij}) \tag{41}$$

1) First cycle: In the first cycle of each iteration of the algorithm, we derive the estimates of the Markov chain initial and transition (prior) probabilities, π_i and π_{hi}, as well as the estimates of the weights, c_{ij}, the means, $\boldsymbol{\mu}_{ij}$, and the degrees of freedom, ν_{ij}, of the component factor analyzers of the emission distributions of the model states. For these purposes, we let the complete data comprise

1) the observable data $\boldsymbol{y}_{mt},\ t = 1, ..., T,\ m = 1, ..., M$
2) their state indicator vectors, $\boldsymbol{s}_{mt}$
3) their state-conditional mixture component indicator vectors, $\boldsymbol{z}_{mt}^{i}$
4) their corresponding precision scalars, u_{ijmt}.

Obviously, under this regard, the first cycle on each iteration of the AECM algorithm for the factor analyzed SHMM reduces to an iteration of the EM algorithm for the "plain" SHMM. Therefore, the required estimates $\gamma_{himt}^{(k)}$, $\gamma_{imt}^{(k)}$, $\xi_{ijmt}^{(k)}$, $r_{ijmt}^{(k)}$, $u_{ijmt}^{(k)}$, $\pi_i^{(k+1)}$, $\pi_{hi}^{(k+1)}$, $c_{ij}^{(k+1)}$, and $\nu_{ij}^{(k+1)}$ are given by (16), (17), (22), (31), (23), (24)-(27), and (29), respectively, with the covariance matrices $\boldsymbol{\Sigma}_{ij}$ of the observation emission probabilities of the model states, $p(\boldsymbol{y}_{mt};\boldsymbol{\Theta}_i)$, now given by (38).

2) Second cycle: In the second cycle of each algorithm iteration, we derive the estimates of the factor loading matrices, $\boldsymbol{B}_{ij}$, and the noise covariance matrices, $\boldsymbol{D}_{ij}$ of the model. For these purposes, and based on (39)-(40), a convenient selection for the complete data comprises

1) the observable data $\boldsymbol{y}_{mt},\ t=1,...,T,\ m=1,...,M$
2) their state indicator vectors, $\boldsymbol{s}_{mt}$
3) their state-conditional mixture component indicator vectors, $\boldsymbol{z}_{mt}^{i}$
4) their corresponding precision scalars, u_{ijmt}
5) their corresponding factor vectors, $\boldsymbol{x}_{ijmt}$.

Then, the expression of $\log p(\boldsymbol{y}_{mt}^{comp};\boldsymbol{\Theta}_i)$ in (11) yields (ignoring constant terms)

$$
\begin{aligned}
p(\boldsymbol{y}_{mt}^{comp};\boldsymbol{\Theta}_i) &\triangleq \prod_{j=1}^{J}\left[c_{ij}p(\boldsymbol{y}_{mt}|\boldsymbol{x}_{ijmt},u_{ijmt};\boldsymbol{\Theta}_i)p(\boldsymbol{x}_{ijmt}|u_{ijmt})p(u_{ijmt};\boldsymbol{\Theta}_i)\right]^{z_{jmt}^{i}} \\
\Rightarrow \log p(\boldsymbol{y}_{mt}^{comp};\boldsymbol{\Theta}_i) &= \sum_{j=1}^{J} z_{jmt}^{i}\Bigg[-\log\Gamma\left(\frac{\nu_{ij}}{2}\right)+\frac{\nu_{ij}}{2}\log\left(\frac{\nu_{ij}}{2}\right)+\frac{\nu_{ij}}{2}(\log u_{ijmt}-u_{ijmt}) \\
&\quad -\frac{u_{ijmt}}{2}(\boldsymbol{y}_{mt}-\boldsymbol{\mu}_{ij}-\boldsymbol{B}_{ij}\boldsymbol{x}_{ijmt})^{T}\boldsymbol{D}_{ij}^{-1}(\boldsymbol{y}_{mt}-\boldsymbol{\mu}_{ij}-\boldsymbol{B}_{ij}\boldsymbol{x}_{ijmt}) \\
&\quad -\frac{1}{2}\log|\boldsymbol{D}_{ij}|+\log c_{ij}\Bigg]
\end{aligned}
\tag{42}
$$

Let us begin with the E-step of this cycle of the algorithm. The Markov chain posterior probabilities γ_{himt} and γ_{imt} are updated using the forward-backward algorithm (eq. (16)-(17)). Concerning the term $E_{\boldsymbol{\Psi}^{(k)}}\left(\log p(\boldsymbol{y}_{mt}^{comp};\boldsymbol{\Theta}_i)|\boldsymbol{y}\right)$, it can be shown (see Appendix B) that its computation again reduces to the computation of the state-conditional posteriors $\xi_{ijmt}^{(k)}$, given by (22), and the expected values of the precision

Algorithm 2 AECM Algorithm for the factor analyzed SHMM

$k := 0$

- **First Cycle**

1) Conduct the forward-backward algorithm to obtain the quantities $a_{imt}^{(k)}$ and $b_{imt}^{(k)}$.
2) Effect the E-step by computing the $\gamma_{himt}^{(k)}$, $\gamma_{imt}^{(k)}$, $\xi_{ijmt}^{(k)}$, $r_{ijmt}^{(k)}$, and $u_{ijmt}^{(k)}$, using (16), (17), (22), (31), and (23), respectively.
3) Effect the M-step by computing the $\pi_i^{(k+1)}$, $\pi_{hi}^{(k+1)}$, $c_{ij}^{(k+1)}$, $\boldsymbol{\mu}_{ij}^{(k+1)}$, and $\nu_{ij}^{(k+1)}$, using (24)-(27) and (29), respectively.

- **Second Cycle**

1) Conduct the forward-backward algorithm to obtain the quantities $a_{imt}^{(k)}$ and $b_{imt}^{(k)}$.
2) Recompute the $\gamma_{himt}^{(k)}$, $\gamma_{imt}^{(k)}$, $\xi_{ijmt}^{(k)}$, $r_{ijmt}^{(k)}$, and $u_{ijmt}^{(k)}$, using (16), (17), (22), (31), and (23), respectively. Finish the E-step of the second cycle of the algorithm iteration by obtaining the estimates of the factor analysis quantities $\boldsymbol{\upsilon}_{ij}^{(k)}$ and $\boldsymbol{\omega}_{ij}^{(k)}$, using (43) and (44), respectively.
3) Effect the M-step by computing the $\boldsymbol{B}_{ij}^{(k+1)}$ and $\boldsymbol{D}_{ij}^{(k+1)}$, using (45) and (46), respectively.
4) If the AECM algorithm converges, **exit**; otherwise increase the iteration counter ($k := k + 1$) and goto *First Cycle*-1.

scalars u_{ijmt}, given by (23), however, it also requires computation of the factor analysis quantities

$$\boldsymbol{\upsilon}_{ij}^{(k)} = (\boldsymbol{B}_{ij}^{(k)} \boldsymbol{B}_{ij}^{(k)^T} + \boldsymbol{D}_{ij}^{(k)})^{-1} \boldsymbol{B}_{ij}^{(k)} \tag{43}$$

$$\boldsymbol{\omega}_{ij}^{(k)} = \boldsymbol{I}_q - \boldsymbol{\upsilon}_{ij}^{(k)^T} \boldsymbol{B}_{ij}^{(k)} \tag{44}$$

Finally, the M-step on the second cycle of each iteration of the algorithm yields the estimates of $\boldsymbol{B}_{ij}$ and $\boldsymbol{D}_{ij}$. These quantities are given by (see Appendix B)

$$\boldsymbol{B}_{ij}^{(k+1)} = \boldsymbol{V}_{ij}^{(k)} \boldsymbol{\upsilon}_{ij}^{(k)} \left(\boldsymbol{\upsilon}_{ij}^{(k)^T} \boldsymbol{V}_{ij}^{(k)} \boldsymbol{\upsilon}_{ij}^{(k)} + \boldsymbol{\omega}_{ij}^{(k)} \right)^{-1} \tag{45}$$

$$\boldsymbol{D}_{ij}^{(k+1)} = \operatorname{diag}\{ \boldsymbol{V}_{ij}^{(k)} - \boldsymbol{V}_{ij}^{(k)} \boldsymbol{\upsilon}_{ij}^{(k)} \boldsymbol{B}_{ij}^{(k+1)^T} \} \tag{46}$$

where, r_{ijmt} is given by (31), and

$$\boldsymbol{V}_{ij}^{(k)} = \frac{\sum_{m=1}^{M} \sum_{t=1}^{T} r_{ijmt}^{(k)} u_{ijmt}^{(k)} (\boldsymbol{y}_{mt} - \boldsymbol{\mu}_{ij}^{(k+1)})(\boldsymbol{y}_{mt} - \boldsymbol{\mu}_{ij}^{(k+1)})^T}{\sum_{m=1}^{M} \sum_{t=1}^{T} r_{ijmt}^{(k)}} \tag{47}$$

An outline of the AECM algorithm for the factor analyzed SHMM can be found in Alg. 2.

V. Experimental Evaluation

In this section, we provide a thorough experimental evaluation of the SHMM and the factor analyzed SHMM, in a series of sequential data modeling applications from diverse domains. For comparison, we also evaluate GHMMs, the selective training method for GHMMs of [8] (ST-HMM), and the large margin CHMM (LM-HMM) of [11]. The evaluated methodologies have been developed in Matlab R2008a, and were executed on a Macintosh platform with an Intel Core 2 Duo 2 GHz CPU, and 2 GB RAM, running Mac OS X 10.5 (Leopard).

A. Experiment on Simulated Data

We begin with a toy example on simulated data, to demonstrate the notion behind the use of the Student's-t distribution as the observations distribution for sequential data modeling in the context of a continuous hidden Markov model. The considered synthetic data was obtained by generating five realizations of 250 samples each. In each realization, the first 100 samples were drawn from the univariate Gaussian distribution $\mathcal{N}(3, 0.25)$, the second 100 samples were drawn from the univariate Gaussian distribution $\mathcal{N}(0, 1)$, while the final 50 samples were drawn from a uniform distribution on the interval $[-10, 10]$ (outliers). In Table I, we provide the two-state hidden Markov models obtained after running the EM algorithm considering both Gaussian and Student's-t observation densities. As we observe, using Gaussian observation densities yields a model with the state 1 output probability distribution mean being far from the correct one, covariances for both state distributions considerably bigger comparing to the actual ones, and less than optimal Markov chain probabilities. On the contrary, Student's-t observation densities obtain correctly the output probability distributions for both the HMM states, and a much improved estimate for the Markov chain probabilities.

B. Bimanual Gesture Recognition

Here we evaluate the SHMM in classification of sequential data. We consider the problem of bimanual gesture recognition, experimenting with the American Sign Language gestures for the words: *against, aim,*

Table I
OBTAINED TWO-STATE MODELS CONSIDERING GAUSSIAN AND STUDENT'S-t OBSERVATION DENSITIES.

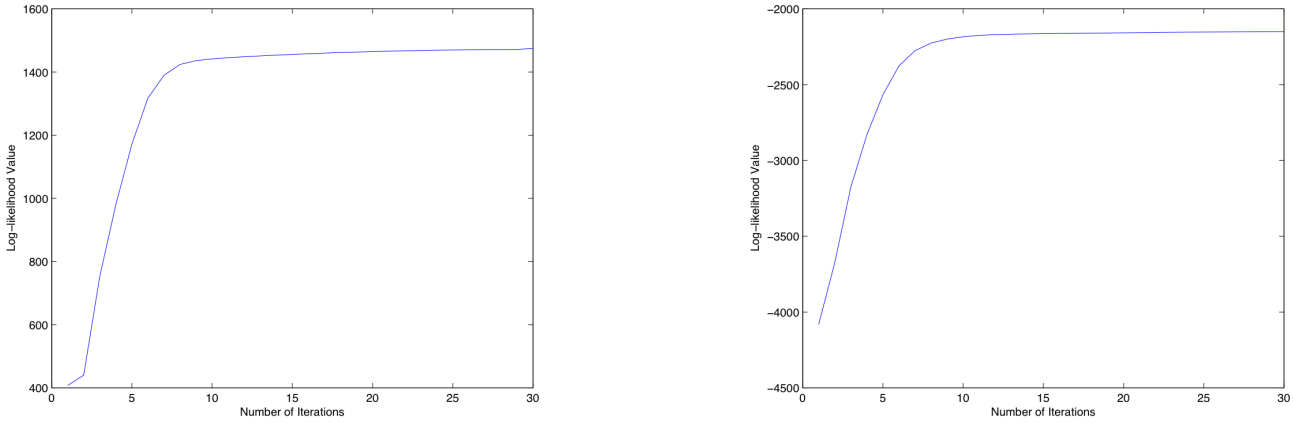
Model	Gaussian	Student's-t
State transition matrix	$\begin{bmatrix} 0.848 & 0.152 \\ 0.308 & 0.692 \end{bmatrix}$	$\begin{bmatrix} 0.8997 & 0.1003 \\ 0.0923 & 0.9077 \end{bmatrix}$
Initial state probabilities	$[0.998\ 0.002]$	$[1\ 0]$
State 1 output probability distribution	$\mathcal{N}(1.85, 2.5413)$	$\mathcal{N}(3.026, 0.63)$
State 2 output probability distribution	$\mathcal{N}(0.13, 20.8467)$	$\mathcal{N}(-0.05, 1.069)$

balloon, bandit, cake, chair, computer, concentrate, cross, deaf, explore, hunt, knife, relay, reverse, and *role*. The used data set, firstly presented in [22], was obtained from four different persons executing each one of these gestures. It comprises a training set, including 30 videos of variable duration per gesture, and a test set composed of 10 videos per gesture[1]. From this data set, we extracted several features representing the relative position of the hands and the face in the images, as well as the shape of the respective skin regions, by means of the complex Zernike moments [23], as described in [22]. This way, a 12-dimensional feature vector was derived from each sequence. To conduct classification of the obtained sequences, we employ 3-state CHMMs ($I = 3$). We try various numbers of component densities per state, J, and the lowest value of J yielding the highest recognition rate for each method is determined. The obtained error rates are depicted in Table II; there, the results for the GHMM, SHMM, and ST-HMM methods are averages over 30 runs of the training algorithm, from different random starts. As we notice, the SHMM provides superior classification performance comparing to its competitors, for a lower number of mixture component densities, J.

Finally, in Fig. 4, we provide the convergence rates of the EM algorithm for the GHMM and SHMM methods. These figures are averages over the conducted 30 runs of the EM algorithm, and over the 16 trained gesture models. As we observe, the EM algorithm converges equally fast for both methods.

Table II
GESTURE RECOGNITION: RECOGNITION ERROR RATES (%) OF THE CONSIDERED METHODS FOR OPTIMAL J VALUES.

Gesture	SHMM ($J = 7$)	GHMM ($J = 9$)	ST-HMM ($J = 9$)	LM-HMM ($J = 9$)
against	0.19	0.97	0.51	0.28
aim	0.25	0.69	0.50	0.31
balloon	4.47	7.02	5.33	4.61
bandit	0.14	0.83	0.71	0.20
chair	11.33	23.64	16.06	13.18
cake	11.08	18.25	14.82	11.95
computer	6.4	6.88	5.50	4.13
concentrate	10.03	16.21	12.85	10.76
cross	15.04	28.13	19.54	16.88
deaf	0.42	4.37	2.19	1.62
explore	4.41	6.62	5.53	4.97
hunt	15.26	28.56	20.83	16.64
knife	11.65	24.12	18.13	13.47
relay	0.42	6.43	3.69	3.36
reverse	3.82	8.50	6.21	5.10
role	0.48	1.84	1.84	1.10
Average	5.96	11.44	8.39	6.79



(a) GHMM ($I = 3, J = 9$)

(b) SHMM ($I = 3, J = 7$)

Figure 4. Bimanual gesture recognition: Average convergence rates of the SHMM and GHMM methods.

Table III
TIMIT SEQUENCE SEGMENTATION: PHONE ERROR RATES (IN %) FOR VARIOUS NUMBERS OF MIXTURE COMPONENT DENSITIES, J.

Model	$J = 1$	$J = 2$	$J = 4$	$J = 8$
SHMM	44.6	41.7	38.3	36.1
GHMM	45.1	43.5	42.7	41.3
ST-HMM	44.9	43.8	42.3	39.4
LM-HMM	44.6	43.3	40.9	37.7

Table IV
TIMIT SEQUENCE SEGMENTATION: EXECUTION TIME OF THE MODEL TRAINING ALGORITHMS.

Model	diagonal SHMM	diagonal GHMM	ST-HMM	LM-HMM
Execution Time (in sec)	1.65	1.23	1.33	904.71

C. Phonetic Recognition in TIMIT Speech Corpus

Further, we consider sequential data segmentation; we perform phonetic recognition experiments, using a subset of the TIMIT speech corpus[2] [24], [25]. The used data set was derived by computing 39-dimensional acoustic feature vectors from 13 mel-frequency cepstral coefficients and their first and second temporal derivatives. In our experiment, we train one CHMM using each one of the methods SHMM, GHMM, ST-HMM, and LM-HMM. In each case, the acoustic feature vectors are labeled by 48 phonetic classes, each represented by one CHMM state. For each method, we compare the phonetic state sequences obtained by Viterbi decoding to the "ground-truth" phonetic transcriptions provided by the TIMIT corpus. To obtain error rates, we map the 48 phonetic state labels down to 39 broader phone categories, using the same standard conventions as in [11]. Then, a relevant error rate can be obtained by aligning the Viterbi and ground truth transcriptions using dynamic programming [25] and summing the substitution, deletion, and insertion error rates from the alignment process. The obtained results can be found in Table III. We notice that the SHMM performs better than the competition, especially for higher numbers of mixture components. In Table IV, we provide the training algorithm execution time of each one of the considered methods, for $J = 1$. We observe that the SHMM, GHMM, and ST-HMM methods impose comparable computational requirements, contrary to the LM-HMM method, which, clearly, imposes a heavy computational burden.

D. Cognitive State Decoding from Brain fMRI Images

Finally, we discuss a case of high-dimensional sequence modeling, to exhibit the advantages of the factor analyzed SHMM over the diagonal one. For these purposes, we consider the task of cognitive

[1]The used data set is publicly and freely available in *http://www.iit.demokritos.gr/∼dkosmo/downloads/gesture/help_gestureDB.htm*

[2]The used data set is publicly available in http://www.cs.berkeley.edu/~feisha/codes/lm_cdhmm/ for demonstration of the LM-HMM [11] method.

state decoding from brain fMRI images [26]. The goal here is to make it possible to detect transient cognitive states using brain fMRI sequences, by training proper HMM classifiers to automatically decode the subject's cognitive state at a single time instant or interval. This problem domain is quite interesting from the perspective of machine learning, because it provides a case study of classifier learning from extremely high dimensional, sparse, and noisy data. In this experiment, we compare the performances of the factor analyzed SHMM, the diagonal SHMM, a factor analyzed form of the GHMM, and the diagonal GHMM.

The used data set[3] is partitioned into trials. For some of these intervals, the subject simply rested, or gazed at a fixation point on a screen. For other trials, the subject was shown a picture and a sentence, and instructed to press a button to indicate whether the sentence correctly described the picture. For these trials, the sentence and picture were presented in sequence, with the picture presented first on half of the trials, and the sentence presented first on the other half of the trials. Forty such trials are available for each subject. Data were collected as follows: One fMRI image was obtained from the subjects every 500msec. Only a fraction of the brain of each subject was imaged. The data is marked up with 25-30 anatomically defined regions (called "Regions of Interest", or ROIs). The timing within each trial is as follows: The first stimulus (sentence or picture) was presented at the beginning of the trail (image=1). Four seconds later (image=9) the stimulus was removed, replaced by a blank screen. Four seconds later (image=17) the second stimulus was presented. This remained on the screen for four seconds, or until the subject pressed the mouse button, whichever came first. A rest period of 15 seconds (30 images) was added after the second stimulus was removed from the screen. Thus, each trial lasted a total of approximately 27 seconds (approximately 54 images).

In our application, we want to train CHMMs to distinguish whether the subjects are viewing a picture or a sentence on the basis of the obtained fMRI sequences. For this purpose, we train two 4×4 CHMMs (one for the picture stimulus sequences, and one for the sentence stimulus sequences), using each one of the

[3]The considered data set is publicly available in http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-81/www/

Table V
COGNITIVE STATE RECOGNITION FROM BRAIN IMAGES: AVERAGE ERROR RATE AND ITS STANDARD DEVIATION (IN %) OVER 30 RUNS OF THE TRAINING ALGORITHM.

Model	factor analyzed SHMM	diagonal SHMM	factor analyzed GHMM	diagonal GHMM
p	70	70	70	70
q	9	-	9	-
Error Rate	23.44 $\pm$ 2.22	29.87 $\pm$ 3.17	35.06 $\pm$ 4.12	38.25 $\pm$ 4.79

Table VI
COGNITIVE STATE RECOGNITION FROM BRAIN IMAGES: AVERAGE EXECUTION TIME OF THE MODEL TRAINING ALGORITHMS.

Model	Execution Time (in sec)
factor analyzed SHMM	3.36
diagonal SHMM	3.33
factor analyzed GHMM	2.50
diagonal GHMM	2.48

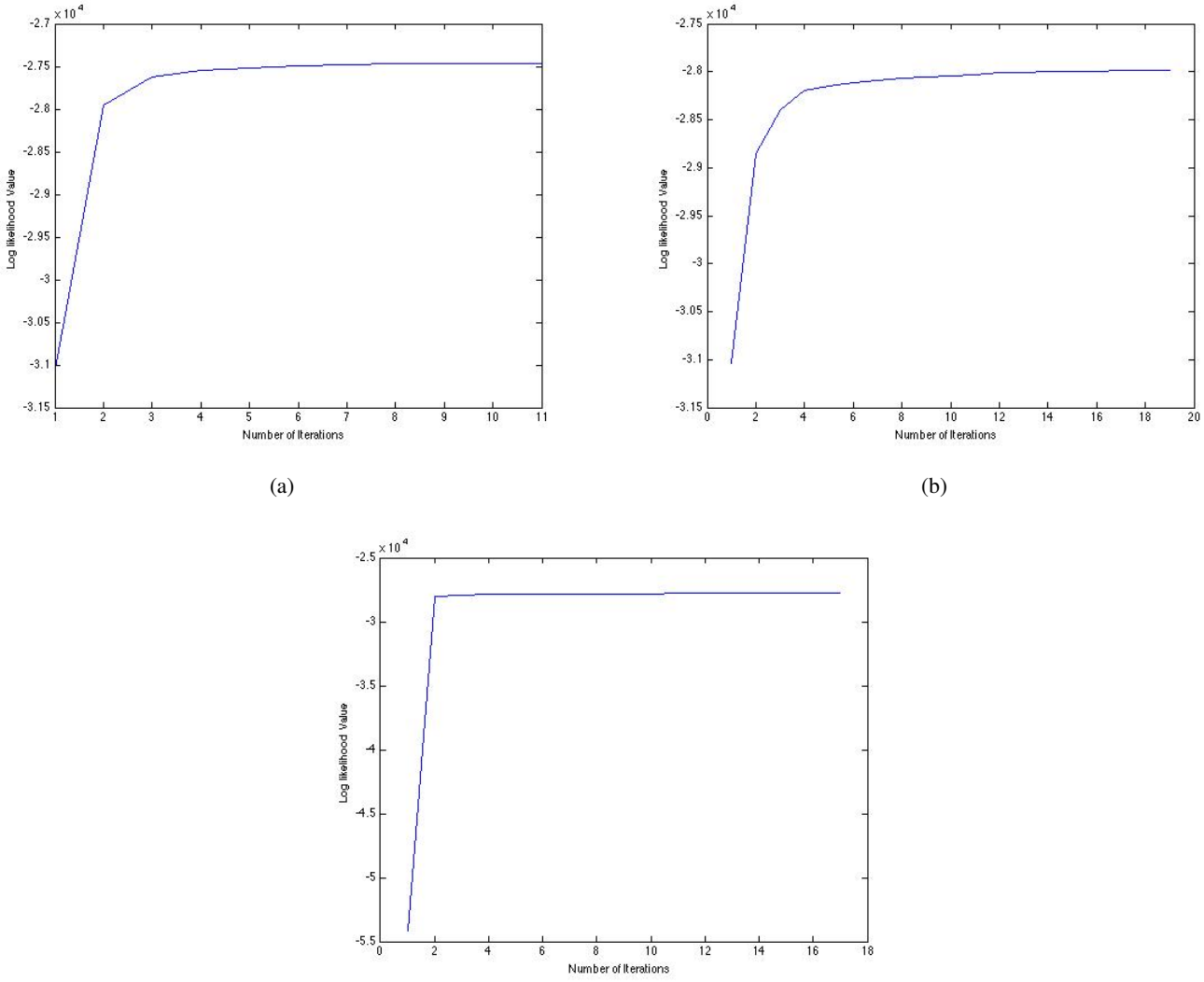


Figure 5. Cognitive state recognition from brain images: Convergence rates of the training algorithm for (a) the factor analyzed SHMM, (b) the diagonal SHMM, and (c) the diagonal GHMM methods.

considered methods. Following the suggestions of the creators of this data set, to train our classifiers we use only the data corresponding to the following Regions of Interest (ROI's): {'CALC' 'LIPL' 'LT' 'LTRIA' 'LOPER' 'LIPS' 'LDLPFC'}. Furthermore, from this data, we retain only the information regarding the p "most active" voxels, by application of the methodology described in [26].

To begin with, we observe that for values of $p > 70$, training of the considered models suffers from difficulties, as we often experience problems with singular or near-singular covariance matrix estimates. For this reason, we select $p = 70$ for this experiment. Concerning proper selection of the number of factors, q, for the factor analyzed SHMM, application of the popular BIC criterion yields $q = 9$. We underline here that the applicability of the BIC criterion in determination of the number of factors in mixtures of factor analyzers has been theoretically proved, given the number of the postulated factor analyzers is known [27]. In Table V, we provide the obtained average recognition error rates of the considered methods, and their standard deviations, over 30 runs of the EM algorithm (from different random starts). In Table VI, we provide the average execution time of the training algorithm for the considered methods. It is obvious that the factor analyzed SHMM completely outperforms its diagonal counterpart only for a nominal computational overhead. Finally, in Fig. 5, we compare the convergence rates of the considered methods. These figures are averages over the 30 runs of the EM algorithm and the two trained models (corresponding to the considered two cognitive states). As we also obtained in Section V.B, we observe that all the methods converge fast, with comparable convergence rates.

VI. Conclusions

In this paper, we proposed the use of Student's-t mixture models as the observation emission densities of continuous hidden Markov models, to offer a more robust methodology for sequential data modeling, comparing to conventional approaches. We provided a multiple token ML treatment of the proposed model, both for the case of full or diagonal covariance matrices, and for the case of factor analyzed covariance matrices. The experimental evaluation of our approach in applications from diverse domains has provided strong evidence towards the superior performance of the SHMM comparing to recently proposed HMM-

based alternatives for robust to outliers sequential data modeling, for comparable computational costs with the GHMM.

APPENDIX A. MULTIPLE TOKEN TREATMENT OF THE SHMM

From (12), and ignoring constant terms, we have

$$\sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{i=1}^{I}\gamma_{imt}^{(k)}E_{\boldsymbol{\Psi}^{(k)}}\left(\log p(\boldsymbol{y}_{mt}^{comp};\boldsymbol{\Theta}_i)|\boldsymbol{y}\right) = \sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{i=1}^{I}\sum_{j=1}^{J}r_{ijmt}^{(k)}\left[\log c_{ij} + Q_{1mt}(\nu_{ij};\boldsymbol{\Psi}^{(k)}) + Q_{2mt}(\boldsymbol{\mu}_{ij},\boldsymbol{\Sigma}_{ij};\boldsymbol{\Psi}^{(k)})\right] \tag{48}$$

where

$$Q_{1mt}(\nu_{ij};\boldsymbol{\Psi}^{(k)}) = \frac{\nu_{ij}}{2}E_{\boldsymbol{\Psi}^{(k)}}(\log u_{ijmt} - u_{ijmt}|\boldsymbol{y}_{mt}) - \log\Gamma\left(\frac{\nu_{ij}}{2}\right) + \frac{\nu_{ij}}{2}\log\left(\frac{\nu_{ij}}{2}\right) \tag{49}$$

$$Q_{2mt}(\boldsymbol{\mu}_{ij},\boldsymbol{\Sigma}_{ij};\boldsymbol{\Psi}^{(k)}) = -\frac{1}{2}\log|\boldsymbol{\Sigma}_{ij}| - \frac{1}{2}E_{\boldsymbol{\Psi}^{(k)}}(u_{ijmt}|\boldsymbol{y}_{mt})(\boldsymbol{y}_{mt}-\boldsymbol{\mu}_{ij})^T\boldsymbol{\Sigma}_{ij}^{-1}(\boldsymbol{y}_{mt}-\boldsymbol{\mu}_{ij}) \tag{50}$$

Hence, to effect the E-step of the multiple token treatment of the SHMM, we need to estimate the updates $\gamma_{imt}^{(k)}$, $\gamma_{himt}^{(k)}$, $\xi_{ijmt}^{(k)}$, $u_{ijmt}^{(k)}$, and $E_{\boldsymbol{\Psi}^{(k)}}\left(\log u_{ijmt}|\boldsymbol{y}_{mt}\right)$. The expressions of the posterior probabilities of the Markov chain, $\gamma_{imt}^{(k)}$ and $\gamma_{himt}^{(k)}$ are given by the forward-backward algorithm (16)-(21) [1], independently of the type of the observation emission probabilities of the Markov chain states. Bayes rule yields the expression of the state-conditional posteriors $\xi_{ijmt}^{(k)}$, given by (22). Finally, the latter two updates have been derived for the case of a single SMM in [4], and can be easily generalized for the SHMM model, yielding (23) and

$$E_{\boldsymbol{\Psi}^{(k)}}\left(\log u_{ijmt}|\boldsymbol{y}_{mt}\right) = \log u_{ijmt}^{(k)} - \log\left(\frac{\nu_{ij}^{(k)}+p}{2}\right) + \psi\left(\frac{\nu_{ij}^{(k)}+p}{2}\right) \tag{51}$$

As we observe, (51) implies that to derive $E_{\boldsymbol{\Psi}^{(k)}}\left(\log u_{ijmt}|\boldsymbol{y}_{mt}\right)$ it suffices to compute the $u_{ijmt}^{(k)}$, which eventually concludes the E-step of the EM fitting of the SHMM model. Concerning the M-step, the expressions of the Markov chain prior probabilities, $\pi_i^{(k)}$, and $\pi_{hi}^{(k)}$, can be found, e.g., in [2], for any type of the emission probabilities of the HMM states. The rest of the model parameters can be easily obtained

by maximization of (48), in a fashion similar to the case of a single SMM in the static setting [4].

APPENDIX B. MULTIPLE TOKEN TREATMENT OF THE FACTOR ANALYZED SHMM

From (42), we have (ignoring constant terms)

$$\begin{aligned}\sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{i=1}^{I}\gamma_{imt}^{(k)}E_{\boldsymbol{\Psi}^{(k)}}\left(\log p(\boldsymbol{y}_{mt}^{comp};\boldsymbol{\Theta}_i)|\boldsymbol{y}\right) = \\ \sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{i=1}^{I}\sum_{j=1}^{J}r_{ijmt}^{(k)}\left[\log c_{ij}+Q_{1mt}(\nu_{ij};\boldsymbol{\Psi}^{(k)})+Q_{2mt}(\boldsymbol{\mu}_{ij},\boldsymbol{B}_{ij},\boldsymbol{D}_{ij};\boldsymbol{\Psi}^{(k)})\right]\end{aligned} \tag{52}$$

where

$$Q_{1mt}(\nu_{ij};\boldsymbol{\Psi}^{(k)}) = -\log\Gamma\left(\frac{\nu_{ij}}{2}\right)+\frac{\nu_{ij}}{2}\log\left(\frac{\nu_{ij}}{2}\right)+\frac{\nu_{ij}}{2}E_{\boldsymbol{\Psi}^{(k)}}(\log u_{ijmt}-u_{ijmt}|\boldsymbol{y}_{mt}) \tag{53}$$

and

$$\begin{aligned}Q_{2mt}(\boldsymbol{\mu}_{ij},\boldsymbol{B}_{ij},\boldsymbol{D}_{ij};\boldsymbol{\Psi}^{(k)}) = &-\frac{1}{2}\log|\boldsymbol{D}_{ij}|-\frac{1}{2}E_{\boldsymbol{\Psi}^{(k)}}\left(u_{ijmt}|\boldsymbol{y}_{mt}\right)(\boldsymbol{y}_{mt}-\boldsymbol{\mu}_{ij})^T\boldsymbol{D}_{ij}^{-1}(\boldsymbol{y}_{mt}-\boldsymbol{\mu}_{ij}) \\ &+(\boldsymbol{y}_{mt}-\boldsymbol{\mu}_{ij})^T\boldsymbol{D}_{ij}^{-1}\boldsymbol{B}_{ij}E_{\boldsymbol{\Psi}^{(k)}}\left(u_{ijmt}\boldsymbol{x}_{ijmt}|\boldsymbol{y}_{mt}\right) \\ &-\frac{1}{2}\mathrm{tr}\left(\boldsymbol{B}_{ij}^T\boldsymbol{D}_{ij}^{-1}\boldsymbol{B}_{ij}E_{\boldsymbol{\Psi}^{(k)}}\left(u_{ijmt}\boldsymbol{x}_{ijmt}\boldsymbol{x}_{ijmt}^T|\boldsymbol{y}_{mt}\right)\right)\end{aligned} \tag{54}$$

Following results from [14], we have

$$\boldsymbol{x}_{ijmt}|\boldsymbol{y}_{mt},u_{ijmt}\sim\mathcal{N}(\boldsymbol{v}_{ij}^T(\boldsymbol{y}_{mt}-\boldsymbol{\mu}_{ij}),\boldsymbol{\omega}_{ij}/u_{ijmt}) \tag{55}$$

and, hence,

$$E_{\boldsymbol{\Psi}^{(k)}}\left(u_{ijmt}\boldsymbol{x}_{ijmt}|\boldsymbol{y}_{mt},u_{ijmt}\right) = u_{ijmt}^{(k)}\boldsymbol{v}_{ij}^{(k)^T}\left(\boldsymbol{y}_{mt}-\boldsymbol{\mu}_{ij}^{(k)}\right) \tag{56}$$

$$\begin{aligned}&E_{\boldsymbol{\Psi}^{(k)}}\left(u_{ijmt}\boldsymbol{x}_{ijmt}\boldsymbol{x}_{ijmt}^T|\boldsymbol{y}_{mt}\right) = \boldsymbol{\omega}_{ij}^{(k)}+ \\ &+u_{ijmt}^{(k)}\boldsymbol{v}_{ij}^{(k)^T}\left(\boldsymbol{y}_{mt}-\boldsymbol{\mu}_{ij}^{(k)}\right)\left(\boldsymbol{y}_{mt}-\boldsymbol{\mu}_{ij}^{(k)}\right)^T\boldsymbol{v}_{ij}^{(k)}\end{aligned} \tag{57}$$

where

$$\boldsymbol{v}_{ij}^{(k)} = (\boldsymbol{B}_{ij}^{(k)}\boldsymbol{B}_{ij}^{(k)^T}+\boldsymbol{D}_{ij}^{(k)})^{-1}\boldsymbol{B}_{ij}^{(k)} \tag{58}$$

$$\boldsymbol{\omega}_{ij}^{(k)} = \boldsymbol{I}_q - \boldsymbol{\upsilon}_{ij}^{(k)^T} \boldsymbol{B}_{ij}^{(k)} \tag{59}$$

Therefore, to effect the E-step on the second cycle on the $(k+1)$th iteration of the algorithm, apart from $\gamma_{imt}^{(k)}$, $\gamma_{himt}^{(k)}$, $\xi_{ijmt}^{(k)}$, and $u_{ijmt}^{(k)}$, we also need to compute the factor analysis quantities $\boldsymbol{\upsilon}_{ij}^{(k)}$ and $\boldsymbol{\omega}_{ij}^{(k)}$. Concerning the M-step on the second cycle on the $(k+1)$th iteration of the algorithm, we have

$$\frac{\partial (\boldsymbol{y}_{mt} - \boldsymbol{\mu}_{ij})^T \boldsymbol{D}_{ij}^{-1} \boldsymbol{B}_{ij} E_{\boldsymbol{\Psi}^{(k)}} \left(u_{ijmt} \boldsymbol{x}_{ijmt} | \boldsymbol{y}_{mt} \right)}{\partial \boldsymbol{B}_{ij}} = \boldsymbol{D}_{ij}^{-1} (\boldsymbol{y}_{mt} - \boldsymbol{\mu}_{ij}) E_{\boldsymbol{\Psi}^{(k)}} \left(u_{ijmt} \boldsymbol{x}_{ijmt} | \boldsymbol{y}_{mt} \right)^T \tag{60}$$

$$\frac{\partial \mathrm{tr} \left(\boldsymbol{B}_{ij}^T \boldsymbol{D}_{ij}^{-1} \boldsymbol{B}_{ij} E_{\boldsymbol{\Psi}^{(k)}} \left(u_{ijmt} \boldsymbol{x}_{ijmt} \boldsymbol{x}_{ijmt}^T | \boldsymbol{y}_{mt} \right) \right)}{\partial \boldsymbol{B}_{ij}} = 2 \boldsymbol{D}_{ij}^{-1} \boldsymbol{B}_{ij} E_{\boldsymbol{\Psi}^{(k)}} \left(u_{ijmt} \boldsymbol{x}_{ijmt} \boldsymbol{x}_{ijmt}^T | \boldsymbol{y}_{mt} \right) \tag{61}$$

from which, it is easy to show that the solution of $\partial Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)}) / \partial \boldsymbol{B}_{ij} = 0$ yields (45). In the same fashion, concerning the estimate of $\boldsymbol{D}_{ij}$, using (54) we deduce that it is easier to derive the partial derivative of $Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)})$ with respect to $\boldsymbol{D}_{ij}^{-1}$. Then, we have

$$\frac{\partial \log |\boldsymbol{D}_{ij}|}{\partial \boldsymbol{D}_{ij}^{-1}} = -\boldsymbol{D}_{ij} \tag{62}$$

$$\frac{\partial (\boldsymbol{y}_{mt} - \boldsymbol{\mu}_{ij})^T \boldsymbol{D}_{ij}^{-1} (\boldsymbol{y}_{mt} - \boldsymbol{\mu}_{ij})}{\partial \boldsymbol{D}_{ij}^{-1}} = (\boldsymbol{y}_{mt} - \boldsymbol{\mu}_{ij}) (\boldsymbol{y}_{mt} - \boldsymbol{\mu}_{ij})^T \tag{63}$$

$$\frac{\partial (\boldsymbol{y}_{mt} - \boldsymbol{\mu}_{ij})^T \boldsymbol{D}_{ij}^{-1} \boldsymbol{B}_{ij} E_{\boldsymbol{\Psi}^{(k)}} \left(u_{ijmt} \boldsymbol{x}_{ijmt} | \boldsymbol{y}_{mt} \right)}{\partial \boldsymbol{D}_{ij}^{-1}} = (\boldsymbol{y}_{mt} - \boldsymbol{\mu}_{ij}) E_{\boldsymbol{\Psi}^{(k)}} \left(u_{ijmt} \boldsymbol{x}_{ijmt} | \boldsymbol{y}_{mt} \right)^T \boldsymbol{B}_i^T \tag{64}$$

$$\frac{\partial \mathrm{tr} \left(\boldsymbol{B}_{ij}^T \boldsymbol{D}_{ij}^{-1} \boldsymbol{B}_{ij} E_{\boldsymbol{\Psi}^{(k)}} \left(u_{ijmt} \boldsymbol{x}_{ijmt} \boldsymbol{x}_{ijmt}^T | \boldsymbol{y}_{mt} \right) \right)}{\partial \boldsymbol{D}_{ij}^{-1}} = \boldsymbol{B}_{ij} E_{\boldsymbol{\Psi}^{(k)}} \left(u_{ijmt} \boldsymbol{x}_{ijmt} \boldsymbol{x}_{ijmt}^T | \boldsymbol{y}_{mt} \right) \boldsymbol{B}_{ij}^T \tag{65}$$

from which we directly obtain that $\partial Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)}) / \partial \boldsymbol{D}_{ij}^{-1} = 0$ yields the maximizer (46).

REFERENCES

[1] O. Cappé, E. Moulines, and T. Rydén, *Inference in Hidden Markov Models*. New York: Springer Series in Statistics, 2005.

[2] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, pp. 245–255, 1989.

[3] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society, B*, vol. 39, no. 1, pp. 1–38, 1977.

[4] D. Peel and G. McLachlan, "Robust mixture modeling using the t distribution," *Statistics and Computing*, vol. 10, pp. 335–344, 2000.

[5] S. Chatzis, D. Kosmopoulos, and T. Varvarigou, "Signal modeling and classification using a robust latent space model based on t distributions," *IEEE Transactions on Signal Processing*, vol. 56, no. 3, pp. 949–963, 2008.

[6] M. Svensén and C. M. Bishop, "Robust Bayesian mixture modelling," *Neurocomputing*, vol. 64, pp. 235–252, 2005.

[7] S. Chatzis and T. Varvarigou, "Factor analysis latent subspace modeling and robust fuzzy clustering using t distributions," *IEEE Transactions on Fuzzy Systems*, Accepted for future publication.

[8] L. Arslan and J. Hansen, "Selective training for hidden Markov models with applications to speech classification," *IEEE Transactions on Speech and Audio Processing*, vol. 7, no. 1, pp. 46–54, 1999.

[9] C.-H. Lee and S.-Y. Lee, "Noise-robust speech recognition using top-down selective attention with an HMM classifier," *IEEE Signal Processing Letters*, vol. 14, no. 7, pp. 489–491, 2007.

[10] J.-T. Chien and C.-P. Liao, "Maximum confidence hidden Markov modeling for face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 4, pp. 606–616, 2008.

[11] F. Sha and L. K. Saul, "Large margin hidden Markov models for automatic speech recognition," in *Advances in Neural Information Processing Systems 19*, B. Schölkopf, J. Platt, and T. Hoffman, Eds. Cambridge, MA: MIT Press, 2007, pp. 1249–1256.

[12] C. Liu and D. Rubin, "ML estimation of the t distribution using EM and its extensions, ECM and ECME," *Statistica Sinica*, vol. 5, no. 1, pp. 19–39, 1995.

[13] D. Andrews and C. Mallows, "Scale mixtures of normal distributions," *J. Royal Stat. Soc. B*, vol. 36, pp. 99–102, 1974.

[14] G. McLachlan, R. Bean, and L. B.-T. Jones, "Extension of the mixture of factor analyzers model to incorporate the multivariate t-distribution," *Comp. Stat. Data Analysis*, vol. 51, no. 11, pp. 5327–5338, 2006.

[15] C. Archambeau and M. Verleysen, "Robust Bayesian clustering," *Neural Networks*, vol. 20, pp. 129–138, 2007.

[16] S. Dharanipragada and K. Visweswariah, "Gaussian mixture models with covariances or precisions in shared multiple subspaces," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, no. 4, pp. 1255–1266, 2006.

[17] J. Lindblom and J. Samuelsson, "Bounded support Gaussian mixture modeling of speech spectra," *IEEE Transactions on Speech and Audio Processing*, vol. 11, no. 1, pp. 88–99, 2003.

[18] J. Frankel and S. King, "Speech recognition using linear dynamic models," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, no. 1, pp. 246–256, 2007.

[19] M. Tipping and C. Bishop, "Mixtures of probabilistic principal component analyzers," *Neural Computation*, vol. 11, no. 2, pp. 443–482, 1999.

[20] Z. Ghahramani and G. Hinton, "The EM algorithm for mixtures of factor analyzers," Department of Computer Science, University of Toronto, Toronto, Canada, M5S 1A4, Tech. Rep. CRGTR- 96-1, 1997.

[21] X. Meng and D. van Dyk, "The EM algorithm - an old folk song sung to a fast new tune (with discussion)," *Journal of the Royal*

Statistical Society B, vol. 59, no. 3, pp. 511–567, 1997.

[22] D. Kosmopoulos and I. Maglogiannis, "Hand tracking for gesture recognition tasks using dynamic Bayesian network," *International Journal of Intelligent Systems and Applications*, vol. 1, no. 3/4, pp. 359–375, 2006.

[23] R. Mukundan and K. R. Ramakrishnan, *Moment Functions in Image Analysis: Theory and Applications.* World Scientific, 1998.

[24] L. F. Lamel, R. H. Kassel, and S. Seneff, "Speech database development: design and analsysis of the acoustic-phonetic corpus," in *Proceedings of the DARPA Speech Recognition Workshop*, 1986, pp. 100–109.

[25] K. F. Lee and H. W. Hon, "Speaker-independent phone recognition using hidden Markov models," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 11, pp. 1641–1648, 1988.

[26] T. M. Mitchell, R. Hutchinson, R. S. Niculescu, F. Pereira, X. Wang, M. Just, and S. Newman, "Learning to decode cognitive states from brain images," *Machine Learning*, vol. 57, no. 1, pp. 145–175, 2004.

[27] G. McLachlan and D. Peel, *Finite Mixture Models*. Wiley Series in Probability and Statistics, 2000.