

# A STRUCTURED SPEECH MODEL WITH CONTINUOUS HIDDEN DYNAMICS AND PREDICTION-RESIDUAL TRAINING FOR TRACKING VOCAL TRACT RESONANCES

*Li Deng, Leo J. Lee, Hagai Attias, and Alex Acero*

Microsoft Research, One Microsoft Way, Redmond WA 98052, USA

{deng, hagai, alexac}@microsoft.com

## ABSTRACT

A novel approach is developed for efficient and accurate tracking of vocal tract resonances, which are natural frequencies of the resonator from larynx to lips, in fluent speech. The tracking algorithm is based on a version of the structured speech model consisting of continuous-valued hidden dynamics and a piecewise-linearized prediction function from resonance frequencies and bandwidths to LPC cepstra. We present details of the piecewise linearization design process and an adaptive training technique for the parameters that characterize the prediction residuals. An iterative tracking algorithm is described and evaluated that embeds both the prediction-residual training and the piecewise linearization design in an adaptive Kalman filtering framework. Experiments on tracking vocal tract resonances in Switchboard speech data demonstrate high accuracy in the results, as well as the effectiveness of residual training embedded in the algorithm. Our approach differs from traditional formant trackers in that it provides meaningful results even during consonantal closures when the supra-laryngeal source may cause no spectral prominences in speech acoustics.

## 1. INTRODUCTION

In recent years, there has been a growing interest in developing accurate, efficient, and compact representations, as well as related statistical models, of speech dynamics. Such representations include articulatory variables [8, 10], vocal tract shapes [4], formants and vocal tract resonances [1, 5, 3, 9]. In this paper, we present a novel technique of tracking vocal tract resonances (VTRs) as a compact representation for time-varying characteristics of speech. VTRs share some common, desirable temporal properties with articulatory variables, such as smoothness and target-directedness, and yet have a lower dimensionality and more intuitive acoustic interpretation. VTRs are related to but are also different from formants. Unlike formants, VTRs do not “disappear”, merge, or split during any part of speech. Rather, they exist at real frequencies at all times, even when the mouth is closed. Defined as the acoustic resonances for the oral portion of the vocal tract when the excitation is forced at the glottis, VTRs correspond to natural frequencies of the physical system. Hence they cannot “disappear” even if the acoustic signal does not directly reveal them. Importantly, VTRs are a smooth function of the articulatory variables, whose movement uniquely determines the time-varying vocal tract area function shaping the dynamics of the acoustic resonances.

While VTRs may not correspond to spectral prominences where zeros in the vocal tract transfer function exist in fricatives, stops, and nasals, they coincide with formants for non-nasalized vowels where no vocal tract side branches and no supra-glottal excitation sources are involved in speech production. Almost all the existing formant tracking techniques (e.g., [7, 6, 12, 11]) rely, directly or indirectly, on the spectral prominence information from speech acoustics only. The new technique presented in this paper exploits additional dynamic prior information, which we call hidden dynamics, to speech acoustics. The prior captures general time-varying properties of VTR trajectories during speech production even if supra-glottal excitation may eliminate acoustic spectral prominences (such as during fricatives and stops). The joint use of the dynamic VTR prior and speech acoustics, as well as of the explicit relationship between the two domains, establishes a type of structured speech model that enables accurate tracking of VTR trajectories at all times and for all manner and voicing classes of speech.

In our earlier work [2], we developed a version of the structured speech model implemented by discretizing the hidden dynamic vectors of VTR. Approximations due to the discretization and the large number of needed discretization levels in the implementation of [2] can be successfully overcome by using continuous-valued hidden dynamics of VTR. In this paper, we will present this new implementation, where a novel technique based on Kalman filtering is developed to perform VTR tracking and to adaptively train the residual parameters in the predictive mapping from the VTR vectors to the acoustics represented by LPC cepstral vectors.

This paper is organized as follows. The general form of the structured speech model and one of its specific forms for use in VTR tracking are presented in Section 2. Detailed design of piecewise linearization of the nonlinear prediction component in the model’s observation equation is given in Section 3. The piecewise linearization enables the use of highly efficient adaptive Kalman-filter based algorithms to track the state variables of VTR. We provide detailed steps of such an algorithm in Section 4. The algorithm is iterative, and it embeds adaptive training of the parameters (means and variances) characterizing the prediction residuals. Experimental results on VTR tracking validating the algorithm are presented in Section 5.

## 2. STRUCTURED SPEECH MODEL WITH CONTINUOUS HIDDEN DYNAMICS

The most general form of the structured speech model is a time-varying nonlinear dynamic system, with carefully designed prediction functions in both the state equation (1) and observation

---

Leo J. Lee (llee@uwaterloo.ca) was a summer student intern at Microsoft Research from Dept. Electrical and Computer Engineering, University of Waterloo, Ontario, Canada.

equation (2) below:

$$\mathbf{x}(k+1) = \mathbf{g}_{s(k)}[\mathbf{x}(k), \mathbf{u}_{s(k)}] + \mathbf{w}(k) \quad (1)$$

$$\mathbf{o}(k) = \mathbf{h}_{s(k)}[\mathbf{x}(k)] + \mathbf{v}(k), \quad (2)$$

where  $s(k)$  is the speech unit at time frame  $k$ , the prediction functions  $\mathbf{g}[\cdot]$  and  $\mathbf{h}[\cdot]$  are time varying according to the changes in the unit  $s(k)$ .  $\mathbf{x}(k) \in \mathbf{R}^n$  is the hidden state vector representing internal speech dynamics at time  $k$ .  $\mathbf{o}(k) \in \mathbf{R}^m$  is the corresponding acoustic observation vector.  $\mathbf{u}_s \in \mathbf{R}^n$  is called the *target* vector, representing the phonetic correlate of the speech unit (denoted by  $s$ , being phones or phonological features).  $\mathbf{w}(k)$  and  $\mathbf{v}(k)$  are uncorrelated Gaussian noises with covariances  $E[\mathbf{w}(k)\mathbf{w}(l)^T] = \mathbf{Q}\delta_{kl}$  and  $E[\mathbf{v}(k)\mathbf{v}(l)^T] = \mathbf{R}\delta_{kl}$ , respectively.

Two key design issues for adopting the above generic structure as a speech model are: 1) to parameterize the time-varying function  $\mathbf{g}[\cdot]$  so that the temporal evolution of the hidden state vector  $\mathbf{x}(k)$  reflect realistic aspects of speech articulation; and 2) to design  $\mathbf{h}[\cdot]$  so that it properly characterizes the “forward” predictive mapping relation from the hidden vector  $\mathbf{x}(k)$  to the observation vector  $\mathbf{o}(k)$ . A specific design of the model for the VTR tracking application is presented now.

First, the prediction function in Eq. 1 is parameterized by the phone-dependent ( $s(k)$ ) “target” vector  $\mathbf{u}_{s(k)}$  and “system” matrix  $\Phi_{s(k)}$ , resulting in the following first-order, target-directed linear state equation

$$\mathbf{x}(k+1) = \Phi_{s(k)}\mathbf{x}(k) + [\mathbf{I} - \Phi_{s(k)}]\mathbf{u}_{s(k)} + \mathbf{w}_s(k). \quad (3)$$

The target-directed property:  $\mathbf{x}(k) \rightarrow \mathbf{u}$  as  $k \rightarrow \infty$  can be readily verified from Eq. 3, so are the smoothness and other desirable properties. The hidden dynamic vector is taken to be the VTR, consisting of frequencies and bandwidths corresponding to the lowest  $P$  poles (dimensionality  $n = 2P$ ):

$$\mathbf{x} = (\mathbf{f}, \mathbf{b})^T = (f_1, f_2, \dots, f_P, b_1, \dots, b_3, b_P)^T. \quad (4)$$

In one specific implementation, we further simplified Eq.3 into

$$\mathbf{x}(k+1) = \Phi\mathbf{x}(k) + [\mathbf{I} - \Phi]\mathbf{u} + \mathbf{w}(k) \quad (5)$$

by removing parameter dependencies on the speech unit. This eliminates the need for phonetic segmentation information for the VTR tracking application, while reducing the phone-specific prior information on VTR to the phone-independent prior distribution for individual components of the VTR vector. For example, in the implementation, we placed the values of the VTR target frequencies at  $\mathbf{u}_{1:4} = (500 \text{ Hz}, 1500 \text{ Hz}, 2500 \text{ Hz}, 3500 \text{ Hz})^T$ . While no phone-specific targets are provided, this gives the useful constraint in VTR tracking that the mean values of the VTR target frequencies are around the above nominal values. The common continuity constraint  $\mathbf{x}(k+1) = \mathbf{x}(k) + \mathbf{w}(k)$  in formant tracking (e.g., [1]) was a special case of (5) and did not provide the prior nominal values for the formant frequencies.

Second, when LPC cepstra are selected as the acoustic observation vector  $\mathbf{o}(k)$ , the prediction function in Eq. 2 can be made phone independent and be determined precisely by an analytical nonlinear function. As derived in detail in [2], the  $i^{\text{th}}$  component of the vector-valued prediction function from the VTR vector to LPC cepstra is:

$$C(i) = \sum_{p=1}^P \frac{2}{i} e^{-\pi i \frac{b_p}{f_s}} \cos(2\pi i \frac{f_p}{f_s}), \quad i = 1, \dots, m \quad (6)$$

where  $f_s$  is the sampling frequency,  $i$  is the order of the cepstrum up to the highest order of  $m$ , and  $p$  is the pole order of the VTR up to the highest order of  $P$ . To account for the predictive modeling error due to zeros and additional poles beyond  $P$ , we introduce the residual vector  $\boldsymbol{\mu}$  in addition to the use of the zero-mean noise  $\mathbf{v}(k)$  in Eq. 2. This gives rise to the following form of the nonlinear observation equation:

$$\mathbf{o}(k) = \mathbf{C}[\mathbf{x}(k)] + \boldsymbol{\mu} + \mathbf{v}(k). \quad (7)$$

In summary, Eqs. 5 and 7 constitute a simplified version of the structured speech model, based on which a novel VTR tracking algorithm is developed and evaluated. The algorithm does not require phone segmentation due to target parameter tying across phones. Note that in contrast to the earlier approach in [2] where the VTR vector  $\mathbf{x}$  of Eq. 4 was discretized,  $\mathbf{x}$  in the current approach is continuous valued.

### 3. PIECEWISE LINEARIZATION OF THE PREDICTION FUNCTION

The adaptive Kalman filter-based algorithm for VTR tracking using the model given by Eqs. 5 and 7 without discretization requires linearization of the nonlinear observation equation (7). One key advantage of using the LPC cepstra as the acoustic measurement is the straightforward design of high-accuracy piecewise linear approximation to the well-behaved nonlinear function Eq. 6.

In our specific implementation of piecewise linearization, we divide each cycle in the sinusoid in each of the  $P = 4$  terms of Eq. 6 into ten non-uniform regions over the frequency axis. For example, for the first-order cepstrum consisting of only half a cycle of a sinusoid, five regions are pre-defined, and as many as 75 regions are used for the cepstrum of order  $m = 15$ . Using the corresponding cepstral values  $c_r, c_{r+1}$  (determined by Eq. 6 for each separate cepstrum order up to 15 and pole order up to 4) for every region boundary pair  $x_r, x_{r+1}$ , we fit the following linear curve ( $c$  vs.  $x$ ) passing through the two points  $[(x_r, c_r), (x_{r+1}, c_{r+1})]$ :

$$\frac{c - c_r}{x - x_r} = \frac{c_{r+1} - c_r}{x_{r+1} - x_r}.$$

From this, we obtain the slope  $\alpha_j$  and intercept  $\beta_j$  for the linearized region  $j$  according to

$$\alpha_r = \frac{c_{r+1} - c_r}{x_{r+1} - x_r}; \quad \beta_r = c_r - \alpha_r x_r.$$

Then, for each cepstral order  $i$ , we have the following linearization for any VTR frequency value inside a region’s boundaries (assuming fixed bandwidths for simplicity in description):

$$c^r(i) \propto \sum_{p=1}^P [\alpha_r(i, p) f_p + \beta_r(i, p)] = \sum_{p=1}^P \alpha_r(i, p) f_p + \gamma_r(i), \quad (8)$$

where

$$\gamma_r(i) = \sum_{p=1}^P \beta_r(i, p).$$

In a matrix form, Eq. 8 becomes the following linear function (conditioned on region  $r$ ):

$$\mathbf{C}^r[\mathbf{f}] = \mathbf{A}_r \cdot \mathbf{f} + \mathbf{d}_r, \quad (9)$$

where

$$\mathbf{A}_r = \begin{bmatrix} \alpha_r(1,1) & \alpha_r(1,2) & \alpha_r(1,3) & \alpha_r(1,4) \\ \alpha_r(2,1) & \alpha_r(2,2) & \alpha_r(2,3) & \alpha_r(2,4) \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_r(15,1) & \alpha_r(15,2) & \alpha_r(15,3) & \alpha_r(15,4) \end{bmatrix}, \quad (10)$$

$$\mathbf{f} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} \quad \text{and} \quad \mathbf{d}_r = \begin{bmatrix} \gamma_r(1) \\ \gamma_r(2) \\ \vdots \\ \gamma_r(15) \end{bmatrix}. \quad (11)$$

Generalizing to the case with variable bandwidths, we have the following piecewise linearized observation equation:

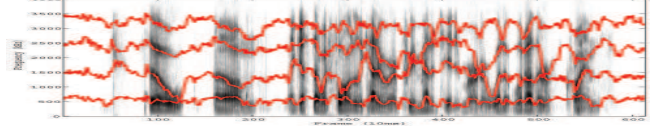
$$\mathbf{o}(k) = \mathbf{A}_r \cdot \mathbf{x}(k) + \mathbf{d}_r + \boldsymbol{\mu} + \mathbf{v}(k), \quad (12)$$

where the ‘‘slope’’ matrix  $\mathbf{A}_r$  and ‘‘intercept’’ vector  $\mathbf{d}_r$  have a somewhat more complicated form than (10) and (11), but they are fixed (i.e., not trained) from the above piecewise linearization procedure based on the known analytical function of Eq. 6. All errors, due to the piecewise linearization approximation or otherwise, are absorbed to the trainable prediction residual parameter,  $\boldsymbol{\mu}$ , in Eq. 12. Note that the ‘‘region’’ index  $r$  (i.e., which ‘‘piece’’ in piecewise linearization) in Eq. 12 is selected based on the approximate value of VTR  $\mathbf{x}$ . In our specific implementation,  $r$  is determined from the prediction step of a ‘‘linearized’’ Kalman filter which we describe in the next section.

#### 4. VTR TRACKING ALGORITHM EMBEDDING PREDICTION-RESIDUAL TRAINING

Once the (piecewise) linearized structured speech model, now consisting of Eqs. 5 and 12, is established, highly efficient adaptive Kalman filtering and smoothing algorithms can be applied directly to track VTRs as the problem of state estimation. To improve the model after a tracking sweep is complete, we use the new VTR estimates to compute the cepstral prediction residuals and then to train the mean and variance parameters for them. The improved model is then used to further improve the VTR tracking. Detailed steps of this adaptive algorithm are provided below:

- Step 1: Initialize the model parameters  $\mathbf{u}$ ,  $\mathbf{Q}$ ,  $\Phi$ ,  $\mathbf{R}$ , and in particular  $\boldsymbol{\mu}(k) = \mathbf{0}$ ;
- Step 2: Kalman filtering (forward pass): For each frame  $k = 1 : N$ 
  - Run Kalman prediction to obtain  $\hat{\mathbf{x}}(k|k-1)$ ;
  - Select region  $\hat{r}$  based on  $\hat{\mathbf{x}}(k|k-1)$ ;
  - Build  $\mathbf{A}_r$  and  $\gamma_r$  in Eq. 12 based on  $\hat{r}$ .
  - Compute Kalman gain and correction to obtain  $\hat{\mathbf{x}}(k|k)$ ;
- Step 3: Kalman smoothing (backward pass): For each frame  $k = N : 1$ , compute  $\hat{\mathbf{x}}(k|N)$ ;
- Step 4: Train residual parameters:
  - Compute predicted cepstra  $\mathbf{C}[\mathbf{x}]$  using Eq. 6 and  $\hat{\mathbf{x}}(k|N)$  for all frames;
  - Compute residuals:  $\mathbf{o}(k) - \mathbf{C}[\hat{\mathbf{x}}(k|N)]$ ;
  - K-mean clustering of all residual frames for the utterance into  $M$  classes;



**Fig. 1.** Tracking VTR frequency ( $f_1$  to  $f_4$ ) trajectories for a typical Switchboard utterance after training the prediction residuals.

- Compute the sample mean and variance for each cluster and use them to update  $\boldsymbol{\mu}(k)$  and  $\mathbf{R}(k)$ ;
- Step 5: Goto to Step 1 using the updated parameters until convergence or a fixed number of iterations is reached.

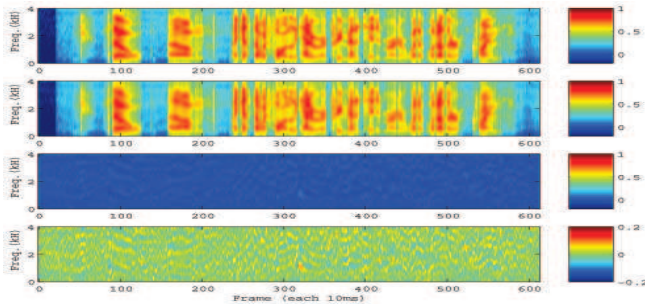
Note that in the above, an assumption was made that the prediction residual follows a mixture-of-Gaussian distribution. The time-varying nature of residual parameters,  $\boldsymbol{\mu}(k)$  and  $\mathbf{R}(k)$ , results from possible switching of the mixture component over frames. If the number of clusters  $M$  is set to be one, then residual parameters,  $\boldsymbol{\mu}$  and  $\mathbf{R}$ , become time invariant. If Step 4 is skipped (Iteration 0), then the algorithm assumes that the analytical nonlinear function for predicting cepstra from VTRs is unbiased.

In our diagnostic experiments, we found that empirical initialization of parameters of  $\mathbf{u}$ ,  $\mathbf{Q}$ , and  $\Phi$  worked satisfactorily well, and hence they were not subject to training in order to reduce computation. However, initialization of  $\mathbf{R}$  (based on the sample residual variance from another fast VTR tracker) and of  $\boldsymbol{\mu}(k) = \mathbf{0}$  did not work well until after the training was carried out. Details of the experiments are presented next.

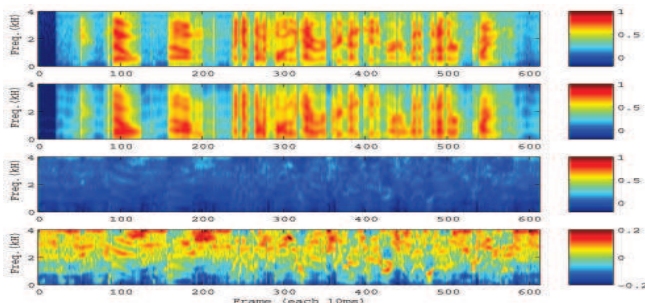
#### 5. EXPERIMENTS AND RESULTS

The algorithm presented in Section 4 has been applied to 249,226 utterances of the Switchboard speech data (training set for a speech recognizer). We have eye-checked several dozens of random utterances among them and found no gross VTR tracking errors. We have also compared our results with the formant tracks from a standard technique in WaveSurfer, and found qualitative improvement mostly in unvoiced sounds and closures. Fig. 3 shows a typical example of the estimated VTR frequency tracks (bandwidths not shown to avoid clutter) with the use of  $M = 10$  Gaussian mixture components and of two iterations of the algorithm described in Section 4. Note that the estimated  $f_1$  usually stays at the normal, low frequency range of the resonance, even if the acoustic spectrum alone does not show prominances.

To examine how accurately the tracked VTRs can provide a compact presentation for speech dynamics, we used the VTR results in Fig. 1 to predict the acoustic spectral trajectory. The prediction was carried out using Eq. 12, but excluding the unpredictable noise or error term  $\mathbf{v}(k)$ . The original speech spectrogram (smoothed by cepstra) is shown in the top panel of Fig. 2, and the predicted spectrogram is shown in the second panel. Excellent match was obtained, and the residual spectrogram, corresponding to the unpredictable noise of  $\mathbf{v}(k)$ , is shown in the third panel of Fig. 2. The magnitude of the prediction error is very low (note the same scaling in plotting the above spectrograms), verifying the strong predicability of the model for the speech data. In the final panel of Fig. 2, we reduced the scaling in order to zoom into the structure of the unpredictable noise. It is clear that not only the unpredictable component of the model is small in magnitude, it



**Fig. 2.** From top to bottom: Cepstral-smoothed spectrogram of the original speech data; Predicted spectrogram from the model; Spectrograms of the unpredictable noise plotted with two different scales. Two training iterations were used.



**Fig. 3.** Same as Fig. 2 but with no prediction-residual training.

also has a more random structure in time and in frequency compared with the original speech signal. Both of these are desirable properties of model prediction.

To examine the role of prediction-residual training, we show in Fig. 3 the same plots as in Fig. 2 except Steps 4 and 5 in the algorithm of Section 4 were eliminated in producing the VTR tracks and in the subsequent prediction of speech acoustics. Deviation of the prediction from the original (comparing the two upper panels) is much larger than that in Fig. 2, resulting in greater and less random prediction errors shown at the bottom two panels of Fig. 3.

To further quantify the effects of prediction-residual training, we computed the prediction error as the sum of squared differences between the original and predicted cepstra over time and cepstral order. The errors as a function of the number of algorithm iterations, with the fixed three Gaussian component for the prediction residual ( $M = 3$ ), are shown in Table 1, where zero-iteration denotes no training of the prediction residual. Dramatic error reduction is seen in the first iteration, and the algorithm quickly converges upon three iterations.

The prediction errors as a function of the number of Gaussian components for the prediction residual, after applying two iterations of the algorithm, are shown in Table 2. Gradual reduction of the prediction error is observed as more components are used.

## 6. SUMMARY AND CONCLUSION

We presented a highly efficient and accurate algorithm for tracking VTRs in natural, fluent speech, which coincide with formants for non-nasalized vowels and they may differ for other types of speech

Iterations	0	1	2	3	4
Pred. Error	670.8	281.7	264.4	258.9	258.8

**Table 1.** Cepstral prediction error versus algorithm iterations.

Mix. Comps. (M)	1	2	3	10	20
Pred. Error	345.6	279.7	264.4	221.8	197.1

**Table 2.** Cepstral prediction error as a function of  $M$ .

sounds. The efficiency is due to the use of an adaptive Kalman filter algorithm, enabled by linearizing the nonlinear component of the speech model. The accuracy is due to the use of a hidden dynamic structure of speech and a physically motivated nonlinear predictive function for speech acoustics, both inherent in the model design. It is also due to the adaptive training for prediction-residual parameters embedded in the tracking algorithm. In many aspects, the new algorithm is superior to an earlier algorithm [2] based on discrete rather than continuous valued hidden VTR dynamics. Because of the elimination of a large number of VTR discretization levels, the new algorithm is more efficient in computation, and it is also generally more accurate as observed in comparative experiments. Our current research involves expanding the current optimization over the VTR dimension alone to joint optimization over both the VTR and speech-unit dimensions in a true spirit of structured speech modeling for speech recognition applications.

## 7. REFERENCES

- [1] I. Bazzi, A. Acero, and L. Deng. "An expectation-maximization approach for formant tracking using a parameter-free non-linear predictor," *Proc. ICASSP*, 2003, pp. 464-467.
- [2] L. Deng, I. Bazzi, and A. Acero. "Tracking vocal tract resonances using an analytical nonlinear predictor and a target-guided temporal constraint," *Proc. Eurospeech*, 2003, Vol. I, pp. 73-76.
- [3] L. Deng and J. Ma, "Spontaneous speech recognition using a statistical coarticulatory model for vocal-tract-resonance dynamics," *J. Acoust. Soc. Am.*, Vol. 108, 2000, pp. 3036-48.
- [4] S. Dusan and L. Deng. "Recovering vocal tract shapes from MFCC parameters," *Proc. ICSLP*, 1998, pp. 3087-90.
- [5] Y. Gao, R. Bakis, J. Huang, and B. Zhang. "Multistage coarticulation model combining articulatory, formant, and cepstral features", *Proc. ICSLP*, Vol. 1, 2000, pp. 25-28.
- [6] G. Kopec. "Formant tracking using HMMs and vector quantization," *IEEE Trans. ASSP*, Vol. 34, 1986, pp. 709-729.
- [7] S. McCandless. "An algorithm for automatic formant extraction using linear prediction spectra," *IEEE Trans. ASSP*, Vol. 22, 1974, pp. 135-141.
- [8] K. Richmond, S. King, and P. Taylor. "Modelling uncertainty in recovering articulation from acoustics," *Computer Speech and Language*, Vol. 17, 2003, pp. 153-172.
- [9] F. Seide, J. Zhou, and L. Deng. "Coarticulation modeling by embedding a target-directed hidden trajectory model into HMM — MAP decoding and evaluation," *Proc. ICASSP*, 2003, pp. 748-751.
- [10] J. Sun, L. Deng, and X. Jing. "Data-driven model construction for continuous speech recognition using overlapping articulatory features," *Proc. ICSLP*, 2000, Vol. I, pp. 437-440.
- [11] D. Talkin. "Speech formant trajectory estimation using dynamic programming with modulated transition costs" *JASA*, S1, 1987, pp. S55.
- [12] L. Welling and H. Ney. "Formant tracking for speech recognition," *IEEE Trans. Speech & Audio Proc.*, Vol. 6, 1998, pp. 36-48.