Modified student's t-hidden Markov model for pattern recognition and classification

Article *in* IET Signal Processing · May 2013

DOI: 10.1049/iet-spr.2012.0315

3 authors:

Hui Zhang
University of Windsor
41 PUBLICATIONS **494** CITATIONS
SEE PROFILE

Q. M. Jonathan Wu
University of Windsor
364 PUBLICATIONS **3,799** CITATIONS
SEE PROFILE

Thanh Minh Nguyen
University of Windsor
54 PUBLICATIONS **467** CITATIONS
SEE PROFILE

Published in IET Signal Processing
Received on 25th October 2012
Revised on 11th February 2013
Accepted on 24th February 2013
doi: 10.1049/iet-spr.2012.0315

ISSN 1751-9675

Modified student's *t*-hidden Markov model for pattern recognition and classification

Hui Zhang[1,2,3], *Qing Ming Jonathan Wu*[2], *Thanh Minh Nguyen*[2]

[1]*Jiangsu Engineering Center of Network Monitoring, Nanjing University of Information Science and Technology, People's Republic of China*
[2]*Department of Electrical and Computer Engineering, University of Windsor, 401 Sunset Avenue, Windsor, ON, Canada N9B 3P4*
[3]*School of Computer and Software, Nanjing University of Information Science & Technology, Nanjing, People's Republic of China*
E-mail: jwu@uwindsor.ca

Abstract: The Gaussian hidden Markov model has been successfully used in pattern recognition and classification applications; however, recently the Student's *t*-mixture model is regarded as an alternative to Gaussian mixture models, as it is more robust for outliers. The model using Student's *t*-mixture distribution as its hidden state is the Student's *t*-hidden Markov model (SHMM). The authors propose a novel Student's *t*-hidden Markov model, which considers the relationship among Markov states, latent components and observations by introducing a regularising scalar exponent in the component densities of the model's emission densities. Moreover, the standard SHMM can be considered as a special case of the modified SHMM with the selection of proper parameter values. Finally, the authors adopt the gradient method to estimate optimal weight parameters. Simultaneously, the expectation–maximisation algorithm is used to fit the modified SHMM. Thus, our model is simple and easy to implement. The experimental results using synthetic and real data demonstrate the improved robustness of the proposed approach.

1 Introduction

Markov models have been widely applied in a series of clustering problems. A special type of Markov model is the hidden Markov model, which provides a convenient way to model sequential observations and tends to cluster among different possible components. Hidden Markov models (sometimes called hidden Markov chains) have been widely used in many applications, such as speech recognition [1, 2], radar image classification [3], active learning [4], image segmentation [5], sequential data modelling [6], and so on. In the literature on hidden Markov models (HMM), the Gaussian mixture model (GMM) is most commonly selected as the hidden state observation emission distribution model, and is referred to as the Gaussian hidden Markov model (GHMM).

As is a well-known method, the GHMM has been successfully and widely used in many applications [7, 8], [9, 10]. GMMs have been widely used in statistical pattern recognition and classification applications [11]. They provide an appealing alternative to non-parametric density estimators for the approximation of unknown distributions, including distributions with multiple modes [12]. In statistical signal modelling and classification applications, their popularity stems from their provision of a sound statistical framework for the approximation of non-Gaussian multimodal distributions.

The major advantage of the GMM [13–17] is that it provides a natural way to cluster data based on the components of the mixture that generated it. Another advantage is that it requires a small number of parameters, and it is simple and easy to implement. However, the GMM is weak in the case of heavy outliers; the model construction may be influenced by the noises. In many practical applications, the robustness for outlying data is crucial, because the mixture of outliers and observable data may severely affect the estimation of the model parameters as well as the model complexity. Thus, additional components are needed to capture the tails of the distributions. When outliers are present in the available fitting data sets (as often happens in real-world applications), GMMs tend to require excessively high numbers of mixture components to capture the long tails of the approximated distributions so as to retain their pattern recognition effectiveness [6]. Given the finite data set with outliers, GMM with higher model order can be used to achieve the similar performance as SMM. However, this higher order model is complex, time-consuming and may cause over fitting.

Student's *t*-mixture model (SMM), recently proposed by Peel and McLachlan [17] as an alternative to GMMs,

provides high robustness against noise. The SMM is more robust than the GMM for heavier tails, and the distribution for each component in the SMM is embedded in a wider class of elliptically symmetric distributions with an additional parameter called degrees of freedom. Hence, the finite mixture model of the long-tailed multivariate Student's *t*-distribution provides a much more robust approach than the GMM.

The significant success of SMM for real training data is experimentally shown in many recent literatures [18–21], which demonstrate that the SMM can model the hidden patterns of the data well, especially in the case of the heavy outliers where GMMs fail. The HMM based on this model is called Student's *t*-HMM (SHMM), in which the hidden state observation emission distributions are modelled by the SMM. The conventional method for estimating the parameters of GHMM and SHMM employs a simple and computationally efficient maximum likelihood (ML) estimation based on either the expectation–maximisation (EM) algorithm [6, 17] or iterative conditional estimation (ICE) [3].

In this paper, we propose a modified SHMM model, considering the relationship among Markov states, latent components and observations by weight parameter β_{ijt} to increase the performance of traditional SHMM for clustering application. Our method is similar to the standard SHMM, and the EM algorithm still can be used for parameter learning. We then adopt the gradient method to estimate these additional optimal parameter β_{ijt}. In Section 2, we briefly review the necessary mathematical background and introduce the modified SHMM. The EM algorithm and the gradient method used for the estimation of parameters of modified SHMM are presented in Section 3. The experimental results are described in Section 4, and Section 5 concludes the paper.

2 Student's *t* distribution

The Student's *t*-distribution provides a heavy-tailed alternative to the Gaussian distribution for potential outliers and therefore can produce a clustering algorithm which is more robust to outliers. The probability density function (pdf) of a Student's *t*-distribution, with mean μ, covariance Σ and degrees of freedom v is [22] (see (1))

$$t(x|\mu, \Sigma, v) = \frac{\Gamma(v/2 + p/2)}{\Gamma(v/2)} \frac{|\Sigma|^{-1/2}}{(\pi v)^{p/2}} \left[1 + \frac{(x-\mu)^T \Sigma^{-1} (x-\mu)}{v}\right]^{-(v+p)/2} \quad (1)$$

where p is the dimensionality of the observations x and $\Gamma(s)$ is the Gamma function

$$\Gamma(s) = \int_0^\infty t^{s-1} e^{-t}\, \mathrm{d}t \quad (2)$$

From [11, 17], it can be shown that the Student's *t*-distribution $t(\mu, \Sigma, v)$ is equivalent to a Gaussian distribution $\mathcal{N}(\mu, \Sigma/u)$, with the precision scaling factor u, given as

$$t(x|\mu, \Sigma, v) \sim \mathcal{N}(x|\mu, \Sigma/u)p(u) \quad (3)$$

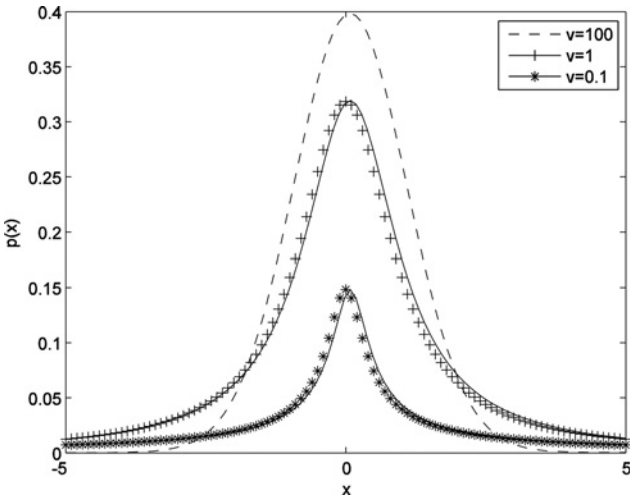

Fig. 1 *Univariate Student's t-distribution with mean = 0 and standard deviation = 1 for various degrees of freedom. As $v \to \infty$ the distribution tends to a Gaussian*

For small values of v, the distribution has heavier tails than a Gaussian

where

$$\mathcal{N}(x|\mu, \Sigma/u) = \left(\frac{u}{2\pi}\right)^{p/2} |\Sigma|^{-1/2} \exp\left[-\frac{u(x-\mu)^T \Sigma^{-1} (x-\mu)}{2}\right] \quad (4)$$

where mean μ and covariance Σ are defined as the same in (1).

Fig. 1 shows the graphical illustration of the Student's *t*-distribution, with mean and standard deviation fixed and for various values of the degrees of freedom v. It can be shown that for $v \to \infty$, the Student's *t*-distribution tends to a Gaussian distribution with the same mean and standard deviation. On the contrary, as v tends to zero, the tails of the distribution become longer, providing a heavy-tailed alternative for potential outliers without affecting the mean or the standard deviation of the distribution. Thus, SMM can provide better performance, better algorithm stability and improved model parameter estimates compared to GMM.

3 Proposed method

This section first introduces the modified SHMM, which incorporates Markov states, latent components and observations with the weight parameter β_{ijt}. Then, we apply the EM algorithm and gradient method for the estimation of parameters.

3.1 Modified Student's t-hidden Markov model

Our idea is based on the previous work in [23]. In their work, the authors introduced a new fuzzy clustering approach by regularising the fuzzy objective function with Kullback–Leibler (KL) divergence. According to [23], the relationship between finite mixture model and fuzzy *c*-means algorithm

doi: 10.1049/iet-spr.2012.0315

can be generated by introducing the KL regularisation. Density estimation with Gaussian mixture model can be considered as a probabilistic approach to clustering, which is strictly confined to Gaussian distributions. On the other hand, fuzzy c-means clustering with regularisation by KL information can be seen as a non-Gaussian version of density estimation [24]. Based on their work, we combined the hidden Markov model and finite mixture model with Student's t-distribution and introduced a modified SHMM (MSHMM). The main contribution of this paper has two parts: (i) Considering HMM for sequential data processing, meanwhile, Student's t-distribution can produce a clustering algorithm which is more robust to outliers. (ii) The parameter λ introduced in [23] equals to β in our paper. However, we define β_{ijt} to incorporate the information among Markov states, latent components and observations. Thus, [23] is a special case of our algorithm when we set β_{ijt} = constant value λ.

Let us consider an N-state modified HMM for all observed data $X = \{x_t\}_{t=1}^{\mathrm{T}}$, where all states have the same number of mixture components, J, and where the hidden emission density of each ith state is defined by Student's t-mixture distribution. Thus, a j-component Student's t-distribution, with mixing proportion c_{ij} belongs to the ith model state, given by

$$p(x_t|\Theta_{it}) = \sum_{j=1}^{J} c_{ij}\gamma_{ijt}t\left(x_t|\mu_{ij}, \Sigma_{ij}, v_{ij}\right)^{\beta_{ijt}} \tag{5}$$

where $\boldsymbol{X} = (x_1, \ldots, x_T)^{\mathrm{T}}$ denotes the observed-data vector and $\Theta_i = \left(c_{ij}, \mu_{ij}, \Sigma_{ij}, v_{ij}\right)_{j=1}^{J}$ are the parameters of the mixture of the jth component ($j = 1, \ldots, J$) and ith model state ($i = 1, \ldots, N$), and γ_{ijt} is the normalised parameter (see equation at the bottom of the page)

satisfying the condition $\int_{-\infty}^{+\infty} \gamma t(x|\mu, \Sigma, v)^{\beta}\,\mathrm{d}x = 1$.

The standard SHMM assumes the conditional probability satisfies the Student's t-distribution. Here, we assume it satisfies the Student's t-distribution to power beta. According to (5), the weight parameters β are incorporated into SHMM to generate a modified SHMM (MSHMM). The MSHMM is quite similar to the standard SHMM, and is thus easy to implement. It can be easily seen that our model degrades to SHMM when β equals one for i, j and t. Thus, SHMM is a special case of our method. With the scaling factor u, the Student's t-distribution can be represented as a mixture of Gaussians with the same mean and with scaled precision

$$p\left(x_t|u_{ijt}, \Theta_{it}\right) = \sum_{j=1}^{J} c_{ij}\gamma_{ijt}\mathcal{N}\left(x_t|\mu_{ij}, \Sigma_{ij}/u_{ijt}\right)^{\beta_{ijt}} \tag{6}$$

where the scaling factors u_{ijt} follow the Gamma distribution, which depends only on the degree of freedom v_{ij} for observation x_t corresponding to the jth component density and ith hidden state distribution

$$u_{ijt} \sim \mathcal{G}\left(\frac{v_{ij}}{2}, \frac{v_{ij}}{2}\right) \tag{7}$$

Let $\{s_{it}\}_{I\times T}$ be a set of state-indicator labels of the observed data, where $s_{it} = 1$ if x_t is emitted from the ith model state, and $s_{it} = 0$, otherwise. Let us denote π_i as the initial state probabilities, and π_{hi} as the state transition probabilities of the Markov chain; thus, the complete data log-likelihood can be written as

$$\begin{aligned}\log L(\Psi) = &\sum_{h=1}^{I} s_{h1}\log\pi_h + \sum_{h=1}^{I}\sum_{i=1}^{I}\sum_{t=1}^{T-1} s_{ht}s_{i,t+1}\log\pi_{hi} \\ &+ \sum_{i=1}^{I}\sum_{t=1}^{T} s_{it}\log p\left(x_t^{\mathrm{comp}}|\Theta_{it}\right)\end{aligned} \tag{8}$$

where the model parameter vector $\Psi = \left\{\Theta_{it}, \pi_i, \pi_{hi}\right\}_{h,i=1}^{I}$ and x_t^{comp} are the complete data of x_t with consideration of latent components. $\log p\left(x_t^{\mathrm{comp}}|\Theta_{it}\right)$ is the complete data log-likelihood of x_t corresponding to the ith hidden state.

The closed-form solution for log-likelihood optimisation of a Student's t-mixture does not exist. However, with the help of scaled factors, the Student's t-distribution can be represented as the Gaussian distribution with a Gamma-distributed scaled factor. Let $\{z_{ijt}\}_{I\times J\times T}$ be a set of state-conditional missing (latent) component labels, where $z_{ijt} = 1$ if x_t is generated from the jth component and emitted from the ith model state, and $z_{ijt} = 0$ otherwise. Thus, we have

$$p\left(x_t^{\mathrm{comp}}|\Theta_{it}\right) = \prod_{j=1}^{J}\left[c_{ij}\gamma_{ijt}\mathcal{N}\left(x_t|\mu_{ij}, \Sigma_{ij}/u_{ijt}\right)^{\beta_{ijt}} p\left(u_{ijt}|v_{ij}\right)^{\beta_{ijt}}\right]^{z_{ijt}} \tag{9}$$

where (see (10))

$$p\left(u_{ijt}|v_{ij}\right) = \frac{1}{\Gamma\left(v_{ij}/2\right)}\left(v_{ij}/2\right)^{v_{ij}/2}\left(u_{ijt}\right)^{v_{ij}/2-1}e^{-u_{ijt}v_{ij}/2} \tag{11}$$

$$\gamma_{ijt} = \left[\frac{\Gamma\left(v_{ij}/2\right)}{\Gamma\left(v_{ij}/2 + p/2\right)}\right]^{\beta_{ijt}} \frac{\Gamma\left(\left(\beta_{ijt}v_{ij} + \beta_{ijt}p\right)/2\right)(\pi)^{p(\beta_{ijt}-1)/2}v_{ij}^{(\beta_{ijt}p-1)/2}}{\Gamma\left(\left(\beta_{ijt}v_{ij} + \beta_{ijt}p - p\right)/2\right)\left(\beta_{ijt}v_{ij} + \beta_{ijt}p - p\right)^{(p-1)/2}\Sigma_{ij}^{-(\beta_{ijt}-1)/2}}$$

$$\mathcal{N}\left(x_t|\mu_{ij}, \Sigma_{ij}/u_{ijt}\right) = \frac{\left|\Sigma_{ij}\right|^{-1/2}u_{ij}^{p/2}}{(2\pi)^{p/2}}\exp\left\{-\frac{1}{2}u_{ijt}\left(x_t - \mu_{ij}\right)^{T}\Sigma_{ij}^{-1}\left(x_t - \mu_{ij}\right)\right\} \tag{10}$$

From (10) and (11), we have (ignoring constant terms) (see (12))

A simple method to choose weight parameters β is to predefine them by experience. In this paper, we use the gradient method to obtain the optimal weight parameter values, as given in the next section. Here, we first evaluate the initial weight parameter values. According to [25], there are many ways to determine the initial weight parameters β. One possible selection is

$$\beta_{ijt} = \alpha + \exp\left(-\left|\log t\left(x_t|\mu_{ij}, \Sigma_{ij}, v_{ij}\right) - \log t\left(x_t|\mu_{ij'}, \Sigma_{ij'}, v_{ij'}\right) + \gamma\right|\right) \tag{13}$$

where 'j' is the component number with the maximum likelihood

$$\log t\left(x_t|\mu_{ij'}, \sum_{ij'}, v_{ij'}\right) \geq \log t\left(x_t|\mu_{ij}, \sum_{ij'}, v_{ij}\right), \quad \text{for } j, j' = 1, 2, \ldots, J \tag{14}$$

In our experiments, we selected a value of 5 for γ and a value of 0.1 for α. It is noticed that a special case of (13) is $\alpha = 1$, $\gamma = \pm\infty$, thus $\beta_{ijt} = 1$, which corresponds to the conventional SHMM model.

3.2 Parameters learning

This subsection is to optimise the parameter set $\Psi = \left\{\pi_i, \pi_{hi}, c_{ij}, \beta_{ijt}, \mu_{ij}, \Sigma_{ij}, v_{ij}\right\}_{h,i,j=1}^{I,I,J}$ in order to maximise the log-likelihood function in (8). To maximise this function, we apply the EM algorithm, which requires calculation of the quantity Q, the current conditional expectation of the complete data log-likelihood function. We first define the state emission and state transition posterior probabilities as follows:

The E-step

$$\omega_{it} = p\left(s_{it} = 1|x_t\right) \tag{15}$$

$$\omega_{hit} = p\left(s_{i,t+1} = 1, s_{ht} = 1|x_t\right) \tag{16}$$

With the help of (15) and (16), the posterior expectation $Q(\Psi)$ of the complete data log-likelihood function, $\log L(\Psi)$, can be calculated as

$$\begin{aligned} Q(\Psi) &= E(\log L(\Psi)) \\ &= \sum_{h=1}^{I} \omega_{h1}^{(k)} \log \pi_h + \sum_{h=1}^{I}\sum_{i=1}^{I}\sum_{t=1}^{T-1} \omega_{hit}^{(k)} \log \pi_{hi} \\ &\quad + \sum_{i=1}^{I}\sum_{t=1}^{T} \omega_{it}^{(k)} E^{(k)}\left(\log p\left(x_t^{\text{comp}}; \Theta_{it}\right)\right) \end{aligned} \tag{17}$$

where k denotes the kth iteration estimation of the corresponding quantities, and (see (18))

Here, the quantities ω_{it} and ω_{hit} can be easily deduced by utilising the forward–backward algorithm [1].

The conditional posterior probabilities of latent component z_{ijt} can be calculated as

$$\begin{aligned} \xi_{ijt} &= E\left(z_{ijt}\right) = p\left(z_{ijt} = 1|x_t, s_{it} = 1\right) \\ &= \frac{p\left(z_{ijt} = 1, x_t|s_{it} = 1\right)}{p\left(x_t|s_{it} = 1\right)} \end{aligned} \tag{19}$$

Using (5), we have

$$\xi_{ijt}^{(k)} = \frac{c_{ij}^{(k)} \gamma_{ijt}^{(k)} t\left(x_t|\mu_{ij}^{(k)}, \Sigma_{ij}^{(k)}, v_{ij}^{(k)}\right)^{\beta_{ijt}^{(k)}}}{\sum_{h=1}^{J} c_{ih}^{(k)} \gamma_{ijt}^{(k)} t\left(x_t|\mu_{ih}^{(k)}, \Sigma_{ih}^{(k)}, v_{ih}^{(k)}\right)^{\beta_{ijt}^{(k)}}} \tag{20}$$

From [17], the posterior probabilities of scaled factors and its log function can be calculated as

$$E\left(u_{ijt}\right) = u_{ijt}^{(k)} = \frac{v_{ij}^{(k)} + p}{v_{ij}^{(k)} + \left(x_t - \mu_{ij}^{(k)}\right)^T \Sigma_{ij}^{-(k)} \left(x_t - \mu_{ij}^{(k)}\right)} \tag{21}$$

$$E\left(\log u_{ijt}\right) = \log u_{ijt}^{(k)} - \log\left(\frac{v_{ij}^{(k)} + p}{2}\right) + \psi\left(\frac{v_{ij}^{(k)} + p}{2}\right) \tag{22}$$

The M-step is calculated by maximising the posterior expectation $Q(\Psi)$ of the complete data log-likelihood. The initial state probabilities can be obtained by maximising the posterior expectation $Q(\Psi)$ over π_h under the constraint $\sum_{h=1}^{I} \pi_h = 1$ for arbitrary h, according to the property of the Markov chain, with the consideration of the Lagrange

$$\begin{aligned} \log p\left(x_t^{\text{comp}}|\Theta_{it}\right) = \sum_{j=1}^{J} z_{ijt} \Bigg\{ &\log c_{ij} + \log \gamma_{ijt} + \beta_{ijt}\left[-\frac{1}{2}\log\left|\Sigma_{ij}\right| - \frac{1}{2} u_{ijt}\left(x_t - \mu_{ij}\right)^T \Sigma_{ij}^{-1}\left(x_t - \mu_{ij}\right)\right] \\ &+ \beta_{ijt}\left[-\log\Gamma\left(\frac{v_{ij}}{2}\right) + \frac{v_{ij}}{2}\left(\log\left(\frac{v_{ij}}{2}\right) + \log u_{ijt} - u_{ijt}\right)\right]\Bigg\} \end{aligned} \tag{12}$$

$$\begin{aligned} &E^{(k)}\left(\log p\left(x_t^{\text{comp}}; \Theta_{it}\right)\right) \\ &= \sum_{j=1}^{J} E\left(z_{ijt}\right)\Bigg\{\log c_{ij}^{(k)} + \log \gamma_{ijt}^{(k)} + \beta_{ijt}^{(k)}\left[-\frac{1}{2}\log\left|\Sigma_{ij}^{(k)}\right| - \frac{1}{2} E\left(u_{ijt}\right)\left(x_t - \mu_{ij}^{(k)}\right)^T \Sigma_{ij}^{-(k)}\left(x_t - \mu_{ij}^{(k)}\right)\right] \\ &\quad + \beta_{ijt}^{(k)}\left[-\log\Gamma\left(\frac{v_{ij}^{(k)}}{2}\right) + \frac{v_{ij}^{(k)}}{2}\left(\log\left(\frac{v_{ij}^{(k)}}{2}\right) + E\left(\log u_{ijt}\right) - E\left(u_{ijt}\right)\right)\right]\Bigg\} \end{aligned} \tag{18}$$

doi: 10.1049/iet-spr.2012.0315

constraint condition. Thus, we have

$$\frac{\partial}{\partial \pi_h^{(k+1)}}\left(Q(\Psi)-\lambda\left(\sum_{i=1}^{I}\pi_h^{(k+1)}-1\right)\right)=0$$

which yields

$$\pi_h^{(k+1)}=\omega_{h1}^{(k)} \tag{23}$$

Similarly, we have

$$\frac{\partial}{\partial \pi_{hi}^{(k+1)}}\left(Q(\Psi)-\sum_{h=1}^{I}\lambda_h\left(\sum_{i=1}^{I}\pi_{hi}^{(k+1)}-1\right)\right)=0$$

which yields

$$\pi_{hi}^{(k+1)}=\frac{\sum_{t=1}^{T-1}\omega_{hit}^{(k)}}{\sum_{t=1}^{T-1}\sum_{i=1}^{I}\omega_{hit}^{(k)}} \tag{24}$$

and the mixing proportion c_{ij} can be calculated as

$$\frac{\partial}{\partial c_{ij}^{(k+1)}}\left(Q(\Psi)-\lambda\left(\sum_{j=1}^{J}c_{ij}^{(k+1)}-1\right)\right)=0$$

which yields

$$c_{ij}^{(k+1)}=\frac{\sum_{t=1}^{T}\omega_{it}^{(k)}\xi_{ijt}^{(k)}}{\sum_{t=1}^{T}\sum_{j=1}^{J}\omega_{it}^{(k)}\xi_{ijt}^{(k)}} \tag{25}$$

To derive the expression for μ_{ij}, we have to perform the maximisation of Q with respect to μ_{ij}. From (17) and (18), we have

$$\frac{\partial Q(\Psi)}{\partial \mu_{ij}^{(k+1)}}=\frac{1}{2}\sum_{t=1}^{T}\omega_{it}^{(k)}\beta_{ijt}^{(k)}\xi_{ijt}^{(k)}u_{ijt}^{(k)}\left(2\Sigma_{ij}^{-(k)}\mu_{ij}^{(k)}-2\Sigma_{ij}^{-(k)}x_t\right) \tag{26}$$

The solution of $\partial Q(\Psi)/\partial \mu_{ij}^{(k+1)}=0$ yields the estimation of μ_{ij}

$$\mu_{ij}^{(k+1)}=\frac{\sum_{t=1}^{T}\omega_{it}^{(k)}\beta_{ijt}^{(k)}\xi_{ijt}^{(k)}u_{ijt}^{(k)}x_t}{\sum_{t=1}^{T}\omega_{it}^{(k)}\beta_{ijt}^{(k)}\xi_{ijt}^{(k)}u_{ijt}^{(k)}} \tag{27}$$

Next, when we consider the partial derivative of Q with respect to Σ_{ij} at the $(t+1)$ iteration step, we have (see (28))

Let $\partial Q(\Psi)/\partial \Sigma_{ij}^{(k+1)}=0$ yields

$$\sum_{ij}^{(k+1)}=\frac{\sum_{t=1}^{T}\omega_{it}^{(k)}\beta_{ijt}^{(k)}\xi_{ijt}^{(k)}u_{ijt}^{(k)}\left(x_t-\mu_{ij}^{(k+1)}\right)\left(x_t-\mu_{ij}^{(k+1)}\right)^T}{\sum_{t=1}^{T}\omega_{it}^{(k)}\beta_{ijt}^{(k)}\xi_{ijt}^{(k)}} \tag{29}$$

Setting the partial derivative of Q with respect to v_{ij}, we have (see (30))

Setting (30) equals zero, so it follows that v_{ij} need to be computed iteratively as solutions of the equations (see (31))

This equation can be solved by MATLAB function fzero (function, parameters).

The final step is to update the estimation of weight parameters β using the gradient method [26] as follows

$$\beta_{ijt}^{(k+1)}=\beta_{ijt}^{(k)}-\eta\frac{\partial Q(\Psi)}{\partial \beta_{ijt}} \tag{32}$$

where η is the learning rate and its value is sufficiently small. In this paper, we have selected $\eta=10^{-5}$ and (see (33))

$$\frac{\partial Q(\Psi)}{\partial \Sigma_{ij}^{-(k+1)}}=\frac{1}{2}\sum_{t=1}^{T}\omega_{it}^{(k)}\beta_{ijt}^{(k)}\xi_{ijt}^{(k)}\left(\Sigma_{ij}^{(k)}-u_{ijt}^{(k)}\left(x_t-\mu_{ij}^{(k+1)}\right)\left(x_t-\mu_{ij}^{(k+1)}\right)^T\right) \tag{28}$$

$$\begin{aligned}\frac{\partial Q(\Psi)}{\partial v_{ij}^{(k+1)}}=\frac{1}{2}\sum_{t=1}^{T}\omega_{it}^{(k)}\beta_{ijt}^{(k)}\xi_{ijt}^{(k)}\Bigg[&\log\left(\frac{v_{ij}}{2}\right)-\psi\left(\frac{v_{ij}}{2}\right)+1.+\log u_{ijt}^{(k)}+\psi\left(\frac{\beta_{ijt}^{(k)}v_{ij}^{(k)}+\beta_{ijt}^{(k)}p}{2}\right)+\frac{\beta_{ijt}^{(k)}p-1}{v_{ij}^{(k)}}\\&-\psi\left(\frac{\beta_{ijt}^{(k)}v_{ij}^{(k)}+\beta_{ijt}^{(k)}p-p}{2}\right)-\frac{\beta_{ijt}^{(k)}(p-1)}{\beta_{ijt}^{(k)}v_{ij}^{(k)}+\beta_{ijt}^{(k)}p-p}-u_{ijt}^{(k)}+\psi\left(\frac{v_{ij}^{(k)}}{2}\right)-\log\left(\frac{v_{ij}^{(k)}+p}{2}\right)\Bigg]\end{aligned} \tag{30}$$

$$\begin{aligned}&\frac{\sum_{t=1}^{T}\omega_{it}^{(k)}\beta_{ijt}^{(k)}\xi_{ijt}^{(k)}}{\sum_{t=1}^{T}\omega_{it}^{(k)}\beta_{ijt}^{(k)}\xi_{ijt}^{(k)}}\left[\log u_{ijt}^{(k)}-u_{ijt}^{(k)}-\frac{\beta_{ijt}^{(k)}(p-1)}{\beta_{ijt}^{(k)}v_{ij}^{(k)}+\beta_{ijt}^{(k)}p-p}+\frac{\beta_{ijt}^{(k)}p-1}{v_{ij}^{(k)}}\right]\\&+\frac{\sum_{t=1}^{T}\omega_{it}^{(k)}\beta_{ijt}^{(k)}\xi_{ijt}^{(k)}}{\sum_{t=1}^{T}\omega_{it}^{(k)}\beta_{ijt}^{(k)}\xi_{ijt}^{(k)}}\left[\psi\left(\frac{\beta_{ijt}^{(k)}v_{ij}^{(k)}+\beta_{ijt}^{(k)}p}{2}\right)-\psi\left(\frac{\beta_{ijt}^{(k)}v_{ij}^{(k)}+\beta_{ijt}^{(k)}p-p}{2}\right)\right]\\&+\log\left(\frac{v_{ij}}{2}\right)-\psi\left(\frac{v_{ij}}{2}\right)+1+\psi\left(\frac{v_{ij}^{(k)}}{2}\right)-\log\left(\frac{v_{ij}^{(k)}+p}{2}\right)=0\end{aligned} \tag{31}$$

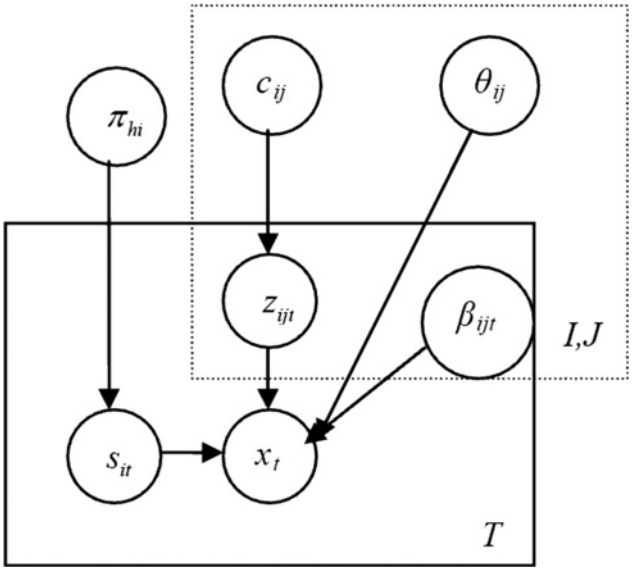

Fig. 2 *Graph of proposed modified SHMM*

Considering the weight parameters β, our model can provide more robust performance, which is shown in the experimental results. The graphical representation of our method is shown in Fig. 2. We conclude our model as follows:

Algorithm: The EM algorithm for the MSHMM is as follows:

1. Initialise the algorithm with the k-means method to obtain initial values and set parameters β_{ijt} using (16), $k:=1$.
2. Use the forward–backward algorithm to calculate the quantities $\omega_{it}^{(k)}$ and $\omega_{hit}^{(k)}$.
3. Compute the $\xi_{ijt}^{(k)}$ and $u_{ijlt}^{(k)}$ according to the E-step, using (20)–(22), respectively.
4. Compute the $\pi_h^{(k+1)}$, $\pi_{hi}^{(k+1)}$, $c_{ij}^{(k+1)}$, $\mu_{ij}^{(k+1)}$, $\Sigma_{ij}^{(k+1)}$ and $v_{ijl}^{(k+1)}$ according to the M-step, using (23)–(31), respectively. Then update $\beta_{ijt}^{(k+1)}$ by using (32).
5. Terminate the iterations if the EM algorithm converges; otherwise, increase the iteration ($k=k+1$) and repeat the steps 2–4.

4 Experimental results

In this section, we experimentally evaluate our method (MSHMM) in a set of synthetic data and real data. For comparison, we also evaluate the fuzzy mixture of the Student's t-distributions model (FSMM) [12], GHMM [1] and the SHMM [6]. The source codes for the FSMM and SHMM algorithms can be downloaded from http://web.mac.com/soteri0s/Sotirios_Chatzis/Software.html. Our experiments have been developed in MATLAB R2009b, and are executed on an Intel Pentium Dual-Core 2.2 GHz CPU, 2 GB RAM.

Table 1 Estimated mean values for three different methods

	Component 1	Component 2	Component 3
FSMM	[0.625–1.694]	[4.783–6.572]	[−2.347–0.667]
SHMM	[0.107–0.184]	[1.206–3.214]	[−3.125–1.395]
MSHMM	[0.064–0.062]	[1.082–2.953]	[−3.034–1.042]

4.1 Synthetic data

The synthetic data set consists of 600 data points from a mixture of three-component bivariate GMMs, where the means of the mixture component densities are

$$\mu_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad \mu_2 = \begin{pmatrix} 1 \\ 3 \end{pmatrix}, \quad \mu_3 = \begin{pmatrix} -3 \\ -1 \end{pmatrix}$$

and their covariances are

$$\sum_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \sum_2 = \begin{pmatrix} 2 & -0.5 \\ -0.5 & 2 \end{pmatrix},$$
$$\sum_3 = \begin{pmatrix} 2 & 0.5 \\ 0.5 & 2 \end{pmatrix}$$

Each component contains 200 data points. The test data is further augmented by 25% outliers, which are drawn from a uniform distribution in the interval [−10, 10]. We use 2×3 (two states and three components) SHMM and MSHMM after running the EM algorithm and gradient method for simulation of the aforementioned synthetic data. For the FSMM method, we adopt $\phi = 1.1$ to obtain the best results, as shown in [12]. All methods are initialised by the same k-means algorithm. The evaluated mean values for three different methods are shown in Table 1, which shows that MSHMM outperforms the FSMM and SHMM methods. The mean values estimated by FSMM are far from correct; however, the MSHMM yields a nearly perfect result, with estimated mean values almost identical to the actual mean values of the modelled data distributions. Finally, in Fig. 3, we show the convergence rates of the MSMM. From Fig. 3, we observe that our method quickly converges with comparable convergence rates.

4.2 UCI data

For experiments with real data, we used four data sets: Wine, Heart, WPBC and WDBC, available from the well-known UCI machine learning repository [27], with varying numbers of samples, dimensionality and components. The Wine data set is used to recognise different wine types according to 13 attributes of chemical analysis. It consists of 178 samples and three components. The Heart data set contains 270 samples and 13 attributes extracted from a larger set of 75 features. The purpose of this data set is to diagnose if there is a presence or absence of heart disease. The WDBC data set is used to diagnose if a breast cancer

$$\frac{\partial Q(\Psi)}{\partial \beta_{ijt}} = \omega_{it}^{(k+1)} \xi_{ijt}^{(k+1)} \left[-\frac{1}{2}\log\left|\Sigma_{ij}^{(k+1)}\right| - \frac{1}{2} u_{ijt}^{(k)} \left(x_t - \mu_{ij}^{(k+1)}\right)^{\mathrm{T}} \Sigma_{ij}^{-(k+1)} \left(x_t - \mu_{ij}^{(k+1)}\right) - \log\Gamma\left(\frac{v_{ij}^{(k+1)}}{2}\right) \right.$$
$$\left. + \frac{v_{ij}^{(k+1)}}{2} \left(\log\left(\frac{v_{ij}^{(k+1)}}{2}\right) + \log u_{ijt}^{(k)} - \log\left(\frac{v_{ij}^{(k+1)} + p}{2}\right) + \psi\left(\frac{v_{ij}^{(k+1)} + p}{2}\right) - u_{ijt}^{(k)} \right) + \frac{\partial \gamma_{ijt}^{(k)}}{\gamma_{ijt}^{(k)} \partial \beta_{ijt}} \right] \tag{33}$$

doi: 10.1049/iet-spr.2012.0315

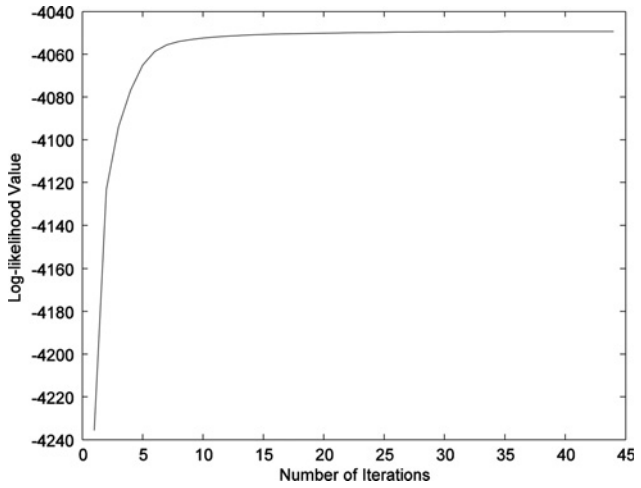

Fig. 3 *Convergence rates of the MSHMM*

is benign or malignant based on 30 attributes and 569 samples. The WPBC data set contains 198 samples and 32 attributes, which are computed from a digitised image of a fine needle aspirate (FNA) of a breast mass. The purpose of this data set is to diagnose if a breast cancer is recurring or non-recurring. Table 2 shows the summary of these four data sets.

The next step is to determine the optimal number of components for our method. The traditional automatic determination methodologies for J are the cross-validation technique [11] and the minimum message length method [28]. These two methods can also be applied in this case. However, for simple implementation, we set the component numbers J as equal to the true class labels for the initialisation of all three methods.

Each data set is first randomly divided into two parts: one for training and another for testing. We then evaluate the results by interpreting the components as clusters and compare them with the ground truth labels. Each data point in the test set is assigned to the component that most likely generated it, and the pattern is classified to the class represented by the component. We can then compute the error rates on the test data. The error rate is defined as

$$\mathrm{ER} = \left(1 - \frac{\bar{T}}{T}\right) \times 100\% \tag{34}$$

where T and $\bar{T}$ are the number of samples of true class labels and estimated components, respectively. Here, the estimated component label is simply selected as the component to which the majority of samples in the component belong. A comparison of the component labels of all samples with the true class labels yields the error rates that are implemented by the CFMATRIX function in MATLAB. The lower error rate gives better classification results. Table 3 shows the mean of the error rate over 10 runs of different methods. It is clear that MSHMM outperforms SHMM and FSMM with lower error rates.

Table 3 Error rates for different methods (%)

	FSMM	SHMM	MSHMM
Wine	28.09	24.72	11.80
Heart	40.74	40.00	37.41
WPBC	36.36	38.89	30.81
WDBC	14.59	13.71	11.77

4.3 Brain fMRI

In this subsection, we evaluate the performance of the GHMM, SHMM and MSHMM for fMRI images [29–31], as described in [6]. The considered data set is publicly available in [32]. The experiment consists of a set of trials, and the data are partitioned into trials. For some of these intervals, the subject simply rested, or gazed at a fixation point on the screen. For other trials, the subject was shown as a picture and a sentence, and was 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 in half of the trials, and the sentence presented in the other half. For each subject, 40 such trials were available.

The images were collected every 500 ms. Only a fraction of the brain of each subject was imaged. The data are marked up with 25–30 anatomically defined regions (called 'Regions of Interest', or ROIs). The timing within each such trial is as follows: (i) The first stimulus (sentence or picture) was presented at the beginning of the trail (image = 1). (ii) Four seconds later (image = 9), the stimulus was removed, and replaced by a blank screen. (iii) Four seconds later (image = 17), the second stimulus was presented. This remained on the screen for 4 s, or until the subject pressed the mouse button, whichever came first. (iv) A rest period of 15 s (30 images) was added after the second stimulus was removed from the screen. Thus, each trial lasted a total of approximately 27 s (approximately 54 images). The goal was 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 instant or interval. Some examples of fMRI image and its interest regions are shown in Figs. 4 and 5.

In this experiment, we trained two 4×3 (four states and three components) chains (one chain for the picture stimulus sequences and the other for the sentence stimulus sequences) for GHMM, SHMM and MSHMM. We selected only the data corresponding to the following ROIs ('CALC', 'LIPL', 'LT', 'LTRIA', 'LOPER', 'LIPS', 'LDLPFC') to train our classifiers, as described by the creators of this data set. In Table 4, we compare the misclassified number for GHMM, SHMM and MSHMM. It is clearly shown that MSHMM outperforms GHMM and

Table 2 Real UCI dataset descriptions

Name	Description	Number of samples, T	Dimensionality of the samples, p	Number of components, J
Wine	wine recognition	178	13	3
Heart	heart disease of statlog	270	13	2
WPBC	Wisconsin prognostic breast cancer	198	32	2
WDBC	Wisconsin diagnostic breast cancer	569	30	2

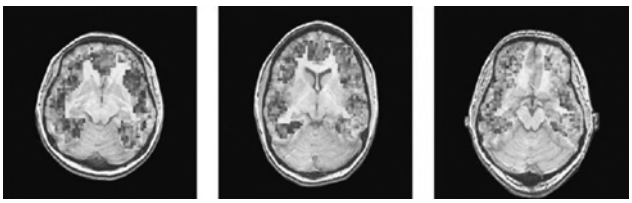

Fig. 4 *Some example images of the fMRI scans*

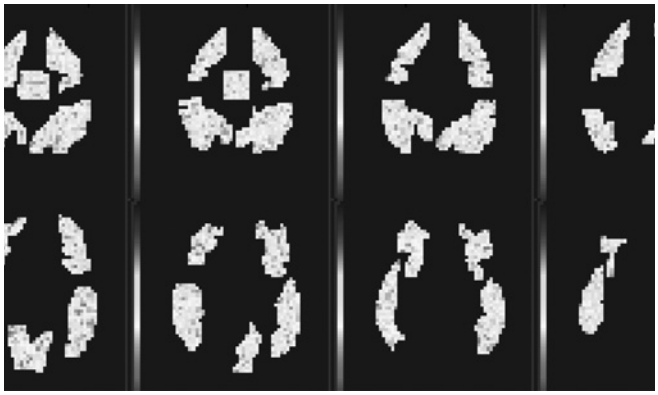

Fig. 5 *Some examples of interest regions*

Table 4 Misclassified number for different ROIs

ROI	GHMM	SHMM	MSHMM
CALC	5	4	1
LIPL	18	16	14
LT	14	13	9
LTRIA	19	16	13
LOPER	12	12	10
LIPS	25	21	17
LDLPFC	16	17	15
computation time, s	5.22	13.16	16.84

SHMM in most ROIs and obtains the same results for 'LTRIA' case.

4.4 Phonetic recognition in TIMIT speech corpus

In the last experiment, we use a subset of the TIMIT speech corpus [33–35] for sequential data segmentation application. 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, the acoustic feature vectors are labelled by 48 phonetic classes, each represented by one HMM 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 [9]. Then, a relevant error rate can be obtained by aligning the Viterbi and ground truth transcriptions using dynamic programming [34] and summing the substitution, deletion and insertion error rates from the alignment process. The obtained results can be found in Table 5. We notice that the MSHMM performs better than the GHMM and SHMM for all different components J.

We also evaluate the computation time for all the methods in the previous experiments. The computation time t (seconds) of the different methods is presented in Tables 4 and 5. It is noted that the computation of GHMM is much faster than that of the other methods. Compared to other methods, our model is the slowest but achieves the best classification results.

5 Conclusions

In this paper, we propose a simple and effective model for clustering. We introduce the robust modified Student's t-hidden Markov model, which exploits the merits of the SMM to offer high robustness against outliers, with consideration of the weight parameters β, while incorporating the Markov states, latent components and observations. The MSHMM can be viewed as an extension of the conventional SHMM; thus, the latter can be considered as a special case of the former. Furthermore, parameter learning is similar to the conventional SHMM; thus, it can be easily implemented by the EM algorithm. However, the EM algorithm does not fit for weight parameters β. Here, the weight parameters are estimated simply and effectively by the gradient method. Our future work will focus on applying the modified SMM to other applications, such as image segmentation.

6 Acknowledgments

This work was supported in part by the Canada Chair Research Programme and the Natural Sciences and Engineering Research Council of Canada. This work was supported in part by the National Natural Science Foundation of China under grant nos. 61105007 and 61103141. This work was supported in part by PAPD (A Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions).

7 References

1 Rabiner, L.R.: 'A tutorial on hidden Markov models and selected applications in speech recognition', *Proc. IEEE*, 1989, **77**, (2), pp. 257–286
2 Arslan, L., Hansen, J.: 'Selective training for hidden Markov models with applications to speech classification', *IEEE Trans. Speech Audio Process.*, 1999, **7**, (1), pp. 46–54
3 Fjørtoft, R., Delignon, Y., Pieczynski, W., Sigelle, M., Tupin, F.: 'Unsupervised classification of radar images using hidden Markov

Table 5 Sequence segmentation: phone error rates for various number components J

	$J=1$, %	$J=2$, %	$J=4$, %	$J=8$, %	Computation time, s
GHMM	45.1	43.5	42.7	41.3	108.1
SHMM	44.6	41.7	38.3	36.1	185.1
MSHMM	43.4	39.9	36.6	36.1	324.8

doi: 10.1049/iet-spr.2012.0315

chains and hidden Markov random fields', *IEEE Trans. Geosci. Remote Sens.*, 2003, **41**, (3), pp. 675–686
4 Ji, S., Krishnapuram, B., Carin, L.: 'Variational Bayes for continuous hidden Markov models and its application to active learning', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2006, **28**, (4), pp. 522–532
5 Pieczynski, W.: 'Pairwise Markov chains', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2003, **25**, (5), pp. 634–639
6 Chatzis, S.P., Kosmopoulos, D.I., Varvarigou, T.A.: 'Robust sequential data modeling using an outlier tolerant hidden Markov model', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2009, **31**, (9), pp. 1657–1669
7 Lee, C.H., Lee, S.Y.: 'Noise-robust speech recognition using top-down selective attention with an HMM classifier', *IEEE Signal Process. Lett.*, 2007, **14**, (7), pp. 489–491
8 Chien, J.T., Liao, C.P.: 'Maximum confidence hidden Markov modeling for face recognition', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2008, **30**, (4), pp. 606–616
9 Sha, F., Saul, L.K.: 'Large margin hidden Markov models for automatic speech recognition', in Schölkopf, B., Platt, J., Hoffman, T. (Eds.): 'Advances in neural information processing systems', MIT Press, 2007, vol. 19, pp. 1249–1256
10 Alon, J., Sclaroff, S., Kollios, G., Pavlovic, V.: 'Discovering clusters in motion time-series data'. IEEE Conf. Computer Vision and Pattern Recognition, 2003
11 McLachlan, G., Peel, D.: 'Finite mixture models' (Wiley, New York, 2000)
12 Chatzis, S.P., Varvarigou, T.A.: 'Robust fuzzy clustering using mixtures of Student's *t*-distributions', *Pattern Recognit. Lett.*, 2008, **29**, (13), pp. 1901–1905
13 Thanh, M.N., Wu, Q.M.J., Ahuja, S.: 'An extension of the standard mixture model for image segmentation', *IEEE Trans. Neural Netw.*, 2010, **21**, (8), pp. 1326–1338
14 Thanh, M.N., Wu, Q.M.J.: 'Gaussian-mixture-model-based spatial neighborhood relationships for pixel labeling problem', *IEEE Trans. Syst. Man Cybern. t B*, 2011, **PP**, (99), pp. 1–10
15 Forbes, F., Peyrard, N.: 'Hidden Markov random field model selection criteria based on mean field-like approximations', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2003, **25**, (9), pp. 1089–1101
16 Celeux, G., Forbes, F., Peyrard, N.: 'EM procedures using mean field-like approximations for Markov model-based image segmentation', *Pattern Recognit.*, 2003, **36**, (1), pp. 131–144
17 Peel, D., McLachlan, G.: 'Robust mixture modeling using the *t* distribution', *Stat. Comput.*, 2000, **10**, (4), pp. 335–344
18 Chatzis, S.P., Kosmopoulos, D.I., Varvarigou, T.A.: 'Signal modeling and classification using a robust latent space model based on *t* distributions', *IEEE Trans. Signal Process.*, 2008, **56**, (3), pp. 949–963
19 Chatzis, S.P., Varvarigou, T.A.: 'Factor analysis latent subspace modeling and robust fuzzy clustering using *t* distributions', *IEEE Trans. Fuzzy Syst.*, 2009, **17**, (3), pp. 505–517
20 Sfikas, G., Nikou, C., Galatsanos, N.: 'Edge preserving spatially varying mixtures for image segmentation', *IEEE Comput. Vis. Pattern Recognit.*, 2008, pp. 1–7. DOI: 10.1109/CVPR.2008.4587416
21 Thanh, M.N., Wu, Q.M.J.: 'Robust Student's *t*-mixture model with spatial constraints and its application in medical image segmentation', *IEEE Trans. Med. Imaging*, 2011, **PP**, (99), pp. 1–14
22 Bishop, C.M.: 'Pattern recognition and machine learning' (Springer, 2006)
23 Ichihashi, H., Honda, K., Tani, N.: 'Gaussian mixture pdf approximation and fuzzy c-means clustering with entropy regularization'. Proc. Fourth Asian Fuzzy Syst. Symp., 2000, pp. 217–221
24 Chen, Z.S., Yang, Y.M.: 'Fault diagnostics of helicopter gearboxes based on multi-sensor mixture hidden Markov models', *ASME J. Vibr. Acoust.*, 2012, **134**, (3), pp. 031010
25 Juang, B.H., Katagiri, S.: 'Discriminative learning for minimum error classification', *IEEE Trans. Signal Process.*, 1992, **40**, pp. 3043–3054
26 Bishop, C.M.: 'Neural networks for pattern recognition' (Oxford University Press, Oxford, UK, 1995)
27 Asuncion, A., Newman, D.: UCI machine learning repository, http://www.ics.uci.edu/~mlearn/MLRepository.html, 2007
28 Figueiredo, M.A.T., Jain, A.K.: 'Unsupervised learning of finite mixture models', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2002, **24**, (3), pp. 381–396
29 Friston, K.J., Holmes, A.P., Worsley, K.J., Poline, J.-B., Firth, C.D., Frackowiak, R.S.J.: 'Statistical parametric maps in functional imaging: a general linear approach', *Hum. Brain Mapp.*, 1995, **2**, pp. 189–210
30 Li, X., Coyle, D., Maguire, L.P., McGinnity, T.M., Watson, D., Benali, H.: 'A least angle regression method for fMRI activation detection in phase-encoded experimental designs', *NeuroImage*, 2010, **52**, (4), pp. 1390–1400
31 Mitchell, T.M., Hutchinson, R., Niculescu, R.S., *et al.*: 'Learning to decode cognitive states from brain images', *Mach. Learn.*, 2004, **57**, (1–2), pp. 145–175
32 http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-81/www/
33 Lamel, L.F., Kassel, R.H., Seneff, S.: 'Speech database development: design and analysis of the acoustic-phonetic corpus'. Proc. DARPA Speech Recognition Workshop, 1986, pp. 100–109
34 Lee, K.F., Hon, H.W.: 'Speaker-independent phone recognition using hidden Markov models', *IEEE Trans. Acoust. Speech Signal Process.*, 1988, **37**, (11), pp. 1641–1648
35 Fei, S., Lawrence, K.S.: 'Large margin Gaussian mixture modeling for phonetic classification and recognition'. Proc. ICASSP, Tolouse, France, 2006, pp. 265–268