A segmental HMM based on a modified emission probability

Stefan Windmann, Reinhold Haeb-Umbach, Volker Leutnant

Dept. of Communications Engineering, University of Paderborn, 33098 Paderborn, Germany
E-Mail: {windmann,haeb,leutnant}@nt.uni-paderborn.de
Web: www-nt.uni-paderborn.de

Abstract

In this paper, a novel segmental Hidden Markov Model (HMM) is proposed. The model is based on a modified emission density where additional statistical dependencies between subsequent frames of the speech signal are considered. In the following we derive an effective search strategy for the modified statistical model. Further an approach to parameter reduction is introduced. Experiments were carried out on the AURORA2 database where consistent improvements were obtained with the segmental HMM.

1 Introduction

The acoustic modelling on the basis of the Hidden Markov Model (HMM) is a wide-spread technique for automatic speech recognition (ASR). However, a frequently cited weak point of the HMM consists in the so-called conditional independence assumption, i.e. in not directly modelling the statistical dependencies between speech features which are extracted from subsequent frames of the speech signal. Segmental HMMs provide a possibility to overcome this drawback by modelling the speech dynamics within speech segments with trajectory models. According to [8], segmental HMMs can be divided with respect to the kind of trajectory model in polynomial segment models, buried Markov models [1], stochastic segment models, trajectory HMMs and switching linear dynamic models (SLDMs). The prevalent techniques in literature are stochastic segment models, where the feature vector trajectory within the segments is described with linear state models (e.g. [2, 5, 4]) and trajectory HMMs (e.g. [6, 9]), where the trajectory is explicitely modeled as a function of the duration and the observations within the current speech segment. These approaches usually require a large number of parameters and are unreliable with respect to the Maximum-Likelihood (ML) training of the parameters. However, Seide et al. [7] have shown that the quantisation of the feature space might be a doorway to efficient algorithms. In their approach a Hidden Trajectory HMM (HTHMM) is applied for acoustic modelling where the speech features are described with a hidden, quantized trajectory beside the emission distribution of the HMM.

The proposed paper is also based on a quantisation of the feature space. However, the prevelant modelling of the emission distribution of the HMM as a Gaussian mixture density, i.e. its quantisation in the standard HMM training, is exploited. In the following, we introduce a modified model, termed Continuous Mixture HMM (CMHMM), where the statistical dependencies are modelled between the mixture components in addition to the dependency on the level of the HMM states (see Section 2).

Based on the Viterbi search for the standard HMM which is described in Section 3, an efficient search algorithm for the CMHMM is introduced in Section 4. Further a method for parameter reduction is adopted in Section 5. Finally we present experimental results in Section 6 and finish with some conclusions in Section 7.

2 Statistical model

In the standard HMM there are assumed transition probabilities between the hidden states q_t and emission distributions of the (cepstral) speech features $\mathbf{x}_t$ at time t (see Fig. 1 a)). In many speech recognition systems the emission density $p(\mathbf{x}_t|q_t)$ is modelled as Gaussian mixture density (GMM)

$$p(\mathbf{x}_t|q_t) = \sum_{m_t} p(\mathbf{x}_t|m_t)P(m_t|q_t), \tag{1}$$

where the probabilities $P(m_t|q_t)$ of the mixture components m_t depend on the HMM states q_t, while the constituent Gaussians $p(\mathbf{x}_t|m_t)$ depend on m_t. In eq. (1) the assumption is made that the mixtures m_t are non-tied mixtures, i.e. that they can be subdivided in disjoint subsets $\mathscr{M}(q_t), q_t = 1,\ldots N_q$, N_q being the number of HMM states:

$$\mathscr{M}(q_t) = \{m_t|P(q_t|m_t) \neq 0\}. \tag{2}$$

Thus, the value of the mixture index uniquely specifies the value of the state variable, i.e.

$$P(q_t|m_t) = \delta(q_t - \hat{q}_t(m_t)), \tag{3}$$

where $\hat{q}_t(m_t)$ denotes the state m_t is assigned to.

The statistical model considered in the paper at hand, which is refered to as CMHMM in the following, is depicted in Fig. 1 b). In the CMHMM a statistical dependency between the mixture components m_{t-1} and m_t is modeled in addition to the dependency between the HMM states q_{t-1} and q_t (Fig. 1 a)). Thus, the emission density of

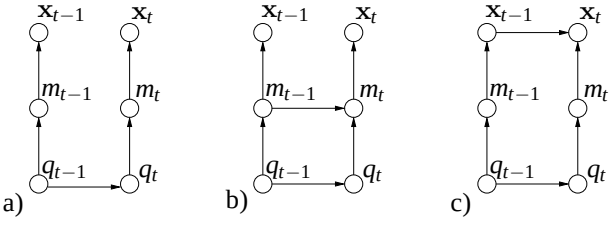

Figure 1: *Model of statistical dependencies: a) HMM b) CMHMM c) Direct modelling of inter-frame correlations*

the CMHMM is assumed to be a HMM with the mixture components m_t as hidden states and the transition probabilities $P(m_t|m_{t-1})$ between the mixture components m_t and m_{t-1} in subsequent frames. It is further assumed that the tuple (q_t, m_t) is as probable as m_t once the condition in (3) holds. This is expressed with the constraint

$$P(q_t, m_t|\mathbf{X}) = \begin{cases} P(m_t|\mathbf{X}) & : q_t = \hat{q}_t(m_t) \\ 0 & : \text{else} \end{cases}. \tag{4}$$

We further neglect the influence of q_t on m_t

$$p(m_t|\mathbf{X},q_t) = \begin{cases} p(m_t|\mathbf{X}) & : q_t = \hat{q}_t(m_t) \\ 0 & : \text{else} \end{cases} . \quad (5)$$

once m_t is conditioned on the observations $\mathbf{X}$.

In contrast to the direct modeling of interframe correlations (see Fig. 1 c)), the modelling of statistical dependencies on the mixture component level allows an (extended) Viterbi search (see Section 4).

3 Viterbi search in the HMM

In order to establish a framework for the search in the CMHMM, first the standard Viterbi search in the HMM is described. The objective is to determine a word sequence $\hat{w}_1^N = \hat{w}_1 \ldots \hat{w}_N$ of length N which maximizes the posterior probability for the given feature vectors $\mathbf{x}_1^T = \mathbf{x}_1 \ldots \mathbf{x}_T$:

$$\begin{aligned} \hat{w}_1^N &= \arg\max_{w_1^N} \{ \max_{q_1^T} \{ p_\alpha(\mathbf{x}_1^T, q_1^T) P(w_1^N) \} \\ &= \arg\max_{w_1^N} \{ \max_{q_1^T} \{ \prod_{t=1}^{T} p_\alpha(\mathbf{x}_t|q_t) P_\alpha(q_t|q_{t-1}) \} \\ &\qquad \prod_{n=1}^{N} P(w_n|w_{n-m+1}^{n-1}) \}, \end{aligned} \quad (6)$$

where $w_1^N = w_1 \ldots w_N$ denotes a possible word sequence described with a language model of order m. In (6), the probability density $p(\mathbf{x}_1^T, q_1^T)$ obtained from the HMM training is scaled with an acoustic scale factor $S_\alpha < 1$, i.e.

$$p_\alpha(\mathbf{x}_1^T, q_1^T) \propto p(\mathbf{x}_1^T, q_1^T)^{S_\alpha} \quad (7)$$

In literature, (6) is often written with a language model scale factor $S_\beta = 1/S_\alpha$. This is possible because the maximation in eq. (6) is invariant to the potentation with a constant factor. However, once sums must be evaluated for the estimation of $p_\alpha(\mathbf{x}_t|q_t)$, the formulation with the acoustic scale factor is more adequate as stated in [10].

In the following, the Viterbi search is described for a bigram language model and a tree structure of the lexicon. In order to account for the possible predecessors v of the current word w for each predecessor v a copy of the lexicon is constructed. A forward variable

$$\alpha_t(v,q_t) = \max_{q_1^{t-1}} P(\mathbf{x}_1^t, q_1^t|v) \quad (8)$$

is assigned to the states q_t of the copies v, which contains the score for the best path for the observed feature sequence $\mathbf{x}_1^t$, which ends in state q_t of the tree copy v at time instance t. Besides, the word boundaries $B_t(v,q_t)$ of the corresponding pathes are tracked.

For each time instance t the forward variable $\alpha_{t-1}(v,q_{t-1})$ and the word boundaries $B_{t-1}(v,q_{t-1})$ are initialized within the tree copies v with the initial states $q_{t-1} = 0$ with the values

$$\begin{aligned} \alpha_{t-1}(v,q_{t-1}=0) &= H(v;t-1) \\ B_{t-1}(v,q_{t-1}=0) &= t-1, \end{aligned} \quad (9)$$

where $H(v;t-1)$ is obtained at the end of time instance $t-1$ as explained beneath. Subsequently $\alpha_t(v,q_t)$ and $B_t(v,q_t)$ are updated for $q_t > 0$ with the forward iteration of a Viterbi algorithm according to the transition probabilities and emission probabilities in the acoustic models:

$$\begin{aligned} \alpha_t(v,q_t) &= \max_{q_{t-1}} \{ \alpha_{t-1}(v,q_{t-1}) P_\alpha(q_t|q_{t-1}) p_\alpha(\mathbf{x}_t|q_t) \} \\ B_t(v,q_t) &= B_{t-1}(v, \arg\max_{q_{t-1}} \{ \alpha_{t-1}(v,q_{t-1}) \\ &\qquad P_\alpha(q_t|q_{t-1}) p_\alpha(\mathbf{x}_t|q_t) \}) \end{aligned} \quad (10)$$

After updating the tree copies for the final states S_w of each word the optimal predecessor

$$v_0(w;t) = \arg\max_v \{ P(w|v) \alpha_t(v, q_t = S_w) \} \quad (11)$$

is determined and stored with the word boundary $B_t(v_0(w;t), q_t = S_w)$. Besides, a forward variable

$$H(w;t) = \max_v \{ P(w|v) \alpha_t(v, q_t = S_w) \} \quad (12)$$

is calculated, which is employed to initialize the forward variable $\alpha_t(v, q_t = 0)$ in the next time instance. At the end of each sentence the most likely word sequence is tracked back using the stored predecessors $v_0(w;t)$ and word boundaries $B_t(v_0(w;t), q_t = S_w)$.

4 Search in the CMHMM

The CMHMM requires an adequate search strategy because the application of the Viterbi algorithm on the mixture component level would lead to a significantly increased computing time as the total number of mixture components of the acoustic model is typically by factors larger than the total number of HMM states. An efficient search without considerable increase in computing time can be achieved with a processing on two levels. The optimal HMM sequence is estimated with a Viterbi algorithm on the HMM state level, while the mixture weights are updated with a causal filter in the forward pass of the Viterbi algorithm, which is based on the observed speech features and the statistical dependencies between the mixture weights in subsequent frames.

Prior to the description of the extended Viterbi algorithm, the required dependencies are derived. The feature vector $\mathbf{x}_t$ depends even for a given HMM state q_t via the mixture weights m_1^{t-1} on the preceding feature vectors $\mathbf{x}_1^{t-1}$. Thus, eq. (6) is written as

$$\begin{aligned} \hat{w}_1^N = \arg\max_{w_1^N} \{ \max_{q_1^T} \{ \prod_{t=1}^{T} p_\alpha(\mathbf{x}_t|\mathbf{x}_1^{t-1}, q_t) \\ P_\alpha(q_t|q_{t-1}) \} \prod_{n=1}^{N} P(w_n|w_{n-m+1}^{n-1}) \}. \end{aligned} \quad (13)$$

where for the emission density holds the relation

$$\begin{aligned} p_\alpha(\mathbf{x}_t|\mathbf{x}_1^{t-1}, q_t) &= \sum_{m_t} p_\alpha(\mathbf{x}_t|\mathbf{x}_1^{t-1}, q_t, m_t) P_\alpha(m_t|\mathbf{x}_1^{t-1}, q_t) \\ &\overset{(5)}{=} \sum_{m_t \in \mathscr{M}(q_t)} p_\alpha(\mathbf{x}_t|m_t) P_\alpha(m_t|\mathbf{x}_1^{t-1}). \end{aligned} \quad (14)$$

In eq. (14), the likelihood $p_\alpha(\mathbf{x}_t|\mathbf{x}_1^{t-1}, q_t, m_t)$ is independend of $\mathbf{x}_1^{t-1}$ (see Fig. 1-b). However, $p_\alpha(\mathbf{x}_t|\mathbf{x}_1^{t-1}, q_t)$ depends on the preceding speech features $\mathbf{x}_1^{t-1}$ via the probabilities $P_\alpha(m_t|\mathbf{x}_1^{t-1})$.

The posterior $P_\alpha(m_t|\mathbf{x}_1^{t-1})$ of the mixture weights can be computed as

$$P_\alpha(m_t|\mathbf{x}_1^{t-1}) = \sum_{q_{t-1}} P_\alpha(q_{t-1}, m_t|\mathbf{x}_1^{t-1}) \tag{15}$$

with the state update equation

$$\begin{aligned} &P_\alpha(q_{t-1}, m_t|\mathbf{x}_1^{t-1}) \\ &= \sum_{m_{t-1}} P_\alpha(m_t|q_{t-1}, m_{t-1}, \mathbf{x}_1^{t-1}) P_\alpha(q_{t-1}, m_{t-1}|\mathbf{x}_1^{t-1}) \\ &\overset{(4)}{=} \sum_{m_{t-1}\in\mathscr{M}(q_{t-1})} P_\alpha(m_t|m_{t-1}) P_\alpha(m_{t-1}|\mathbf{x}_1^{t-1}), \end{aligned} \tag{16}$$

where we have used the fact that m_t is independent of $\mathbf{x}_1^{t-1}$ once m_{t-1} is given, see Fig. 1-b).

Inserting (15) and (16) into the Viterbi approximation (10) with the modified likelihood (14) leads to

$$B_t(v, q_t) = B_{t-1}(v, \hat{q}_{t-1}(q_t)) \tag{17}$$

with

$$\begin{aligned} \hat{q}_{t-1}(q_t) = \arg\max_{q_{t-1}} \{ &\alpha_{t-1}(v, q_{t-1}) P_\alpha(q_t|q_{t-1}) \\ &\sum_{m_t\in\mathscr{M}(q_t)} p_\alpha(\mathbf{x}_t|m_t) P_\alpha(m_t|\mathbf{x}_1^{t-1}) \} \\ = \arg\max_{q_{t-1}} \{ &\alpha_{t-1}(v, q_{t-1}) P_\alpha(q_t|q_{t-1}) \\ &\sum_{m_t\in\mathscr{M}(q_t)} p_\alpha(\mathbf{x}_t|m_t) \sum_{\tilde{q}_{t-1}} P_\alpha(\tilde{q}_{t-1}, m_t|\mathbf{x}_1^{t-1}) \}. \end{aligned} \tag{18}$$

In light of the Viterbi approximation, it is consequent to replace the summation over the predecessor states $\tilde{q}_{t-1}$ in (18) by the single 'best' predecessor state:

$$\begin{aligned} \hat{q}_{t-1}(q_t) \approx \arg\max_{q_{t-1}} \{ &\alpha_{t-1}(v, q_{t-1}) P_\alpha(q_t|q_{t-1}) \\ &\sum_{m_t\in\mathscr{M}(q_t)} p_\alpha(\mathbf{x}_t|m_t) P_\alpha(q_{t-1}, m_t|\mathbf{x}_1^{t-1}) \}. \end{aligned} \tag{19}$$

To obtain $\alpha_t(v, q_t)$, we have to replace the argmax()-operations in (18) and (19) by max()-operations. Using the argument $\hat{q}_{t-1}(q_t)$ which is known to maximize (19) we obtain

$$\begin{aligned} \alpha_t(v, q_t) &= \alpha_{t-1}(v, \hat{q}_{t-1}(q_t)) P_\alpha(q_t|\hat{q}_{t-1}(q_t)) \\ &\quad \sum_{m_t\in\mathscr{M}(q_t)} p_\alpha(\mathbf{x}_t|m_t) P_\alpha(m_t|\mathbf{x}_1^{t-1}) \qquad (20) \\ &\approx \alpha_{t-1}(v, \hat{q}_{t-1}(q_t)) P_\alpha(q_t|\hat{q}_{t-1}(q_t)) \\ &\quad \sum_{m_t\in\mathscr{M}(q_t)} p_\alpha(\mathbf{x}_t|m_t) P_\alpha(\hat{q}_{t-1}(q_t), m_t|\mathbf{x}_1^{t-1}). \qquad (21) \end{aligned}$$

The comparison of eq. (20) and eq. (21) shows that the approximation in eq. (21) implies the approximation

$$P_\alpha(m_t|\mathbf{x}_1^{t-1}) \approx P_\alpha(\hat{q}_{t-1}(q_t), m_t|\mathbf{x}_1^{t-1}). \tag{22}$$

The second and final step in the recursive update of the mixture weight posterior is the measurement update:

$$P_\alpha(m_t|\mathbf{x}_1^{t}) \propto P_\alpha(m_t|\mathbf{x}_1^{t-1}) p_\alpha(\mathbf{x}_t|m_t). \tag{23}$$

The integration of the state equation (16) and the measurement equation (23) in the Viterbi algorithm, which was described in Section 3, leads to the following modification of the Viterbi strategy. At time instance $t = 0$, the mixture weights $m_0 \in \mathscr{M}(q_0)$ are initialized for all HMM states q_0:

$$P_\alpha(m_0|\mathbf{x}_1^0) = P_\alpha(m_0) = P_\alpha(m_0|\hat{q}_0(m_0)). \tag{24}$$

The forward iteration of the standard Viterbi search within words (10) is replaced by the following steps:

Algorithm 1 Forward iteration in the CMHMM

```
for all q_t do
    for all q_{t-1} do
        for all m_t ∈ M(q_t) do
            Compute P_α(q_{t-1}, m_t | x_1^{t-1}) with eq. (16).
        end for
    end for
    Compute q̂_{t-1}(q_t) with eq. (19).
    Compute α_t(v, q_t) with eq. (21).
    for all m_t ∈ M(q_t) do
        Update P_α(m_t | x_1^{t-1}) with eq. (22)
        Perform the measurement update in eq. (23).
    end for
end for
```

5 Memory-efficient state update

Other issues to be addressed are the memory requirements. The statistical dependencies between the mixture components m_{t-1} and m_t can be captured by $P(m_t|m_{t-1})$. As the number of mixture components is large, the storage requirements are considerable. In order to avoid the estimation and storage of this matrix an alternative way of modeling the dependencies among successive mixture components is proposed. It employs switching linear dynamical models (SLDMs) to describe the feature trajectory, a modelling approach which has been successfully applied to speech feature enhancement. The idea consists in mapping the mixture components m_{t-1} of state q_{t-1} to the feature domain, resulting in $\mu_{\mathbf{x}_{t-1}}(q_{t-1})$, predicting the next feature vector $\mu_{\mathbf{x}_t}(q_t)$ with linear state models and finally mapping it back to the mixture space resulting in an estimate of $P_\alpha(m_t|\mathbf{x}_1^{t-1})$ which avoids the necessity of $P(m_t|m_{t-1})$. More precise, the following steps are required in order to update the mixture weights m_t:

$$\begin{aligned} \mu_{\mathbf{x}_{t-1}}(q_{t-1}) &= \sum_{m_{t-1}} P_\alpha(m_{t-1}|\mathbf{x}_1^{t-1}) \mu_{\mathbf{x}_{t-1}}(q_{t-1}, m_{t-1}) \\ \mu_{\mathbf{x}_t}(q_t) &= \sum_{s_t} P(s_t|q_t)(\mathbf{A}(s_t)\mu_{\mathbf{x}_{t-1}}(q_{t-1}) + b(s_t)) \\ P_\alpha(m_t|\mathbf{x}_1^{t-1}) &\propto P(m_t|q_t) p_\alpha(\mu_{x_t}(q_t)|q_t, m_t), \end{aligned} \tag{25}$$

$$\begin{aligned} \text{with} \quad \mu_{\mathbf{x}_{t-1}}(q_{t-1}, m_{t-1}) &= E[p(\mathbf{x}_{t-1}|q_{t-1}, m_{t-1})], \\ \mu_{\mathbf{x}_{t-1}}(q_{t-1}) &= E[p(\mathbf{x}_{t-1}|q_{t-1}, \mathbf{x}_1^{t-1})], \\ \mu_{\mathbf{x}_t}(q_t) &= E[p(\mathbf{x}_t|q_t, \mathbf{x}_1^{t-1})]. \end{aligned} \tag{26}$$

The prediction in eq. (25) is carried out with a set of linear state models s_t which are assigned to the HMM states via the dependency $P(s_t|q_t)$, where s_t denotes the state or regime variable which identifies which out of the M LDMs

to be used. The model parameters $(\mathbf{A}(s_t), \mathbf{b}(s_t), \mathbf{C}(s_t))$ can be obtained from clean speech training data as described in [3]. The training of the statistical dependency $P(s_t|q_t)$ is considered in [11]. The projection of the feature vector on the mixture weights is obtained by application of the Bayes's theorem.

6 Experimental results

The experiments were performed on test set A and test set B of the AURORA2 database. The AURORA2 database is a subset of the TI Digits recognition task to which noise was artificially added at different SNR levels. Each of the two test sets A and B consists of four different noise types. On AURORA2 results given in the tables are average over the SNR levels 0, 5, 10, 15, 20dB. Training has been carried out on clean speech. We modified the ETSI standard front-end extraction in the same manner as in [3] by replacing the energy feature with c_0 and using the squared power spectral density rather than the spectral magnitude as the input of the Mel-frequency filter-bank.

First the standard Viterbi search with a HMM was tested. The speech recognition with the standard front-end (SFE + HMM) modified in the described way yielded an overall recognition accuracy of 60.37% on test set A (Tab. 1) and 56.37% on test set B of the AURORA2 database (Tab. 2). For speech feature enhancement, a switching linear dynamic model approach proposed in [3] with $M = 16$ models was applied (SLDM-M16 + HMM). With the SLDM-M16 recognition rates of $79,87\%$ and $79,16\%$ were obtained on the two test sets of the AURORA2 database. The CMHMM requires the estimation of additional parameters. As described in Section 5, $M = 16$ state models with static speech features and dynamic features of order one and two, i.e. overall 39 components, were estimated on clean speech training data and mapped to the HMM states. The application of the CMHMM lead to recognition rates of $81,24\%$ on test set A and $79,67\%$ on test set B (SLDM-M16 + CMHMM).

	Sub.	Bab.	Car	Exh.	Avg.
SFE + HMM	68,06	44,74	59,97	68,73	60,37
SLDM-M16 + HMM	80.19	72.56	84.28	82.43	79.87
SLDM-M16 + CMHMM	82.31	74.21	86.30	82.15	81.24

Table 1: SLDM-M16: Recognition rates on test set A of the AURORA2 database

	Res.	Str.	Air.	Tra.	Avg.
SFE + HMM	52,07	65,50	52,72	55,19	56,37
SLDM-M16 + HMM	74,39	79,29	79,05	83,91	79,16
SLDM-M16 + CMHMM	74,95	79,98	79,48	84,27	79,67

Table 2: SLDM-M16: Recognition rates on test set B of the AURORA2 database

7 Conclusions

In this paper, a novel segmental HMM based on a modified emission probability was proposed. We presented an efficiant search strategy where the mixture weights were updated with a causal filter. The number of parameters could be reduced by the application of SLDMs which are also used for a model-based speech feature enhancement. Compared to the HMM, the application of the CMHMM leads to improved recognition rates on the AURORA2 database with similar computational requirements, while the memory requirements are only increased by storage for the state mapping table $P(s_t|q_t)$.

Acknowledgment

The research was partly supported by the DFG Research Training Group GK-693 of the Paderborn Institute for Scientific Computation (PaSCo).

Literatur

[1] J.A. Bilmes. Buried markov models: a graphical-modeling approach to automatic speech recognition. In *Computer Speech and Language*, volume 17, pages 213–231, 2003.

[2] L. Digalakis. *Segment-Based Stochastic Models of Spectral Dynamics for Continuous Speech Recognition*. PhD thesis, Boston University, 1992.

[3] J. Droppo and A. Acero. Noise robust speech recognition with a switching linear dynamic model. In *ICASSP*, pages 953–956, 2004.

[4] J. Frankel and S. King. Speech recognition using linear dynamic models. In *IEEE Transactions on Audio Speech and Language Processing*, volume 15, pages 213–231, 2007.

[5] J. Ma and L. Deng. Target-directed mixture linear dynamic models for spontaneous speech recognition. *IEEE Trans. Speech Audio Processing*, 12:47–58, 2004.

[6] M.-J. Russel and W.-J. Holmes. Linear trajectory segmental hmms. *IEEE signal processing letters*, 4:72–74, 1997.

[7] F. Seide, J. Zhou, and L. Deng. Coarticulation modeling by embedding a target-directed hidden trajectory model into hmm-map decoding and evaluation. In *Proc. Int. Conf. Acoustics, Speech and Signal Processing*, volume 1, pages 748–751, 2003.

[8] K.C. Sim and M.J.F. Gales. Discriminative semi-parametric trajectory model for speech recognition. In *Computer Speech and Language*, volume 21, pages 669–687, 2007.

[9] Zen H. Tokuda, K. and T. Kitamura. Trajectory modeling based on hmms with the explicit relationship between static and dynamic features. In *Eurospeech*, 2003.

[10] F. Wessel. *Word Posterior Probabilities for Large Vocabulary Continuous Speech Recognition*. PhD thesis, RWTH Aachen, 2002.

[11] S. Windmann and R. Haeb-Umbach. An iterative approach to speech feature enhancement and recognition. In *Interspeech*, pages 1086–1089, 2007.