PAPER

A Bayesian Approach to Image Recognition Based on Separable Lattice Hidden Markov Models

Kei SAWADA[†a)], ***Student Member***, **Akira TAMAMORI**[†b)], ***Member***, **Kei HASHIMOTO**[†c)], ***Nonmember***, **Yoshihiko NANKAKU**[†d)], ***and*** **Keiichi TOKUDA**[†e)], ***Members***

SUMMARY This paper proposes a Bayesian approach to image recognition based on separable lattice hidden Markov models (SL-HMMs). The geometric variations of the object to be recognized, e.g., size, location, and rotation, are an essential problem in image recognition. SL-HMMs, which have been proposed to reduce the effect of geometric variations, can perform elastic matching both horizontally and vertically. This makes it possible to model not only invariances to the size and location of the object but also nonlinear warping in both dimensions. The maximum likelihood (ML) method has been used in training SL-HMMs. However, in some image recognition tasks, it is difficult to acquire sufficient training data, and the ML method suffers from the over-fitting problem when there is insufficient training data. This study aims to accurately estimate SL-HMMs using the maximum a posteriori (MAP) and variational Bayesian (VB) methods. The MAP and VB methods can utilize prior distributions representing useful prior information, and the VB method is expected to obtain high generalization ability by marginalization of model parameters. Furthermore, to overcome the local maximum problem in the MAP and VB methods, the deterministic annealing expectation maximization algorithm is applied for training SL-HMMs. Face recognition experiments performed on the XM2VTS database indicated that the proposed method offers significantly improved image recognition performance. Additionally, comparative experiment results showed that the proposed method was more robust to geometric variations than convolutional neural networks.

key words: *image recognition, hidden Markov models, separable lattice hidden Markov models, Bayesian approach, deterministic annealing*

1. Introduction

Image recognition is a technique for identifying objects in an image. Typical applications include biometrics authentication, e.g., fingerprint and face, optical character recognition (OCR), and video recognition. Statistical approaches have been successfully applied in the field of image recognition. In particular, principal component analysis (PCA) based approaches, such as the eigenface method [1] and the subspace method [2], attain good recognition performance. For conventional statistical approaches, however, it is normal to apply a pre-processing method for normalizing image variations, e.g., geometric variations such as size, location, and rotation. This is because many classifiers cannot deal with such image variations. The accuracy of these normalization processes affects recognition performance. Task-dependent normalization techniques have thus been developed for each image recognition task. However, the final objective of image recognition is not to accurately normalize image variations for human perception but to achieve high recognition performance. It is therefore a good idea to integrate the normalization processes into classifiers and optimize them on the basis of a consistent criterion.

Geometric variations of an object to be recognized are an essential problem in image recognition. Therefore, much research work has been conducted on this problem. Scale-invariant feature transform (SIFT) [3] and histograms of oriented gradients (HOG) [4] have been proposed to detect and describe local features that are invariant to local geometric variation. Unfortunately, these methods cannot grasp global information. In recent years, convolutional neural network (CNN) based techniques have achieved significant improvements [5], [6]. In addition to the structure of the standard feed-forward neural networks as classifiers, CNNs have geometric invariants based on multiple convolutional and pooling layers. However, since pooling is independently performed in each local window, it is difficult to represent global geometric transforms over an entire image. Another way to address geometric variations is using hidden Markov models (HMMs) [7], [8]. Geometric matching between input images and model parameters is represented by discrete hidden variables and the normalization process is included in the calculation of probabilities. However, the extension of HMMs to two dimensions for two-dimensional data generally leads to an exponential increase in the amount of computation needed for training. To overcome this problem, several low computational complexity models have been proposed [9]–[15]. Among them, separable lattice HMMs (SL-HMMs) have been proposed to reduce computational complexity while retaining outstanding properties that model two-dimensional data [15]. SL-HMMs can perform an elastic matching in both the horizontal and vertical directions, which makes it possible to model not only invariances to the size and location of an object but also nonlinear warping in both dimensions. One of advantages of SL-HMMs against CNNs is explicit modeling of generative process which can represent geometric variations over an entire image. Furthermore, some extensions to structures representing typical geometric variations which are seen in many

Manuscript received March 17, 2016.
Manuscript revised August 20, 2016.
Manuscript publicized September 5, 2016.

[†]The authors are with the Department of Scientific and Engineering Simulation, Nagoya Institute of Technology, Nagoya-shi, 466–8555 Japan.
a) E-mail: swdkei@sp.nitech.ac.jp
b) E-mail: mataki@sp.nitech.ac.jp
c) E-mail: bonanza@sp.nitech.ac.jp
d) E-mail: nankaku@nitech.ac.jp
e) E-mail: tokuda@nitech.ac.jp

DOI: 10.1587/transinf.2016EDP7112

image recognition tasks have already been proposed, e.g., a structure for rotational variations [16], a structure with multiple horizontal and vertical Markov chains [17], explicit state duration modeling [18], trajectory modeling [19], and integration SL-HMMs and factor analyzers [20]. By selecting an appropriate model structures reflecting data generation process for a target task, human knowledge can effectively be utilized as prior information and this makes it possible to construct classifiers with a small amount of training data. It is also an interesting property of SL-HMMs that various size images can directly be used as inputs without size normalization.

In some image recognition tasks, only a small amount of training data is available and so efforts to achieve high generalization ability are required. The maximum likelihood (ML) criterion has typically been used in image recognition using SL-HMMs. However, although SL-HMMs can be trained from a relatively small amount of training data, since the ML criterion produces a point estimate of model parameters, the estimation accuracy may be degraded due to the over-fitting problem when there is insufficient training data. In this study, we focus on estimating SL-HMMs with high generalization ability by using the Bayesian criterion. The Bayesian criterion assumes that model parameters are random variables, and high generalization ability can be obtained by marginalizing all model parameters in estimating predictive distributions. Moreover, the Bayesian criterion can utilize prior distributions representing useful prior information on model parameters. Therefore, the Bayesian criterion has higher generalization ability than the ML criterion when there is insufficient training data. However, the Bayesian criterion requires complicated integral and expectation computations to obtain posterior distributions when models have hidden variables. To overcome this problem, the maximum a posteriori (MAP) method [21] and the variational Bayesian (VB) method [22] have been proposed as approximation methods. We applied the MAP and VB methods to image recognition based on SL-HMMs, and obtained significantly better performance than the ML method [23]. The additional contributions of this paper are 1) further evaluation of SL-HMMs based on the MAP and VB methods, 2) improvement of the training algorithm by applying the deterministic annealing expectation maximization (DAEM) algorithm [24], [25], and 3) comparison with CNN-based approaches in image recognition experiments. The DAEM algorithm can alleviate the local maximum problem dependent on the initial parameter. We show that the MAP and VB methods applying the DAEM algorithm can improve the performance in image recognition experiments. Additionally, comparative experiments results show that the proposed method is more robust to geometric variations than CNNs.

The rest of the paper is organized as follows. Section 2 briefly explains the structure of SL-HMMs, and Sect. 3 describes training criteria in the Bayesian statistics. A training algorithm for SL-HMMs using the ML (conventional) method is described in Sect. 4. In Sect. 5, we derive Bayesian (proposed) approach for SL-HMMs. Section 6 presents face recognition experiments we did on the XM2VTS database [26] and we conclude the paper with a summary of key points in Sect. 7.

2. Separable Lattice Hidden Markov Models

In the case that observations are two-dimensional data, e.g., pixel values of an image, observations are assumed to be given on a two-dimensional lattice as:

$$\boldsymbol{o} = \{\boldsymbol{o}_{\boldsymbol{t}} \,|\, \boldsymbol{t} = (t^{(1)}, t^{(2)}) \in \boldsymbol{T}\}, \tag{1}$$

where $\boldsymbol{T} = \{(1,1),(1,2),\ldots,(1,T^{(2)}),(2,1),\ldots,(t^{(1)},t^{(2)}),\ldots,(T^{(1)},T^{(2)})\}$ denotes the two-dimensional image lattice, $\boldsymbol{t}$ denotes the two-dimensional coordinate lattice, $t^{(m)}$ is the coordinate of the m-th dimension, $T^{(m)}$ is the maximum coordinate of the m-th dimension, and $m \in \{1,2\}$ denotes dimension index. In two-dimensional HMMs, observation $\boldsymbol{o}_{\boldsymbol{t}}$ is emitted from a state indicated by hidden variable $\boldsymbol{z}_{\boldsymbol{t}}$. The hidden variables $\boldsymbol{z}_{\boldsymbol{t}} \in \boldsymbol{K}$ can take one of $K^{(1)}K^{(2)}$ states, which are assumed to be arranged on a two-dimensional state lattice $\boldsymbol{K} = \{(1,1),(1,2),\ldots,(1,K^{(2)}),(2,1),\ldots,(K^{(1)},K^{(2)})\}$, where $K^{(m)}$ is the maximum state of the m-th dimension. In other words, a set of hidden variables represents a segmentation of observations into the $K^{(1)}K^{(2)}$ states and each state corresponds to a segmented region in which the observation vectors are assumed to be generated from the same distribution. The number of possible state sequences in two-dimensional HMMs is $(K^{(1)}K^{(2)})^{T^{(1)}T^{(2)}}$. Therefore, standard two-dimensional HMMs demand high computational costs.

Separable lattice hidden Markov models (SL-HMMs) have been proposed to reduce computational complexity [15]. In SL-HMMs, to reduce the number of possible state sequences, hidden variables are constrained to be composed of two Markov chains as follows:

$$\boldsymbol{z} = \{\boldsymbol{z}^{(1)}, \boldsymbol{z}^{(2)}\}, \tag{2}$$

$$\boldsymbol{z}^{(m)} = \{z^{(m)}_{t^{(m)}} \,|\, 1 \le t^{(m)} \le T^{(m)}\}, \tag{3}$$

where $\boldsymbol{z}^{(m)}$ is the Markov chain along with the m-th coordinate, and $z^{(m)}_{t^{(m)}} \in \{1,\ldots,K^{(m)}\}$. The composite structure of hidden variables in SL-HMMs is defined as the product of hidden state sequences as:

$$\boldsymbol{z}_{\boldsymbol{t}} = (z^{(1)}_{t^{(1)}}, z^{(2)}_{t^{(2)}}). \tag{4}$$

This means that hidden state sequences are independent of each dimension and the segmented regions of observations are constrained to rectangles. That is, it allows an observation lattice to be elastic both horizontally and vertically. Using this structure, the number of possible state sequences can be reduced from $(K^{(1)}K^{(2)})^{T^{(1)}T^{(2)}}$ to $(K^{(1)})^{T^{(1)}}(K^{(2)})^{T^{(2)}}$.

Figures 1 and 2 respectively show the graphical model representation and the model structure of SL-HMMs in face image modeling. The joint likelihood of observations $\boldsymbol{o}$ and hidden variables $\boldsymbol{z}$ can be written as:

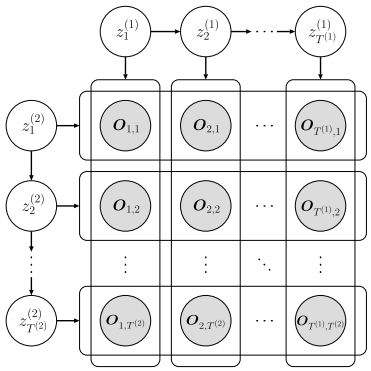


Fig. 1 Graphical model representation of SL-HMMs. The rounded boxes represent a group of variables, and the arrow to each box represents the dependency in regard to all variables in the box instead of drawing arrows to all the variables.

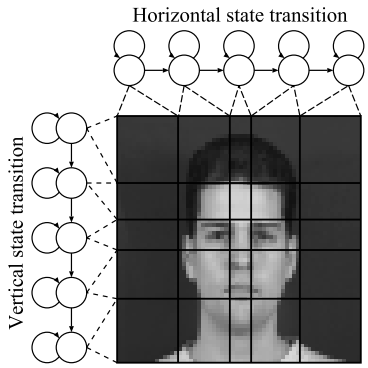


Fig. 2 Model structure of SL-HMMs in face image modeling.

$$P(\boldsymbol{o}, z \,|\, \boldsymbol{\Lambda}) = \prod_{m=1}^{2} \Big[P(\boldsymbol{z}^{(m)} \,|\, \boldsymbol{\Lambda}) \Big] P(\boldsymbol{o} \,|\, z, \boldsymbol{\Lambda})$$

$$= \prod_{m=1}^{2} \left[P(z_1^{(m)} \,|\, \boldsymbol{\Lambda}) \prod_{t^{(m)}=2}^{T^{(m)}} P(z_{t^{(m)}}^{(m)} \,|\, z_{t^{(m)}-1}^{(m)}, \boldsymbol{\Lambda}) \right] \prod_{\boldsymbol{t}} P(\boldsymbol{o}_{\boldsymbol{t}} \,|\, \boldsymbol{z}_{\boldsymbol{t}}, \boldsymbol{\Lambda}), \quad (5)$$

where $\boldsymbol{\Lambda}$ is a set of model parameters. The model parameters of SL-HMMs are summarized as follows:

$$\boldsymbol{\Lambda} = \{\boldsymbol{\pi}^{(1)}, \boldsymbol{\pi}^{(2)}, \boldsymbol{a}^{(1)}, \boldsymbol{a}^{(2)}, \boldsymbol{b}\}. \quad (6)$$

1) $\boldsymbol{\pi}^{(m)} = \{\pi_i^{(m)} \,|\, 1 \le i \le K^{(m)}\}$: an initial state probability distribution. The probability of state i at $t^{(m)} = 1$ is represented by $\pi_i^{(m)} = P(z_1^{(m)} = i \,|\, \boldsymbol{\Lambda})$.
2) $\boldsymbol{a}^{(m)} = \{a_{ij}^{(m)} \,|\, 1 \le i, j \le K^{(m)}\}$: a state transition probability matrix. The probability of moving from state i to state j is represented by $a_{ij}^{(m)} = P(z_{t^{(m)}}^{(m)} = j \,|\, z_{t^{(m)}-1}^{(m)} = i, \boldsymbol{\Lambda})$.
3) $\boldsymbol{b} = \{\boldsymbol{b}_{\boldsymbol{k}}(\boldsymbol{o}_{\boldsymbol{t}}) \,|\, \boldsymbol{k} \in \boldsymbol{K}\}$: an output probability distribution. The probability of an observation $\boldsymbol{o}_{\boldsymbol{t}}$ being generated from a state $\boldsymbol{k}$ is represented by $\boldsymbol{b}_{\boldsymbol{k}}(\boldsymbol{o}_{\boldsymbol{t}}) = P(\boldsymbol{o}_{\boldsymbol{t}} \,|\, \boldsymbol{z}_{\boldsymbol{t}} = \boldsymbol{k}, \boldsymbol{\Lambda})$, where $\boldsymbol{k}$ denotes the two-dimensional state index in the two-dimensional state lattice $\boldsymbol{K}$. In this study, the output probability distribution is assumed to be a single Gaussian distribution $P(\boldsymbol{o}_{\boldsymbol{t}} \,|\, \boldsymbol{z}_{\boldsymbol{t}} = \boldsymbol{k}, \boldsymbol{\Lambda}) = \mathcal{N}(\boldsymbol{o}_{\boldsymbol{t}} \,|\, \boldsymbol{\mu}_{\boldsymbol{k}}, \boldsymbol{\Sigma}_{\boldsymbol{k}})$, where $\boldsymbol{\mu}_{\boldsymbol{k}}$ and $\boldsymbol{\Sigma}_{\boldsymbol{k}}$ respectively denote the mean vector and the diagonal covariance matrix in the state $\boldsymbol{k}$.

In the application of image modeling, SL-HMMs can perform an elastic matching in both the horizontal and vertical directions by assuming the transition probabilities with left-to-right and top-to-bottom topologies, which makes it possible to model not only invariances to the size and location of an object but also nonlinear warping in both dimensions.

3. Training Criterion

In Bayesian statistics, it is important to estimate high generalization ability from training data. This section explains the estimation criteria of Bayesian statistics.

3.1 Maximum Likelihood Criterion

The maximum likelihood (ML) criterion has typically been used to train statistical models. The optimal model parameters are estimated in the ML criterion by maximizing the likelihood of training data $P(\boldsymbol{o} \,|\, \boldsymbol{\Lambda})$ as:

$$\boldsymbol{\Lambda}^{(\mathrm{ML})} = \arg\max_{\boldsymbol{\Lambda}} P(\boldsymbol{o} \,|\, \boldsymbol{\Lambda}). \quad (7)$$

The predictive distribution of testing data $\boldsymbol{x}$ in the testing stage is given by $P(\boldsymbol{x} \,|\, \boldsymbol{\Lambda}^{(\mathrm{ML})})$. However, since the ML criterion produces a point estimate of model parameters, the estimation accuracy may be decreased due to the over-fitting problem when there is insufficient training data.

3.2 Maximum a Posteriori Criterion

The optimal model parameters in the maximum a posteriori (MAP) criterion are estimated by maximizing the posterior probability for given training data as:

$$\boldsymbol{\Lambda}^{(\mathrm{MAP})} = \arg\max_{\boldsymbol{\Lambda}} P(\boldsymbol{o} \,|\, \boldsymbol{\Lambda}) P(\boldsymbol{\Lambda}), \quad (8)$$

where $P(\boldsymbol{\Lambda})$ is a prior distribution for model parameters $\boldsymbol{\Lambda}$. The MAP criterion can utilize prior distribution $P(\boldsymbol{\Lambda})$, and can be seen as an extension of the ML criterion. Testing in the MAP criterion is conducted using predictive distribution $P(\boldsymbol{x} \,|\, \boldsymbol{\Lambda}^{(\mathrm{MAP})})$. However, it still suffers from the over-fitting problem because of point estimates, when there is insufficient training data.

3.3 Bayesian Criterion

The predictive distribution of the Bayesian criterion is given by:

$$P(\boldsymbol{x} \,|\, \boldsymbol{o}) = \int P(\boldsymbol{x} \,|\, \boldsymbol{\Lambda}) P(\boldsymbol{\Lambda} \,|\, \boldsymbol{o}) \mathrm{d}\boldsymbol{\Lambda}. \quad (9)$$

Posterior distribution $P(\boldsymbol{\Lambda} \,|\, \boldsymbol{o})$ for a set of model parameters $\boldsymbol{\Lambda}$ can be written with the Bayes theorem:

$$P(\boldsymbol{\Lambda} \,|\, \boldsymbol{o}) = \frac{P(\boldsymbol{o} \,|\, \boldsymbol{\Lambda}) P(\boldsymbol{\Lambda})}{P(\boldsymbol{o})}, \quad (10)$$

Table 1 Training criteria. The c denotes an object class index.

Criterion	Training	Testing
ML criterion	$\boldsymbol{\Lambda}_c^{(\mathrm{ML})} = \arg\max_{\boldsymbol{\Lambda}} P(\boldsymbol{o}_c \mid \boldsymbol{\Lambda})$	$c^{(\mathrm{ML})} = \arg\max_c P(\boldsymbol{x} \mid \boldsymbol{\Lambda}_c^{(\mathrm{ML})})$
MAP criterion	$\boldsymbol{\Lambda}_c^{(\mathrm{MAP})} = \arg\max_{\boldsymbol{\Lambda}} P(\boldsymbol{o}_c \mid \boldsymbol{\Lambda}) P(\boldsymbol{\Lambda})$	$c^{(\mathrm{MAP})} = \arg\max_c P(\boldsymbol{x} \mid \boldsymbol{\Lambda}_c^{(\mathrm{MAP})})$
Bayesian criterion	$P(\boldsymbol{\Lambda} \mid \boldsymbol{o}_c) = \dfrac{P(\boldsymbol{o}_c \mid \boldsymbol{\Lambda}) P(\boldsymbol{\Lambda})}{P(\boldsymbol{o}_c)}$	$c^{(\mathrm{Bayes})} = \arg\max_c \int P(\boldsymbol{x} \mid \boldsymbol{\Lambda}) P(\boldsymbol{\Lambda} \mid \boldsymbol{o}_c) \mathrm{d}\boldsymbol{\Lambda}$

where $P(\boldsymbol{o})$ is evidence. The model parameters are integrated out in Eq. (9) so that the effect of over-fitting is mitigated. That is, the Bayesian criterion has higher generalization ability than the ML and MAP criteria when there is insufficient training data. However, the Bayesian criterion requires complicated integral and expectation computations to obtain posterior distributions when models have hidden variables. A Markov chain Monte Carlo (MCMC) [27] and variational Bayesian (VB) [22] methods have been proposed as approaches to approximation to overcome this problem. The training criteria are summarized in Table 1.

4. SL-HMMs Using Maximum Likelihood Method

4.1 Expectation Maximization Algorithm

Since SL-HMMs have hidden variables z, it is difficult to obtain an analytic solution to Eq. (7). The parameters of SL-HMMs can be estimated via the expectation maximization (EM) algorithm [28], which is an iterative procedure. This procedure maximizes the expectation of the complete-data log-likelihood so-called Q-function:

$$Q(\boldsymbol{\Lambda}, \boldsymbol{\Lambda}^{(\mathrm{old})}) = \sum_z P(z \mid \boldsymbol{o}, \boldsymbol{\Lambda}^{(\mathrm{old})}) \ln P(\boldsymbol{o}, z \mid \boldsymbol{\Lambda}), \tag{11}$$

where $\boldsymbol{\Lambda}^{(\mathrm{old})}$ denotes the current parameters. The likelihood of the training data is guaranteed to increase by increasing the value of the Q-function. The EM algorithm starts with some initial model parameters $\boldsymbol{\Lambda}^{(\mathrm{old})}$ and iterates between the following two steps.

$$\begin{aligned}&\text{(E-step): compute } Q(\boldsymbol{\Lambda}, \boldsymbol{\Lambda}^{(\mathrm{old})})\\ &\text{(M-step): } \boldsymbol{\Lambda}^{(\mathrm{new})} = \arg\max_{\boldsymbol{\Lambda}} Q(\boldsymbol{\Lambda}, \boldsymbol{\Lambda}^{(\mathrm{old})})\end{aligned}$$

The E-step computes the posterior probabilities of the hidden variables $P(z \mid \boldsymbol{o}, \boldsymbol{\Lambda}^{(\mathrm{old})})$ while keeping model parameters $\boldsymbol{\Lambda}^{(\mathrm{old})}$ fixed to current values. Then, the Q-function is computed by using $P(z \mid \boldsymbol{o}, \boldsymbol{\Lambda}^{(\mathrm{old})})$. The M-step estimates the re-estimation parameters $\boldsymbol{\Lambda}^{(\mathrm{new})}$ by maximizing the Q-function. These steps are iterated until convergence of the log-likelihood by replacing $\boldsymbol{\Lambda}^{(\mathrm{old})} \leftarrow \boldsymbol{\Lambda}^{(\mathrm{new})}$. By maximizing the Q-function with respect to model parameter $\boldsymbol{\Lambda}$, the re-estimation parameters $\boldsymbol{\Lambda}^{(\mathrm{new})}$ in the M-step can be easily derived. By contrast, the calculation of the posterior probabilities $P(z \mid \boldsymbol{o}, \boldsymbol{\Lambda}^{(\mathrm{old})})$ in the E-step is computationally intractable due to the combination of hidden variables.

4.2 Variational Method

Variational methods have been used to approximate the ML method in probabilistic graphical models with hidden variables [29]. An approximate posterior distribution is estimated by maximizing the lower bound of the log-marginal likelihood instead of the true log-likelihood. The lower bound of the log-marginal likelihood $\mathcal{F}^{(\mathrm{ML})}$ is defined by using Jensen's inequality as:

$$\begin{aligned}\ln P(\boldsymbol{o} \mid \boldsymbol{\Lambda}) &= \ln \sum_z Q(z) \frac{P(\boldsymbol{o}, z \mid \boldsymbol{\Lambda})}{Q(z)}\\ &\geq \sum_z Q(z) \ln \frac{P(z \mid \boldsymbol{\Lambda}) P(\boldsymbol{o} \mid z, \boldsymbol{\Lambda})}{Q(z)}\\ &\triangleq \mathcal{F}^{(\mathrm{ML})}(Q, \boldsymbol{\Lambda}),\end{aligned} \tag{12}$$

where $Q(z)$ is an arbitrary distribution. The difference between the true log-likelihood $\ln P(\boldsymbol{o} \mid \boldsymbol{\Lambda})$ and the lower bound $\mathcal{F}^{(\mathrm{ML})}$ is given by the Kullback-Leibler (KL) divergence between the arbitrary distribution $Q(z)$ and the true posterior distribution $P(z \mid \boldsymbol{o}, \boldsymbol{\Lambda})$ as:

$$\ln P(\boldsymbol{o} \mid \boldsymbol{\Lambda}) - \mathcal{F}^{(\mathrm{ML})}(Q, \boldsymbol{\Lambda}) = \mathrm{KL}[Q(z) \,\|\, P(z \mid \boldsymbol{o}, \boldsymbol{\Lambda})], \tag{13}$$

Since the true log-likelihood $\ln P(\boldsymbol{o} \mid \boldsymbol{\Lambda})$ is independent of $Q(z)$, maximizing the lower bound $\mathcal{F}^{(\mathrm{ML})}$ is equivalent to minimizing the KL divergence. In other words, $Q(z)$ can be regarded as an approximation of true posterior distribution $P(z \mid \boldsymbol{o}, \boldsymbol{\Lambda})$. The variational method iteratively maximizes $\mathcal{F}^{(\mathrm{ML})}$ with respect to $Q(z)$ and $\boldsymbol{\Lambda}$ holding the other parameters fixed.

$$\begin{aligned}&\text{(E-step): } Q^{(\mathrm{new})}(z) = \arg\max_{Q(z)} \mathcal{F}^{(\mathrm{ML})}(Q(z), \boldsymbol{\Lambda}^{(\mathrm{old})})\\ &\text{(M-step): } \boldsymbol{\Lambda}^{(\mathrm{new})} = \arg\max_{\boldsymbol{\Lambda}} \mathcal{F}^{(\mathrm{ML})}(Q^{(\mathrm{new})}(z), \boldsymbol{\Lambda})\end{aligned}$$

The E- and M-step are iterated until convergence of the lower bound $\mathcal{F}^{(\mathrm{ML})}$ is obtained by replacing $\boldsymbol{\Lambda}^{(\mathrm{old})} \leftarrow \boldsymbol{\Lambda}^{(\mathrm{new})}$.

To reduce computational complexity, hidden variables are assumed to be conditionally independent of one another, i.e.,

$$Q(z) \approx Q(z^{(1)}) Q(z^{(2)}), \tag{14}$$

where $\sum_{z^{(m)}} Q(z^{(m)}) = 1$, and $Q(z^{(m)})$ is called variational posterior distribution. In the E-step, the optimal variational posterior distributions $Q(z^{(m)})$ that maximize objective function $\mathcal{F}^{(\mathrm{ML})}$ are given as:

$$Q(z^{(m)}) \propto \exp\left[\sum_{z^{(\bar{m})}} Q(z^{(\bar{m})}) \ln P(z^{(m)} \mid \boldsymbol{\Lambda}) P(\boldsymbol{o} \mid z, \boldsymbol{\Lambda})\right], \tag{15}$$

where $\bar{m}$ represents the $\bar{m}$-th dimension, which is an alternative to the m-th dimension. The details of E- and M-step are

described in Appendix A.1.

5. Bayesian Approach for SL-HMMs

The ML method can efficiently estimate model parameters. However, since the ML method produces a point estimate of model parameters, the estimation accuracy may be degraded due to the over-fitting problem when there is insufficient data. In this paper, we propose the Bayesian approach for the training of SL-HMMs. The Bayesian approach has two advantageous properties for the training: using prior distributions and marginalization of model parameters. Therefore, the Bayesian approach can be expected higher generalization ability than the ML method. The MAP method focus on using prior distributions. On the other hand, the VB method has the advantageous properties of both using prior distributions and marginalization of model parameters. The training algorithms based on the MAP and VB method are derived in this section.

5.1 Maximum a Posteriori Method

The MAP method is derived in the same way as the ML method. The lower bound of the log-marginal likelihood $\mathcal{F}^{(\mathrm{MAP})}$ is defined by using Jensen's inequality as:

$$\begin{aligned}\ln P(\boldsymbol{o}\,|\,\boldsymbol{\Lambda})P(\boldsymbol{\Lambda}) &= \ln \sum_{z} Q(z)\frac{P(\boldsymbol{o}, z\,|\,\boldsymbol{\Lambda})P(\boldsymbol{\Lambda})}{Q(z)} \\ &\geq \sum_{z} Q(z)\ln\frac{P(z\,|\,\boldsymbol{\Lambda})P(\boldsymbol{o}\,|\,z,\boldsymbol{\Lambda})P(\boldsymbol{\Lambda})}{Q(z)} \\ &\triangleq \mathcal{F}^{(\mathrm{MAP})}(Q,\boldsymbol{\Lambda}),\end{aligned} \tag{16}$$

The MAP method can be seen as an extension of the ML method by using prior distributions $P(\boldsymbol{\Lambda})$. The re-estimation parameters using the MAP method are shown in Appendix B.1.

5.1.1 Prior Distribution

The MAP and VB methods have an advantage in that they can utilize prior distributions representing useful prior information on model parameters. Although arbitrary distributions can be used as prior distributions, conjugate prior distributions are widely used as prior distributions. A conjugate prior distribution is a distribution where the resulting posterior distribution belongs to the same distribution family as the prior distribution. The conjugate prior distributions of an SL-HMM are defined as:

$$\begin{aligned}P(\boldsymbol{\Lambda}) = &\prod_{m=1}^{2}\left[\mathcal{D}(\boldsymbol{\pi}^{(m)}\,|\,\boldsymbol{\phi}^{(m)})\prod_{i=1}^{K^{(m)}}\mathcal{D}(\boldsymbol{a}_i^{(m)}\,|\,\boldsymbol{\alpha}_i^{(m)})\right] \\ &\times\prod_{\boldsymbol{k}}\mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{k}}\,|\,\boldsymbol{\nu}_{\boldsymbol{k}},\xi_{\boldsymbol{k}}^{-1}\boldsymbol{\Sigma}_{\boldsymbol{k}})\mathcal{W}(\boldsymbol{\Sigma}_{\boldsymbol{k}}^{-1}\,|\,\eta_{\boldsymbol{k}},\boldsymbol{R}_{\boldsymbol{k}}),\end{aligned} \tag{17}$$

where $\mathcal{D}(\cdot)$ is a Dirichlet distribution and $\mathcal{N}(\cdot)\mathcal{W}(\cdot)$ is a Gauss-Wishart distribution. These distributions can be represented by a set of hyper-parameters $\{\boldsymbol{\phi}^{(1)},\boldsymbol{\phi}^{(2)},\boldsymbol{\alpha}_i^{(1)},\boldsymbol{\alpha}_i^{(2)},\boldsymbol{\nu}_{\boldsymbol{k}},\xi_{\boldsymbol{k}},\eta_{\boldsymbol{k}},\boldsymbol{R}_{\boldsymbol{k}}\}$.

Since the prior distributions of model parameters affect the estimation of posterior distributions in the MAP and VB methods, determining prior distributions is a serious problem in estimating appropriate models. We set the prior distribution as:

$$P(\boldsymbol{\Lambda}) \propto P(\boldsymbol{o}^{(\mathrm{prior})}\,|\,\boldsymbol{\Lambda})^{\frac{1}{\tau}}, \tag{18}$$

where $\boldsymbol{o}^{(\mathrm{prior})}$ is data given in advance (we called this prior data). We can control the degree of influence the prior distribution has on the posterior distribution by adjusting tuning parameter τ. The hyper-parameters based on prior data are given as Appendix B.2.

5.2 Variational Bayesian Method

5.2.1 Posterior Distribution

An approximate posterior distribution is estimated in the VB method by maximizing the lower bound of log-marginal likelihood instead of the true likelihood. The lower bound of log-marginal likelihood $\mathcal{F}^{(\mathrm{VB})}$ is defined by using Jensen's inequality:

$$\begin{aligned}\ln P(\boldsymbol{o}) &= \ln \sum_{z}\int Q(z,\boldsymbol{\Lambda})\frac{P(\boldsymbol{o},z,\boldsymbol{\Lambda})}{Q(z,\boldsymbol{\Lambda})}\mathrm{d}\boldsymbol{\Lambda} \\ &\geq \sum_{z}\int Q(z,\boldsymbol{\Lambda})\ln\frac{P(z\,|\,\boldsymbol{\Lambda})P(\boldsymbol{o}\,|\,z,\boldsymbol{\Lambda})P(\boldsymbol{\Lambda})}{Q(z,\boldsymbol{\Lambda})}\mathrm{d}\boldsymbol{\Lambda} \\ &\triangleq \mathcal{F}^{(\mathrm{VB})}(Q),\end{aligned} \tag{19}$$

where $Q(z,\boldsymbol{\Lambda})$ is an arbitrary distribution. The relation between the log-marginal likelihood and the lower bound $\mathcal{F}^{(\mathrm{VB})}$ is represented by using the KL divergence between $Q(z,\boldsymbol{\Lambda})$ and true posterior distribution $P(z,\boldsymbol{\Lambda}\,|\,\boldsymbol{o})$:

$$\ln P(\boldsymbol{o}) - \mathcal{F}^{(\mathrm{VB})}(Q) = \mathrm{KL}[Q(z,\boldsymbol{\Lambda})\,\|\,P(z,\boldsymbol{\Lambda}\,|\,\boldsymbol{o})]. \tag{20}$$

Eq. (20) means that the arbitrary distribution $Q(z,\boldsymbol{\Lambda})$ approximate true posterior distribution $P(z,\boldsymbol{\Lambda}\,|\,\boldsymbol{o})$.

To reduce computational complexity, random variables are assumed to be conditionally independent of one another, i.e.,

$$Q(z,\boldsymbol{\Lambda}) \approx Q(z^{(1)})Q(z^{(2)})Q(\boldsymbol{\Lambda}), \tag{21}$$

$$Q(\boldsymbol{\Lambda}) \approx \prod_{m=1}^{2}\left[Q(\boldsymbol{\pi}^{(m)})\prod_{i=1}^{K^{(m)}}Q(\boldsymbol{a}_i^{(m)})\right]\prod_{\boldsymbol{k}}Q(\boldsymbol{b}_{\boldsymbol{k}}), \tag{22}$$

where $\sum_{z^{(m)}}Q(z^{(m)}) = 1$ and $\int Q(\boldsymbol{\Lambda})\mathrm{d}\boldsymbol{\Lambda} = 1$. Under this assumption, the optimal variational posterior distributions $Q(z^{(m)})$ and $Q(\boldsymbol{\Lambda})$ that maximize the objective function $\mathcal{F}^{(\mathrm{VB})}$ are given as:

$$\begin{aligned}&Q(z^{(m)}) \\ &\propto \exp\left[\sum_{z^{(\bar{m})}}\int Q(z^{(\bar{m})})Q(\boldsymbol{\Lambda})\ln P(z^{(m)}\,|\,\boldsymbol{\Lambda})P(\boldsymbol{o}\,|\,z,\boldsymbol{\Lambda})\mathrm{d}\boldsymbol{\Lambda}\right],\end{aligned} \tag{23}$$

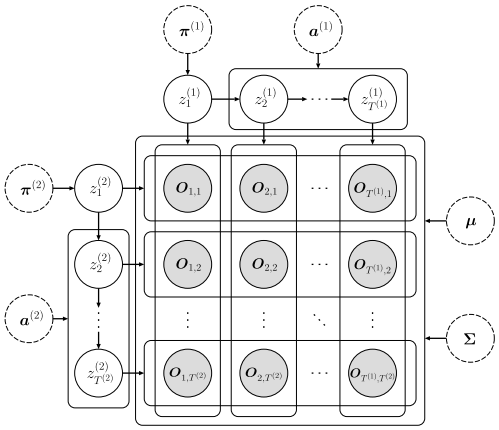


Fig. 3 Graphical model representation with model parameters of SL-HMMs using the ML method. The dashed circles represent model parameters.

$$Q(\boldsymbol{\Lambda}) \propto P(\boldsymbol{\Lambda}) \exp\left[\sum_{z} Q(z) \ln P(z\,|\,\boldsymbol{\Lambda}) P(\boldsymbol{o}\,|\,z, \boldsymbol{\Lambda})\right], \tag{24}$$

Since the approximate posterior distributions obtained, i.e., $Q(\boldsymbol{z}^{(m)})$ and $Q(\boldsymbol{\Lambda})$, are dependent on each other, these updates need to be iterated as with the EM algorithm.

$$\text{(VB E-step):}\ Q^{(\text{new})}(z) = \arg\max_{Q(z)} \mathcal{F}^{(\text{VB})}(Q(z)Q^{(\text{old})}(\boldsymbol{\Lambda}))$$

$$\text{(VB M-step):}\ Q^{(\text{new})}(\boldsymbol{\Lambda}) = \arg\max_{Q(\boldsymbol{\Lambda})} \mathcal{F}^{(\text{VB})}(Q^{(\text{new})}(z)Q(\boldsymbol{\Lambda}))$$

The update equations increase the value of the objective function $\mathcal{F}^{(\text{VB})}$ at each iteration until convergence by replacing $Q^{(\text{old})}(\boldsymbol{\Lambda}) \leftarrow Q^{(\text{new})}(\boldsymbol{\Lambda})$.

When conjugate prior distributions are used for prior distributions, the posterior distributions are represented by the same set of parameters $\{\hat{\boldsymbol{\phi}}^{(1)}, \hat{\boldsymbol{\phi}}^{(2)}, \hat{\boldsymbol{\alpha}}_i^{(1)}, \hat{\boldsymbol{\alpha}}_i^{(2)}, \hat{\boldsymbol{\nu}}_k, \hat{\xi}_k, \hat{\eta}_k, \hat{\boldsymbol{R}}_k\}$. Figures 3 and 4 show the graphical model representation with model parameters of SL-HMMs. The details of VB E- and M-step are described in Appendix C.1.

5.2.2 Predictive Distribution

Predictive distribution $P(\boldsymbol{x}\,|\,\boldsymbol{o})$ is estimated using Eq. (9) in the testing stage of the VB method. Since $Q(\boldsymbol{\Lambda})$ is an approximation of posterior distribution $P(\boldsymbol{\Lambda}\,|\,\boldsymbol{o})$, $Q(\boldsymbol{\Lambda})$ can be substituted for $P(\boldsymbol{\Lambda}\,|\,\boldsymbol{o})$ in Eq. (9). Although Eq. (9) includes a complicated expectation calculation, the same approximation as that in training can be applied.

In image recognition based on SL-HMMs using VB method, posterior distributions $P(\boldsymbol{\Lambda}\,|\,\boldsymbol{o}_c)$ are trained for each class c, i.e., subject, separately. Then, the likelihood of testing data $\boldsymbol{x}$, which is calculated by the predictive distribution $P(\boldsymbol{x}\,|\,\boldsymbol{o}_c)$, is compared among all subjects. The class $c^{(\text{Bayes})}$ which obtains the highest likelihood is chosen as the identification result.

5.3 Training Algorithm with Deterministic Annealing

An iterative procedure, such as an EM algorithm, suffers from the local maximum problem dependent on the initial parameter value. A deterministic annealing EM (DAEM) algorithm has been proposed to overcome this problem [24], [25]. We apply the DAEM algorithm to the training of SL-HMMs using the MAP (see Appendix B.3) and VB methods.

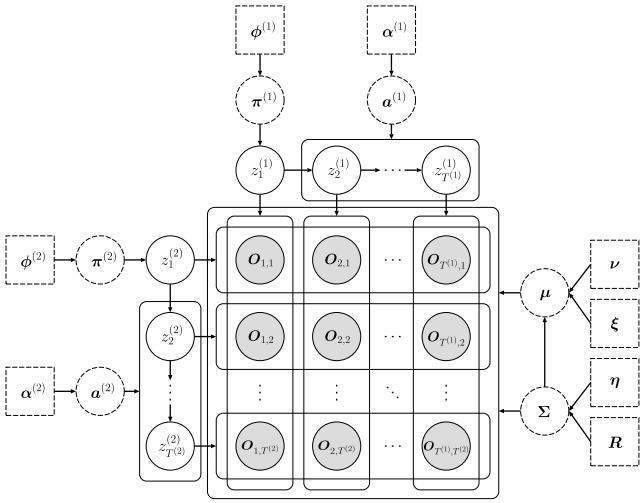


Fig. 4 Graphical model representation with model parameters of SL-HMMs using the VB method. The dashed rectangles represent hyperparameters.

5.3.1 Deterministic Annealing EM Algorithm

In this paper, instead of the simultaneous distribution $P(\boldsymbol{o}, z, \boldsymbol{\Lambda})$, another distribution $f(\boldsymbol{o}, z, \boldsymbol{\Lambda})$ is defined by using three temperature parameters as:

$$f(\boldsymbol{o}, z, \boldsymbol{\Lambda}) \triangleq P^{\beta_1}(z\,|\,\boldsymbol{\Lambda}) P^{\beta_2}(\boldsymbol{o}\,|\,z, \boldsymbol{\Lambda}) P^{\beta_3}(\boldsymbol{\Lambda}), \tag{25}$$

where β_1, β_2, and β_3 are respectively a temperature parameter of the initial and state transition probability distributions, the output probability distributions, and the prior distributions. Instead of Eq. (19), a lower bound $\mathcal{F}^{(\text{VBDA})}$ is defined by using Jensen's inequality:

$$\mathcal{F}^{(\text{VBDA})}(Q) \triangleq \sum_{z} \int Q(z, \boldsymbol{\Lambda}) \ln \frac{P^{\beta_1}(z\,|\,\boldsymbol{\Lambda}) P^{\beta_2}(\boldsymbol{o}\,|\,z, \boldsymbol{\Lambda}) P^{\beta_3}(\boldsymbol{\Lambda})}{Q(z, \boldsymbol{\Lambda})} \mathrm{d}\boldsymbol{\Lambda}. \tag{26}$$

Random variables are assumed to be conditionally independent of one another, which is the same as Eqs. (21) and (22). The optimal variational posterior distributions $Q(\boldsymbol{z}^{(m)})$ and $Q(\boldsymbol{\Lambda})$ that maximize the objective function $\mathcal{F}^{(\text{VBDA})}$ are given by the variational method as:

$$Q(\boldsymbol{z}^{(m)}) \propto \exp\left\{\sum_{z^{(\bar{m})}} \int Q(\boldsymbol{z}^{(\bar{m})}) Q(\boldsymbol{\Lambda}) \times \ln P^{\beta_1}(\boldsymbol{z}^{(m)}\,|\,\boldsymbol{\Lambda}) P^{\beta_2}(\boldsymbol{o}\,|\,z, \boldsymbol{\Lambda}) \mathrm{d}\boldsymbol{\Lambda}\right\}, \tag{27}$$

$$Q(\boldsymbol{\Lambda}) \propto P^{\beta_3}(\boldsymbol{\Lambda}) \exp\left\{\sum_{z} Q(z) \ln P^{\beta_1}(z\,|\,\boldsymbol{\Lambda}) P^{\beta_2}(\boldsymbol{o}\,|\,z, \boldsymbol{\Lambda})\right\}. \tag{28}$$

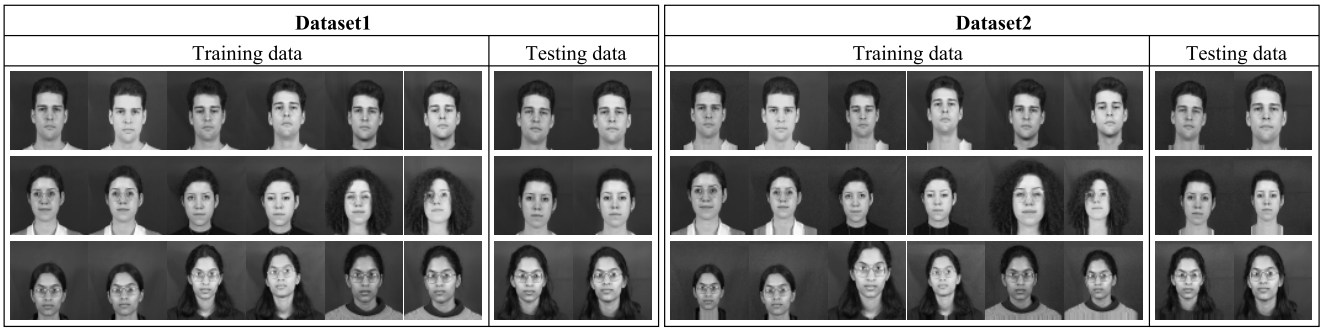


Fig. 5 Examples of images for the experiments.

By applying the deterministic annealing, the temperature parameters β_l are attached to the original variational posterior distributions, where $l = 1, 2, 3$ denotes the temperature parameter index. In the deterministic annealing process, the temperature parameters β_l are gradually increased from $\beta_l \simeq 0$ to $\beta_l = 1$. When $\beta_l \simeq 0$, the variational posterior distributions take a form with nearly uniform distribution. While the temperature parameter is increasing, the form of variational posterior distributions becomes close to that of the original variational posterior distributions. Finally at $\beta_l = 1$, the variational posterior distributions take the form of the original variational posterior distributions, and the reliable model parameters can be estimated without the effect of the local maximum problem. The re-estimation parameters are derived in Appendix C.2.

6. Experiments

6.1 Conditions

Face recognition experiments on the XM2VTS database [26] were conducted to evaluate the effectiveness of the proposed method. The experimental conditions are summarized in Table 2. We prepared two datasets for these experiments. **Dataset1** did not include many size and location variations, while **Dataset2** did. The cropping image sizes and center coordinates of cropping were randomly generated by Gaussian distributions in **Dataset2**. Figure 5 shows some examples of images for the experiments.

In the prior distributions, we used all training samples for all subjects as prior data in the research discussed. This is the same idea as that in a universal background model (UBM). We controlled the degree of influence the prior distribution by adjusting tuning parameter τ.

The temperature parameter $\beta_l(e)$ was updated by

$$\beta_l(e) = \left(\frac{e}{E}\right)^{\theta_l}, \tag{29}$$

where $e = 1, \ldots, E$ denotes the number of iterations of temperature updates. In these experiments, the number of temperature parameter updates was set to $E = 20$ and the schedule of temperature θ_l was varied to $\theta_l = 2^{\omega} (\omega = -6, \ldots, 6)$. Figure 6 shows plots of the schedules of temperature θ_l.

Table 2 Experimental conditions.

Database	XM2VTS [26]	
Original image size	720 × 576	
Dataset	**Dataset1**	**Dataset2**
Cropping image size	550 × 550	480 × 480–720 × 720
Center coordinates of cropping	(360, 288)	(360 ± 80, 288 ± 20)
Subsampling image size	64 × 64, grayscale	
Subject (class)	100	
Training data	6, 4, 2 images per person	
Testing data	2 images per person	
Method	**ML-EM**, **ML-DAEM**, **MAP-EM**, **MAP-DAEM**, **VB-EM**, **VB-DAEM**	
HMM structure	Left-to-right and top-to-bottom without skip transitions	
HMM state	16 × 16, 24 × 24, 32 × 32, 40 × 40, 48 × 48, 56 × 56	
Prior distribution	Universal background model	
Tuning parameter τ	100, 500, 1000, 2000, 3000, 4000, 5000, 6000, 7000, 8000, 9000, 10000	
Schedule of temperature θ_l	$2^{-6}, 2^{-4}, 2^{-2}, 2^{-1}, 2^{0}, 2^{1}, 2^{2}, 2^{4}, 2^{6}$	

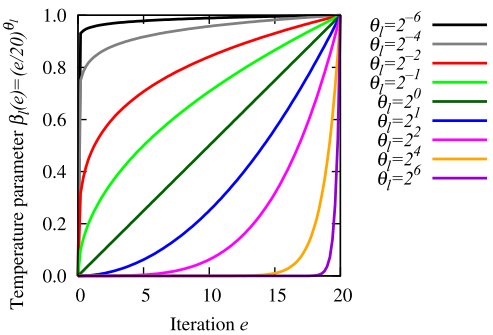


Fig. 6 Schedule of temperature θ_l.

The training recipe of **VB-DAEM** is summarized in Table 3. Steps 1–2 are a setting of prior distributions, steps 3–4 are an initialization, and steps 5–9 are an iterative procedure.

We performed two convolutional neural network (CNN)-based approaches [5], [6] (**CNN** and **CaffeNet**) in order to compare with the proposed method. In **CNN**, a CNN was trained by using the Caffe [30] based on each **Dataset1** and **Dataset2**. In **CaffeNet**, a pre-trained CNN (CaffeNet) [6], [30], which was trained by using the ImageNet Large Scale Visual Recognition Challenge 2012

Table 3 Training recipe of **VB-DAEM**.

1. Train UBM from all training samples for all subjects with flat hyper-parameters.
2. Set hyper-parameters Eqs. (A· 18)–(A· 23) from UBM by adjusting tuning parameter τ.
3. Set Eqs. (A· 2)–(A· 3) to flat probabilities.
4. Compute Eqs. (A· 33)–(A· 38).
5. Update temperature parameters $\beta_l(e)$ Eq. (29).
6. (VB E-step): Update Eqs. (A· 29)–(A· 31) and Eqs. (A· 2)–(A· 3).
7. (VB M-step): Update Eqs. (A· 39)–(A· 44).
8. Go to step 6 until convergence of lower bound $\mathcal{F}^{(\mathrm{VBDA})}$ Eq. (26).
9. Go to step 5 by adding 1 to e until $e = E$.

(ILSVRC2012) dataset [31], was used to extract image features. The details of CNN approaches are as follows:

CNN: The architecture of the CNN was $I(64, 1) - C(128, 10, 1, 55) - P(3, 2, 27) - C(256, 5, 1, 23) - P(3, 2, 11) - F(800) - F(600) - F(400) - O(100)$, where $I(i, d)$ indicates a input layer with d dimensional $i \times i$ size image, $C(f, w, s, o)$ indicates a convolutional layer with f filters of $w \times w$ size window with a stride of s and $o \times o$ size output, $P(w, s, o)$ indicates a pooling layer, $F(n)$ indicates a fully-connected layer with n units, and $O(c)$ indicate a output layer with c classes. The ReLU function and dropout with probability 0.5 were used in the convolutional and fully-connected layers.

CaffeNet: The image-feature vectors were composed of 4096 dimensions extracting the pre-trained CaffeNet of the 7$^{\mathrm{th}}$ fully-connected layer. The one-nearest neighbor was then used as the classifier.

6.2 Results

Mean vectors have been given in Fig. 7 to demonstrate what effect prior distributions had in the VB method. In the Fig. 7, (a) presents all training images for one subject and (b) presents mean vectors ($\boldsymbol{\mu}$ or $\boldsymbol{\nu}$) of the model obtained with the UBM, **VB-DAEM**, and **ML-DAEM**. Although from Fig. 7 (b) it can be seen that the UBM roughly represents a facial shape, it is difficult to identify the characteristics of a particular subject. As tuning parameter τ is increased in **VB-DAEM**, the mean vector gradually changes from the UBM to the image of the subject in Fig. 7 (a). If the UBM is used as the prior distribution with appropriate tuning parameter τ, similar state alignments are expected to be obtained for all subject models and this therefore avoids the over-fitting problem. It can actually be seen in Fig. 7 (b) that **VB-DAEM** ($\tau = 1000$) preserved the shape of the face using the UBM, even though the shape of **ML-DAEM** had collapsed due to over-fitting. However, tuning parameter τ needs to be carefully determined because the optimal value depends on the amount of training data and the number of states.

Figures 8 (a) and 8(b) respectively show the recognition rates using six images as training data for each subject on **Dataset1** and **Dataset2**. A tuning parameter with which the highest recognition was obtained was used in each state.

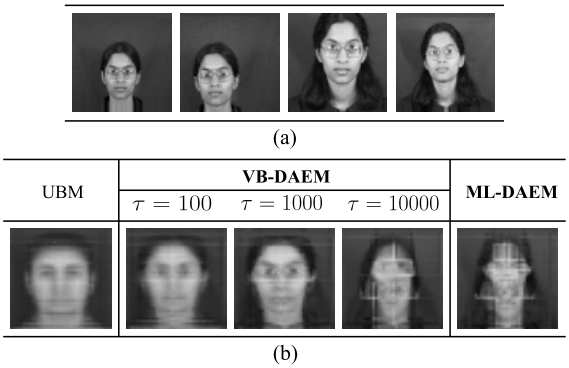


Fig. 7 Effect on mean vectors of prior distributions in **Dataset2**. The training data comprised four images and there were 40×40 HMM states. (a) All training images of one subject. (b) Mean vectors ($\boldsymbol{\mu}$ or $\boldsymbol{\nu}$) of the model obtained with the UBM, **VB-DAEM**, and **ML-DAEM**.

The schedule of temperature $\theta_1 = 2^0$, $\theta_2 = 2^0$, and $\theta_3 = 2^{-6}$ were used which could stably be obtained high recognition performance each method in preliminary experiments. In the EM algorithm, we can see from the results that **MAP-EM** and **VB-EM** achieved significantly better recognition rates than **ML-EM**. Similarly, in the DAEM algorithm, **MAP-DAEM** and **VB-DAEM** outperformed **ML-DAEM**. These results suggest that proposed methods, i.e., the MAP and VB methods for SL-HMMs, mitigated the over-fitting problem and achieved higher generalization ability than the ML method. Comparing the computational cost per one iteration, the VB method is on the same computational order as the ML method. However, in the VB method, convergence of lower bound requires a lot of iterations because of marginalization of model parameters. Similar performance was obtained under all conditions by comparing the MAP and VB methods. However, the VB method was slightly better than MAP when the appropriate number of states was selected. The highest recognition rates for **MAP-DAEM** (**Dataset1**: 85.0%, **Dataset2**: 81.0%) and **VB-DAEM** (**Dataset1**: 85.5%, **Dataset2**: 82.0%) were obtained at 40×40 states. Therefore, we confirmed that the use of a prior distribution was more effective than the marginalization of model parameters in this task. In addition, much improvement was obtained by the DAEM algorithm compared with the EM algorithm in each method. This is because the performance of the EM algorithm was degraded by the local maximum problem, and the DAEM algorithm was able to reduce the effect of unreliable initial parameters in the training of SL-HMMs.

Figures 8 (c) and 8 (d) respectively show the recognition rates obtained when the numbers of images were changed for **Dataset1** and **Dataset2**. The HMM states and tuning parameters with which the highest recognition was obtained were used for each number of training data images, and $\theta_1 = 2^0$, $\theta_2 = 2^0$, and $\theta_3 = 2^{-6}$ were used for the schedule of temperature. The MAP and VB methods achieved higher recognition rates than ML method for all numbers of training images. The difference between ML method and MAP/VB methods became especially larger when small

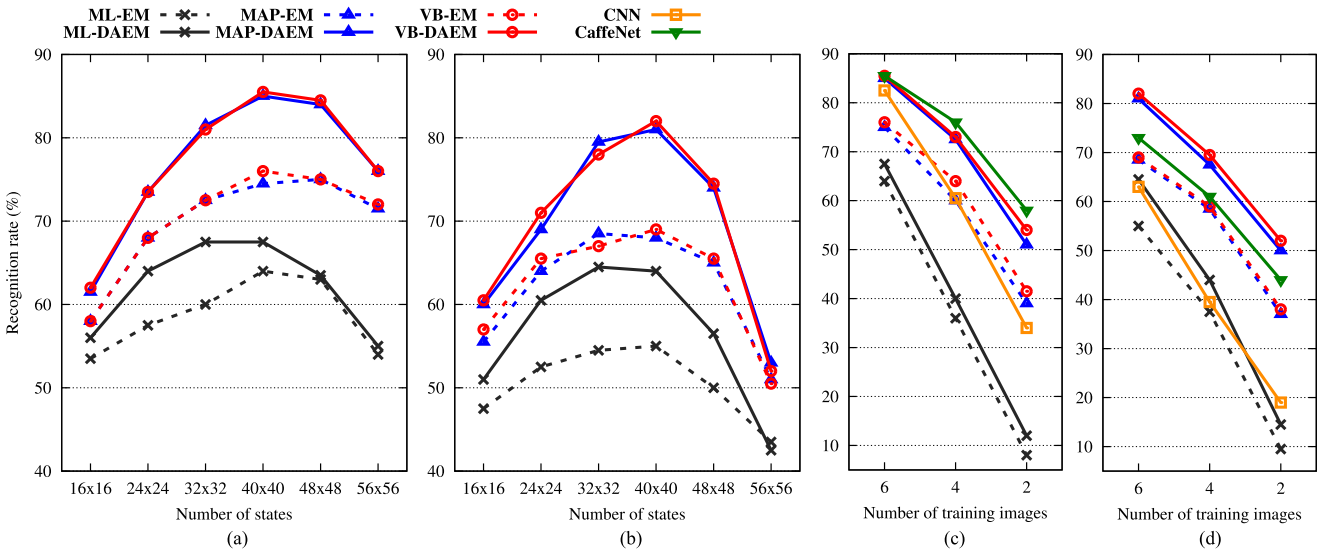


Fig. 8 Recognition rates obtained in image recognition experiments. (a) The effect of the number of HMM states on **Dataset1**. (b) The effect of the number of HMM states on **Dataset2**. (c) The effect of the amount of training data on **Dataset1**. (d) The effect of the amount of training data on **Dataset2**.

Table 4 Recognition rates obtained in schedule of temperature experiments. Where θ_1, θ_2, and θ_3 are respectively a schedule of the initial and state transition probability distributions, the output probability distributions, and the prior distributions. Bold numbers indicate recognition rates of 87.0% or more.

θ_1		2^0	2^1	2^2	2^4	2^6	2^0	2^1	2^2	2^4	2^6	2^0	2^1	2^2	2^4	2^6
θ_2				2^0					2^1					2^2		
θ_3	2^{-6}	85.5	85.0	85.0	85.5	85.5	85.0	84.5	85.5	86.0	86.5	84.5	85.5	85.5	**87.0**	86.5
	2^{-4}	84.0	83.5	83.5	83.5	83.5	85.5	85.0	84.5	86.5	86.5	85.0	85.5	84.5	86.0	86.0
	2^{-2}	85.5	84.5	85.0	85.0	85.0	86.0	86.0	86.0	86.0	86.0	85.5	86.0	85.0	85.5	85.5
	2^{-1}	80.0	80.5	81.0	80.0	80.5	85.5	**87.0**	**87.5**	**88.0**	**88.0**	85.0	86.0	**87.0**	**87.0**	**87.0**
	2^0	77.5	78.0	78.0	77.5	78.0	85.0	85.0	84.5	84.5	84.5	85.5	86.0	86.0	86.5	86.5

numbers of training images were used. These results clearly show that the proposed methods can estimate high generalization ability when there is insufficient training data. By contrast, it is considered that the difference ML method and MAP/VB methods become smaller when there is sufficient training data. Although the MAP and the VB methods had almost the same recognition rates, **VB-DAEM** (**Dataset1**: 54.0%, **Dataset2**: 52.0%) obtained better recognition rates than **MAP-DAEM** (**Dataset1**: 51.0%, **Dataset2**: 50.0%) when only two training images were used. Comparing the proposed method with **CNN**, **VB-DAEM** achieved better recognition rates than **CNN**. These results indicate that the proposed method is more effective than **CNN** when the amount of training data is insufficient. However, the number of training images in the experiments was small to train the CNN. Therefore, in the future, we should perform on large datasets. Although **VB-DAEM** and **CaffeNet** had almost the same recognition rates in **Dataset1**, **VB-DAEM** outperformed **CaffeNet** in **Dataset2**. These results suggest that the proposed method is more robust to geometric variations than **CaffeNet**.

The effect of the schedule of temperature θ_l was evaluated in **VB-DAEM**. Table 4 shows recognition rates obtained in the experiments. Six images in **Dataset1** were used as training data for each subject. SL-HMMs with 40×40 states and tuning parameter $\tau = 4000$ were used in the experiments. Recognition rates was improved by using the appropriate schedule of temperature θ_l. By contrast, an inappropriate schedule caused a decrease in the performance of generalization ability. Statistics of UBM used in prior distributions are high reliability because it is trained in advance. On the other hand, statistics of training data are low reliability in early stage of training. Therefore, high recognition rates was obtained by bringing the schedule of temperature $\theta_l = 1$ in the order of θ_3, θ_2, and θ_1. As future work, a method of automatically determining the schedule will be needed to obtain an appropriate schedule.

7. Conclusion

This paper proposed an image recognition method based on separable lattice hidden Markov models (SL-HMMs) using the maximum a posteriori (MAP) and variational Bayesian (VB) methods. An improved training algorithm using the deterministic annealing expectation maximization (DAEM) algorithm was also derived. Face recognition experiments performed on the XM2VTS database showed that the MAP and VB methods offer better recognition performance than

the maximum likelihood (ML) method. These results suggest that the MAP and VB methods are useful for image recognition applications based on SL-HMMs. The use of prior distributions was more effective than the marginalization of model parameters in this task. The DAEM algorithm was able to reduce the effect of unreliable initial parameters in the training of SL-HMMs. Additionally, comparative experiment results showed that the proposed method was more robust to geometric variations than convolutional neural networks. Subjects for future work include applying the Bayesian criterion to image recognition based on hidden Markov eigen-image models [20], which integrate SL-HMMs and factor analyzers, and performing experiments on various image recognition tasks.

Acknowledgments

This work was supported by Grant-in-aid for JSPS Fellows Grant Number 15J08391 and the Hori Sciences and Arts Foundation.

References

[1] M.A. Turk and A.P. Pentland, "Face recognition using eigenfaces," Conference on Computer Vision and Pattern Recognition, pp.586–591, 1991.
[2] S. Watanabe and N. Pakvasa, "Subspace method of pattern recognition," International Joint Conference on Pattern Recognition, pp.25–32, 1973.
[3] D.G. Lowe, "Distinctive image features from scale-invariant keypoints," Int. J. Comput. Vision., vol.60, no.2, pp.91–110, 2004.
[4] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," Conference on Computer Vision and Pattern Recognition, pp.886–893, 2005.
[5] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proc. IEEE, vol.86, no.11, pp.2278–2324, 1998.
[6] A. Krizhevsky, I. Sutskever, and G.E. Hinton, "ImageNet classification with deep convolutional neural networks," Conference on Neural Information Processing Systems, pp.1097–1105, 2012.
[7] F.S. Samaria, Face recognition using hidden Markov models, Ph. D. dissertation, University of Cambridge, 1994.
[8] A.V. Nefian and M.H. Hayes, "Hidden Markov models for face recognition," International Conference on Acoustics, Speech and Signal Processing, vol.5, pp.2721–2724, 1998.
[9] S.-S. Kuo and O.E. Agazzi, "Keyword spotting in poorly printed documents using pseudo 2-D hidden Markov models," IEEE Trans. Pattern Anal. Machine Intell., vol.16, no.8, pp.842–848, 1994.
[10] A.V. Nefian and M.H. Hayes, "Maximum likelihood training of the embedded HMM for face detection and recognition," International Conference on Image Processing, vol.1, pp.33–36, 2000.
[11] X. Ma, W.A. Pearlman, J.W. Woods, D. Schonfeld, A. Khokhar, and L. Lu, "Image segmentation and classification based on a 2D distributed hidden Markov model," Society of Photo-optical Instrumentation Engineers, vol.6822, 2008.
[12] J. Li, A. Najmi, and R.M. Gra, "Image classification by a two dimensional hidden Markov model," IEEE Trans. Signal Process., vol.48, no.2, pp.517–533, 2000.
[13] H. Othman and T. Aboiilnasr, "A simplified second-order HMM with application to face recognition," International Symposium on Circuits and Systems, vol.2, pp.161–164, 2001.
[14] J.-T. Chien and C.-P. Liao, "Maximum confidence hidden Markov modeling for face recognition," IEEE Trans. Pattern Anal. Mach. Intell., vol.30, no.4, pp.606–616, 2008.
[15] D. Kurata, Y. Nankaku, K. Tokuda, T. Kitamura, and Z. Ghahramani, "Face recognition based on separable lattice HMMs," International Conference on Acoustics, Speech and Signal Processing, vol.5, pp.737–740, 2006.
[16] A. Tamamori, Y. Nankaku, and K. Tokuda, "An extension of separable lattice 2-D HMMs for rotational data variations," IEICE Trans. Inf. & Syst., vol.E95-D, no.8, pp.2074–2083, 2012.
[17] K. Kumaki, Y. Nankaku, and K. Tokuda, "Face recognition based on extended separable lattice 2-D HMMs," International Conference on Acoustics, Speech and Signal Processing, pp.2209–2212, 2012.
[18] Y. Takahashi, A. Tamamori, Y. Nankaku, and K. Tokuda, "Face recognition based on separable lattice 2-D HMM with state duration modeling," International Conference on Acoustics, Speech and Signal Processing, pp.2162–2165, 2010.
[19] A. Tamamori, Y. Nankaku, and K. Tokuda, "Image recognition based on separable lattice trajectory 2-D HMMs," IEICE Trans. Inf. & Syst., vol.E97-D, no.7, pp.1842–1854, 2014.
[20] Y. Nankaku and K. Tokuda, "Face recognition using hidden Markov eigenface models," International Conference on Acoustics, Speech and Signal Processing, vol.2, pp.469–472, 2007.
[21] J.-L. Gauvain and C.-H. Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," IEEE Trans. Speech Audio Process., vol.2, no.2, pp.291–298, 1994.
[22] H. Attias, "Inferring parameters and structure of latent variable models by variational Bayes," Conference on Uncertainty in Artificial Intelligence, pp.21–30, 1999.
[23] K. Sawada, A. Tamamori, K. Hashimoto, Y. Nankaku, and K. Tokuda, "Face recognition based on separable lattice 2-D HMMs using variational bayesian method," International Conference on Acoustics, Speech and Signal Processing, pp.2205–2208, 2012.
[24] N. Ueda and R. Nakano, "Deterministic annealing EM algorithm," Neural Networks, vol.11, no.2, pp.271–282, 1998.
[25] K. Katahira, K. Watanabe, and M. Okada, "Deterministic annealing variant of variational Bayes method," Journal of Physics: Conference Series, vol.95, 012015, 2008.
[26] K. Messer, J. Matas, J. Kittler, and K. Jonsson, "XM2VTSDB: The extended M2VTS database," International Conference on Audio and Video-based Biometric Person Authentication, pp.72–77, 1999.
[27] B.P. Carlin and S. Chib, "Bayesian model choice via Markov chain Monte Carlo," J. Royal Statistical Society, vol.57, no.3, pp.473–484, 1995.
[28] A.P. Dempster, N.M. Laird, and D.B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," J. Royal Statistical Society: Series B, vol.39, no.1, pp.1–38, 1977.
[29] M.I. Jordan, Z. Ghahramani, T.S. Jaakkola, and L.K. Saul, "An introduction to variational methods for graphical models," Machine Learning, vol.37, no.2, pp.183–233, 1999.
[30] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R.B. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," ACM international conference on Multimedia, pp.675–678, 2014.
[31] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A.C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," Int. J. Comput. Vision., vol.115, no.3, pp.211–252, 2015.
[32] L.R. Rabiner and B.H. Juang, "An introduction to hidden Markov models," IEEE ASSP Magazine, vol.3, no.1, pp.4–16, 1986.

Appendix A: Derivation of ML Method for SL-HMMs

A.1 EM Algorithm for ML Method

The variational posterior distribution $Q(z^{(m)})$ can be represented as follows:

$$Q(\boldsymbol{z}^{(m)}) \propto \exp\left[\sum_{i=1}^{K^{(m)}} z_{i,1}^{(m)} \ln \pi_i^{(m)}\right] \times \exp\left[\sum_{t^{(m)}=2}^{T^{(m)}} \sum_{i=1}^{K^{(m)}} \sum_{j=1}^{K^{(m)}} z_{i,t^{(m)}-1}^{(m)} z_{j,t^{(m)}}^{(m)} \ln a_{ij}^{(m)}\right] \times \exp\left[\sum_{\boldsymbol{t}} \sum_{i=1}^{K^{(m)}} \sum_{j=1}^{K^{(\bar{m})}} z_{i,t^{(m)}}^{(m)} \langle z_{j,t^{(\bar{m})}}^{(\bar{m})} \rangle_{Q(\boldsymbol{z}^{(\bar{m})})} \ln \boldsymbol{b}_{\boldsymbol{k}}(\boldsymbol{o}_{\boldsymbol{t}})\right]. \quad (A\cdot 1)$$

The expectation value with respect to $Q(\boldsymbol{z}^{(m)})$ is computed in the E-step by the following equations:

$$\langle z_{i,t^{(m)}}^{(m)} \rangle_{Q(\boldsymbol{z}^{(m)})} = \sum_{\boldsymbol{z}^{(m)}} Q(\boldsymbol{z}^{(m)}) z_{i,t^{(m)}}^{(m)}, \quad (A\cdot 2)$$

$$\langle z_{i,t^{(m)}-1}^{(m)} z_{j,t^{(m)}}^{(m)} \rangle_{Q(\boldsymbol{z}^{(m)})} = \sum_{\boldsymbol{z}^{(m)}} Q(\boldsymbol{z}^{(m)}) z_{i,t^{(m)}-1}^{(m)} z_{j,t^{(m)}}^{(m)}, \quad (A\cdot 3)$$

$$\langle z_{\boldsymbol{k},\boldsymbol{t}} \rangle_{Q(\boldsymbol{z})} = \sum_{\boldsymbol{z}^{(1)}} \sum_{\boldsymbol{z}^{(2)}} Q(\boldsymbol{z}^{(1)}) Q(\boldsymbol{z}^{(2)}) z_{i,t^{(1)}}^{(1)} z_{j,t^{(2)}}^{(2)}, \quad (A\cdot 4)$$

where $\langle\cdot\rangle_{Q(\cdot)}$ denotes the expectation with respect to the posterior distribution $Q(\cdot)$ and $z_{i,t^{(m)}}^{(m)}$ is the Kronecker delta function:

$$z_{i,t^{(m)}}^{(m)} = \delta(z_{t^{(m)}}^{(m)}, i) = \begin{cases} 0 & (z_{t^{(m)}}^{(m)} \neq i) \\ 1 & (z_{t^{(m)}}^{(m)} = i) \end{cases}. \quad (A\cdot 5)$$

The variational posterior distribution $Q(\boldsymbol{z}^{(m)})$ in Eq. (A· 1) has a Markovian structure as the likelihood function of an standard one-dimensional HMM. Therefore, Eqs. (A· 2) and (A· 3) can be computed efficiently by the forward-backward algorithm given in [32].

In the M-step, the model parameters of the SL-HMMs can be updated by sufficient statistics of the training data as follows:

$$\pi_i^{(m)} = \langle z_{i,1}^{(m)} \rangle_{Q(\boldsymbol{z}^{(m)})}, \quad (A\cdot 6)$$

$$a_{ij}^{(m)} = \frac{N_{ij}^{(m)}}{\sum_{t^{(m)}=2}^{T^{(m)}} \langle z_{i-1,t^{(m)}}^{(m)} \rangle_{Q(\boldsymbol{z}^{(m)})}}, \quad (A\cdot 7)$$

$$\boldsymbol{\mu}_{\boldsymbol{k}} = \boldsymbol{F}_{\boldsymbol{k}}, \quad (A\cdot 8)$$

$$\boldsymbol{\Sigma}_{\boldsymbol{k}} = \boldsymbol{S}_{\boldsymbol{k}}, \quad (A\cdot 9)$$

where statistics $N_{ij}^{(m)}$, $\boldsymbol{F}_{\boldsymbol{k}}$, and $\boldsymbol{S}_{\boldsymbol{k}}$ are represented as follows:

$$N_{ij}^{(m)} = \sum_{t^{(m)}=2}^{T^{(m)}} \langle z_{i,t^{(m)}-1}^{(m)} z_{j,t^{(m)}}^{(m)} \rangle_{Q(\boldsymbol{z}^{(m)})}, \quad (A\cdot 10)$$

$$N_{\boldsymbol{k}} = \sum_{\boldsymbol{t}} \langle z_{\boldsymbol{k},\boldsymbol{t}} \rangle_{Q(\boldsymbol{z})}, \quad (A\cdot 11)$$

$$\boldsymbol{F}_{\boldsymbol{k}} = \frac{1}{N_{\boldsymbol{k}}} \sum_{\boldsymbol{t}} \langle z_{\boldsymbol{k},\boldsymbol{t}} \rangle_{Q(\boldsymbol{z})} \boldsymbol{o}_{\boldsymbol{t}}, \quad (A\cdot 12)$$

$$\boldsymbol{S}_{\boldsymbol{k}} = \frac{1}{N_{\boldsymbol{k}}} \sum_{\boldsymbol{t}} \langle z_{\boldsymbol{k},\boldsymbol{t}} \rangle_{Q(\boldsymbol{z})} (\boldsymbol{o}_{\boldsymbol{t}} - \boldsymbol{F}_{\boldsymbol{k}})(\boldsymbol{o}_{\boldsymbol{t}} - \boldsymbol{F}_{\boldsymbol{k}})^{\top}. \quad (A\cdot 13)$$

Appendix B: Derivation of MAP Method for SL-HMMs

B.1 EM Algorithm for MAP Method

The model parameters of SL-HMMs using the MAP method can be updated by sufficient statistics and hyper-parameters as follows:

$$\pi_i^{(m)} = \frac{\langle z_{i,1}^{(m)} \rangle_{Q(\boldsymbol{z}^{(m)})} + \phi_i^{(m)} - 1}{1 + \sum_{i'=1}^{K^{(m)}} (\phi_{i'}^{(m)} - 1)}, \quad (A\cdot 14)$$

$$a_{ij}^{(m)} = \frac{N_{ij}^{(m)} + \alpha_{ij}^{(m)} - 1}{\sum_{t^{(m)}=2}^{T^{(m)}} \langle z_{i-1,t^{(m)}}^{(m)} \rangle_{Q(\boldsymbol{z}^{(m)})} + \sum_{j'=1}^{K^{(m)}} (\alpha_{ij'}^{(m)} - 1)}, \quad (A\cdot 15)$$

$$\boldsymbol{\mu}_{\boldsymbol{k}} = \frac{N_{\boldsymbol{k}} \boldsymbol{F}_{\boldsymbol{k}} + \xi_{\boldsymbol{k}} \boldsymbol{\nu}_{\boldsymbol{k}}}{N_{\boldsymbol{k}} + \xi_{\boldsymbol{k}}}, \quad (A\cdot 16)$$

$$\boldsymbol{\Sigma}_{\boldsymbol{k}} = \frac{1}{N_{\boldsymbol{k}} + \eta_{\boldsymbol{k}} - D} \Big[\sum_{\boldsymbol{t}} \langle z_{\boldsymbol{k},\boldsymbol{t}} \rangle_{Q(\boldsymbol{z})} (\boldsymbol{o}_{\boldsymbol{t}} - \boldsymbol{\mu}_{\boldsymbol{k}})(\boldsymbol{o}_{\boldsymbol{t}} - \boldsymbol{\mu}_{\boldsymbol{k}})^{\top} + \xi_{\boldsymbol{k}} (\boldsymbol{\mu}_{\boldsymbol{k}} - \boldsymbol{\nu}_{\boldsymbol{k}})(\boldsymbol{\mu}_{\boldsymbol{k}} - \boldsymbol{\nu}_{\boldsymbol{k}})^{\top} + \boldsymbol{R}_{\boldsymbol{k}} \Big], \quad (A\cdot 17)$$

where D is a dimension of observation $\boldsymbol{o}$.

B.2 Hyper-Parameters

The hyper-parameters based on prior data are given as:

$$\phi_i^{(m)} = \frac{\langle \tilde{z}_{i,t^{(m)}}^{(m)} \rangle_{Q(\boldsymbol{z}^{(m)})}}{\tau} + 1, \quad (A\cdot 18)$$

$$\alpha_{ij}^{(m)} = \frac{\tilde{N}_{ij}^{(m)}}{\tau} + 1, \quad (A\cdot 19)$$

$$\boldsymbol{\nu}_{\boldsymbol{k}} = \tilde{\boldsymbol{F}}_{\boldsymbol{k}}, \quad (A\cdot 20)$$

$$\xi_{\boldsymbol{k}} = \frac{\tilde{N}_{\boldsymbol{k}}}{\tau}, \quad (A\cdot 21)$$

$$\eta_{\boldsymbol{k}} = \frac{\tilde{N}_{\boldsymbol{k}}}{\tau} + D, \quad (A\cdot 22)$$

$$\boldsymbol{R}_{\boldsymbol{k}} = \frac{\tilde{N}_{\boldsymbol{k}}}{\tau} \tilde{\boldsymbol{S}}_{\boldsymbol{k}}, \quad (A\cdot 23)$$

where $\tilde{\cdot}$ denotes statistics of prior data and τ is the tuning parameter.

B.3 DAEM Algorithm for MAP Method

The model parameters of SL-HMMs using the MAP method with the DAEM algorithm can be updated by sufficient statistics and hyper-parameters as follows:

$$\pi_i^{(m)} = \frac{\beta_1 \langle z_{i,1}^{(m)} \rangle_{Q(\boldsymbol{z}^{(m)})} + \beta_3 (\phi_i^{(m)} - 1)}{\beta_1 + \beta_3 \sum_{i'=1}^{K^{(m)}} (\phi_{i'}^{(m)} - 1)}, \quad (A\cdot 24)$$

$$a_{ij}^{(m)} = \frac{\beta_1 N_{ij}^{(m)} + \beta_3 (\alpha_{ij}^{(m)} - 1)}{\beta_1 \sum_{t^{(m)}=2}^{T^{(m)}} \langle z_{i-1,t^{(m)}}^{(m)} \rangle_{Q(\boldsymbol{z}^{(m)})} + \beta_3 \sum_{j'=1}^{K^{(m)}} (\alpha_{ij'}^{(m)} - 1)}, \quad (A\cdot 25)$$

$$\boldsymbol{\mu}_{\boldsymbol{k}} = \frac{\beta_2 N_{\boldsymbol{k}} \boldsymbol{F}_{\boldsymbol{k}} + \beta_3 \xi_{\boldsymbol{k}} \boldsymbol{\nu}_{\boldsymbol{k}}}{\beta_2 N_{\boldsymbol{k}} + \beta_3 \xi_{\boldsymbol{k}}}, \quad (A\cdot 26)$$

$$\boldsymbol{\Sigma}_{\boldsymbol{k}} = \frac{1}{\beta_2 N_{\boldsymbol{k}} + \beta_3(\eta_{\boldsymbol{k}} - D)} \times\Bigg[\beta_2 \sum_{\boldsymbol{t}} \langle z_{\boldsymbol{k},\boldsymbol{t}}\rangle_{Q(z)}(\boldsymbol{o}_{\boldsymbol{t}} - \boldsymbol{\mu}_{\boldsymbol{k}})(\boldsymbol{o}_{\boldsymbol{t}} - \boldsymbol{\mu}_{\boldsymbol{k}})^\top + \beta_3\xi_{\boldsymbol{k}}(\boldsymbol{\mu}_{\boldsymbol{k}} - \boldsymbol{\nu}_{\boldsymbol{k}})(\boldsymbol{\mu}_{\boldsymbol{k}} - \boldsymbol{\nu}_{\boldsymbol{k}})^\top + \beta_3\boldsymbol{R}_{\boldsymbol{k}}\Bigg]. \quad (\mathrm{A}\cdot 27)$$

Appendix C: Derivation of VB Method for SL-HMMs

C.1 EM Algorithm for VB Method

In the VB E-step, the optimal variational posterior distributions $Q(z^{(m)})$ that maximize the objective function $\mathcal{F}^{(\mathrm{VB})}$ are given as:

$$Q(z^{(m)}) \propto \exp\left[\sum_{i=1}^{K^{(m)}} z_{i,1}^{(m)} \langle \ln \pi_i^{(m)}\rangle_{Q(\boldsymbol{\pi}^{(m)})}\right] \times \exp\left[\sum_{t^{(m)}=2}^{T^{(m)}}\sum_{i=1}^{K^{(m)}}\sum_{j=1}^{K^{(m)}} z_{i,t^{(m)}-1}^{(m)} z_{j,t^{(m)}}^{(m)} \langle \ln a_{ij}^{(m)}\rangle_{Q(\boldsymbol{a}_i^{(m)})}\right] \times \exp\left[\sum_{\boldsymbol{t}}\sum_{i=1}^{K^{(m)}}\sum_{j=1}^{K^{(\tilde{m})}} z_{i,t^{(m)}}^{(m)} \langle z_{j,t^{(\tilde{m})}}^{(\tilde{m})}\rangle_{Q(z^{(\tilde{m})})} \langle \ln \boldsymbol{b}_{\boldsymbol{k}}(\boldsymbol{o}_{\boldsymbol{t}})\rangle_{Q(\boldsymbol{b}_{\boldsymbol{k}})}\right], \quad (\mathrm{A}\cdot 28)$$

The updates of the expectations of model parameters are derived as:

$$\langle \ln \pi_i^{(m)}\rangle_{Q(\boldsymbol{\pi}^{(m)})} = \Psi(\hat{\phi}_i^{(m)}) - \Psi\left(\sum_{i'=1}^{K^{(m)}} \hat{\phi}_{i'}^{(m)}\right), \quad (\mathrm{A}\cdot 29)$$

$$\langle \ln a_{ij}^{(m)}\rangle_{Q(\boldsymbol{a}_i^{(m)})} = \Psi(\hat{\alpha}_{ij}^{(m)}) - \Psi\left(\sum_{j'=1}^{K^{(m)}} \hat{\alpha}_{ij'}^{(m)}\right), \quad (\mathrm{A}\cdot 30)$$

$$\langle \ln \boldsymbol{b}_{\boldsymbol{k}}(\boldsymbol{o}_{\boldsymbol{t}})\rangle_{Q(\boldsymbol{b}_{\boldsymbol{k}})} = -\frac{1}{2}\Bigg\{D\ln\pi + \frac{D}{\hat{\xi}_{\boldsymbol{k}}} - \sum_{d=1}^{D}\Psi\left(\frac{\hat{\eta}_{\boldsymbol{k}} + 1 - d}{2}\right) + \ln|\hat{\boldsymbol{R}}_{\boldsymbol{k}}| + \mathrm{Tr}\left[\hat{\eta}_{\boldsymbol{k}}\hat{\boldsymbol{R}}_{\boldsymbol{k}}^{-1}(\boldsymbol{o}_{\boldsymbol{t}} - \hat{\boldsymbol{\nu}}_{\boldsymbol{k}})(\boldsymbol{o}_{\boldsymbol{t}} - \hat{\boldsymbol{\nu}}_{\boldsymbol{k}})^\top\right]\Bigg\}, \quad (\mathrm{A}\cdot 31)$$

where $\Psi(\cdot)$ is a digamma function. The expectation value in Eqs. (A· 2) and (A· 3) can be computed efficiently by the forward-backward algorithm given in [32].

In the VB M-step, the optimal variational posterior distributions $Q(\boldsymbol{\Lambda})$ that maximize the objective function $\mathcal{F}^{(\mathrm{VB})}$ are given as:

$$Q(\boldsymbol{\Lambda}) \propto P(\boldsymbol{\Lambda})\prod_{m=1}^{2}\Bigg\{\exp\left[\sum_{i=1}^{K^{(m)}}\langle z_{i,t^{(m)}}^{(m)}\rangle_{Q(z^{(m)})}\ln\pi_i^{(m)}\right] \times \exp\left[\sum_{t^{(m)}=2}^{T^{(m)}}\sum_{i=1}^{K^{(m)}}\sum_{j=1}^{K^{(m)}}\langle z_{i,t^{(m)}-1}^{(m)} z_{j,t^{(m)}}^{(m)}\rangle_{Q(z^{(m)})}\ln a_{ij}^{(m)}\right]\Bigg\} \times \exp\left[\sum_{\boldsymbol{t}}\sum_{\boldsymbol{k}}\langle z_{\boldsymbol{k},\boldsymbol{t}}\rangle_{Q(z)}\ln\boldsymbol{b}_{\boldsymbol{k}}(\boldsymbol{o}_{\boldsymbol{t}})\right], \quad (\mathrm{A}\cdot 32)$$

The posterior distribution of model parameters $Q(\boldsymbol{\Lambda})$ can be updated by statistics of the training data as follows:

$$\hat{\phi}_i^{(m)} = \langle z_{i,1}^{(m)}\rangle_{Q(z^{(m)})} + \phi_i^{(m)}, \quad (\mathrm{A}\cdot 33)$$

$$\hat{\alpha}_{ij}^{(m)} = N_{ij}^{(m)} + \alpha_{ij}^{(m)}, \quad (\mathrm{A}\cdot 34)$$

$$\hat{\boldsymbol{\nu}}_{\boldsymbol{k}} = \frac{N_{\boldsymbol{k}}\boldsymbol{F}_{\boldsymbol{k}} + \xi_{\boldsymbol{k}}\boldsymbol{\nu}_{\boldsymbol{k}}}{N_{\boldsymbol{k}} + \xi_{\boldsymbol{k}}}, \quad (\mathrm{A}\cdot 35)$$

$$\hat{\xi}_{\boldsymbol{k}} = N_{\boldsymbol{k}} + \xi_{\boldsymbol{k}}, \quad (\mathrm{A}\cdot 36)$$

$$\hat{\eta}_{\boldsymbol{k}} = N_{\boldsymbol{k}} + \eta_{\boldsymbol{k}}, \quad (\mathrm{A}\cdot 37)$$

$$\hat{\boldsymbol{R}}_{\boldsymbol{k}} = N_{\boldsymbol{k}}\boldsymbol{S}_{\boldsymbol{k}} + \frac{N_{\boldsymbol{k}}\xi_{\boldsymbol{k}}}{N_{\boldsymbol{k}} + \xi_{\boldsymbol{k}}}(\boldsymbol{F}_{\boldsymbol{k}} - \boldsymbol{\nu}_{\boldsymbol{k}})(\boldsymbol{F}_{\boldsymbol{k}} - \boldsymbol{\nu}_{\boldsymbol{k}})^\top + \boldsymbol{R}_{\boldsymbol{k}}. \quad (\mathrm{A}\cdot 38)$$

C.2 DAEM Algorithm for VB Method

In the VB E-step, the forward-backward algorithm is applied by taking temperature parameters into account. In the VB M-step, the posterior distribution of model parameters $Q(\boldsymbol{\Lambda})$ can be updated by statistics of the training data as follows:

$$\hat{\phi}_i^{(m)} = \beta_1\langle z_{i,1}^{(m)}\rangle_{Q(z^{(m)})} + \beta_3(\phi_i^{(m)} - 1) + 1, \quad (\mathrm{A}\cdot 39)$$

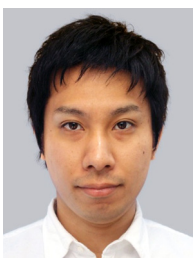
$$\hat{\alpha}_{ij}^{(m)} = \beta_1 N_{ij}^{(m)} + \beta_3(\alpha_{ij}^{(m)} - 1) + 1, \quad (\mathrm{A}\cdot 40)$$

$$\hat{\boldsymbol{\nu}}_{\boldsymbol{k}} = \frac{\beta_2 N_{\boldsymbol{k}}\boldsymbol{F}_{\boldsymbol{k}} + \beta_3\xi_{\boldsymbol{k}}\boldsymbol{\nu}_{\boldsymbol{k}}}{\beta_2 N_{\boldsymbol{k}} + \beta_3\xi_{\boldsymbol{k}}}, \quad (\mathrm{A}\cdot 41)$$

$$\hat{\xi}_{\boldsymbol{k}} = \beta_2 N_{\boldsymbol{k}} + \beta_3\xi_{\boldsymbol{k}}, \quad (\mathrm{A}\cdot 42)$$

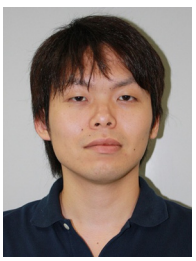
$$\hat{\eta}_{\boldsymbol{k}} = \beta_2 N_{\boldsymbol{k}} + \beta_3(\eta_{\boldsymbol{k}} - D) + D, \quad (\mathrm{A}\cdot 43)$$

$$\hat{\boldsymbol{R}}_{\boldsymbol{k}} = \beta_2 N_{\boldsymbol{k}}\boldsymbol{S}_{\boldsymbol{k}} + \frac{\beta_2\beta_3 N_{\boldsymbol{k}}\xi_{\boldsymbol{k}}}{\beta_2 N_{\boldsymbol{k}} + \beta_3\xi_{\boldsymbol{k}}}(\boldsymbol{F}_{\boldsymbol{k}} - \boldsymbol{\nu}_{\boldsymbol{k}})(\boldsymbol{F}_{\boldsymbol{k}} - \boldsymbol{\nu}_{\boldsymbol{k}})^\top + \beta_3\boldsymbol{R}_{\boldsymbol{k}}. \quad (\mathrm{A}\cdot 44)$$

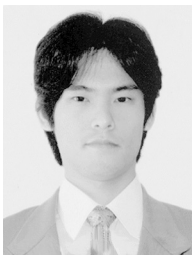


Kei Sawada received the B.E. and M.E. degrees in Computer Science and Scientific and Engineering Simulation from Nagoya Institute of Technology, Nagoya, Japan, in 2011 and 2013. He is currently a Ph.D. candidate at Nagoya Institute of Technology. From June 2014 to November 2014, he was a visiting researcher at the University of Edinburgh. Since April 2015, he has been a Research Fellow of the Japan Society for the Promotion of Science (DC2) at Nagoya Institute of Technology, Nagoya, Japan. His research interests include image recognition, speech recognition and synthesis, and machine learning. He is a student member of the Institute of Electronics, Information and Communication Engineers (IEICE), the Acoustical Society of Japan (ASJ) and International Speech Communication Association (ISCA).

Akira Tamamori received the B.E. degree in Computer Science, the M.E. and Ph.D. degrees in the Department of Scientific and Engineering Simulation from Nagoya Institute of Technology, Nagoya, Japan, in 2008, 2010, 2014, respectively. He is now a Visiting Assistant Professor at the same institute. His research interests include statistical machine learning, image recognition, and speech signal processing. He is a member of the Institute of Electronics, Information and Communication Engineers (IEICE) and the Acoustical Society of Japan (ASJ).



Kei Hashimoto received the B.E., M.E., and Ph.D. degrees in computer science, computer science and engineering, and scientific and engineering simulation from Nagoya Institute of Technology, Nagoya, Japan in 2006, 2008, and 2011, respectively. From October 2008 to January 2009, he was an intern researcher at National Institute of Information and Communications Technology (NICT), Kyoto, Japan. From April 2010 to March 2012, he was a Research Fellow of the Japan Society for the Promotion of Science (JSPS) at Nagoya Institute of Technology, Nagoya, Japan. From May 2010 to September 2010, he was a visiting researcher at University of Edinburgh and Cambridge University. From April 2012, he is now an Assistant Professor at Nagoya Institute of Technology, Nagoya, Japan. His research interests include statistical speech synthesis and speech recognition. He is a member of the Acoustical Society of Japan.



Yoshihiko Nankaku received his B.E. degree in Computer Science, and his M.E. and Ph.D. degrees in the Department of Electrical and Electronic Engineering from Nagoya Institute of Technology, Nagoya, Japan, in 1999, 2001, and 2004 respectively. After a year as a postdoctoral fellow at the Nagoya Institute of Technology, he became an Associate Professor at the same Institute. He was a visiting researcher at the Department of Engineering, University of Cambridge, UK, from May to October 2011. His research interests include statistical machine learning, speech recognition, speech synthesis, image recognition, and multi-modal interface. He is a member of the Institute of Electronics, Information and Communication Engineers (IEICE) and the Acoustical Society of Japan (ASJ).

Keiichi Tokuda received the B.E. degree in electrical and electronic engineering from Nagoya Institute of Technology, Nagoya, Japan, the M.E. and Dr.Eng. degrees in information processing from the Tokyo Institute of Technology, Tokyo, Japan, in 1984, 1986, and 1989, respectively. From 1989 to 1996 he was a Research Associate at the Department of Electronic and Electric Engineering, Tokyo Institute of Technology. From 1996 to 2004 he was a Associate Professor at the Department of Computer Science, Nagoya Institute of Technology as Associate Professor, and now he is a Professor at the same institute. He is also an Honorary Professor at the University of Edinburgh. He was an Invited Researcher at ATR Spoken Language Translation Research Laboratories, Japan from 2000 to 2013 and was a Visiting Researcher at Carnegie Mellon University from 2001 to 2002. He published over 80 journal papers and over 190 conference papers, and received six paper awards and three achievement awards. He was a member of the Speech Technical Committee of the IEEE Signal Processing Society from 2000 to 2003, a member of ISCA Advisory Council and an associate editor of IEEE Transactions on Audio, Speech & Language Processing, and acts as organizer and reviewer for many major speech conferences, workshops and journals. He is a IEEE Fellow and ISCA Fellow. His research interests include speech coding, speech synthesis and recognition, and statistical machine learning.