Spectral Conversion Based on Statistical Models Including Time-Sequence Matching

Yoshihiko Nankaku†, Kenichi Nakamura†, Tomoki Toda‡ and Keiichi Tokuda†

† Nagoya Institute of Technology, Gokiso-cho, Showa-ku, Nagoya, Aichi, 466-8555 Japan
‡ Nara Institute of Science and Technology, 8916-5 Takayama, Ikoma, Nara, 630-0101 Japan

{nankaku, k-n, tokuda}@sp.nitech.ac.jp, tomoki@is.naist.jp

Abstract

This paper proposes a spectral conversion technique based on a new statistical model which includes time-sequence matching. In conventional GMM-based approaches, the Dynamic Programming (DP) matching between source and target feature sequences is performed prior to the training of GMMs. Although a similarity measure of two frames, e.g., the Euclid distance is typically adopted, this might be inappropriate for converting the spectral features. The likelihood function of the proposed model can directly deal with two different length sequences, in which a frame alignment of source and target feature sequences is represented by discrete hidden variables. In the proposed algorithm, the maximum likelihood criterion is consistently applied to the training of model parameters, sequence matching and spectral conversion. In the subjective preference test, the proposed method is superior than the conventional GMM-based method.

1. Introduction

In recent years, voice conversion especially the statistical model based approaches are widely investigated. This technique can modify speech characteristics using conversion rules statistically extracted from a small amount of training data. As a typical spectral conversion method, a mapping algorithm based on the Gaussian Mixture Model (GMM) has been proposed [1]. In this method, the mapping between spectral features of the source and target is determined based on GMMs. In each mixture component, the conditional mean vector of target features given source features is calculated as a simple linear transformation using the covariance matrix of the concatenated feature vector. The converted vector is defined as the weighted sum of the conditional mean vectors, and the conditional occupancy probabilities of mixture components are used as weights. A more accurate formulation of spectral conversion based on ML (Maximum Likelihood) criterion has been presented [2]. The ML-based conversion is a sophisticated technique because all processes in the algorithm is derived based on the single objective function.

In these GMM-based method, GMMs are trained using joint feature vectors which are references of mapping rules, and the DP matching between feature sequences of source and target are conducted prior to the training of GMMs. Typically the similarity measure of two frames is adopted independently of the training of GMMs, e.g., Euclid distance. However, this might be inappropriate for converting the spectral features. To avoid this problem, we propose a voice conversion technique based on a new statistical model including temporal matching between source and target feature sequences. The likelihood function can directly deal with two different length sequences, in which a frame alignment between two sequences is represented by discrete hidden variables. In the proposed voice conversion technique, the ML criterion is consistently applied to the training of model parameters, sequence matching and spectral conversion.

The parameters of the proposed model can be estimated via the expectation maximization (EM) algorithm for approximating the Maximum Likelihood (ML) estimate. However, a complex model structure leads to an exponential increase in the amount of computation for its training algorithm and the exact expectation step is computationally intractable. To derive a feasible algorithm, we applied the variational EM algorithm [4], [5]. Variational methods approximate the posterior distribution over the hidden variables by a tractable distribution. A structure approximation is presented in which the hidden variables of GMMs and the temporal matching are decoupled. However, the convergence point of the EM algorithm depends on the initial model parameters. Moreover, in the variational EM algorithm for the proposed model, the decoupled posterior distributions are updated individually based on the other distributions which are unreliable at an early stage of training. To overcome these problems, we applied the deterministic annealing EM (DAEM) algorithm [6] to the variational algorithm for the proposed model.

The paper is organized as follows. Section 2 explains the conventional voice conversion technique based on GMMs. Section 3 describes a new statistical model including temporal matching, and section 4 explains its training algorithm. Voice conversion based on the proposed model is presented in section 5 and experimental results are reported in Section 6. Finally, conclusions and future works are given in Section 7.

2. GMM-based Spectral Conversion

To convert spectral feature sequences of a source speaker to that of a target speaker, the joint probability density of two features are modeled by GMM [2]. Let a vector \( \mathbf{O}_t = \begin{bmatrix} \mathbf{O}_t^{(1)} \top, \mathbf{O}_t^{(2)} \top \end{bmatrix} \) be a joint feature vector of the source one \( \mathbf{O}_t^{(1)} \) and the target one \( \mathbf{O}_t^{(2)} \) at time \( t \). An alignment between two feature sequences is obtained by the Dynamic Programming (DP) matching. In the GMM-based voice conversion, the vector sequence \( \mathbf{O} = \begin{bmatrix} \mathbf{O}_1 \top, \ldots, \mathbf{O}_T \top, \ldots, \mathbf{O}_T \top \end{bmatrix} \) is modeled by GMM to learn a relation between source and target fea-
tures. The output probability of $O$ given GMM $\lambda$ can be written as follows:

$$P(O | \lambda) = \prod_{t=1}^{T} \sum_{i=1}^{M} w_i N(O_t; \mu_i, \Sigma_i)$$

(1)

where

$$\mu_i = \begin{bmatrix} \mu_i^{(1)} \\ \mu_i^{(2)} \end{bmatrix}, \Sigma_i = \begin{bmatrix} \Sigma_i^{(1,1)} & \Sigma_i^{(1,2)} \\ \Sigma_i^{(2,1)} & \Sigma_i^{(2,2)} \end{bmatrix}.$$  (2)

and $M$ means the number of mixtures, $w_i = P(i | \lambda)$ is the mixture weight of the $i$-th component, $\mu_i$ and $\Sigma_i$ are the mean vector and covariance matrix, respectively. These model parameters are estimated via the Expectation Maximization (EM) algorithm.

2.1. Maximum likelihood spectral conversion

In the maximum likelihood spectral conversion, the optimal converted feature sequence $O^{(2)} = \begin{bmatrix} O_1^{(2),T}, \ldots, O_t^{(2),T}, \ldots, O_T^{(2),T} \end{bmatrix}^T$ given a source feature sequence $O^{(1)} = \begin{bmatrix} O_1^{(1),T}, \ldots, O_t^{(1),T}, \ldots, O_T^{(1),T} \end{bmatrix}^T$ is obtained by maximizing the following conditional distribution:

$$P(O^{(2)} | O^{(1)}, \lambda) = \sum_m \left[ P(m | O^{(1)}, \lambda) \prod_{t=1}^{T} P(O_t^{(2)} | O_t^{(1)}, m_t, \lambda) \right]$$

(3)

where $m = \{m_1, m_2, \ldots, m_T\}$ is a mixture number sequence. The conditional distribution can also be written as GMM, and its output probability distribution is presented as follows:

$$P(O_t^{(2)} | O_t^{(1)}, m_t = i, \lambda) = \mathcal{N}(O_t^{(2)}; E_i(t), D_i)$$

(4)

where

$$E_i(t) = \mu_i^{(2)} + \Sigma_i^{(2,1)} \Sigma_i^{(1,1)^{-1}} (O_i^{(1)} - \mu_i^{(1)})$$

(5)

and

$$D_i = \Sigma_i^{(2,2)} - \Sigma_i^{(2,1)} \Sigma_i^{(1,1)^{-1}} \Sigma_i^{(1,2)}.$$  (6)

Since the equation (3) includes latent variables, the optimal sequence of $O^{(2)}$ is estimated via the EM algorithm. The EM algorithm is an iterative method for approximating the maximum likelihood estimation. It maximizes the expectation of the complete data log-likelihood so called Q-function (auxiliary function):

$$Q(O^{(2)}, \hat{O}^{(2)}) = \sum_{\{m|O\}} \left[ \sum_m \left[ P(O^{(2)} | m, O^{(1)}, \lambda) \right] \times \ln P(O^{(2)} | m, O^{(1)}, \lambda) \right]$$

(7)

Taking the derivative of the Q-function, the spectral sequence $\hat{O}^{(2)}$ which maximizes the Q-function is given by

$$\hat{O}^{(2)} = \left( \overline{D}^{-1} E \right)^{-1} \overline{D}^{-1} E$$

(8)

where

$$\overline{D}^{-T} = \text{diag} [\overline{D}_{11}^{-T}, \overline{D}_{21}^{-T}, \ldots, \overline{D}_{T1}^{-T}]$$

(9)

$$\overline{D}_t^{-1} = \sum_{i=1}^{M} \gamma_i(t) D_i^{-1}$$

(10)

$$\overline{D}^{-1} E = \left[ \overline{D}^{-1} E_1, \overline{D}^{-1} E_2, \ldots, \overline{D}^{-1} E_T \right]^T.$$  (11)

$$\overline{D}^{-1} E_t = \sum_{i=1}^{M} \gamma_i(t) D_i^{-1} E_i(t)$$

(12)

$$\gamma_i(t) = P(m_t = i | O^{(1)}, O^{(2)}, \lambda)$$

(13)

3. Statistical Model Including Time-Sequence Matching

3.1. Definition of Model Structure

In the conventional method, the DP matching is conducted based on a similarity measure between two frames. However, this matching might not be optimal for spectral conversion. To overcome this problem, we define the likelihood function $P(O^{(1)}, O^{(2)} | \lambda)$ including the structure of sequence matching. The simultaneous optimization is performed for DP matching and training of model parameters based on the ML criterion. The advantage of the the proposed model can directly deal with two different length sequences $O^{(1)} = \begin{bmatrix} O_1^{(1),T}, \ldots, O_t^{(1),T}, \ldots, O_T^{(1),T} \end{bmatrix}^T$ and $O^{(2)} = \begin{bmatrix} O_1^{(2),T}, \ldots, O_t^{(2),T}, \ldots, O_T^{(2),T} \end{bmatrix}^T$. The likelihood function of observation sequences $O = (O^{(1)}, O^{(2)})$ is written as follows:

$$P(O | \lambda) = \sum_{m,a} \left[ P(m | \lambda) P(O^{(1)} | m, \lambda) \times P(a | \lambda) P(O^{(2)} | O^{(1)}, m, a, \lambda) \right]$$

(14)

where $m = \{m_1, \ldots, m_{t^{(1)}}, \ldots, m_{T^{(1)}}\}$ is a mixture number sequence and its element $m_{t^{(1)}}$ means the mixture number of the observation $O^{(1)}$ at time $t^{(1)}$. The variable $a = \{a_1, \ldots, a_{t^{(2)}}, a_{t^{(2)}}, \ldots, a_{T^{(2)}}\}$ is a transition cost function of the temporal matching and $a_{t^{(2)}} \in \{1, \ldots, T^{(1)}\}$ indicates the frame number of source sequence $O^{(1)}$ which corresponds to the frame number of target sequence $O^{(2)}$. Each element of the observation data likelihood is defined as follows:

$$P(m | \lambda) = \prod_{t^{(1)}} P(m_{t^{(1)}} | \lambda)$$

(15)

$$P(O^{(1)} | m, \lambda) = \prod_{t^{(1)}} \mathcal{N} \left( O_{t^{(1)}}^{(1)}; \mu_{m_{t^{(1)}}}^{(1)}, \Sigma_{m_{t^{(1)}}}^{(1)} \right)$$

(16)

$$P(a | \lambda) = \prod_{t^{(2)}} P(a_{t^{(2)}} | a_{t^{(2)}-1}, \lambda)$$

(17)

$$P(O^{(2)} | O^{(1)}, m, a, \lambda) = \prod_{t^{(2)}} \mathcal{N} \left( O_{t^{(2)}}^{(2)}; W_{m_{t^{(2)}}, a_{t^{(2)}}}, \Sigma_{m_{t^{(2)}}}^{(2)} \right)$$

(18)
where
\[
W_i = \begin{bmatrix} \mu_i & W_i \end{bmatrix}
\] (19)
\[
\tilde{O}_{(1)}^i = \begin{bmatrix} 1 & O_{(1)}^{iT} \end{bmatrix}^T
\] (20)

The model parameters of the proposed model are summarized as follows:

1. \( w = \{w_i | 1 \leq i \leq M \} \) : the mixture weights of the GMM which generates the source feature sequence \( O^{(1)} \), where \( w_i = P(m_{i(1)} = i | \Lambda) \) is the probability of \( i \)-th mixture.

2. \( B^{(1)} = \{\beta_i^{(1)} | 1 \leq i \leq M \} \) : the output probability distributions of source feature \( O^{(1)} \), where \( \beta_i^{(1)} = P(O_{(1)}^{i(1)} | m_{i(1)} = i) \) is the probability of source feature vector \( O_{(1)}^{i(1)} \) at \( i \)-th mixture and which is assumed to be a Gaussian distribution: \( N(O_{(1)}^{i(1)}; \mu^{(1)}_i, \Sigma^{(1)}_i) \) where \( \mu^{(1)}_i \) and \( \Sigma^{(1)}_i \) are the mean vector and covariance matrix, respectively.

3. \( c = \{c_n | 1 \leq n \leq N \} \) : the transition probabilities of the sequence matching where \( c_n \) indicates the probability \( P(a_{n(2)} = a_{n(2)−1} + n | a_{n(2)−1}) \). This parameter corresponds to the cost function in the DP matching.

4. \( B^{(2)} = \{\beta_i^{(2)} | 1 \leq i \leq M \} \) : the output distributions of the target features \( O^{(2)} \), where \( \beta_i^{(2)} = P(O_{(2)}^{i(2)} | O_{(1)}^{i(1)}, m_{i(1)} = i, a_{n(2)} = t^{(1)}) \) is the probability of target feature vector \( O_{(2)}^{i(2)} \) given the corresponding source feature vector \( O_{(1)}^{i(1)} \) at \( i \)-th mixture. This conditional distribution is assumed to be a Gaussian distribution: \( N(O_{(2)}^{i(2)}; W, O_{(1)}^{i(1)} + \mu^{(2)}_i, \Sigma^{(2)}_i) \) where \( \mu^{(2)}_i \) and \( \Sigma^{(2)}_i \) are the mean vector and covariance matrix, respectively.

Using shorthand notation, the proposed model is defined as \( \Lambda = \{w, c, B^{(1)}, B^{(2)}\} \). Figure 1 shows the generative process of observations \( O^{(1)}, O^{(2)} \) by the proposed model. First, a mixture number sequence \( m \) is determined according to the weight \( P(m | \Lambda) \) and a source feature sequence \( O^{(1)} \) is generated from Gaussian distribution \( P(O^{(1)} | m, \Lambda) \). Second, the frame matching between \( O^{(1)} \) and \( O^{(2)} \) is determined according to \( P(a | \Lambda) \). Finally, the target feature sequence \( O^{(2)} \) is generated according to the conditional Gaussian distribution \( P(O^{(2)} | O^{(1)}, m, a, \Lambda) \) given the source feature sequence.

4. Training Algorithm

The parameters of the proposed model can be estimated via the expectation maximization (EM) algorithm which is an iterative procedure for approximating the Maximum Likelihood (ML) estimate. This procedure maximizes the expectation of the complete data log-likelihood so called \( Q \)-function:

\[
Q(\Lambda, \Lambda') = \sum_{m, a} P(m, a | O, \Lambda) \ln P(O, m, a | \Lambda')
\] (21)

The likelihood of the training data is guaranteed to increase by increasing the value of the \( Q \)-function:

\[
Q(\Lambda, \Lambda') \geq Q(\Lambda, \Lambda) \Rightarrow P(O | \Lambda') \geq P(O | \Lambda)
\] (22)

The EM algorithm starts with some initial model parameters and iterates between the following two steps:

\[(E \text{ step}) : \quad \text{compute } Q(\Lambda^{(k)}, \Lambda) \quad \text{(M step)} : \quad \Lambda^{(k+1)} = \arg \max_{\Lambda} Q(\Lambda^{(k)}, \Lambda)\]

where \( k \) denotes the iteration number. The E-step computes the posterior probabilities over the hidden variables while keeping model parameters \( \Lambda \) fixed to current values. The M-step uses these probabilities to calculate the expected log-likelihood of the training data as a function of the parameters and maximizes the \( Q \)-function with respect to model parameters \( \Lambda \). In this procedure, each step increases the value of the \( Q \)-function; hence the likelihood of the training data is also guaranteed to increase or remain unchanged on each iteration.

By maximizing the \( Q \)-function, the re-estimation formulae in the M-step are derived as follows:

\[
w_i = \frac{1}{N(1)} \sum_{i(1)} \gamma^{(1)}_i(i) \] (23)
\[
\mu^{(1)}_i = \frac{1}{N(1)} \sum_{i(1)} \gamma^{(1)}_i(i) O^{(1)}_{i(1)} \] (24)
\[
\Sigma^{(1)}_i = \frac{1}{N(1)} \sum_{i(1)} \gamma^{(1)}_i(i) (O^{(1)}_{i(1)} - \mu^{(1)}_i)(O^{(1)}_{i(1)} - \mu^{(1)}_i)^T \] (25)
\[
c_n = \frac{1}{N(2)} \sum_{i(2)} \sum_{i(1)} \gamma^{(2)}_{n(2)}(t^{(1)}, n) \] (26)
\[
\bar{W}_i = \left( \frac{1}{N(2)} \sum_{i(2)} \sum_{i(1)} \gamma^{(2)}_{n(2)}(t^{(1)}, i) O^{(2)}_{i(2)} O^{(1)}_{i(1)}^T \right) \times \left( \sum_{i(2)} \sum_{i(1)} \gamma^{(2)}_{n(2)}(t^{(1)}, i) O^{(1)}_{i(1)} O^{(2)}_{i(2)}^T \right)^{-1} \] (27)
\[
\Sigma^{(2)}_i = \frac{1}{N(2)} \sum_{i(2)} \sum_{i(1)} \gamma^{(2)}_{n(2)}(t^{(1)}, i) \times (O^{(2)}_{i(2)} - \bar{W} \bar{O}^{(1)}_{i(1)}) (O^{(2)}_{i(2)} - \bar{W} \bar{O}^{(1)}_{i(1)})^T \] (28)

where \( \gamma \) and \( \xi \) denote the expectations with respect to the posterior distribution over the hidden variables. These expectations
are computed in the E-step by the following equations.

\[ \gamma_{(1)}(i) = \frac{P(s_{(1)} = i \mid O, \Lambda)}{\sum_{m, a} P(m, a \mid O, \Lambda) \delta(m_{(1)}, i) \ln Q(m_{(1)} | \Lambda)} \]

\[ \gamma_{(2)}(t^{(1)}, i) = \frac{P(a_{(2)} = t^{(1)} \mid O, \Lambda)}{\sum_{m, a} P(m, a \mid O, \Lambda) \times \delta(m_{(1)}, i) \delta(a_{(2)}, t^{(1)}) \ln Q(a_{(2)} | \Lambda)} \]

\[ \gamma_{(1)}(i) \]

\[ \gamma_{(2)}(t^{(1)}, i) \]

\[ \gamma_{(1)}(i) \]

\[ \gamma_{(2)}(t^{(1)}, i) \]

\[ \gamma_{(1)}(i) \]

\[ \gamma_{(2)}(t^{(1)}, i) \]

\[ \gamma_{(1)}(i) \]

\[ \gamma_{(2)}(t^{(1)}, i) \]

and \( N^{(1)} \) and \( N^{(2)} \) mean the total number of frames of source and target feature sequences, respectively, and \( N^{(1)} \) and \( N^{(2)} \) are the occupancy counts of \( i \)-th mixture which can be written as follows:

\[ N^{(1)} = \sum_{i(1)} \gamma_{(1)}(i), \quad N^{(2)} = \sum_{t(1)} \sum_{i(1)} \gamma_{(2)}(t^{(1)}, i) \]

(32)

where \( \delta(\cdot) \) is the Kronecker delta function: \( \delta(u, v) = 1 \) if \( u = v \), \( \delta(u, v) = 0 \) otherwise. If we compute expectations in the exact E-step directly according to (29)–(31), we need to consider summations over all the combinations of \( m \) and \( a \). Therefore, the complexity of the E-step becomes \( O(M^{(1)} F(1)^{2(1)}) \) and it is infeasible due to the number of hidden variables.

4.1. Variational approximation

Variational methods have been used for approximate maximum likelihood estimation in probabilistic graphical models with hidden variables. We present a structure approximation algorithm for arbitrary structure of graphical models. Here we consider a constrained variational EM algorithm dependent on a constraint to the posterior distribution \( Q(m, a) \) and it should be determined for each structure of graphical models. Here we consider a constrained family of variational distributions by assuming that \( Q(m, a) \) factorizes over \( m \) and \( a \), so that

\[ Q(m, a) = Q(m)Q(a) \]

(36)

where \( \sum_m Q(m) = 1, \sum_a Q(a) = 1 \). To make the bound as tight as possible, we use elementary calculus of variations to take functional derivatives of the lower bound with respect to \( Q(m) \) and \( Q(a) \). In this case, the Euler-Lagrange equation can be solved simply by taking partial derivatives with respect to one of the distributions

\[ \frac{\partial \mathcal{F}}{\partial Q(m = m')} = \sum_a Q(a) \ln P(O \mid m', a) - \ln Q(m') + 1 \]

\[ \frac{\partial \mathcal{F}}{\partial Q(m = m')} = \sum_a Q(a) \ln P(O^{(2)} \mid O^{(1)}, m', a) + \ln P(m' \mid a) + \ln P(O^{(1)} \mid m', a) - \ln Q(m') + \text{const} \]

(37)

The maximum of \( \mathcal{F} \) occurs at a critical point subject to the constraint that \( \sum_m Q(m) = 1 \), and can be found using a Lagrange multiplier \( \lambda_m \). By setting for each of mixture number sequence \( m \)

\[ \frac{\partial \mathcal{F}}{\partial Q(m)} + \lambda_m = 0 \]

(38)
the optimal approximation of the posterior distribution is derived as

\[ Q(m) \propto P(m | \varLambda) P(O^{(1)} | m, \varLambda) \times \exp \left[ \ln P(O^{(2)} | O^{(1)}, m, a, \varLambda) \right]_{Q(m)} \]  

(39)

Similarly to the distribution \( Q(m) \), the optimal distribution of sequence matching can be obtained as

\[ Q(a) \propto P(a | \varLambda) \times \exp \left[ \ln P(O^{(2)} | O^{(1)}, m, a, \varLambda) \right]_{Q(m)} \]  

(40)

By inspection, equation (39) has the same structure as the posterior distribution of standard GMMs, therefore it can be easily calculated. Moreover, equation (40) is composed of a first-order dynamic programming (forward-backward algorithm in the training of hidden Markov models). Using these approximate distributions, a new set of expectations can be compute as follows:

\[ \gamma^{(1)}_{(1)}(i) = \sum_{m} Q(m) \delta(m_{(1)}, i) \]  

(41)

\[ \gamma^{(2)}_{(2)}(t^{(1)}) = \sum_{a} Q(a) \delta(a_{(2)}, t^{(1)}) \]  

(42)

\[ \gamma^{(2)}_{(2)}(t^{(1)}, i) = \sum_{a} Q(a) \delta(a_{(2)}, t^{(1)}) \]  

(43)

\[ \xi^{(2)}_{(2)}(t^{(1)}, n) = \sum_{a} Q(a) \delta(a_{(2)}-1, t^{(1)}) \times \delta(a_{(2)}, t^{(1)} + n) \]  

(44)

4.2. Variational DAEM algorithm

The EM algorithm has the problem that the solution converges to a local optimum and the convergence point depends on the initial model parameters. In the variational EM algorithm, the decoupled posterior distributions are updated individually based not only on the initial model parameters but also on the other distributions, both of which are unreliable at an early stage of training. To avoid this problem, we apply the DAEM algorithm to the algorithm derived in the previous section.

In the DAEM algorithm, the problem of maximizing the log-likelihood is reformulated as minimizing the thermodynamic free energy defined as

\[ L_{\beta} = -\frac{1}{\beta} \ln \sum_{m, a} P^{\beta}(O, m, a | \varLambda) \]  

(45)

where \( 1/\beta \) called the “temperature” and this cost function can be rewritten by using Jensen’s inequality:

\[ -L_{\beta} = \frac{1}{\beta} \ln \sum_{m, a} Q_{\beta}(m, a) \frac{P^{\beta}(O, m, a | \varLambda)}{Q_{\beta}(m, a)} \geq \sum_{m, a} Q_{\beta}(m, a) \ln P(O, m, a | \varLambda) - \frac{1}{\beta} \sum_{m, a} Q_{\beta}(m, a) \ln Q_{\beta}(m, a) \]  

(46)

\[ = F_{\beta}(Q_{\beta}, \varLambda) \]  

(47)

where \(-F_{\beta}(Q_{\beta}, \varLambda)\) is the same form as the free energy in statistical physics, and maximizing \( F_{\beta}(Q_{\beta}, \varLambda) \) with a fixed temperature can be interpreted as the approach to thermodynamic equilibrium. In the algorithm, the temperature is gradually decreased and the function is deterministically optimized at each temperature. The procedure of the DAEM algorithm can be summarized as follows:

1. Give an initial model and set \( \beta = \beta_{\min} \)
2. Iterate EM-steps with \( \beta \) fixed until \( F_{\beta} \) converged:

   \[ \text{(E step)} : \quad Q_{\beta+k+1}(m) = \arg \max_{Q_{\beta+k+1}} F_{\beta+k+1}(Q_{\beta+k+1}, \varLambda) \]  

   \[ \text{(M step)} : \quad \varLambda_{k+1} = \arg \max_{\varLambda} F_{\beta+k+1}(Q_{\beta+k+1}, \varLambda) \]  

3. Increase \( \beta \).
4. If \( \beta > 1 \), stop the procedure. Otherwise go to step 2.

where \( 1/\beta_{\min} \) is an initial temperature and should be chosen as a high enough value that the EM-steps can achieve a single global maximum of \( F_{\beta} \). At the initial temperature, the entropy of \( Q_{\beta} \) is intended to be maximized rather than the \( Q \) function (the first term of equation (46)); therefore \( Q_{\beta} \) takes a form nearly uniform distribution. While the temperature is decreasing, the form of \( Q_{\beta} \) changes from uniform to the original posterior and at the final temperature \( 1/\beta = 1 \), the negative free energy \( F_{\beta} \) becomes equal to the lower bound \( F \), accordingly the DAEM algorithm agrees with the original EM algorithm.

Similarly to the variational EM algorithm, the optimal distribution which maximizes \( F_{\beta} \) is given by

\[ Q_{\beta}(m) \propto P^{\beta}(m | \varLambda) P^{\beta}(O^{(1)} | m, \varLambda) \times \exp \left[ \ln P(O^{(2)} | O^{(1)}, m, a, \varLambda) \right]_{Q(m)} \]  

(48)

\[ Q_{\beta}(a) \propto P^{\beta}(a | \varLambda) \times \exp \left[ \ln P(O^{(2)} | O^{(1)}, m, a, \varLambda) \right]_{Q(m)} \]  

(49)

5. ML-Based Spectral Conversion

The converted feature sequence \( O^{(2)} \) can be obtained by maximizing the lower bound of the likelihood. Taking the derivative of \( F \) with respect to \( O^{(2)} \), the optimal sequence is given as the following equation.

\[ \hat{O}^{(2)}_{i(2)} = \left( \sum_{i(1)} \sum_{\gamma^{(1)}_{(1)}(i)} \gamma^{(2)}_{(2)}(t^{(1)}) \Sigma_{i(1)}^{-1} \right)^{-1} \times \left( \sum_{i(1)} \gamma^{(1)}_{(1)}(i) \gamma^{(2)}_{(2)}(t^{(1)}) \Sigma_{i(1)}^{-1} \left( W \tilde{O}^{(1)}_{i(2)} \right) \right) \]  

(50)

Although the proposed method can represent different length sequences of source and target features, the transition probability \( P(\alpha | \varLambda) \) assumed in this paper is insufficient to generate the duration of the converted feature sequence. Therefore, one to one frame matching is used in the conversion process (i.e. \( a_{(2)} = t^{(2)} \)). Under this assumption, if \( t^{(1)} = t^{(2)} \), \( \gamma^{(2)}_{(2)}(t^{(1)}) = 1 \), otherwise \( \gamma^{(2)}_{(2)}(t^{(1)}) = 0 \), therefore equation (50) can be rewritten as

\[ \hat{O}^{(2)}_{i(2)} = \left( \sum_{i} \gamma^{(1)}_{(1)}(i) \Sigma_{i}^{-1} \right)^{-1} \times \left( \sum_{i} \gamma^{(1)}_{(1)}(i) \Sigma_{i}^{-1} \left( W \tilde{O}^{(1)}_{i(2)} \right) \right) \]  

(51)
Given the temporal matching $a$, the optimal converted feature sequence still depends on the posterior distribution of mixture number sequence $Q(m)$. Therefore, an iterative update procedure is required. The conversion procedure is summarized as follows:

1. Compute the expectation $\gamma^{(1)}_i(i)$ for each frame of the source feature sequence $O^{(1)}$ (omitting the last term of equation (39), that is, $Q(m) \propto P(m | \lambda) P(O^{(1)} | m, \lambda)$ and equation (41)).
2. The converted feature sequence $\hat{O}^{(2)}$ is obtained by using $\gamma^{(1)}_i(i)$ (equation (51)).
3. Update the expectation $\gamma^{(1)}_i(i)$ by using both the source feature sequence $O^{(1)}$ and the converted feature sequence $\hat{O}^{(2)}$ (equation (39), (41)).
4. If $\mathcal{F}$ is converged, stop the procedure. Otherwise go to 2.

6. Experiments

Voice conversion experiments on the ATR Japanese speech database were conducted. Two male speakers are selected as a source and a target speaker (source:mtk target:mht). Twenty sentences uttered by the both speakers were used for training and 200 sentences were used for evaluation. The speech data were down-sampled from 20KHz to 16KHz, windowed at a 5-ms frame rate using a 25-ms Blackman window, and parameterized into 24 mel-cepstral coefficients excepting the zero-th coefficients and their first order derivative were used as the dynamic features.

Although voice similarity to target speakers is primarily required in voice conversion, we conducted subjective preference tests in speech quality because the proposed method is expected to improve speech quality. In preliminary experiments, it is confirmed that the proposed method obtained the almost same or higher performance in voice similarity than the conventional method. The number of mixtures was set to four which achieved better performance than the conventional GMM-based method "CONV1," "DPGMM" was still better than "CONV2." This is because the DP matching and training GMMs are simultaneously optimized based on the integrated objective measure. It could also be an advantage that "DPGMM" utilizes all frame combination of source and target features, not dependent on the posterior distribution of mixture number sequence. Thus, the proposed method obtained the almost same or higher performance in voice similarity than the conventional GMM-based approach.

8. References