Adaptive Kalman Filtering and Smoothing for Tracking Vocal Tract Resonances Using a Continuous-Valued Hidden Dynamic Model

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

***Abstract*—A novel Kalman filtering/smoothing algorithm is presented for efficient and accurate estimation of vocal tract resonances or formants, which are natural frequencies and bandwidths of the resonator from larynx to lips, in fluent speech. The algorithm uses a hidden dynamic model, with a state-space formulation, where the resonance frequency and bandwidth values are treated as continuous-valued hidden state variables. The observation equation of the model is constructed by an analytical predictive function from the resonance frequencies and bandwidths to LPC cepstra as the observation vectors. This nonlinear function is adaptively linearized, and a residual or bias term, which is adaptively trained, is added to the nonlinear function to represent the iteratively reduced piecewise linear approximation error. Details of the piecewise linearization design process are described. An iterative tracking algorithm is presented, which embeds both the adaptive residual training and piecewise linearization design in the Kalman filtering/smoothing framework. Experiments on estimating resonances in Switchboard speech data show accurate estimation results. In particular, the effectiveness of the adaptive residual training is demonstrated. Our approach provides a solution to the traditional "hidden formant problem," and produces meaningful results even during consonantal closures when the supra-laryngeal source may cause no spectral prominences in speech acoustics.**

***Index Terms*—Adaptive piecewise linearization, adaptive residual parameter learning, continuous dynamics, formant analysis, hidden dynamic model, nonlinear prediction, speech processing, state-space model, vocal tract resonance.**

I. Introduction

DEVELOPMENT of accurate, efficient, and compact representations of the speech signal and its dynamic behavior has been actively pursued by many speech researchers. The representations investigated include articulatory or pseudo-articulatory variables [2], [12], [19], [22], [25], [28], vocal tract shapes [4], [5], [11], [26] formants and vocal tract resonances [1], [8], [10], [13], [15], [27], [31]. In recent years, we have focused this area of our research on vocal tract resonances (VTRs) as a compact representation for hidden time-varying characteristics of speech, in the context of hidden dynamic modeling for speech recognition. VTRs share some common, desirable temporal properties with articulatory variables but they have much lower dimensionality. (Examples of these temporal properties are smoothness, and the tendency for each VTR component to move toward the respective spatial "target" point within each phonetic segment.[1]) VTRs also have more intuitive acoustic interpretations in terms of spectral prominences in speech acoustics, commonly known as formants, when there are no narrow local constrictions and no side acoustic branches in the vocal tract (i.e., for non-nasal vowels and sonorant consonants). In the recent work reported in [7], [10], a technique for VTR tracking was developed based on a version of the hidden dynamic model where the hidden VTR variables are quantized. This discrete-valued model has inherent quantization errors which are difficult to quantify. And additional errors are introduced by the approximations needed to reduce the otherwise prohibitive amount of computation due to the combinatorics which would result from a very large number of quantization levels. The continuous-valued hidden dynamic model presented in this paper is free from both of these problems due to its elimination of VTR quantization. The difference between the discrete-valued model in [7], [10] and the continuous-valued model described in this paper is analogous to that of the discrete-output HMM and continuous-observation-density HMM elaborated in [21].

Although VTRs may not correspond to spectral prominences where zeros in the vocal tract transfer function exist in fricatives, stops, and nasals, they largely coincide with the spectral prominences or formants for non-nasalized vowels and semivowels. In these speech sounds, no vocal tract side branches and no supra-glottal excitation sources are involved in speech production in general. Almost all the existing formant tracking techniques (e.g., [16], [18], [29], [30], [32]) 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, in the form of VTR hidden dynamics, to speech acoustics. This prior information 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, enables the hidden dynamic model to accurately track VTR trajectories at all times and for all manner and voicing classes of speech.

[1] We call this tendency a "target-directed" property.

Manuscript received September 23, 2004; revised January 13, 2006. A preliminary version of this paper was presented at ICASSP 2004 (rated top in its category) [4]. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Rainer Martin.

L. Deng, H. Attias, and A. Acero are with Microsoft Research, Redmond, WA 98052 USA (e-mail: deng@microsoft.com).

L. J. Lee is with Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1 Canada.

Digital Object Identifier 10.1109/TASL.2006.876724

When the prior information about continuous-valued hidden VTR dynamics is expressed in a recursive form (i.e., state equation), which we have used in the current work, and the relationship between the VTR vector and the acoustic observation vector is represented in a memoryless fashion (i.e., observation equation), a state-space formulation of the hidden dynamic model is established. This formulation allows the powerful and efficient Kalman filtering/smoothing algorithm to apply to the VTR tracking problem. To enable this application, we carry out adaptive piecewise linearization for the nonlinear observation equation. In the past, Kalman filter with linearization has also been used for tracking formants or resonances [20], [23], [24]. The work presented in this paper differs from the previous work in at least three significant ways. First, all the previous work used LPC coefficients as the output vector, resulting in much more complicated nonlinear observation equations than our observation equation with LPC cepstra as the output vector (as a function of the state vector of VTR frequencies and bandwidths). Second, due to the availability of the analytical form of our observation equation, which was lacking in all previous Kalman filtering techniques for formant tracking, we are able to perform direct analysis of the nonlinear observation equation. This allows us to partition the nonlinear function, having linearization of each partitioned region with high approximation accuracy and in an adaptive manner. In contrast, the use of an extended Kalman filter in the past, exemplified in the work of [23], was based on first-order Taylor series approximation and gave unknown approximation accuracy for the linearization of the observation equation.[2] Third, in addition to using carefully designed piecewise linearization to reduce approximation errors in the observation equation, we further introduce an iteratively and adaptively learned residual term to minimize approximation errors in the observation equation as well as VTR tracking errors. None of the earlier formant tracking work based on Kalman filtering used this adaptive mechanism, whose effectiveness will be demonstrated in this paper.

The remainder of this paper is organized as follows. In Section II, we outline the general form of the continuous-valued hidden dynamic model and one of its specific forms for use in VTR tracking as the focus of this paper. We devote Section III to a detailed description of the design process that provides accurate piecewise linearized approximation to the nonlinear observation equation in the hidden dynamic model that maps from the hidden VTR vector to the acoustic observation vector in the form of LPC cepstra. A simplified case is presented first, where only VTR frequencies are partitioned in the design process. This is followed by the general case where both VTR frequencies and bandwidths are subject to partitioning and functional linearization. Given piecewise linearization of the hidden dynamic model, a novel adaptive Kalman filtering/smoothing algorithm is developed and described in Section IV for hidden state estimation; i.e., VTR tracking. Both the region in the piecewise linearization and the cepstrum-prediction residual parameters (mean and variance) are adaptively learned in an iterative procedure. Finally, experimental results on VTR tracking are presented in Section V, demonstrating the effectiveness of the new VTR tracking technique.

II. Continuous-Valued Hidden Dynamic Model

In a general form, the hidden dynamic model, where no quantization is applied to any variables, is a time-varying nonlinear dynamic system with carefully designed prediction functions in both the state (1) and observation (2)

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

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

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

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

A. Prior Model of Hidden Dynamics

The recursive prediction function in (1) is parameterized by the phone-dependent "target" vector $\mathbf{u}_{s(k)}$ and "system" matrix $\mathbf{\Phi}_{s(k)}$, resulting in the following first-order, target-directed linear state equation

$$\mathbf{x}(k+1) = \mathbf{\Phi}_{s(k)}\mathbf{x}(k) + \left[\mathbf{I} - \mathbf{\Phi}_{s(k)}\right]\mathbf{u}_{s(k)} + \mathbf{w}_s(k). \tag{3}$$

The target-directed property: $\mathbf{x}(k) \rightarrow \mathbf{u}$ as $k \rightarrow \infty$ under zero process noise is readily verified from (3), as is the smoothness property (both across and within speech units). The hidden dynamic vector is taken to be the VTR, consisting of resonance frequencies and bandwidths corresponding to the lowest P poles (i.e., dimensionality equals $n = 2P$)

$$\mathbf{x} = (\mathbf{f}, \quad \mathbf{b})' = (f_1, f_2, \ldots, f_P, b_1, b_2, \ldots, b_P)'. \tag{4}$$

For VTR tracking applications, in order to remove the requirement of knowing the phonetic sequence (as well as segmentation) underlying the utterance, we further simplify (3) into

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

[2] While this weakness was recently overcome by the use of particle filtering [31], greater computation was incurred and lower estimation accuracy was observed from the spectrographic displays (overlaying with the estimation results) compared with our approach presented with the same displays as shown in this paper.

by removing parameter dependencies on the speech unit. This gives a weaker prior model than (3) since, for example, the phone-specific VTR targets are no longer provided as the prior information. That is, the simplified prior model (5) reduces the phone-specific prior information on VTR to the phone-independent prior distribution for individual components of the VTR vector. This simplification permits the use the conventional Kalman filter/smoother for efficient state estimation; otherwise, much more costly algorithms would be needed [17].

In our implementation of model (5), we choose the eight-dimensional values (i.e., $P = 4$) of the VTR target frequencies and bandwidths of

$$\mathbf{u} = (500\text{ Hz}, 1500\text{ Hz}, 2500\text{ Hz}, 3500\text{ Hz}, 80\text{ Hz}, 120\text{ Hz}, 150\text{ Hz}, 200\text{ Hz})'. \quad (6)$$

Although no phone-specific targets are provided, (6) gives a useful constraint in VTR tracking that the mean values of the VTR target frequencies and bandwidths are around the above nominal values. Note that the common continuity constraint

$$\mathbf{x}(k+1) = \mathbf{x}(k) + \mathbf{w}(k)$$

in formant or VTR tracking (e.g., [1], [31]) was a special case of (5) and did not provide prior nominal values for the formant frequencies. The work of [10] also used this highly simplified VTR dynamic prior model, and in addition, it quantized the VTR vector $\mathbf{x}(k)$ into discrete values to facilitate the search for optimal VTR values in absence of the Kalman filtering framework.

B. Observation Model

In the current work, LPC cepstra are chosen as the acoustic observation vector, $\mathbf{o}(k)$, in (2). Then, as detailed in [7], the prediction function of (2) can be shown to be phone independent and have a relatively simple analytical nonlinear form based on an all-pole speech model. In this function, the i^{th}-order LPC cepstrum (up to the highest order of I) is expressed as

$$C(i) = \sum_{p=1}^{P} \frac{2}{i} e^{-\pi i \frac{b_p}{f_s}} \cos\left(2\pi i \frac{f_p}{f_s}\right), \quad i = 1, \dots, I \quad (7)$$

where f_s is the sampling frequency, 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 which are not incorporated in (7), we introduce the residual vector $\boldsymbol{\mu}$, also phone independent, in addition to the use of zero-mean noise $\mathbf{v}(k)$ in (2). This gives rise to the following form of the nonlinear observation equation, which we use throughout this work:

$$\mathbf{o}(k) = \mathbf{C}[\mathbf{x}(k)] + \boldsymbol{\mu} + \mathbf{v}(k). \quad (8)$$

In summary, (5) and (8) constitute a version of the continuous-valued nonlinear hidden dynamic model, based on which a novel VTR tracking algorithm within the Kalman filtering and smoothing framework is developed and evaluated, as will be presented in Sections IV and V. The algorithm does not require information of phone labels and segmentations due to model parameter tying across phones (i.e., speech-unit independent). Note that in contrast to the earlier approach in [7], [10] where the VTR vector $\mathbf{x}$ in (4) was discretized, $\mathbf{x}$ in the current approach is continuously valued.

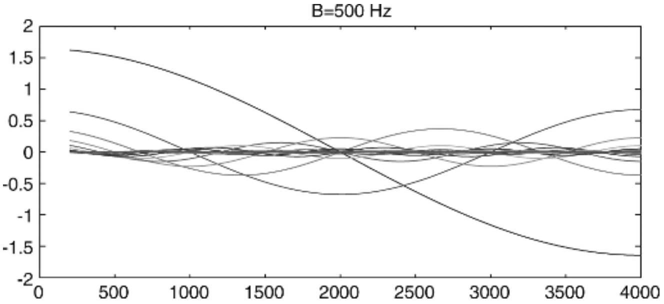

Fig. 1. Plot of one term $(2/i)e^{-\pi i(b/f_s)}\cos(2\pi i(f/f_s))$ in (7) as a function of VTR frequency f (with fixed bandwidth $b = 500$ Hz) for $i = 1, \dots, 15$.

III. Partitioning and Linearization of the Observation Model

The adaptive Kalman filter-based algorithm for VTR tracking using the model given by (5) and (8) without state-variable quantization requires linearization of the nonlinear observation (8). One key advantage of using the LPC cepstra as the acoustic observation vector is the straightforward design of high-accuracy piecewise linear approximation to the well-behaved nonlinear function (7). This design starts with partitioning the input VTR vector space on a component-by-component basis. The partitioning depends on the desired accuracy of linear approximation to (7) for each partition or region.

To illustrate the general property of the nonlinear function (7), we show in Fig. 1 one of the terms (for $i = 1, \dots, 12$) in (7) as a function of the VTR frequency, with the fixed bandwidth of $b = 500$ Hz and fixed sampling frequency of $f_s = 8000$ Hz. Each of the curves is sinusoidal, with an amplitude inversely proportional to the cepstral order. This smooth, well-behaved nonlinearity makes it possible to achieve a piecewise-linear approximation with precontrolled and arbitrarily high accuracy.

In our specific implementation of piecewise linearization, we divide each cycle in the sinusoid, shown in Fig. 1, in each of the $P = 4$ terms of (7) 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 predefined, and as many as 75 regions are used for the highest-order cepstrum. Fewer regions are used for the cepstra of lower orders, since they are less cyclic and hence the partitioning can be made coarser with the same level of approximation accuracy. In the remainder of this section, for simplicity in description, we first derive the piecewise linearized observation equation when only the VTR frequencies are included as the state vector (with a dimension of $P = 4$) which are subject to partitioning, linearization, and estimation; that is, we assume the bandwidths are fixed and are thus not part of the state vector. We then describe the more complicated case when both VTR frequencies and bandwidths are included as the state vector (with a dimension of $2P = 8$) for partitioning, linearization, and estimation.

A. Piecewise Linearized Observation Equation I: Partitioning Frequencies Only

In this simplified case, we have the following single-resonance cepstral function expressed in terms of a sinusoidal function for partitioning and piecewise linearization:

$$c(i) = \underbrace{\frac{2}{i} e^{-\pi i \frac{b}{f_s}}}_{B(i)} \cos\left(2\pi i \frac{f}{f_s}\right) = B(i) \cos\left(2\pi i \frac{f}{f_s}\right). \quad (9)$$

We partition the frequency axis f into R regions: $r = 1, 2, \ldots, R$, where R is 75 for cepstrum with order $i = 15$, and is gradually reduced to 5 for cepstrum with order $i = 1$. For each pair of the partitioned region boundaries f_r, f_{r+1} in VTR frequencies, we have the corresponding cepstral values c_r and c_{r+1} as determined by (9). Within each region, we fit the following linear curve (c versus x) passing through the two points $[(f_r, c_r), (f_{r+1}, c_{r+1})]$:

$$\frac{c - c_r}{f - f_r} = \frac{c_{r+1} - c_r}{f_{r+1} - f_r}.$$

From this, we obtain the slope α_j and intercept β_j for the linearized region j according to

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

Then, for each cepstral order i, we have the following linearized cepstral function (with P terms corresponding to P resonances) for any VTR frequency value inside the region's boundaries:

$$c^r(i, P) \approx \sum_{p=1}^{P} [\alpha_r(i,p) f_p + \beta_r(i,p)] = \sum_{p=1}^{P} \alpha_r(i,p) f_p + g_r(i) \quad (10)$$

where

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

In a matrix form, (10) becomes the following linear function (conditioned on region r):

$$\mathbf{c}^r[\mathbf{f}] = \mathbf{G}_r \cdot \mathbf{f} + \mathbf{g}_r \quad (11)$$

where

$$\mathbf{G}_r = \begin{bmatrix} \alpha_r(1,1) & \alpha_r(1,2) & \cdots & \alpha_r(1,P) \\ \alpha_r(2,1) & \alpha_r(2,2) & \cdots & \alpha_r(2,P) \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_r(M,1) & \alpha_r(M,2) & \cdots & \alpha_r(M,P) \end{bmatrix} \quad (12)$$

$$\mathbf{f} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_P \end{bmatrix} \text{ and } \mathbf{g}_r = \begin{bmatrix} g_r(1) \\ g_r(2) \\ \vdots \\ g_r(M) \end{bmatrix}. \quad (13)$$

This then gives rise to the piecewise linearized observation equation

$$\mathbf{o}(k) \approx \mathbf{G}_r \cdot \mathbf{x}(k) + \mathbf{g}_r + \boldsymbol{\mu} + \mathbf{v}(k) \quad (14)$$

where the state vector is $\mathbf{x}(k) = \mathbf{f}(k)$.

B. Piecewise Linearized Observation Equation II: Partitioning Both Frequencies and Bandwidths

In this general case, we need to partition and then linearize both the sinusoidal and exponential functions in the following single-resonance cepstral expression

$$c(i) = \frac{2}{i} e^{-\pi i \frac{b}{f_s}} \cos\left(2\pi i \frac{f}{f_s}\right). \quad (15)$$

For each of the P VTR bandwidths, we partition its axis uniformly from 0 Hz to 500 Hz with an increment of 50 Hz; that is, $r = 1, 2, \ldots, R$, where the total number of regions is $R = 10$. Given a fixed region r, we carry out the same linearization process as before, except now for both the sinusoidal and exponential functions (omit region index r and resonance index p for brevity)

$$\begin{aligned} \cos\left(2\pi i \frac{f}{f_s}\right) &\approx \alpha(i) f + \beta(i) \\ e^{-\pi i \frac{b}{f_s}} &\approx \gamma(i) b + \delta(i). \end{aligned} \quad (16)$$

This gives a (piecewise) linear approximation to the single-resonance ith-order cepstrum:

$$\begin{aligned} c(i) &\approx \frac{2}{i} \{[\alpha(i) f + \beta(i)] [\gamma(i) b + \delta(i)]\} \\ &= \frac{2}{i} \{\alpha(i)\gamma(i) \cdot f \cdot b + \alpha(i)\delta(i) \cdot f \\ &\qquad + \beta(i)\gamma(i) \cdot b + \beta(i)\delta(i)\} \\ &= \frac{2}{i} \{\alpha(i)\gamma(i) \cdot (f_0 + \Delta f) \cdot (b_0 + \Delta b) \\ &\qquad + \alpha(i)\delta(i) \cdot f + \beta(i)\gamma(i) \cdot b + \beta(i)\delta(i)\} \quad (17) \\ &= \frac{2}{i} \{\alpha(i)\gamma(i)[f_0 \cdot b_0 + \Delta f \Delta b + f_0 \Delta b + b_0 \Delta f] \\ &\qquad + \alpha(i)\delta(i) \cdot f + \beta(i)\gamma(i) \cdot b + \beta(i)\delta(i)\} \\ &\approx \frac{2}{i} \{\alpha(i)\gamma(i)\,[f_0 \cdot b_0 + f_0 \cdot (b - b_0) + b_0 \cdot (f - f_0)] \\ &\qquad + \alpha(i)\delta(i) \cdot f + \beta(i)\gamma(i) \cdot b + \beta(i)\delta(i)\} \quad (18) \\ &= \frac{2}{i} \{\alpha(i)\gamma(i)[f_0 \cdot b + b_0 \cdot f] + \alpha(i)\delta(i) \cdot f \\ &\qquad + \beta(i)\gamma(i) \cdot b + \beta(i)\delta(i) - \alpha(i)\gamma(i) f_0 b_0\} \\ &= \frac{2}{i} \left\{ \underbrace{[\alpha(i)\gamma(i) b_0 + \alpha(i)\delta(i)]}_{\phi(i)} \cdot f \right. \\ &\qquad + \underbrace{[\alpha(i)\gamma(i) f_0 + \beta(i)\gamma(i)]}_{\psi(i)} \cdot b \\ &\qquad \left. + \underbrace{\beta(i)\delta(i) - \alpha(i)\gamma(i) f_0 b_0}_{d(i)} \right\} \\ &= \frac{2}{i} \{\phi(i) \cdot f + \psi(i) \cdot b + d(i)\}. \quad (19) \end{aligned}$$

In (17), we used $f = f_0 + \Delta f$ and $b = b_0 + \Delta b$ where f_0 and b_0 are the VTR frequency and bandwidth at the left-side boundary of the partitioned region. And in (18), the higher-order term $\Delta f \Delta b$ was ignored.

Given (19), the general expression for the P-resonance cepstrum, with order $i = 1, 2, \ldots, I$, now has the following form of piecewise linear approximation (region denoted by r):

$$\begin{aligned} c^r(i,P) &\approx \sum_{p=1}^{P} \frac{2}{i} \left\{ \phi_r(i,p) f_p + \psi_r(i,p) b_p + d_r(i,p) \right\} \\ &= \sum_{p=1}^{P} \frac{2}{i} \left\{ \phi_r(i,p) f_p + \psi_r(i,p) b_p \right\} + h_r(i) \quad (20) \end{aligned}$$

where

$$h_r(i) = \frac{2}{i} \sum_{p=1}^{P} d_r(i,p).$$

In a matrix form, (20) becomes the region-r dependent linear function

$$\mathbf{c}^r[\mathbf{x}] = \mathbf{H}_r \cdot \mathbf{x} + \mathbf{h}_r \qquad (21)$$

where [see (22) at the bottom of the page], and

$$\mathbf{x} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_P \\ b_1 \\ b_2 \\ \vdots \\ b_P \end{bmatrix} \text{ and } \mathbf{h}_r = \begin{bmatrix} h_r(1) \\ h_r(2) \\ \vdots \\ h_r(I) \end{bmatrix}. \qquad (23)$$

The final result for the piecewise linearized observation equation becomes

$$\mathbf{o}(k) \approx \mathbf{H}_r \cdot \mathbf{x}(k) + \mathbf{h}_r + \boldsymbol{\mu} + \mathbf{v}(k) \qquad (24)$$

where the state vector is $\mathbf{x}(k) = (\mathbf{f}(k), \mathbf{b}(k))'$.

Note that the "slope" matrix $\mathbf{H}_r$ and "intercept" vector $\mathbf{h}_r$ have no free parameters. They are obtained from the above piecewise linearization procedure based on the known analytical function of (7). All errors, due to the piecewise linearization approximation, as well as the approximation of (7) to real speech cepstral data, are absorbed to the trainable prediction residual parameter $\boldsymbol{\mu}$ in (14) and (24). The "region" index r (i.e., which "piece" in piecewise linearization) in (14) and (24) is selected based on the approximate value of the state vector $\mathbf{x}$. In our specific implementation, r is determined from the prediction step of a "linearized" Kalman filter which we will describe in Section IV.

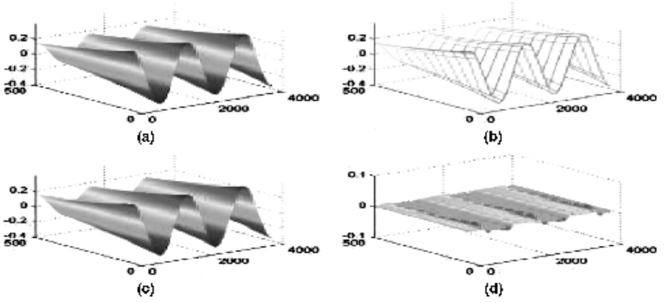

Fig. 2. Illustration of approximations to (one term of) the analytical nonlinear function of (7) with predesigned input regions for VTR frequencies and bandwidths. Each region represents a separate linear approximation to the nonlinear function, giving rise to the overall piecewise linear approximation. See text for details of the four subplots. (a) Exact nonlinear mapping. (b) Linearization points. (c) Piecewise linear mapping. (d) Error surface.

We have described in this section two ways of linearizing the observation equation in the hidden dynamic model—one deterministically fixing the VTR bandwidth and the other treating the VTR bandwidth as the random vector. Fixing the bandwidth values makes the estimation algorithm and implementation much simpler, and it has the state vector with a lower dimension. However, since the fixed bandwidth values may be inaccurate (they are empirically chosen as shown in Section II-A used in our experiments), the resulting estimates may be affected by this inaccuracy. In contrast, when the VTR bandwidths are included as part of the state vector, they are simultaneously estimated with the VTR frequencies. Although this implementation as detailed in Section III-B is more complex, it does not suffer from the empirical choice of the bandwidth values. Because there is no standard database available with correctly annotated VTR values, we have not been able to systematically assess these two implementations experimentally. Visual inspection of the estimation results indicates that the latter implementation including the bandwidth in the state vector is slightly superior to former implementation.

C. Illustration of Piecewise Linearization

Predesign of the input regions for piecewise linear approximation to the observation equation in the state-space based hidden dynamic model is the most significant aspect of our new approach. Fig. 2(a)–(d) provides an example of the result of this design process. The example is taken for the 5th order cepstrum, where the exact nonlinear mapping for a single-resonance term in (7) is shown in (a) and the predesigned linearization regions are shown in (b). The VTR frequency is plotted which ranges from 0 to 4000 Hz, and the bandwidth from 0 to 500 Hz, covering the typical resonances in speech sounds. The piecewise linearized function using the predesigned regions is given in (c), which can be seen to be virtually the same as the original function (a). The very small errors due to the approximation are plotted in (d); note the enlarged scale in the plot in order to show the errors.

$$\mathbf{H}_r = 2 \cdot \begin{bmatrix} \phi_r(1,1) & \phi_r(1,2) & \cdots & \phi_r(1,P) & \psi_r(1,1) & \psi_r(1,2) & \cdots & \psi_r(1,P) \\ \frac{1}{2}\phi_r(2,1) & \frac{1}{2}\phi_r(2,2) & \cdots & \frac{1}{2}\phi_r(2,P) & \frac{1}{2}\psi_r(2,1) & \frac{1}{2}\psi_r(2,2) & \cdots & \frac{1}{2}\psi_r(2,P) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{1}{I}\phi_r(I,1) & \frac{1}{I}\phi_r(I,2) & \cdots & \frac{1}{I}\phi_r(I,P) & \frac{1}{I}\psi_r(I,1) & \frac{1}{I}\psi_r(I,2) & \cdots & \frac{1}{I}\psi_r(I,P) \end{bmatrix} \qquad (22)$$

IV. Adaptive Kalman Filter and Smoother Embedding Prediction-Residual Training

After the piecewise linearized hidden dynamic model, consisting of (5) and (14) [or (24)], is established, highly efficient adaptive Kalman filtering and smoothing algorithm can be applied to track VTRs as this problem has now been reduced to a special case of the well-known problem of minimal-mean-square-error state estimation (e.g., Chapter 5 in [6]). The novel aspect of adaptive learning in the algorithm developed here consists of two key elements. First, the choice of the linearized region, r, which determines the model parameters in the linearized observation (14) ($\mathbf{G}_r$ and $\mathbf{g}_r$) or (24) ($\mathbf{H}_r$ and $\mathbf{h}_r$), is learned adaptively based on the predictor stage of the Kalman filter. Second, the prediction-residual parameters, $\boldsymbol{\mu}$ and $\mathbf{Q}_v$, in the linearized observation (14) and (24) are adaptively learned after each iteration of the VTR tracking sweep is complete with the new VTR estimates available. The improved observation equations with updated parameters of $\boldsymbol{\mu}$ and $\mathbf{Q}_v$ are then used to further improve VTR tracking. Detailed steps of this adaptive algorithm are provided below.

Adaptive VTR racking algorithm

```
Step 1) Fix model parameters $\mathbf{u}$, $\mathbf{Q}_w$, and $\mathbf{\Phi}$;
Step 2) Initialize $\boldsymbol{\mu}(k)$ and $\mathbf{Q}_v(k)$;
Step 3) Kalman filtering (forward pass): For all frames
        $k = 1, 2, \ldots, N$
          • Run Kalman predictor to obtain $\hat{\mathbf{x}}(k|k-1)$;
          • Choose region $\hat{r}$ based on $\hat{\mathbf{x}}(k|k-1)$;
          • Choose $\mathbf{H}_r$ and $\mathbf{h}_r$ in (24) based on $\hat{r}$.
          • Compute Kalman gain and correction to
            obtain $\hat{\mathbf{x}}(k|k)$;
Step 4) Kalman smoothing (backward pass): For frames
        $k = N, N-1, \ldots, 1$, compute $\hat{\mathbf{x}}(k|N)$;
Step 5) Adapt residual parameters in observation
        equation:
          • Compute predicted cepstra $\mathbf{C}[\mathbf{x}]$ using (7)
            with $\hat{\mathbf{x}}(k|N)$ as input for all frames;
          • Compute residual vectors: $\mathbf{o}(k) - \mathbf{C}[\hat{\mathbf{x}}(k|N)]$;
          • K-mean clustering (using Euclidean distance)
            of all residual frames for the utterance into
            $M$ classes and index each frame with the
            associated class $m$;
          • Compute the sample means and variances for
            each cluster and use them to update $M$ sets
            of $\boldsymbol{\mu}(m)$ and $\mathbf{Q}_v(m)$, and assign each frame
            with the updated mean and variance based on
            the indexed class;
Step 6) Go to Step 2 using the updated frame-dependent
        mean $\boldsymbol{\mu}(k)$ and variance $\mathbf{Q}_v(k)$ parameters until
        convergence or a fixed number of iterations is
        reached. The output of the algorithm is $\hat{\mathbf{x}}(k|N)$
        at the final iteration.
```

In the above, an assumption is made that the cepstral prediction residual from VTR follows a mixture-of-Gaussian distribution. In Step 2 above, for all mixture components, the mean vectors $\boldsymbol{\mu}$ are initialized to be zero and the diagonal covariance matrices are initialized identically according to sample variance computation.[3] After each iteration with the updated residual's M-component mixture's means and variances, for each frame, we select one of the M sets of the residual means and variances, according to the minimum cepstral prediction error. (This selection is easily carried out after indexing each frame with the class label in the K-mean clustering procedure.) The selected, frame-dependent residual means and variances are used in Kalman filter and smoother in Steps 3 and 4 for the next iteration.[4] We now provide detailed computation for these Steps 3 and 4 below:

Kalman filter using piecewise linearized hidden dynamic model (Step 3)

For $k = 1, 2, \ldots, N$, and for the given adaptively selected region $r(\hat{\mathbf{x}}_{k+1|k})$ and residual parameters,

Kalman Prediction

$$\begin{aligned}\hat{\mathbf{x}}_{k+1|k} &= \mathbf{\Phi}\hat{\mathbf{x}}_{k|k} + [\mathbf{I} - \mathbf{\Phi}]\mathbf{u} \\ \mathbf{\Sigma}_{k+1|k} &= \mathbf{\Phi}\Sigma_{k|k}\mathbf{\Phi}^{\mathrm{Tr}} + \mathbf{Q}_w\end{aligned}$$

Kalman Gain

$$\mathbf{K}_{k+1} = \mathbf{\Sigma}_{k+1|k}\mathbf{H}_r^{\mathrm{Tr}}\left[\mathbf{H}_r\mathbf{\Sigma}_{k+1|k}\mathbf{H}_r^{\mathrm{Tr}} + \mathbf{Q}_v\right]^{-1}$$

Kalman Correction

$$\begin{aligned}\hat{\mathbf{x}}_{k+1|k+1} &= \hat{\mathbf{x}}_{k+1|k} + \mathbf{K}_{k+1} \\ &\quad \times \left[\mathbf{o}(k+1) - \mathbf{H}_r \cdot \hat{\mathbf{x}}_{k+1|k} - \boldsymbol{\mu}(k) - \mathbf{h}_r\right] \\ \mathbf{\Sigma}_{k+1|k+1} &= \mathbf{\Sigma}_{k+1|k} - \mathbf{K}_{k+1} \\ &\quad \times \left[\mathbf{H}_r\mathbf{\Sigma}_{k+1|k}\mathbf{H}_r^{\mathrm{Tr}} + \mathbf{Q}_v(k)\right]\mathbf{K}_{k+1}^{\mathrm{Tr}}.\end{aligned}$$

Kalman smoother using piecewise linearized hidden dynamic model (Step 4)

Given the Kalman filter results $\hat{\mathbf{x}}_{k|k-1}$, $\hat{\mathbf{x}}_{k|k}$, $\mathbf{\Sigma}_{k|k-1}$, and $\mathbf{\Sigma}_{k|k}$, the smoothed VTR estimate defined as $\hat{\mathbf{x}}_{k|N} = E(\mathbf{x}(k)|\mathbf{o}_1^N)$ is computed for $k = N-1, N-2, \ldots, 1$ recursively by

$$\begin{aligned}\hat{\mathbf{x}}_{k|N} &= \hat{\mathbf{x}}_{k|k} + \mathbf{L}_k(\hat{\mathbf{x}}_{k+1|N} - \hat{\mathbf{x}}_{k+1|k}) \\ \mathbf{\Sigma}_{k|N} &= \mathbf{\Sigma}_{k|k} + \mathbf{L}_k(\mathbf{\Sigma}_{k+1|N} - \mathbf{\Sigma}_{k+1|k})\mathbf{L}_k^{\mathrm{Tr}},\end{aligned}$$

where $\mathbf{L}_k \equiv \mathbf{\Sigma}_{k|k}\mathbf{\Phi}^{\mathrm{Tr}}\mathbf{\Sigma}_{k+1|k}^{-1}$.

[3] The sample variances are based on a small set of training data and from the cepstral prediction errors computed using (7) with the VTR tracker developed in [10].

[4] The use of the frame-dependent residual means and variances makes the algorithm efficient, requiring only one instead of M times of running Steps 3 and 4 (Kalman filter/smoother) for the next iteration. A more rigorous method would be to run Kalman filter/smoother M times in the next iteration, one for each of the M frame-independent Gaussian residual parameter sets. This may give higher accuracy but is much more expensive in computation.

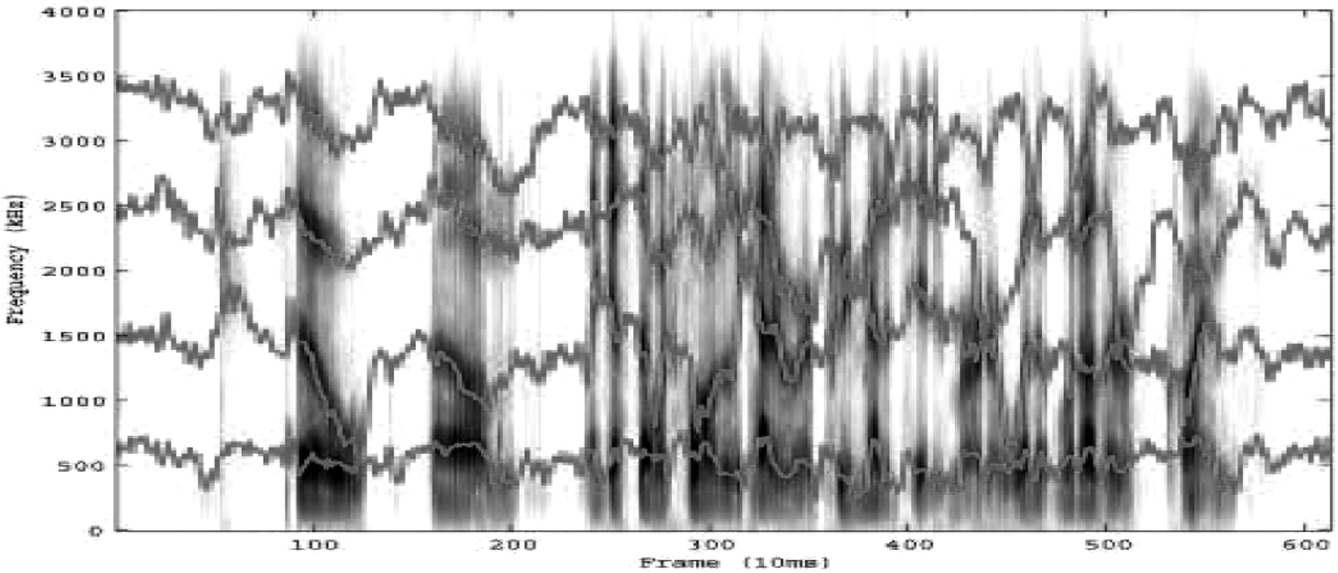

Fig. 3. Tracking VTR frequency (f_1 to f_4) trajectories for a typical Switchboard utterance after two iterations of adaptive training of the prediction-residual parameters with $M = 10$ Gaussian mixture components.

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

V. Experiments and Results

The adaptive Kalman filtering and smoothing algorithm presented in Section IV has been implemented in Matlab and applied to 249,226 utterances of the Switchboard speech data [5] to obtain the estimates of VTR f_1 to f_4 (as well as b_1 to b_4) sequences in these data. We have eye-checked several dozens of random utterances among them and found no gross VTR tracking errors based on overlaid plots of the computed VTR tracks with high-quality spectrogram displays. We have also compared our results with the formant tracks from a standard formant tracking technique in WaveSurfer, and found qualitative improvement in unvoiced sounds and in 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 five-step algorithm described in Section IV. Note that the estimated f_1 typically stays at the normal, low frequency range of the resonance, even if the acoustic spectrum alone does not show prominences in this range.

To examine the degree to which the tracked VTRs can accurately provide a compact representation for speech dynamics, we use the VTR results in Fig. 3 to predict the acoustic spectral trajectory based on the observation equation of the hidden dynamic model. The prediction is carried out using observation (14), 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. 4, and the predicted spectrogram is shown in the second panel. The predicted spectrogram (log magnitude as plotted) is obtained by performing inverse Fourier transform on the sum of the residual mean vector and the output of (14) using the tracked VTR frequencies and bandwidths as the input. Excellent match to the data spectrogram is observed, and the spectrogram corresponding to the unpredictable noise of $\mathbf{v}(k)$, is shown in the third panel of Fig. 4. The magnitude of the prediction error is very low (note the same scaling in plotting the above spectrograms), verifying the strong predictability of the model for the speech data. In the final panel of Fig. 4, we reduce 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 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 the adaptive prediction-residual training, we show in Fig. 5 the same plots as in Fig. 4 except Steps 5 and 6 in the VTR tracking algorithm of Section IV are eliminated in producing the VTR tracks and in the subsequent prediction of speech acoustics; that is, the residual mean vector $\boldsymbol{\mu}$ is set to zero in initialization and not subsequently adapted. Comparing the two upper panels of Fig. 5, we observe that the difference between the data spectrogram and the predicted spectrogram is considerably larger than that in Fig. 4. This results in greater and less random prediction errors shown at the bottom two panels of Fig. 5.

To further quantify the effects of adaptive prediction-residual training, we compute the cepstral prediction error as the sum of squared differences between the original and predicted cepstra over time and over cepstral order. The errors as a function of the number of algorithm iterations, with the fixed three Gaussian components for the prediction residual ($M = 3$), are shown in Table I, 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 two to three iterations.

[5] This data set is used as the training data for a speech recognizer.

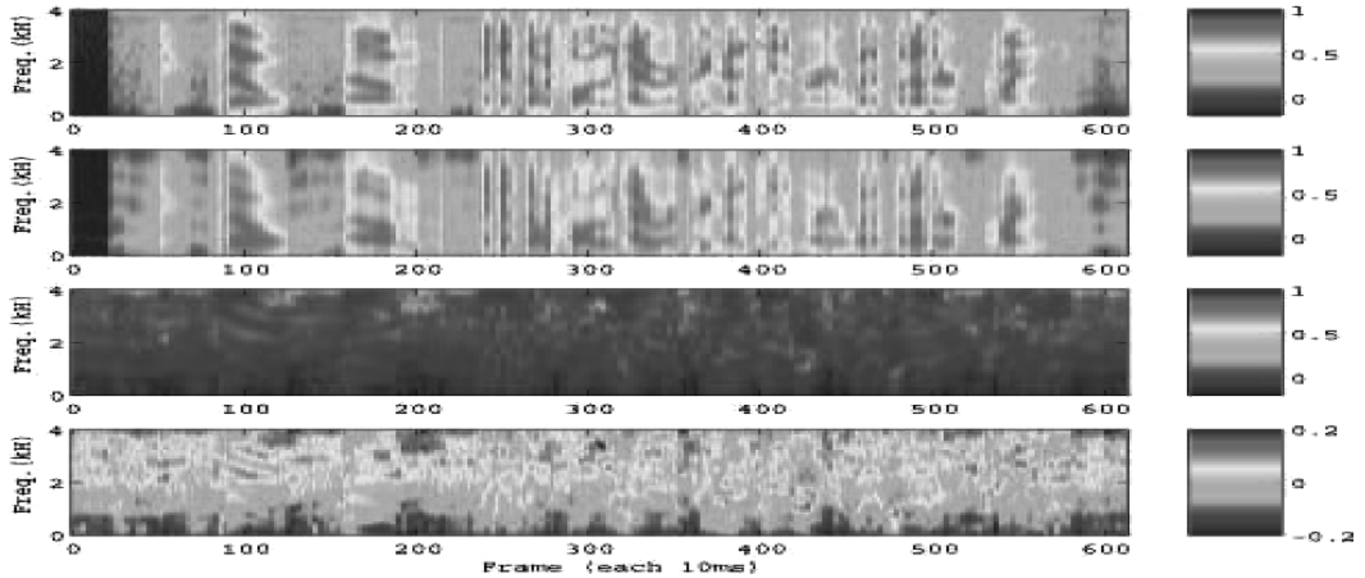

Fig. 4. From top to bottom: cepstral-smoothed spectrogram of the original speech data; predicted spectrogram from the model; spectrograms of the unpredictable noise plotted with two different scales. Two adaptive training iterations are used to track the VTR frequencies that are used for predicting the cepstral sequence and then spectrogram by inverse fourier transformation.

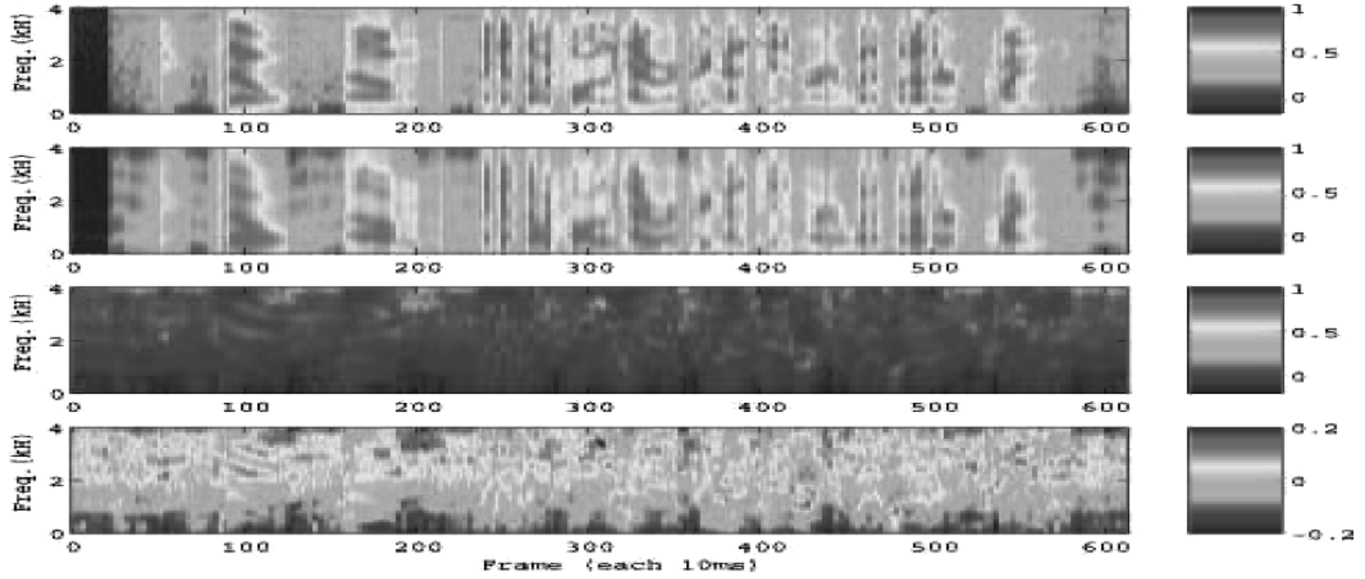

Fig. 5. Same as Fig. 4 but with no adaptive prediction-residual training.

TABLE I
CEPSTRAL PREDICTION ERRORS AS A FUNCTION OF THE NUMBER OF ALGORITHM ITERATIONS FOR ADAPTIVE TRAINING OF THE PREDICTION RESIDUAL PARAMETERS (MEANS AND VARIANCES)

No. of Iterations	0	1	2	3	4
Prediction Error	670.8	281.7	264.4	258.9	258.8

TABLE II
CEPSTRAL PREDICTION ERRORS AS A FUNCTION OF THE NUMBER OF MIXTURE COMPONENTS M FOR THE PREDICTION RESIDUAL

No. of Mixture Components (M)	1	2	3	10	20
Prediction Error	345.6	279.7	264.4	221.8	197.1

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 II. Gradual reduction of the prediction error is observed as more components are used. However, the error reduction due to the increase of the number of mixture components is more dramatic when the number is low (e.g., from $M = 1$ to $M = 2$) than when the number becomes high (e.g., from $M = 10$ to $M = 20$).

The findings reported above in this section are consistent among all the utterances that we have examined. We present a further example utterance here in Fig. 6 for the VTR frequency tracking results with the number of mixture components M being set at 10 in the prediction residual and with the iterative algorithm being run at convergence. As a contrast, we show

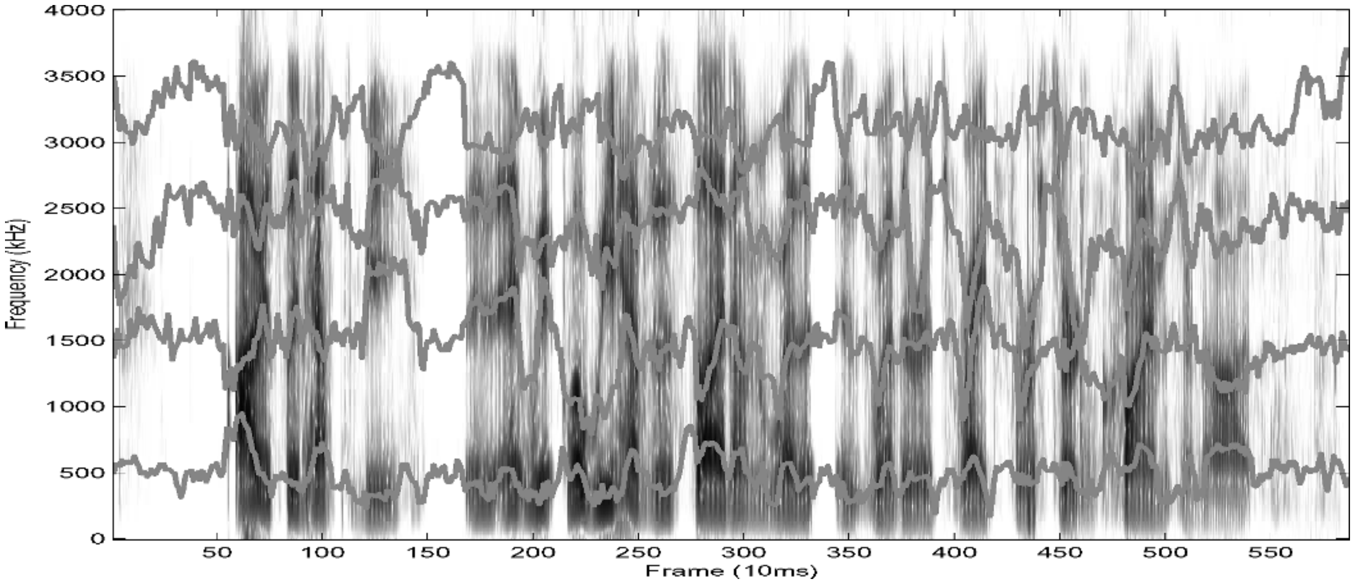

Fig. 6. VTR frequency (f_1 to f_4) tracking results for another switchboard utterance with adaptive training at convergence and with $M = 10$.

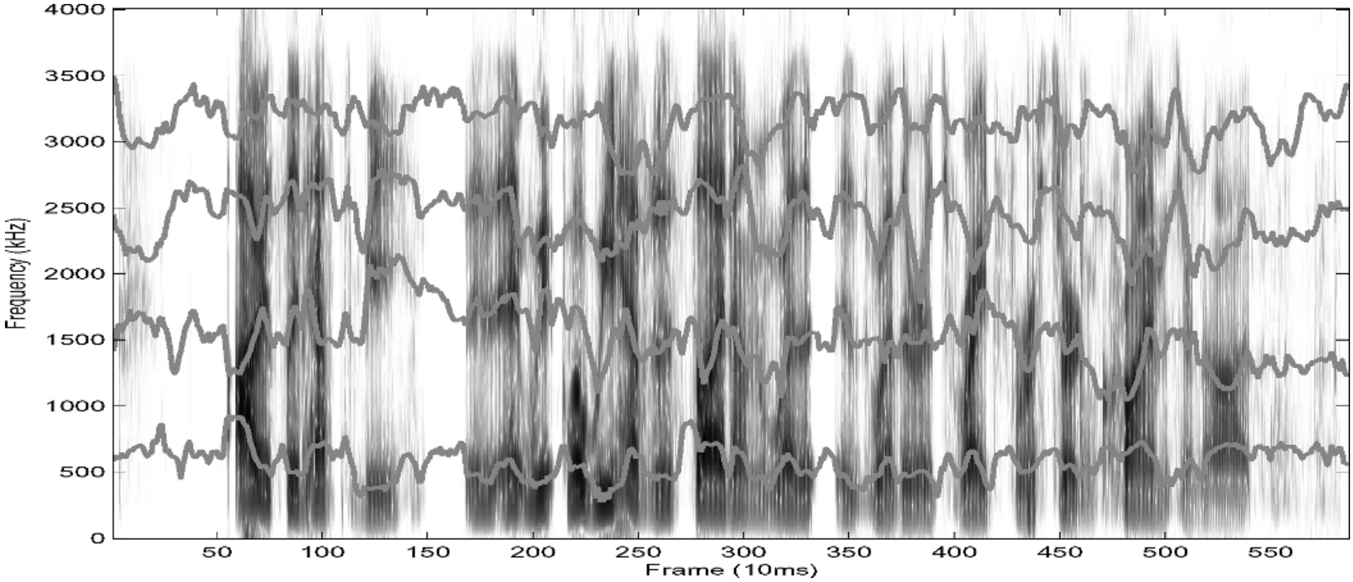

Fig. 7. VTR frequency (f_1 to f_4) tracking results for the same utterance at in Fig. 6 but with no adaptive training applied to the prediction residual parameters.

in Fig. 7 the VTR frequency tracking results with no adaptive training applied (i.e., $\boldsymbol{\mu} = 0$ which is not split and updated). As can be seen, several fast moving resonance frequencies, such as those clearly visible for f_2 (e.g., around frames 220, 370, 410, and 430), are not tracked accurately. The mis-tracking of VTRs of such a type accounts for large cepstral prediction errors since the wrong input to the prediction function (7) necessarily creates the output cepstra that are far away from the true cepstra in the data. In general, as illustrated in this example, we have found that the adaptive training of prediction residual parameters is more effective when the utterance contains the VTRs with a faster rate of movement and with a wider range of local changes.

For all the speech utterances we have experimented and examined, the reported tracking algorithm provides meaningful results even during consonantal closures when the supra-laryngeal source may cause no spectral prominences in speech acoustics. The tracked VTRs correspond ideally to the underlying resonances in the vocal tract with or without direct acoustic evidence. But in absence of the true resonance data that would need to be computed from the speaker's vocal tract shapes, indirect evidence supporting such ideal correspondence can be gleaned from the results shown in Fig. 3 and Fig. 6. For example, realistic transitions in the tracked VTR frequencies in Fig. 6 are visible from the consonantal closure or constriction regions into and out of the adjacent vocalic sounds where spectral prominences in speech acoustics are evident (e.g., between frames 140 to 180 for f_2). In traditional formant tracking approaches, however, since the closure regions have little acoustic energy around resonance frequencies, the tracking algorithms usually give random or no estimates (e.g., [16], [18]). Such a "hidden formant problem" is eliminated in our approach, where the tracked VTRs are provided throughout the entire utterances as shown in the examples provided above.

VI. Summary and Discussion

In this paper, we present a novel algorithm for high-accuracy tracking of VTRs in natural, fluent speech, which coincide with formants or spectral prominences for non-nasalized vowels and sonorant consonants but they may differ for other types of speech sounds. The main novelty is in the use of an adaptive Kalman filter algorithm, which is enabled by linearizing the nonlinear component of the continuous-valued hidden dynamic model. The algorithm is based on the state-space model comprising a target-directed dynamic structure of speech and a physically motivated nonlinear predictive function for speech acoustics.

One key innovation of the work presented in this paper is the elaborate design of piecewise linearization on the parameter-free, analytical nonlinear function in (7) in a new adaptive Kalman filtering and smoothing framework. While this nonlinear function has been used for formant tracking in the past, our work generalizes earlier work in significant ways. The work of [3] was inspired by the same relationship between the VTR variables and the LPC cepstra as we have reported in this paper, but it used one single linear function to (very crudely) approximate the nonlinearity. This approximation was improved by the later work reported in [14], where a piecewise linear model was used which divides the entire frequency range of each formant into four bands. This gave a 4-piece piecewise linear approximation. The success of this extension adds support to the importance of dealing with nonlinearity in the analytical relationship between the formants and cepstra. Our work presented in this paper can be considered as a further generalization of the work of [3], [14] by using cepstrum-order dependent linearization. The number of linear "pieces" in the functional approximation varies from five to 75, designed according to detailed properties of the nonlinear function. In addition, the new, powerful computational framework of adaptive Kalman filtering and smoothing is used as the basis for the estimation, with direct incorporation of the hidden speech dynamics as the prior information. Such prior information was missing in all the earlier work.

Moreover, in many aspects, the new algorithm presented in this paper is also superior to our earlier algorithm [7], [10] designed based on discrete-valued hidden 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 empirical comparisons.

It is worth discussing some key properties of the hidden dynamic model presented in Section II-B which underlies our tracking technique. Since (7) is derived based on a low-order all-pole or auto-regressive (AR) model of the speech waveform, many consonants which have large and varied non-AR effects will create model inadequacy. Examples of such non-AR effects are pole-zero cancellation during fricatives and stop bursts, cancellation of F1 during aspiration, the changed relationship between the VTR bandwidth and amplitude caused by nasal zeros (as well as formant splits), and the extra spectral tilt caused by breathy voice during /h/. Our model assumes that all these non-AR effects are represented by a mean vector and zero-mean Gaussian noise, which seems implausible. We have empirically fixed this inadequate representation by using the adaptive mean vector as described in Section IV. Preliminary evaluation as shown in Table II demonstrates the effectiveness of this ad-hoc technique, and confirms that a reasonably large number of vectors are needed to represent the non-AR effects. An alternative proposal in [31] for dealing with the non-AR effects is to use an empirical scaling constant in the exponent of the AR model (exponentially weighted AR model). While this treatment of the relationship between cepstra and VTR is more desirable than our observation equation, the tracking results shown in [31] appear to be less accurate than our results based on spectrogram inspection. (It is not clear whether this difference in the results is due to the much simplified state equation in [31] or due to the approximations used to implement the tracking algorithm.) In any case, how to adequately represent the non-AR effectives in the type of the model presented in this paper is an interesting research direction.

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.

Acknowledgment

The authors would like to thank the anonymous reviewers who provided constructive comments that significantly improved the quality of this paper.

References

[1] I. Bazzi, A. Acero, and L. Deng, "An expectation-maximization approach for formant tracking using a parameter-free non-linear predictor," in *Proc. ICASSP*, 2003, pp. 464–467.

[2] C. Blackburn and S. Young, "Towards improved speech recognition using a speech production model," in *Proc. Eurospeech*, 1995, vol. 2, pp. 1623–1626.

[3] D. Broad and F. Clermont, "Formant estimation by linear transformation of the LPC cepstrum," *J. Acoust. Soc. Amer.*, vol. 86, pp. 2013–2017, 1989.

[4] L. Deng, "A dynamic, feature-based approach to the interface between phonology and phonetics for speech modeling and recognition," *Speech Commun.*, vol. 24, no. 4, pp. 299–323, 1998.

[5] L. Deng, "Computational models for speech production," in *Computational Models of Speech Pattern Processing*, K. Ponting, Ed. Berlin, Germany: Springer, 1999, pp. 199–213.

[6] L. Deng and D. O'Shaughnessy, *Speech Processing—A Dynamic and Optimization-Oriented Approach*. New York: Marcel Dekker, 2003.

[7] L. Deng, I. Bazzi, and A. Acero, "Tracking vocal tract resonances using an analytical nonlinear predictor and a target-guided temporal constraint," in *Proc. Eurospeech*, 2003, vol. I, pp. 73–76.

[8] L. Deng and J. Ma, "Spontaneous speech recognition using a statistical coarticulatory model for vocal-tract-resonance dynamics," *J. Acoust. Soc. Amer.*, vol. 108, pp. 3036–3048, 2000.

[9] L. Deng, L. J. Lee, H. Attias, and A. Acero, "A structured speech model with continuous hidden dynamics and prediction-residual training for tracking vocal tract resonances," in *Proc. ICASSP*, 2004, vol. I, pp. 557–560.

[10] L. Deng, A. Acero, and I. Bazzi, "Tracking vocal tract resonances using a quantized nonlinear function embedded in a temporal constraint," *IEEE Trans. Speech Audio Process.*, vol. 14, no. 2, pp. 425–434, Mar. 2006.

[11] S. Dusan and L. Deng, "Recovering vocal tract shapes from MFCC parameters," in *Proc. ICSLP*, 1998, pp. 3087–3090.

[12] J. Frankel and S. King, "ASR—Articulatory speech recognition," in *Proc. Eurospeech*, 2001, vol. 1, pp. 599–602.

[13] Y. Gao, R. Bakis, J. Huang, and B. Zhang, "Multistage coarticulation model combining articulatory, formant, and cepstral features," in *Proc. ICSLP*, 2000, vol. 1, pp. 25–28.

[14] J. Hogberg, “Prediction of formant frequencies from linear combinations of filterbank and cepstral coefficients,” KTH-STL Quarterly Progress Rep, Royal Inst. Technol. Stockholm, Sweden, 1997, pp. 41–49.

[15] W. Holmes, “Segmental HMMs: modeling dynamics and underlying structure in speech,” in *Mathematical Foundations of Speech Recognition and Processing, Volume X in IMA Volumes in Mathematics and Its Applications*, M. Ostendorf and S. Khudanpur, Eds. New York: Springer-Verlag, 2002.

[16] G. Kopec, “Formant tracking using hidden Markov models and vector quantization,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-34, no. 4, pp. 709–729, Aug. 1986.

[17] L. J. Lee, H. Attias, L. Deng, and P. Fieguth, “A multimodal variational approach to learning and inference in switching state space models,” in *Proc. ICASSP*, 2004, vol. V, pp. 505–508.

[18] S. McCandless, “An algorithm for automatic formant extraction using linear prediction spectra,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-22, no. 2, pp. 135–141, Apr. 1974.

[19] R. McGowan and A. Faber, “Speech production parameters for automatic speech recognition,” *J. Acoust. Soc. Amer.*, vol. 101, p. 28, 1997.

[20] M. Niranjan and I. Cox, “Recursive tracking of formants in speech signals,” in *Proc. ICASSP*, 1994, vol. II, pp. 205–208.

[21] L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[22] K. Richmond, S. King, and P. Taylor, “Modelling uncertainty in recovering articulation from acoustics,” *Comput. Speech Lang.*, vol. 17, pp. 153–172, 2003.

[23] G. Rigoll, “A new algorithm for estimation of formant trajectories directly from the speech signal based on an extended Kalman-filter,” in *Proc. ICASSP*, 1986, pp. 1229–1232.

[24] ——, “Formant tracking with quasilinearization,” in *Proc. ICASSP*, 1988, pp. 307–310.

[25] R. Rose, J. Schroeter, and M. Sondhi, “The potential role of speech production models in automatic speech recognition,” *J. Acoust. Soc. Amer.*, vol. 99, pp. 1699–1709, 1996.

[26] J. Schroeter and M. Sondhi, “Techniques for estimating vocal-tract shapes from the speech signal,” *IEEE Trans. Speech Audio Process.*, vol. 2, no. 1, pp. 133–150, Jan. 1994.

[27] 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. ICASSP*, 2003, pp. 748–751.

[28] J. Sun, L. Deng, and X. Jing, “Data-driven model construction for continuous speech recognition using overlapping articulatory features,” in *Proc. ICSLP*, 2000, vol. 1, pp. 437–440.

[29] D. Talkin, “Speech formant trajectory estimation using dynamic programming with modulated transition costs,” *J. Acoust. Soc. Amer.*, vol. S1, p. S55, 1987.

[30] L. Welling and H. Ney, “ Formant estimation for speech recognition,” *IEEE Trans. Speech Audio Process.*, vol. 6, no. 1, pp. 36–48, Jan. 1998.

[31] Y. Zheng and M. Hasegawa-Johnson, “Formant tracking by mixture state particle filter,” in *Proc. ICASSP*, 2004, vol. 1, pp. 565–568.

[32] P. Zolfaghari, S. Watanabe, A. Nakamura, and S. Katagiri, “Bayesian modelling of the speech spectrum using mixture of Gaussians,” in *Proc. ICASSP*, 2004, vol. 1, pp. 556–559.

Li Deng (M’86–SM’91–F’04) received the Ph.D. degree in electrical engineering from the University of Wisconsin, Madison, in 1986.

In 1989, he joined the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada, as an Assistant Professor, where he became Full Professor in 1996. From 1992 to 1993, he conducted sabbatical research at the Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, and from 1997 to 1998, at the ATR Interpreting Telecommunications Research Laboratories, Kyoto, Japan. In 1999, he joined Microsoft Research, Redmond, WA, as a Senior Researcher, where he is currently a Principal Researcher. He is also an Affiliate Professor in the Department of Electrical Engineering, University of Washington, Seattle. His research interests include automatic speech and speaker recognition, statistical methods and machine learning, neural information processing, machine intelligence, human speech production and perception, acoustic phonetics, auditory speech processing, noise robust speech processing, speech synthesis and enhancement, spoken language understanding systems, multimedia signal processing, and multimodal human–computer interaction. In these areas, he has published over 250 refereed papers in leading international conferences and journals, 12 book chapters, and has given keynotes, tutorials, and lectures worldwide. He has been granted over a dozen U.S. or international patents in acoustics, speech/language technology, and signal processing. He authored two books in speech processing.

Dr. Deng served on the Education Committee and the Speech Processing Technical Committee of the IEEE Signal Processing Society from 1996 to 2000, and was an Associate Editor for the IEEE TRANSACTIONS ON SPEECH AND AUDIO PROCESSING from 2002 to 2005. He currently serves on the Society’s Multimedia Signal Processing Technical Committee and editorial board of the IEEE *Signal Processing Magazine*. He was a Technical Chair of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP04), and is the General Chair of the IEEE Workshop on Multimedia Signal Processing (2006). He is a Fellow of the Acoustical Society of America.

Leo J. Lee (S’01–M’05) received the B.E. degree in electronic engineering from Tsinghua University, Beijing, China, in 1995 and the M.A.Sc. and Ph.D. degrees in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada in 1999 and 2004.

He is currently an NSERC Postdoctoral Fellow in the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON, Canada. He worked in the Speech Technology Group, Microsoft Research, Redmond, WA, during 2002 and 2003. His research interests include statistical signal processing, computational biology, graphical models, and machine learning.

Hagai Attias, photograph and biography not available at the time of publication.

Alex Acero (S’85–M’90–SM’00–F’04) received the M.S. degree from the Polytechnic University of Madrid, Madrid, Spain, in 1985, the M.S. degree from Rice University, Houston, TX, in 1987, and the Ph.D. degree from Carnegie Mellon University, Pittsburgh, PA, in 1990, all in electrical engineering.

He worked in Apple Computer’s Advanced Technology Group from 1990 to 1991. In 1992, he joined Telefonica I+D, Madrid, as Manager of the Speech Technology Group. In 1994, he joined Microsoft Research, Redmond, WA, where he became Senior Researcher in 1996 and Manager of the Speech Research Group in 2000. Since 2005, he has been Research Area Manager overseeing speech, natural language, communication, and collaboration. He is currently an affiliate Professor of electrical engineering at the University of Washington, Seattle. He is author of the books *Acoustical and Environmental Robustness in Automatic Speech Recognition* (Norwell, MA: Kluwer, 1993) and *Spoken Language Processing* (Englewood Cliffs, NJ: Prentice-Hall, 2001), has written invited chapters in three edited books and over 100 technical papers, and has given keynotes, tutorials, and other invited lectures worldwide. He holds 12 U.S. patents. His research interests include speech recognition, synthesis and enhancement, speech denoising, language modeling, spoken language systems, statistical methods and machine learning, multimedia signal processing, and multimodal human–computer interaction.

Dr. Acero served on the Speech Technical Committee of the IEEE Signal Processing Society from 1996 to 2002, chairing the committee from 2000 to 2002. He was Publications Chair of ICASSP’98, Sponsorship Chair of the 1999 IEEE Workshop on Automatic Speech Recognition and Understanding, and General Co-Chair of the 2001 IEEE Workshop on Automatic Speech Recognition and Understanding. He has served as Associate Editor for IEEE SIGNAL PROCESSING LETTERS and is presently Associate Editor for the IEEE TRANSACTIONS OF SPEECH AND AUDIO PROCESSING and member of the editorial board of *Computer Speech and Language*. He was member of the Board of Governors of the IEEE Signal Processing Society from 2003 to 2005. He is a 2006 Distinguished Lecturer of the IEEE Signal Processing Society.