# From HMM's to Segment Models: A Unified View of Stochastic Modeling for Speech Recognition

Mari Ostendorf, *Member, IEEE*, Vassilios V. Digalakis, and Owen A. Kimball, *Member, IEEE*

***Abstract*—In recent years, many alternative models have been proposed to address some of the shortcomings of the hidden Markov model (HMM), which is currently the most popular approach to speech recognition. In particular, a variety of models that could be broadly classified as segment models have been described for representing a variable-length sequence of observation vectors in speech recognition applications. Since there are many aspects in common between these approaches, including the general recognition and training problems, it is useful to consider them in a unified framework. Thus, the goal of this paper will be to describe a general stochastic model that encompasses most of the models proposed in the literature, pointing out similarities of the models in terms of correlation and parameter tying assumptions, and drawing analogies between segment models and HMM's. In addition, we summarize experimental results assessing different modeling assumptions and point out remaining open questions.**

## I. Introduction

TO date, the most successful speech recognition systems have been based on the hidden Markov model (HMM) [1], [2], and the use of HMM's for acoustic modeling dominates the continuous speech recognition field. Although HMM's will continue to play a role in most recognition systems for a long time to come, many alternative models have been proposed in recent years to address some of the shortcomings of HMM's. These new higher order models tend to require more computation than HMM's but with the increase in computational power and the broad use of progressive search techniques, they are viable and of interest for current systems. Unfortunately, the research on new models has tended to proceed in isolated pockets, and the proliferation of terms used to describe different modeling assumptions has made it difficult to appreciate the common themes across the various proposals. The goal of this paper is thus to bring together a variety of work under a common framework in order to make it easier for different researchers to benefit from the successes of others in developing robust estimation techniques and making appropriate assumptions about variable dependence and parameter tying.

Manuscript received June 20, 1995; revised February 22, 1996. This work was funded by ARPA and ONR under grant number ONR-N00014-92-J-1778, with additional support for M. Ostendorf provided by ATR. The associate editor coordinating the review of this paper and approving it for publication was Dr. Douglas D. O'Shaughnessy.

M. Ostendorf is with the Electrical, Computer, and Systems Engineering Department, Boston University, 44 Cummington St., Boston, MA 02215 USA.

V. Digalakis is with the Technical University of Crete, Hania, Greece.

O. A. Kimball is with the BBN HARK Systems Corp., Boston, MA, USA.

Publisher Item Identifier S 1063-6676(96)06718-1.

Broadly speaking, there are three HMM limitations that various models have tried to address: weak duration modeling, the assumption of conditional independence of observations given the state sequence, and the restrictions on feature extraction imposed by frame-based observations. The limitation that an HMM state duration model is implicitly given by a geometric distribution has been addressed by introducing models with explicit state duration distributions [3], [4]. Relaxation of the assumption of conditional independence of observations, which is widely recognized to be practically useful but unrealistic, has been the subject of several studies. A simple mechanism for capturing time dependence is to augment the observation space with feature derivatives. In addition, several variations of HMM's have been proposed to explicitly model correlation, including conditionally Gaussian HMM's [5]–[7] and "segmental" HMM's [8], [9]. Finally, the goal of using segmental rather than frame-based features, probably the initial motivating factor for development of "segmental" acoustic models, led to the work of Bush and Kopec [10] and Zue and colleagues [11], [12]. However, the stochastic modeling problem becomes more difficult when segmental or fixed-length features are used, requiring heuristic weightings and/or the use of posterior distributions. Excluding the posterior distribution models, we shall show that many of the proposed models are special cases of a more general segment model (SM),[1] which facilitates comparison of the different modeling assumptions.

The remainder of the paper is organized as follows. In Section II, we address the problem of modeling frame-based features, introducing the segment model as a generalization of an HMM. We describe stochastic segment models in general terms, giving recognition and training algorithms and showing differences with respect to the standard HMM algorithms for these problems. Next, in Section III, we discuss specific distribution assumptions that can be made to model the dynamics of feature vectors, show that many of the different models can be seen as special cases of a dynamical system model, and draw analogies to different HMM extensions. After treating frame-based features, in Section IV, we move to the problem of modeling fixed-length segmental features and discuss issues in the use of posterior distributions for segment modeling. Finally, Section V concludes with a discussion of several questions in segment modeling that are unresolved by current studies.

[1] We have avoided the term "stochastic segment model" (SSM), which we have used in much of our own work, to make clear that the term SM includes the work of others, although the modifier "stochastic" would still apply.

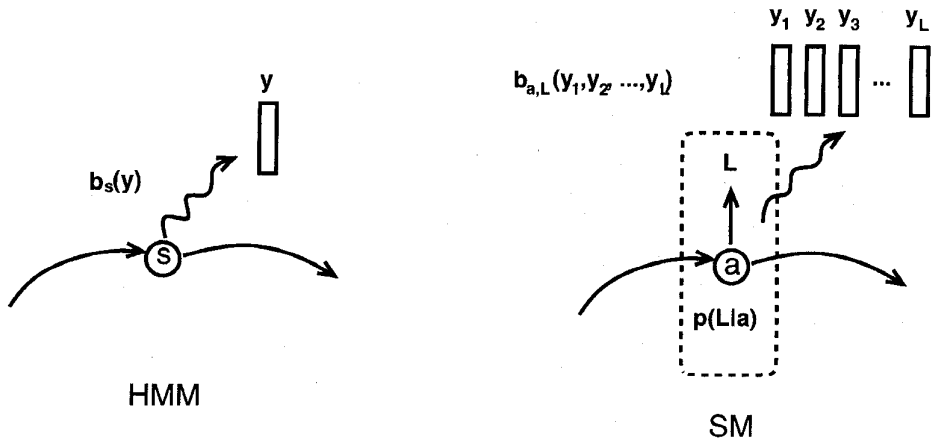



Fig. 1. HMM and an SM illustrated as generative processes: One frame $(y)$ is generated by an HMM state, and a variable-length sequence of frames $(y_1, \cdots, y_l)$ is generated by a SM "state" associated with random length $l$.

## II. Segmental and Hidden Markov Models

Taking a statistical approach, the general problem of recognizing a word sequence involves finding the sequence of labels $a_1^N = \{a_1, \cdots, a_N\}$ that is most likely given the sequence of $T$ $D$-dimensional feature vectors $y_1^T = \{y_1, \cdots, y_T\}$

$$\hat{a}_1^N = \arg\max_{N, a_1^N} \mathrm{p}(a_1^N | y_1^T) = \arg\max_{N, a_1^N} \mathrm{p}(a_1^N)\mathrm{p}(y_1^T | a_1^N)$$

where, for example, $a_i$ corresponds to a phone, and the recognized phone sequence is constrained to pronunciations in a lexicon. (Note that $a \in \mathcal{A}$ need not be a phone label; it could be any unit that can provide a mapping to a word sequence, e.g., a "triphone" or other linguistic unit conditioned on phonetic context or an automatically learned unit.) Using class-conditional distributions,[2] as in the equation above, we need a language model $\mathrm{p}(a_1^N)$ and an acoustic model $\mathrm{p}(y_1^T | a_1^N)$. Here, we focus on options for the acoustic model $\mathrm{p}(y_1^T | a_1^N)$.

In hidden Markov modeling, the fundamental observation distribution model is at the frame level: $b_s(y) = \mathrm{p}(y|s)$ for $s \in \mathcal{S}$, where $\mathcal{S}$ is the set of discrete HMM states, and a phone is typically represented by a sequence of states. In segment modeling, the fundamental distribution model $b_{a,l}(y_1^l) = \mathrm{p}(y_1^l | a, l)$ represents a segment $y_1^l = [y_1, \cdots, y_l]$, where $l$ is a random variable, and $a \in \mathcal{A}$, where $\mathcal{A}$ is the set of segment labels. (Without loss of generality, we assign time $t = 1$ to the first frame in the segment.) Fig. 1 illustrates this difference between the HMM and the SM from the perspective of generative models, i.e., generating a single observation versus a sequence of observations. From coding theory, we know that quantizing sequences of observations makes it possible to achieve lower distortion for a given bit rate than quantizing individual samples. If a segmental coding strategy is more efficient than a frame-based coding strategy, it seems reasonable to expect that a segmental recognition strategy will be more effective than a frame-based recognition strategy since minimum distortion is related to maximum likelihood (ML) in a Gaussian model.

[2]The terminology "class-conditional distribution" refers to the probability of observations given a class label, e.g., $\mathrm{p}(y|a)$, whereas "posterior distribution" refers to the probability of a class label given an observation, e.g., $\mathrm{p}(a|y)$.

In acoustic modeling, a "segment" might correspond to a phone-sized unit, but SM's have also been used to represent subphone units [13]–[15], diphones [16], and syllables [18]. We therefore use the term "segment" here in a more general sense than the typical linguistic association of "segment" with phonetic units. The unit size does not affect the probabilistic formalism, although it does have an impact on the computational costs of the model because of the greater length variability that must be accounted for in longer units. In both HMM's and SM's, the discrete "state" sequence $s_1^T$ and $(a, l)_1^N$, respectively, is typically modeled as a Markov chain.

With an HMM, there are several options for modeling the distribution $\mathrm{p}(y|s)$, including discrete distributions, full or diagonal covariance Gaussian densities, Gaussian mixtures, and Laplacian distributions, all of which have been used in speech recognition. Similarly, there are many possible distribution assumptions with SM's, in fact, many more options because of the large number of degrees of freedom in the model. However, in both cases, there are general recognition and training algorithms that can be described for all distribution assumptions. In this section, we describe the model at this general level for the particular case of class-conditional distributions.

### A. General Modeling Framework

A general segment model provides a joint model for a random-length sequence of observations $y_1^l = [y_1, \cdots, y_l]$, generated by unit $a$ according to the density

$$\mathrm{p}(y_1, \cdots, y_l, l|a) = \mathrm{p}(y_1, \cdots, y_l | l, a)\mathrm{p}(l|a) = b_{a,l}(y_1^l)\mathrm{p}(l|a). \tag{1}$$

Letting $\mathcal{L}$ be the set of possible observation lengths (in frames), a segment model for label $a \in \mathcal{A}$ is characterized by 1) a *duration distribution* $\mathrm{p}(l|a)$ that gives the likelihood of segment length $l \in \mathcal{L}$ and, thereby, the likelihood of a particular segmentation of an utterance and 2) a *family of output densities* $\{b_{a,l}(y_1^l); l \in \mathcal{L}\}$ that describes observation sequences of different lengths. In addition, a Markov assumption for sequences of $a_i$ is made either implicitly or explicitly by embedding

phone segments in a word pronunciation network or other probabilistic finite-state network.

Before elaborating on this abstract definition, let us consider two simple extensions of HMM's that fit this model. The simplest distribution assumption for a segment model uses a single output distribution and assumes that successive observed frames are independent and identically distributed within given segment boundaries. In this case, the probability of the segment given label $a$ and length $l$ is the product of the probability of each observation $y_i$

$$b_{a,l}(y_1^l) = \prod_{i=1}^{l} \mathrm{p}(y_i|a),$$

and the segment model reduces to a one-state HMM with an explicit duration model $\mathrm{p}(l|a)$, as opposed to the typical implicit geometric HMM duration model. (See Fig. 4 for this and other relationships between distribution assumptions.) This simple segment model is also known as a hidden "semi-Markov" model [3], as well as a continuously-variable duration HMM [4], [19], and a segment model [20]. By introducing an explicit state duration distribution, these models have the added complexity of hypothesizing segmentations in recognition and training. If we can accept this additional cost, then it is natural to move beyond the simple single-region segment model to more complex segment models since the benefit of an explicit length distribution is small relative to the gains possible with less restrictive distribution assumptions.

To make the model slightly more complex, we can use multiple distribution regions $r = 1, \cdots, R$ but still assume that observations are conditionally independent given the segment length, as in, e.g., [21] and [22]. In this case, the probability of a segment given label $a$ and duration $l$ becomes

$$b_{a,l}(y_1^l) = \prod_{i=1}^{l} \mathrm{p}(y_i|a, r_i) \tag{2}$$

where the specific distribution used for vector $y_i$ corresponds to region $r_i$. If the sequence of regions is constrained by some length-dependent mapping, this particular segment model can be thought of as a HMM with a complex topology (parallel paths for different lengths) and state parameter tying specified by the mapping to the distribution regions.

The segment model can be further generalized in a variety of ways. Here, we give the framework to represent a broad class of segment models, leaving more specific examples for Section III. The segment **duration distribution** $\{\mathrm{p}(l|a); l \in \mathcal{L}\}$ can be either parametric or nonparametric. Parametric models investigated have included the Poisson distribution [3], the Gamma distribution [4], a speaking-rate-normalized Gamma distribution [23], and context-dependent clustered Gamma models [24]. The nonparametric model simply uses smoothed relative frequencies, e.g., [22]. For phone-sized units, any reasonable assumption works well empirically, probably because the contribution of the duration model is small relative to the segment observation probability, which is in a much higher dimensional space. The **family of output densities** $\{b_{a,l}(y_1^l); l \in \mathcal{L}\}$ represents $l$-length trajectories in vector space ($y_i \in \Re^d$) with a sequence of distributions that can be thought of as dividing the segment into separate regions in time. Observations may be correlated within and across regions, but distribution parameters are time invariant within a region. In this sense, a segment distribution region is similar to an HMM state. A collection of distribution mappings (or time-warping transformations) $\{T_l(i); i = 1, \cdots, l; l \in \mathcal{L}\}$ associate each frame $y_i$ in the variable length observation $y_1^l$ with one of the model regions. Together, the mapping and the region-dependent distributions provide a means of specifying $b_{a,l}(y_1^l)$ for a large range of $l$ with a small number of parameters.

The mapping $T_l$ is a key component needed to specify the distribution family. $T_l$ can be deterministic or dynamic. Two variations of the *deterministic mapping* are possible: either a) to a fixed number of distributions using a table lookup or b) to a continuum of models determined by sampling a segment trajectory, as shown in Fig. 2. Trajectory sampling is more appealing for units that have smooth trajectories since it avoids assumption of piecewise constant dynamics. On the other hand, the constraint of a fixed number of distributions allows for automatic mapping estimation, as discussed later. A *dynamic mapping*, as used in [16] and [17], is implemented using dynamic programming to find the ML mapping to a fixed number of regions. If the distribution family is given by (2), then a segment model with an unconstrained dynamic mapping is equivalent to an HMM network (e.g., [25]), except for the explicit duration distribution. Deterministic mappings have the advantage of reduced computation relative to dynamic programming, and for phone-sized units and smaller, they work quite well in practice. In addition, there is evidence for systematic intrasegmental timing patterns in speech [26] that supports the use of a deterministic mapping, although not a simple linear mapping.

To further explain the segment model and to illustrate the relationship between segment models and HMM's, consider the problem of computing the probability of a phone sequence in continuous speech. A phone can be represented by a sequence of HMM states or by a segment model. In a HMM, the $T$-length observation sequence $y_1^T$ is connected to the $N$-length phone sequence $a_1^N$ via the state sequence $s_1^T$

$$\begin{aligned}\mathrm{p}(y_1^T|a_1^N) &= \sum_{s_1^T} \mathrm{p}(y_1^T, s_1^T|a_1^N) \\ &= \sum_{s_1^T} \mathrm{p}(y_1^T|s_1^T, a_1^N)\mathrm{p}(s_1^T|a_1^N),\end{aligned}$$

where

$$\mathrm{p}(y_1^T|s_1^T, a_1^N) = \prod_{t=1}^{T} \mathrm{p}(y_t|s_t) = \prod_{t=1}^{T} b_{s_t}(y_t), \tag{3}$$

$$\mathrm{p}(s_1^T|a_1^N) = I(s_1^T, a_1^N) \prod_{t=1}^{T} \mathrm{p}(s_t|s_{t-1}) \tag{4}$$

and $I(s_1^T, a_1^N)$ is an indicator function that equals one if the state sequence $s_1^T$ is permissible by the phone sequence $a_1^N$ and zero otherwise. The equations above require the usual HMM assumptions that an observation vector is conditionally independent from other observations and states given the current state (see (3)) and that the state sequence is Markov

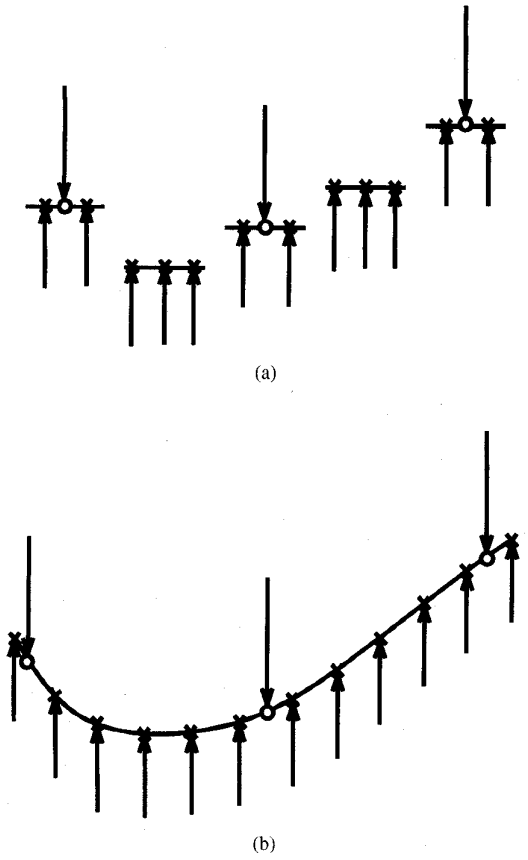


Fig. 2. Distribution mapping to (a) a fixed number of model regions (five here) versus (b) a continuum of distributions via trajectory sampling, illustrated in both cases for a linear-time warping with three-frame (o, down arrows) and 12-frame (×, up arrows) observations. The line represents the distribution mean as a function of time for 1-D observations.

(see (4)). Feature variability is captured by the frame-level observation distributions associated with each state $b_s(\cdot)$, and time variability is represented in the state sequence model $p(s_i|s_{i-1})$, which puts a geometric distribution on the time spent in each state.

Alternatively, in a segment model, we map to $a_1^N$ from $y_1^T$ via a segmentation. Since the segmentation can be uniquely specified by the sequence of segment lengths $l_1^N = \{l_1, \cdots, l_N\}$, then

$$\begin{aligned} \mathrm{p}(y_1^T|a_1^N) &= \sum_{l_1^N} \mathrm{p}(y_1^T, l_1^N|a_1^N) \\ &= \sum_{l_1^N} \mathrm{p}(y_1^T|l_1^N, a_1^N)\mathrm{p}(l_1^N|a_1^N), \end{aligned}$$

where

$$\begin{aligned} \mathrm{p}(y_1^T|l_1^N, a_1^N) &= \prod_{i=1}^{N} \mathrm{p}(y_{t(i-1)+1}^{t(i)}|l_i, a_i) \\ &= \prod_{i=1}^{N} b_{a_i,l_i}(y_{t(i-1)+1}^{t(i)}), \end{aligned} \tag{5}$$

$$\mathrm{p}(l_1^N|a_1^N) = \prod_{i=1}^{N} \mathrm{p}(l_i|a_i, l_{i-1}, a_{i-1}) \tag{6}$$

where $t(i)$ is the ending time of the $i$th segment, and $l_i = t(i) - t(i-1)$ is the segment length. The assumptions required to get (5) and (6) are the same as those used in (3) and (4), respectively, with the definition of state expanded to include the label-duration pair.[3] In the segment model, feature variability is represented by a more general probability distribution conditioned on the segment label and length $b_{a,l}(\cdot)$, and time variability is represented in the duration probabilities $p(l|a)$ and the length-dependent mapping $T_l$ that specifies the distribution sequence within a segment.

The same HMM versus SM comparison can be made for continuous word recognition, where a word is represented by a network of subword units. In this case, the HMM state sequence again serves as an intermediary between observations and words since the HMM states map uniquely to a word sequence. For the SM, the segmentation and segment labels form the intermediate stage between words and observations. In training and recognition problems, where the HMM state sequence or the SM segmentation is hidden, the impact of the difference in modeling assumptions is a larger state space for the SM, as we will see in the remainder of this section.

### *B. Recognition Algorithms*

The recognition algorithm for the segment model is similar to that used for HMM's, using dynamic programming to find the most likely "state" sequence (i.e., Viterbi decoding). In this section, we provide details about this algorithm together with some techniques for reducing computation that serve to counter the high cost of search with segment models.

*1) Viterbi Decoding:* The standard recognition solution for HMM's with a large state space (e.g., large vocabulary continuous speech recognition) involves finding the most likely state sequence

$$\hat{s}_1^T = \arg\max_{s_1^T} \mathrm{p}(y_1^T|s_1^T)\mathrm{p}(s_1^T)$$

via Viterbi decoding (dynamic programming) and then mapping the state sequence to the appropriate word sequence $\hat{w}_1^K = h(\hat{s}_1^T)$. Assuming that the state network used in decoding has been defined by a word/pronunciation network, the mapping then gives a unique word sequence.

For segment models, the solution is analogous, but in this case, the state includes both the segment label and duration. In other words, using the notation $q$ for a state in general, the segment state is $q = (a, l) \in \mathcal{A} \times \mathcal{L} = \mathcal{Q}$, whereas the HMM state is $q = s \in \mathcal{S} = \mathcal{Q}$. Segment-based recognition then involves finding

$$(\hat{N}, \hat{a}_1^{\hat{N}}) = \arg\max_{N, a_1^N} \{\max_{l_1^N} \mathrm{p}(y_1^T|l_1^N, a_1^N)\mathrm{p}(l_1^N|a_1^N)\mathrm{p}(a_1^N)\} \tag{7}$$

again using a dynamic programming algorithm and then mapping the segment label sequence to the appropriate word sequence $\hat{w}_1^K = f(\hat{a}_1^N)$, as in HMM decoding. The key difference between the SM and HMM search algorithms is

[3] In the segment model definition in (1), we make the stronger assumption that $\mathrm{p}(l_i|a_i, l_{i-1}, a_{i-1}) = \mathrm{p}(l_i|a_i)$ to simplify the presentation and reduce the number of free parameters.

the explicit evaluation of different segmentations, which adds an extra dimension to the dynamic programming search, as described below.[4]

Define $\mathcal{A}_t$ to be the set of segment models that are active at time $t$, which is determined by the word grammar, phoneme pronunciation networks, and optional search pruning. Define $\rho(t,j)$ to be the set of allowable segment boundaries for a segment with label $j$ ending at time $t$, which is determined by the utterance length and constraints on allowable phone duration time. Finally, define $\delta_t(i)$ to be the log probability of the most likely segmentation/label sequence ending with segment label $i$ for observations $y_1^t = \{y_1, \cdots, y_t\}$

$$\delta_t(i) = \max_{n, l_1^n, a_1^{n-1}} \log \mathrm{p}(y_1^t, l_1^n, a_1^{n-1}, a_n = i).$$

The traceback information is stored in $\psi_t(i)$, which contains the ending time and label of the best previous segment that led to $\delta_t(i)$. Then, the dynamic programming SM recognition algorithm involves the following:

---

**Initialize:** $t = 1$

$\delta_1(i) = \log \mathrm{p}(y_1|l,i)\mathrm{p}(l|i)p(i) \quad \forall i \in \mathcal{A}_1, l = 1$

**Iterate:** $t = 2, \cdots, T; \forall i \in \mathcal{A}_t, l_\tau = t - \tau$

$\delta_t(i) = \max_{j \in \mathcal{A}_\tau, \tau \in \rho(t,i)} \delta_\tau(j) +$
$\log[\mathrm{p}(y_{\tau+1}, \cdots, y_t|l_\tau, i)\mathrm{p}(l_\tau|i)\mathrm{p}(i|j)]$

$\psi_t(i) = \arg\max_{j \in \mathcal{A}_\tau, \tau \in \rho(t,i)} \delta_\tau(j) +$
$\log[\mathrm{p}(y_{\tau+1}, \cdots, y_t|l_\tau, i)\mathrm{p}(l_\tau|i)\mathrm{p}(i|j)]$

**Traceback:**

$\hat{a}_T = \arg\max_{i \in \mathcal{A}_T} \delta_T(i), \quad t = T$

Iterate while $t > 0$: 1) $(\hat{a}_{t'}, t') = \psi_t(\hat{a}_t)$, 2) $t \leftarrow t'$

---

Note that the recognized segment labels $\hat{a}_t$ are indexed by their ending time since the total number of recognized segments is not known until traceback is finished.

Ignoring the effect of pruning, which can be implemented for both HMM's and SM's, the state space for the segment model is determined by the product of the model set size $|\mathcal{A}|$ and the number of allowable segment duration start times, which is roughly the maximum allowable segment duration $L_{\max}$ (e.g., $L_{\max} = 60$ for read speech and a 10ms frame rate). The comparable state space for HMM's is $|\mathcal{S}|$, which is typically only 3–5 times $|\mathcal{A}|$. On top of this difference is the possibly higher cost of SM probability evaluations if one of the models from Section III is used. As a consequence, several SM cost reduction techniques have been considered. For the case where the segment model uses an assumption of conditional independence, it is possible to use distribution score caching with the SM to eliminate redundant Gaussian computations. The resulting SM will then have computational requirements comparable with the analogous HMM, plus an additional (nontrivial) cost associated with the overhead of tracking segment structure. Segment-level score caching is also useful, especially for more general distribution assumptions. A second approach to reducing computation is segment "pruning" or eliminating unlikely phone candidates based on partial segment likelihoods. In one experiment, where candidates are pruned according to a threshold based on the contributions to the likelihood of successive frames, a 75% reduction in computation was obtained with no loss in recognition accuracy [28].

*2) Reduced Search Spaces:* Although it is useful to reduce the cost of segment evaluations, it is perhaps more important to reduce the number of segment evaluations by reducing the search space. There are two basic strategies for shrinking the search space: 1) reducing the set of segmentations considered and 2) SM rescoring in a multipass search framework.

For an utterance of length $T$, there are approximately $2^T$ segmentations that must be considered in an optimal search. Reducing the size of this set can significantly reduce segment modeling search costs. Initial work in this area [11] involved using hierarchical clustering of frames using a similarity measure that resulted in a dendrogram representation of a set of possible segmentations. This reduced set of segmentations is then searched with a dynamic programming algorithm, where the dendrogram specifies the segmentation constraints $\{\rho(t,i)\}$. A different strategy is the local search algorithm proposed in [28], which starts with an initial segmentation and iteratively adjusts segmentation times and segment labels to most improve overall likelihood at each step. Possible adjustments—e.g., splits, merges, and combination split-merge moves—define the local neighborhood searched; therefore, the possible segmentation set is reduced but determined dynamically as part of recognition.

Eliminating segmentations from the search space may introduce errors early in the search process if the ultimate goal is word recognition (or some other higher level unit); therefore, alternative approaches have also been investigated. In particular, one can use an HMM (i.e., a simpler model) to provide a set of sentence hypotheses that are subsequently rescored by a segment model. The sentence hypotheses can be described as an $N$-best list or a word lattice. In either case, the recognition algorithm is the same as that given in Section II-B-1, but the set of segment labels $\mathcal{A}_t$ to evaluate at each time $t$ is reduced. Rescoring time can be further reduced (by a factor of 30 without loss in performance) if the HMM segmentation times are available, in which case, $\{\rho(t,i)\}$ is given by a window of times around the corresponding HMM start time for phone $i$ ending within some window of $t$. Together, rescoring and segmentation time constraints make large vocabulary recognition with segment models feasible, but computation can be further reduced via rescoring a lattice with time constraints rather than the equivalent $N$-best list [29] (by a factor of 3–10, depending on the size of the $N$-best list). Of course, all methods of reducing the search space introduce errors, and it is an empirical question as to which approach is more effective. In our experience, HMM hypothesis rescoring has been most effective for word recognition in terms of both accuracy and computation reduction.

An additional advantage of the rescoring framework is that it provides a simple mechanism for combining knowledge sources. In $N$-best rescoring [30]–[32], each knowledge

[4]The algorithm described here is similar to that in [27], except that here, we include the duration likelihood, the Markov label assumption, and pruning notation and do not include the heuristic length penalty introduced to handle fixed-length features.

source separately scores each hypothesis, the scores are linearly combined, and the hypotheses are reranked based on the combined score. The weights used in the score combination can be estimated automatically given $N$-best hypotheses and an optimization criterion such as minimum word error rate in the top ranking sentence. Weight estimation is an unconstrained multidimensional optimization problem, which can be solved using gradient techniques such as Powell's method, as in [30], or by a grid-based search that chooses among different local optima [33].

### C. Parameter Estimation Algorithms

The hidden state component that is common to all the acoustic models presented in Section II–A complicates parameter estimation, requiring some form of iterative algorithm for ML estimation. In this section, we will first present generalizations of the two most common iterative schemes used in speech recognition, which are applicable for either segmental or HMM's. One estimates the conditional probability distribution of the hidden state sequence and is an instance of the expectation-maximization (EM) algorithm [34]. The other finds the most likely hidden state sequence at each iteration and is sometimes called "Viterbi training," where the segmental $k$-means algorithm in the HMM case [35] is an example.

Once a solution to the hidden-state problem is provided, the second step of the iterative algorithm—which is estimation of the state transition probabilities and "output" distribution parameters (either segmental or frame-based)—is typically straightforward, with the exception of some complex models of segmental dynamics. We discuss the problem of estimating the distribution mapping in the latter part of this section, further discuss parameter estimation issues in Section III when introducing specific distribution assumptions, and reserve the detailed parameter update equations for Appendix A.

*1) Discrete State Estimation—The Generalized Forward-Backward Algorithm:* The EM algorithm was applied to HMM's by Baum and colleagues [36] and is now a standard estimation tool in speech recognition. For many of the new variations of HMM's that have been proposed, extensions of the Baum–Welch algorithm have been derived, including continuously variable duration HMM's [4], segmental HMM's [37], and conditionally Gaussian HMM's [5]–[7]. Here, we give a solution for a more general notion of a discrete hidden state that handles all of these models, using what we shall refer to as the *generalized forward-backward* algorithm in the expectation (E) step as an extension of the so-called HMM forward-backward algorithm to compute the posterior probability of a state given the observed data.

Following the approach in [1] but with a general unobserved state $q \in \mathcal{Q}$, the ML estimate for model parameters $\theta$ are obtained by maximizing the marginal distribution

$$\mathrm{p}_\theta(y_1^T) = \sum_{q_1^M \in h^{-1}[w_1^K]} \mathrm{p}(q_1^M, y_1^T)$$

where the summation is over all admissible discrete state sequences $q_1^M \in h^{-1}[w_1^K]$ for the given "transcription" of the training data $w_1^K$. The length $M$ term in $q_1^M$ is used to indicate that the state sequence is not necessarily at the same time scale as the observation sequence, allowing variable-length observations. The solution to this maximization problem can be obtained by the EM algorithm, treating $q_1^M$ as missing data and maximizing at each iteration

$$E_\theta\{\log \mathrm{p}_{\theta'}(q_1^M, y_1^T)|y_1^T\} = \sum_{M, q_1^M} \mathrm{p}_\theta(q_1^M|y_1^T) \log \mathrm{p}_{\theta'}(q_1^M, y_1^T) \tag{8}$$

with respect to parameters $\theta'$ given parameters $\theta$ from the previous iteration. If the discrete state sequence has the Markov property, then the terms needed for maximizing (8) are

$$\mathrm{p}_\theta(q_t = q, y_1^T) = \sum_{M, q_1^M \in Q_t^1} \mathrm{p}(q_1^M, y_1^T)$$

$$\mathrm{p}_\theta(q_t = q, q_{p(t)} = q', y_1^T) = \sum_{M, q_1^M \in Q_t^2} \mathrm{p}(q_1^M, y_1^T)$$

where $q_t$ is aligned with time $t$ (e.g., ending at time $t$), $q_{p(t)}$ is the state preceding $q_t$, $Q_t^1(q) = \{q_1^M : q_t = q\}$ is the set of all state sequences that pass through state $q$ at time $t$, and $Q_t^2(q, q') = \{q_1^M : q_t = q, q_{p(t)} = q'\}$ is similarly defined but constrains a state transition at time $t$. (Note that many state sequences will not have a segment ending at time $t$, in which case, $q_t$ has a null value.) These probabilities can be computed efficiently with an extension of the forward-backward algorithm, as shown below for $\mathrm{p}(q_t, y_1^T)$, where the subscript $\theta$ is dropped to simplify the notation. The recursions for the second probability $\mathrm{p}(q_t, q_{p(t)}, y_1^T)$ can be derived similarly and are omitted for brevity.

Let $q_t^-$ and $q_t^+$ be the state sequences before and after the state $q_t$, and let $Y(\cdot)$ represent the contiguous observation sequence that corresponds to a particular state or sequence of states. Thus, $Y(q_t)$ represents the observation sequence associated with state $q_t$. Nonoverlapping state sequences are assumed to have nonoverlapping observation sequences. The state likelihood is given by

$$\begin{aligned}\mathrm{p}(q_t = q, y_1^T) &= \mathrm{p}(q_t = q, Y(q_t^-, q_t, q_t^+)) \\ &= \mathrm{p}(q_t = q, Y(q_t^-, q_t), Y(q_t^+)) \\ &= \mathrm{p}(q_t = q, Y(q_t^-, q_t))Y(q_t^+)|q_t \\ &= q, Y(q_t^-, q_t)) = \alpha_t^G(q)\beta_t^G(q)\end{aligned}$$

where $\alpha_t^G(q)$ and $\beta_t^G(q)$ ($G$ for general) are calculated using the recursive "generalized" forward and backward algorithms given below. Since the label sequence is not associated with the same time scale as the observations, we denote previous and next states by $q_{p(t)}$ and $q_{n(t)}$, respectively, and their corresponding times at the observation level as $t_p$ and $t_n$. (Note that although $q_1^M$ is Markov, the sequence $\{q_t; t = 1, \cdots, T\}$ is not Markov, hence, the "semi-Markov" terminology in [3].)

$$\begin{aligned}\alpha_t^G(q) &= \mathrm{p}(q_t = q, Y(q_t^-), Y(q_t)) \\ &= \sum_{q'} \mathrm{p}(q_t = q, q_{p(t)} = q', Y(q_t^-), Y(q_t)) \\ &= \sum_{q'} \mathrm{p}(Y(q_t)|q_t = q, q_{p(t)} = q', Y(q_t^-))\end{aligned}$$

$$\cdot \mathrm{p}(q_t = q | q_{p(t)} = q', Y(q_t^-)) \mathrm{p}(q_{p(t)} = q', Y(q_t^-))$$
$$= \sum_{q'} \mathrm{p}(Y(q_t) | q_t = q, Y(q_{p(t)})) \mathrm{p}(q_t = q | q_{p(t)} = q') \cdot \alpha_{t_p}^G(q') \quad (9)$$

$$\beta_t^G(q) = \mathrm{p}(Y(q_t^+) | q_t = q, Y(q_t^-, q_t))$$
$$= \sum_{q'} \mathrm{p}(Y(q_{n(t)}^+), Y(q_{n(t)}), q_{n(t)} = q' | q_t = q \; Y(q_t^-, q_t))$$
$$= \sum_{q'} \mathrm{p}(Y(q_{n(t)}^+) | q_{n(t)} = q', q_t = q, Y(q_t^-, q_t), \; Y(q_{n(t)})) \cdot \mathrm{p}(Y(q_{n(t)}) | q_{n(t)} = q', q_t = q, Y(q_t^-, q_t)) \cdot \mathrm{p}(q_{n(t)} = q' | q_t = q, Y(q_t^-, q_t))$$
$$= \sum_{q'} \beta_{t_n}^G(q') \mathrm{p}(Y(q_{n(t)}) | q_{n(t)} = q', Y(q_t)) \cdot \mathrm{p}(q_{n(t)} = q' | q_t = q). \quad (10)$$

In the last step of both derivations, we use the assumptions that 1) the general observations $Y(q)$ are conditionally Markov given the current state, and 2) the label sequence is Markov.

These equations easily simplify to the standard HMM results by letting $q_t = s \in \mathcal{S}, Y(q_t) = y_t, t_p = t-1$, and so forth. For the segment model, $q_t = (a, l) \in \mathcal{A} \times \mathcal{L}, Y(q_t) = [y_{t-l+1}, \cdots, y_t], t_p = t-l$, etc., and the state likelihood $\mathrm{p}(q_t = (a, l), y_1^T)$ is interpreted as the probability that a segment of label $a$ and length $l$ ends at time $t$. The term $\alpha_t^S(a, l)$ (where $S$ stands for segment model) corresponds to the probability of the partial observation sequence $y_1^t$ with a segment of label $a$ and length $l$ ending at time $t$. Similarly, $\beta_t^S(a, l)$ corresponds to the probability of the partial observation sequence $y_{t+1}^T$, given that a segment of label $a$ and length $l$ ends at time $t$ and the preceding observations $y_1^t$. The terms $\alpha_t^S(a, l)$ and $\beta_t^S(a, l)$ are calculated using recursive forward and backward algorithms:

$$\alpha_t^S(a, l) = \mathrm{p}(a_t = a, l_t = l, y_1^t)$$
$$= \sum_{a'} \sum_{l'} \mathrm{p}(y_{t-l+1}, \cdots, y_t | a, l) \mathrm{p}(a, l | a', l') \cdot \alpha_{t-l}^S(a', l') \quad (11)$$

$$\beta_t^S(a, l) = \mathrm{p}(y_{t+1}^T | a_t = a, l_t = l, y_1^t)$$
$$= \sum_{a'} \sum_{l'} \beta_{t+l'}^S(a', l') \mathrm{p}(y_{t+1}, \cdots, y_{t+l'} | a', l') \cdot \mathrm{p}(a', l' | a, l) \quad (12)$$

where we have made the additional (but unnecessary) assumption that segment observations are conditionally independent given the segment label sequence

$$\mathrm{p}(Y(q_t) | q_t = q, Y(q_{p(t)})) = \mathrm{p}(Y(q_t) | q_t = q).$$

(The more general form allows for conditionally Gaussian models across segments.) In addition, (11) and (12) simplify further if we include the earlier assumptions that segment lengths depend only on the current segment label, and segment labels are Markov

$$\mathrm{p}(a, l | a', l') = \mathrm{p}(l | a) \mathrm{p}(a | a')$$

and are then equivalent to those in [37].

*2) Discrete State Estimation: the Most Likely State Sequence:* An alternative approach is to jointly estimate the *most-likely-state-sequence* (MLSS) $\hat{q}_1^M$ and the model parameters $\theta$, thereby maximizing the likelihood

$$\mathrm{p}_\theta(y_1^T, \hat{q}_1^M) = \max_{q_1^M \in h^{-1}[w_1^K]} \mathrm{p}(y_1^T, q_1^M)$$

over $\theta$. This maximization can be performed by alternating between the following two steps:

1) Find the most likely state sequence given the current parameter estimates.
2) Re-estimate the model parameters $\theta$ using the newly obtained state sequence.

To find the most likely state sequence in the case of segment models, we must find the most likely segment label sequence and segmentation (or sequence of segment lengths) jointly. As in recognition, we use a dynamic programming algorithm, which differs from that described in Section II-B-1 in that the possible label sequence is constrained according to the known word sequence, effectively reducing the sets of active segment labels $\{\mathcal{A}_t\}$. Describing the algorithm in terms of the dynamic programming algorithm in Section II-B-1 represents a slight generalization of that described in [27] since we do not constrain the segmentation to a specific phone sequence but allow for alternate pronunciations for words when such are available from the recognition lexicon. For the parameter re-estimation step, we can unify the treatment of the EM and the MLSS algorithms by using degenerate conditional distributions of the discrete state sequence, given the current model parameters and the observations.

This MLSS re-estimation procedure has been applied to both HMM's [35] and segmental models [27]. For HMM's, it has been shown that under certain conditions, the MLSS procedure will yield asymptotically identical results with the Baum–Welch algorithm [38]. Although this has been debated, in practice, the MLSS procedure provides satisfactory estimates when good initial estimates and enough training data are used. In particular, it provides a practical alternative for SM training, for which the generalized forward-backward algorithm can be very costly. Starting from an HMM segmentation, segment model training requires only a few iterations of MLSS training.

*3) Robust Parameter Estimation:* The main difficulty in modeling context for segment recognition is that the SM has a large number of free parameters, which requires a significant amount of training. In representing context, the number of models increases, and therefore, the effective amount of training per model is reduced. In addition, the interesting SM distribution assumptions are often not amenable to simple smoothing techniques, such as co-occurrence smoothing used in discrete distribution HMM's [39] or variance clipping [40] and Bayesian smoothing [41] used in continuous distribution HMM's. An alternative solution is parameter tying, i.e., assuming that some model parameters are shared across models and/or regions. Parameters can be tied based on heuristic rules using knowledge of the application, as in [42] and [43], or can be determined automatically through distribution clustering.

Parameter tying via distribution clustering has been used successfully in both segment modeling [22], [44] and extensively in hidden Markov modeling (e.g., [45]–[48]). A general approach used in many recognition systems is divisive clustering to maximize the likelihood of the training data represented by the clustered models (or alternatively to minimize entropy). The algorithm uses a greedy search to successively add models through binary splits of subsets of the data as in decision tree design [49]. Using an ML criterion, each possible split is evaluated in terms of the likelihood ratio of one versus two distributions for representing the data at that node of the tree. Distribution clustering is used with the MLSS training algorithm described above in the parameter re-estimation step of the algorithm or as an intermediate step, which uses MLSS segmentation information to design model topology but is followed by EM re-estimation [47], [50].

Examples of ML node evaluation functions are given in [44] for different cases of parameter tying with frame-level Gaussian distributions, i.e., assuming that observations are conditionally independent given the state/region sequence. In principle, the algorithm can be extended to any distribution assumption that assumes conditional independence across but not necessarily within regions, but the node evaluation functions may be costly for complex distribution assumptions. As a consequence, some combination of heuristics, clustering, and experimentation will probably be needed to solve the parameter tying problem for segment models.

*4) Distribution Mapping Estimation:* If $T_l(i)$ is a deterministic function that provides the distribution regions for the $l$-long observation (typically frames $i \in \{1, \cdots, l\}$), then it must be defined somehow. The mapping can be chosen heuristically for both the fixed region and trajectory sampling approaches or automatically for the fixed region approach where $T_l(j) \in \{1, \cdots, R\}$, where $R$ is the number of regions. Heuristics that have been used successfully include linear time warping for phones (e.g., [21], [27]), linear sampling of the cepstral vector trajectory for phones [27], and functions of consonant-vowel structure for syllables [43]. However, there is evidence that intraphone timing, although systematic, is nonlinear [26]; therefore, better performance may be obtained by deriving the mapping automatically. Here, two approaches are outlined, which could in principle be combined: divisive distribution clustering in the temporal domain and trajectory estimation, both based on a ML criterion.

The first approach uses ML distribution clustering as described above but in the temporal domain. Starting with one region per segment, data is successively partitioned to add regions in the segment model where they most increase the overall training likelihood, using questions like "is $i/l < \gamma$?" or "is $l < \gamma$?" (where $\gamma$ is some threshold learned in clustering). The resulting mapping is model-independent with a constant number of regions per model, if clustering is based on the statistics of all models. Alternatively, clustering can be phone dependent, in which case, different phones can be assigned different numbers of regions. Temporal clustering can also be combined with distribution clustering to define parameter sharing over different triphones, resulting in an overall algorithm very similar to successive state splitting [50].

The second approach assumes a known number of regions for each model and again can be used for general or model-dependent warpings. The algorithm finds $T_l$ separately for each length $l \in \mathcal{L}$ (and optionally for each group of models $\mathcal{A}_k \subset A$) such that

$$T_l^* = \arg\max_{T_l} \sum_{a \in \mathcal{A}_k} \sum_{Y_i \in \mathcal{Y}_{a,l}} \log \mathrm{p}(Y_i | a, T_l)$$

$$= \arg\max_{T_l} \sum_{a \in \mathcal{A}_k} \sum_{Y_i \in \mathcal{Y}_{a,l}} \sum_{j=1}^{l} \log \mathrm{p}(y_j^i | a, T_l(j)) \quad (13)$$

where $\mathcal{Y}_{a,l} = \{Y_i : a_i = a, l_i = l\}$, and $y_j^i$ is the $j$th feature vector in segment $Y_i$. Equation (13) can be maximized using dynamic programming (e.g., [51]), assuming that the mapping is constrained to be monotonic in model indices, i.e., $T_l(j) T_l(j+1)$, and successive frames are either conditionally independent or Markov given length $l$. Given the new warping $T_l^*$, the distribution parameters are then re-estimated, and the process can be iterated within step (2) of the MLSS algorithm. The initial warping might be a simple heuristic or one obtained by divisive clustering. A disadvantage of this algorithm is that it has no generalization mechanism for unobserved lengths $l$.

## III. Models of Feature Dynamics

Since an HMM is a special case of a segment model, the segment model is capable of achieving at least the same level of performance as an HMM, and experiments have shown that performance is similar for equivalent distribution assumptions and numbers of free parameters [52]. However, the segment model allows for more general families of distributions than with an HMM, particularly distributions that implicitly or explicitly model feature dynamics. There are many possible distribution assumptions that can represent feature dynamics, each with advantages and disadvantages that must be weighed experimentally. In this section, we outline several different alternatives, including constrained mean, Gauss–Markov, and more general linear models, as well as segmental mixture models. For each case, we describe analogous HMM assumptions. Some insights into experimental tradeoffs are provided, but the space of parameterizations has not been explored enough to draw strong conclusions about the relative advantages of the different assumptions.

### A. Constrained Mean Trajectory

The simplest distribution assumption is given by (2), where a segment model is characterized by distribution regions, and frame-based observations are assumed to be conditionally independent, given the region sequence (or the state sequence,for an HMM). Unlike an HMM, the region sequence within a segment can be constrained by a deterministic distribution mapping $T_l$, hence the term "constrained mean." Two issues determine the particular type of constrained-mean trajectory model: whether the distribution mapping $T_l$ is a trajectory sampling function or an indexing function to a fixed set of regions and whether the mean trajectory is parametric or

nonparametric.[5] By parametric, we mean that the mean is specified by a constant, linear, or higher order polynomial trajectory, and distributions for specific regions correspond to points along the trajectory. Nonparametric trajectory models, on the other hand, have distribution parameters that are separately estimated for each model region. While both parametric and nonparametric models can use either type of deterministic distribution mapping, it is most common for the parametric trajectory to be used with trajectory sampling and the nonparametric trajectory to be used with the fixed set of region-dependent models.

The first frame-based stochastic segment models were nonparametric, i.e., [27] for fixed-length observations and [21] for variable-length observations. Further work at Boston University continued in this vein, exploring different frame-level distribution and parameter tying assumptions with deterministic distribution mappings, e.g., [42], [54], [44], yielding performance comparable with many state-of-the-art HMM systems in an unlimited vocabulary dictation task (e.g., 10.0–11.5% word accuracy on the 1994 ARPA benchmarks [55]). Nonparametric trajectory models using dynamic mappings include [16] and [17]. Because time correlation can be captured implicitly through the use of derivative features and because robust parameter estimation is easier at the frame level, this model has been difficult to improve upon in terms of performance. Since the distributions across a segment are not constrained in parameter estimation, this is also the least explicit model of feature dynamics.

Parametric trajectory segment models were introduced separately by Gish and Ng [56] (as a segment model) and Deng *et al.* [15] (as a nonstationary-state HMM).[6] In both cases, the mean trajectory is parameterized by a polynomial in $D$-dimensional vector space, and frame-level observations are assumed to be conditionally independent given the segment length. Specifically, the sequence of distributions used in computing the likelihood of an $l$-length segment $y_1^l$ is described by the sequence of means $[\mu_1 \cdots \mu_l] = BZ_l$, where $B$ is a $D \times (m+1)$ matrix of coefficients for polynomial order $m$, and $Z_l$ is an $(m+1) \times l$ time sampling matrix. For $m = 0$, $Z_l$ is a row vector of 1's; for $m = 1$, $Z_l$ has a vector of 1's in the first row and a vector of normalized times $\tilde{t}$ in the second row. As an example, the value of the $i$th component of the mean vector at normalized time $\tilde{t}$ is given in the quadratic case $(m = 2)$ by $\mu_{ti} = b_{i1} + b_{i2}\tilde{t} + b_{i3}\tilde{t}^2$, where $b_{ij}$ is an element of the coefficient matrix $B$. The two approaches differ in the representation of time. Gish and Ng define the observed segment to be a linear sampling of a complete trajectory; therefore, $\tilde{t} \in [0, 1]$, as illustrated in Fig. 2(b) for a scalar trajectory. Deng *et al.* use absolute time; therefore, $\tilde{t} \propto j$ for the $j$th frame in the segment, i.e., the trajectory varies with segment length. Using absolute time has the advantage of efficient recognition and segmentation algorithms since the Markov assumption holds within and across segments, but it is only reasonable for subphonetic units (a phone of length $l$ generally does not correspond to the first half of a phone of length $2l$).[7] Both approaches reduce training costs by taking advantage of an assumption that the covariance is identical for all frames in the segment, although this assumption may have associated tradeoffs in speech recognition performance. (In nonparametric trajectory modeling, we find that covariance determinants vary as a function of the region in a phone, i.e., there is more variation at the beginning and end of a phone than in the middle.) In experiments on the TIMIT corpus comparing constant, linear, and quadratic mean functions, the different researchers both find error rate reduction with higher order models (approximately 10% for vowel classification [56] and 20% for phone recognition [59]), and both find that only a subset of sounds require quadratic trajectories (i.e., diphthongs [56] or transitional subphonetic units defined by articulatory features [59]).

The parametric and nonparametric approaches each have their respective advantages, and parameter estimation equations for both are given in the Appendix. The parametric approach is well motivated by the smooth trajectories in many speech units, assuming that units that do not vary smoothly in time (e.g., stop consonants) are represented by multiple segments (or "states"). The nonparametric approach has computational (and/or storage) advantages since distribution means can be stored in a small table and score caching can be used for reducing computation. Parametric models tend to have fewer parameters than nonparametric models, but nonparametric models may be better suited to parameter tying (which has been most successful at the subsegment level) and distribution mapping estimation. Further research is needed to assess the relative benefits.

### B. Conditionally Gaussian Models

After conditional independence of observations, the next simplest distribution assumption is the Markov property. For Gaussian distributions,[8] this corresponds to a Gauss–Markov assumption within and optionally across segment regions or HMM states, e.g., for segments

$$b_{a,l}(y_1^l) = \prod_{t=1}^{l} \mathrm{p}(y_t | y_{t-1}, a, r_t). \tag{14}$$

Researchers have long observed that the HMM assumption of conditional independence is not valid and have investigated alternative assumptions. Early work with Markov assumptions, referred to here as conditionally Gaussian HMM's,[9] was due to Wellekens [5], who described extensions to the Viterbi and

[5] The parametric versus nonparametric terminology is borrowed from Goldenthal and Glass [53], although we classify their nonparametric model with those described in Section IV that use fixed-length features.

[6] A parametric trajectory model is also proposed by Krishnan and Rao [57], but they represent probabilities of regression terms rather than observations, and thus, their model fits with those in Section IV.

[7] This deficiency of the model has been addressed somewhat by a state-dependent time scaling term in [58], although normalized time is still proportional to the frame time.

[8] Discrete observation Markov assumptions are explored in [60].

[9] We have used the term "conditionally Gaussian HMM" to distinguish between these models and autoregressive (or hidden filter) HMM's [61], [62]. The autoregressive model represents conditional dependence within a fixed-dimensional vector of waveform samples using scalar linear prediction. The conditionally Gaussian HMM represents conditional dependence across vectors in a variable-length sequence, using vector linear prediction.

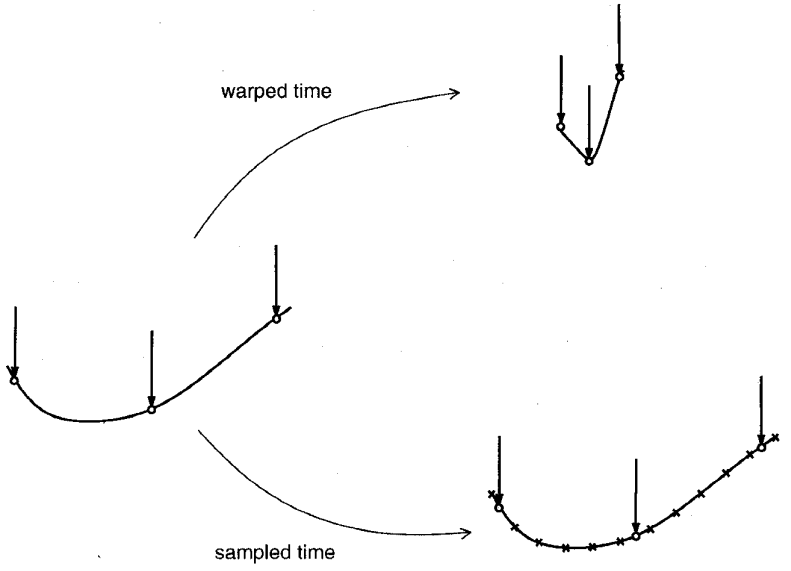



Fig. 3. Illustration of two possible correlation assumptions: time warping (correlation between successive observations) versus time sampling (correlation between hidden regions).

Baum–Welch algorithms for this case, and Brown, [6] who also explored these models experimentally.

Conditionally Gaussian models were rediscovered by Kenny *et al.* [7], who (like Brown) found a benefit to the conditionally Gaussian model for simple cepstral features but not for features augmented with derivatives. The analogous (Gauss–Markov) assumption for segment models was explored by Digalakis *et al.* [52] with similar conclusions. Deng and colleagues extended the parametric trajectory model [15] to the Gauss–Markov case [63], providing the first positive results for the Gauss–Markov assumption with cepstral derivatives. The 10–40% reduction in error rate (smaller as the number of states increased) obtained on a speaker-dependent CVC syllable recognition task was not simply due to the addition of the parametric trajectory mean; the combination of the parametric trajectory mean and the Gauss-Markov assumption outperformed either assumption alone.

For HMM's, further work with explicit time correlation modeling has generated more encouraging results. Woodland [64] achieves improved performance by using higher order vector predictors and discriminant output distributions. Takahashi *et al.* [65] also obtained good results by conditioning on a quantized version of the previous observation and modeling conditional dependence in a mixture framework. Segment modeling research, on the other hand, took a different approach to solving this problem by adding an observation noise term, as described in the next section.

Training the Gauss–Markov parameters for the segment model is analogous to the HMM solution. For segment modeling, however, it is more convenient to use MLSS training to associate observations to model regions than to deal with the added complexity of the hidden segmentation; therefore, the MLSS update equations are given in the Appendix. In addition, for segment modeling, there is the question of whether correlation is represented between observations or between some sequence of regions in a hidden trajectory, as illustrated by time warping versus time sampling respectively in Fig. 3, with time sampling requiring a more complex parameter estimation process.

### C. Dynamical System Model

A stochastic, linear dynamical system (DS) is, in general, described by the equations

$$x_{t+1} = F_t x_t + w_t \tag{15}$$

$$y_t = H_t x_t + v_t \tag{16}$$

where $x_t$ is an unobserved state vector, $y_t$ is an observed feature vector, and $w_t$ and $v_t$ are uncorrelated Gaussian vector processes with mean and autocovariance functions $(\mu_W(t), C_W(t,u) = Q_W(t)\delta_{t,u})$ and $(\mu_V(t), C_V(t,u) = Q_V(t)\delta_{t,u})$, where $\delta_{tu}$ is the Kronecker delta. The initial state $x_0$ is also Gaussian, with mean and covariance $(\mu_0, \Sigma_0)$. The dynamical system model is widely used for estimation and control problems with nonstationary signals but was introduced as a speech recognition model by Digalakis *et al.* [13], [66]. In order to use the DS in a multiregion segment modeling framework, a parameter set $\Theta = \{H, \mu_V, Q_V, F, \mu_W, Q_W, \mu_0, \Sigma_0\}$ is defined for each region of each segment model, assuming that the system parameters are locally time-invariant within a region. The probability of a segment is computed using the innovation sequence $\{e_t\}$

$$b_{a,l}(y_1^l) = \prod_{t=1}^{l} \mathrm{p}(e_t|a, r_t) \tag{17}$$

where $\mathrm{p}(e_t|a, r_t)$ is Gaussian with zero mean and covariance $\Sigma_{e_t}$, and $e_t$ and $\Sigma_{e_t}$ are found using the same equations given in the Appendix as part of parameter estimation. Within a region, the DS model can be viewed as a continuous-state

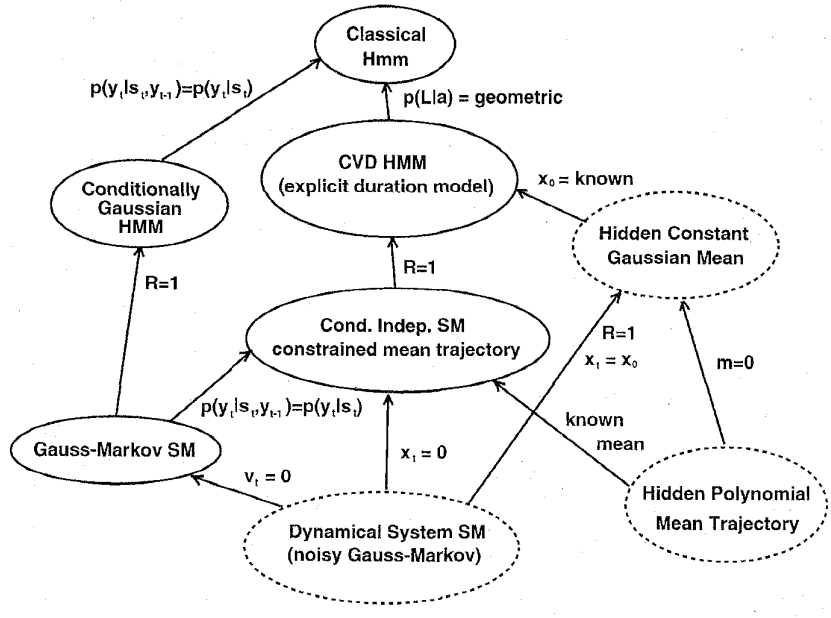



Fig. 4. Family tree of stochastic models for a variable-length frame-based observation sequence. The arrows indicate model simplifications, with $s_t$ representing both HMM state and SM region. The four cases in the top half of the figure correspond to a single region (or "stationary") SM. The three models with the dashed boundaries have a continuous hidden state.

HMM since the hidden trajectory vectors $x_t$ are continuous valued. Taken as a whole, the DS segment model combines both a continuous unobserved state $x_t$ with a discrete state that is the index to the model region $r_t$.

One view of the hidden trajectory is as a filtered series of targets $(\mu_W)$, and in this sense, the DS model is similar to that proposed by Bakis [67] with the exception that Bakis' approach uses minimum error rather than ML for training and recognition. From another perspective, the stochastic process generated by this model can be thought of as a scaled, noisy observation (specified by the observation equation (16)) of a Gauss–Markov process (described by the state equation (15)). Thus, it includes the Gauss–Markov process as a special case $(H = I, v_t = 0, y_t = x_t)$. Many of the other modeling assumptions described here can also be viewed as special cases of the DS model (see Fig. 4). For example, if the unobserved state $x_t$ is taken to be zero, then the terms $v_t$ provide the distributions for the regions $(y_t = v_t)$, and the multiregion DS model corresponds to the constrained-mean trajectory assumption. The constrained mean may be parametric or not, depending on the definition of $v_t$. In Section III-E, we shall describe additional special cases.

Training is equivalent to ML identification of a stochastic dynamical system. The classical method to obtain ML estimates requires the integration of adjoint equations, which can become too involved under certain distribution assumptions and for the large number of models typically used in speech. Alternatively, the EM algorithm provides a simpler solution by viewing the state variables $x_t$ as a hidden continuous state [66]. Specifically, the EM algorithm involves iteratively computing expected moments of the hidden state process and re-estimating the model parameters using multivariate regression. Further details are provided in the Appendix. Note that this iterative algorithm is embedded in the parameter re-estimation step of the general iterative MLSS training algorithm.

Because there are so many options for the structure of this model, parameter tying becomes very important but also more difficult. Thus far, most parameter tying has been based on experimentation and heuristic assumptions using knowledge of the units being modeled. In [66], it was assumed that $H = I, \mu_W = 0$ and $Q_V$ was tied over all regions and all phones (arguing that "observation noise" was independent of phone label). This structure was compared experimentally to other variations that included the nonparametric constrained mean and Gauss–Markov assumptions in context-independent phone classification experiments, and the DS model with the time warping correlation assumption gave the best results with 6–13% reduction in error rate relative to other models [81]. (Presumably, the benefit of the DS model would be greater for the context-dependent modeling case because of the reduced variance of the initial state. In addition, other results [68] suggest that segmental models are better suited to representing detailed contexts.) In contrast, Bakis [67] chose parameters equivalent to making $H$ model-dependent, $\mu_V = 0$, tying $F$ over all phones (arguing that $F$ represented rate of movement of the articulators and was therefore phone-independent), and letting $\mu_W$ represent hidden targets. Ross and Ostendorf [69] defined parameter tying for intonation modeling based on linguistic studies of different factors affecting intonation. Clearly, there are many untested options within this framework, and further experimental work is needed for assessing broad topology assumptions as well as parameter tying.

### D. Nonlinear Models

The dynamical system model described in the previous section can be further generalized by allowing for nonlin-

ear models, as proposed in [70]. Other possibilities include predictive neural network models, e.g., [71]–[74], generalized to the segment framework. Of course, the use of nonlinear models introduces an additional computational burden, particularly in automatic training, which is a cost that is not clearly justified in all speech modeling problems. In our own studies [13], we compared the performance of linear versus nonlinear regression in explaining the variance of a particular observation within a segment, where the nonlinear regression is based on the alternating conditional expectation (ACE) algorithm [75]. In addition to validating the widely held belief that observations within the same segment are highly correlated, the results showed that the percentage of the variance explained by the linear regression method was in most cases very close to that explained by the ACE method. Thus, linear models (i.e., Gaussian assumptions) are probably adequate for modeling intrasegmental dependencies, at least for cepstral parameters and phone or subphone units. Linear regression did not work well for predictions across phones; therefore, nonlinear models may be useful for diphone units and/or other feature sets. In addition, nonlinear models may be useful in combination with linear models, e.g., for mapping from a low-dimensional, linear hidden trajectory space [76] or for modeling the trajectory over longer time spans [77].

### *E. Segment-Level Mixtures*

Since mixture distributions have been used so successfully in HMM's (e.g., [78]), a natural extension of any of the models described so far is to segmental mixtures. The direct analogy to HMM general Gaussian mixtures (which are often called continuous density HMM's) is a discrete mixture of segmental distributions with the mixture mode specifying which of the mixture components generated the segment observations. Essentially, any of the models given in Fig. 4 can be extended by introducing a discrete mixture mode. Alternatively, one can envision a continuous mixture mode by defining a prior on a parametric trajectory. In either case, the correlation among the sequence of random variables in a segment is represented through the mixture mode. If the advantage of frame-level mixture distributions stems from systematic variation in speech, then segmental mixtures may be able to represent the systematic component via a framework that keeps the mixture mode constant across the segment. In contrast, the frame-level mixture model allows mixture modes to change randomly at each time step. Of course, if the advantage of frame-level mixtures is simply that Gaussian models do not fit the data well, then frame-based mixtures will be a more efficient representation than segmental mixtures. This question must be answered empirically and remains open at this point, although our intuition and preliminary experiments favor the systematic variation interpretation.

*1) Discrete Mixture Modes:* The discrete-mode segmental mixture model attempts to represent systematic variation by generalizing the SM to have a finite collection of segment-level distributions, which are combined with mixture weights that correspond to the probability of observing a particular trajectory in a segment. Specifically, the probability of $y_1^l$ given unit $a$ and length $l$ is

$$b_{a,l}(y_1^l) = \sum_{j=1}^{N_C} \mathrm{p}(c_j|a)\mathrm{p}(y_1^l|c_j,a,l) = \sum_{j=1}^{N_C} \mathrm{p}(c_j|a) b_{a,l}^j(y_1^l|c_j). \quad (18)$$

For each of the $N_C$ mixture components, $c_j, b_{a,l}^j(y_1^l|c_j)$ gives the probability of the complete segment conditioned on that component, and the probability $\mathrm{p}(c_j|a)$ is the mixture weight. Initial studies used constrained mean trajectory models as the mixture components $b_{a,l}^j(y_1^l|c_j)$, although in principle, any of segmental distribution assumptions could be used. Nonparametric trajectory segmental mixtures are introduced in [54] and [79], and parametric trajectory mixtures are described in [56]. (A segment-level mixture model is also proposed in [80], but in this case, fixed-length segment observations are used, as in [27], and the model does not strictly follow the framework described here.) One problem with segmental mixture models is the greater number of free parameters, which can lead to overtraining and difficulties in estimating robust context-dependent models, as described below. The number of free parameters can be reduced, if necessary, by using parametric constrained mean trajectories and/or by using mixtures at the subsegment level as explored in [13] and [14].

As for the dynamical system model, the iterative EM algorithm is required to estimate the mixture model parameters since the mixture mode is hidden. In this case, the E-step requires computation of the posterior probabilities of each mixture component for each hypothesized segment, and the M-step uses these probabilities to weight the observations in updating the parameters. These steps are repeated until adequate convergence is observed: typically a few iterations. The EM algorithm for estimating frame-level mixture distributions is sensitive to issues of initialization and unbounded likelihoods, and the problems for segmental mixtures can be more severe because of the higher dimensional space. As a consequence, techniques like variance clipping are important for obtaining good results.

Experiments with segmental mixtures of nonparametric constrained trajectory models [79] give very good performance for context-independent phone modeling on the Resource Management task, outperforming single constrained-mean and frame-level mixture models by 20–30% [14], [79]. For the context-dependent modeling case, however, more training data and/or further work on parameter tying is needed before this approach outperforms the frame-level mixture model.

*2) Continuous Mixture Modes:* An alternative approach to segmental mixture modeling is to represent a continuum of possibilities by putting a prior on some parameter of the trajectory, such as the mean in the parametric constrained-mean trajectory model. The simplest such model assumes a constant mean throughout the segment $y_t \sim N(\mu, \Sigma)$, where the mean is modeled by a Gaussian prior $\mu \sim N(\mu_0, \Sigma_0)$. This model was proposed by Russell [8] and Gales and Young [37], [9] as a "segmental HMM" and by Ostendorf and Digalakis [81], [13] as a "target state SM." The constant Gaussian mean

model is again a special case of the dynamical system model, where the state is constant for all $t, x_t = x_0 = \mu$, i.e., $F = I, \mu_V = 0, w_k = 0$ and $Q_V = \Sigma$. In addition, it can be viewed as a sophisticated version of variable frame rate analysis, as shown in [8] and [68], where the segment mean is used rather than the first observed value.

The three separate developments of the constant, random mean model resulted in three approaches for computing the probability of a segment. Russell [8] proposed an approximation of the segment probability based on the most likely trajectory given $y_1^l$, whereas Gales and Young [9] give a formula for the exact probability:

$$\log b_{a,l}(y_1^l) = K - \frac{1}{2}\left[ l \log |\Sigma| + \log |\Sigma_0| - \log |\Sigma_n| + \mu_0^{\mathrm{T}}\Sigma_0^{-1}\mu_0 + \sum_{j=1}^{l} y_j^{\mathrm{T}}\Sigma^{-1}y_j - \mu_n^{\mathrm{T}}\Sigma_n^{-1}\mu_n \right]$$

where $K$ includes the $2\pi$ terms, T indicates transpose (versus $T$ for observation length), and

$$\Sigma_n^{-1} = \Sigma_0^{-1} + l\Sigma^{-1} \quad \text{and}$$
$$\mu_n = \Sigma_n\left(\Sigma_0^{-1}\mu_0 + \Sigma^{-1}\sum_{j=1}^{l} y_j\right).$$

Under the dynamical system interpretation, the exact solution can be obtained recursively by taking the product of the innovation probabilities as in Section III-C, which can save computation because it allows for segment pruning. However, the recursive solution is only efficient when the terms $K_i$ and $\Sigma_{e_i}$ (from the Appendix) are prestored for all lengths, which is not practical for long segments.

More general parametric trajectories can also be used in a continuous mixture, as Holmes and Russell [82] have shown in an extension that puts a prior on linear (or potentially higher order) trajectories. When the hidden mean is a polynomial trajectory with $m > 0$, which is the hidden polynomial mean model in Fig. 4, there is no direct analogy to the DS model unless the assumption that $w_t$ are uncorrelated is relaxed.

Initial experimental results by all sites on context-independent modeling were discouraging, perhaps because of the constant mean and covariance assumptions. However, initial work in context-dependent phone modeling has led to some gain in performance on a digit recognition task: 20–30% reduction in error rate relative to a standard frame-based HMM but smaller improvements relative to variable-frame-rate and mixture HMM's [68]. In addition, the use of higher order trajectories could lead to further improvements.

## IV. Segmental Features and Posterior Distributions

As mentioned earlier, one of the initial motivations for considering segmental models is the potential for incorporating segmental features. In this section, we return to the problem of representing segmental features and show that it is problematic for stochastic modeling in general. Since the most promising approach to incorporating segmental features in a valid statistical model is through posterior distributions, we then highlight some modeling issues critical to this approach.

A segmental feature is any transformation $f(y_1^l, l)$ of the variable-length segment observation sequence $y_1^l$, such as a vector of average formant frequencies. In approaches using conditional distribution models with feature transformations, Equation (5) becomes

$$\mathrm{p}(y_1^T | l_1^N, a_1^N) = \prod_{i=1}^{N} \mathrm{p}(y_{t(i-1)+1}^{t(i)} | l_i, a_i) \equiv \prod_{i=1}^{N} \mathrm{p}(f(Y_i, l_i) | l_i, a_i)$$

where $Y_i = y_{t(i-1)+1}^{t(i)}$ represents the observations in the $i$th segment. (In this section, we use "$\equiv$" to indicate a modeling assumption.) One common type of "segmental feature" is a fixed-length, sampled version of the observation sequence, different versions of which are used in [27], [95], and [80]. Unfortunately, the fixed-length feature mapping changes the dimensionality of the probability space of the whole sequence so that it is proportional to the number of hypothesized segments, and thus, fewer segments are favored because of the smaller number of probability terms. The dimensionality problem can be addressed heuristically with a length-dependent weighting factor, as in, e.g., [27], but it reflects a more general problem of conditioning on different events for different segmentations, as illustrated next with the posterior distribution equations. Moreover, for context-dependent models and the nonparametric constrained-mean distribution, Kimball [79] finds that the fixed-length feature assumption hurts performance, even with the appropriate heuristic length weighting factor.

The dimensionality difference is not so obviously a problem with posterior distribution models. Looking at the problem of phone *classification*, it seems reasonable to assume that there is a piecewise constant mapping from the space of segment observations to phone likelihoods, as in $\mathrm{p}(a|Y) \equiv \mathrm{p}(a|f(Y,l))$. For the problem of phone *recognition*, however, segment boundaries are not known *a priori*, and this modeling assumption requires feature processing to depend on the segmentation. Using posterior distributions, the segment recognition problem (see (7)) becomes

$$\hat{a}_1^N = \arg\max_{N, a_1^N} \{\max_{l_1^N} \mathrm{p}(a_1^N | l_1^N, y_1^T)\mathrm{p}(l_1^N | y_1^T)\} \equiv \arg\max_{N, a_1^N} \{\max_{l_1^N} \mathrm{p}(a_1^N | F(y_1^T, l_1^N))\mathrm{p}(l_1^N | y_1^T)\} \quad (19)$$

where $F(y_1^T, l_1^N) = \{f(Y_i, l_i)\}$. Since the feature processing assumption necessarily depends on the segmentation, the features differ as a function of the segmentation, and the overall conditioning event $F(y_1^T, l_1^N)$ is not unique. In this case, the foundation of statistical detection theory is lost since the theory holds for comparing $p(a|z)$ to $p(a'|z)$ and not for the comparison of $p(a|z)$ to $p(a'|w)$. Thus, segmental features are problematic in general for joint segmentation and recognition

problems and are most useful for the more restricted case of rescoring hypotheses with the same segmentation.

However, posterior distribution models need not be restricted to fixed-length features, and there is much interest in such models because they provide a broader and potentially more powerful class of discriminant functions. In particular, both decision trees [49] and neural networks can be used to estimate posterior distributions, providing general nonparametric models of the mapping from observations to class labels. (For neural networks, see [83] for a connection to HMM's and [84] for a more general discussion.) Posterior distributions have been used successfully in HMM's (e.g. [85]–[88]) and, under certain assumptions, can be shown to be mathematically consistent [89]. For segmental posterior distributions, however, some additional difficulties are encountered, again because of the segmentation dependence. There are fundamentally two problems: segmentation likelihood modeling and appropriate independence assumptions for context-dependent models. Various methods for computing the segmentation likelihood ($p(l_1^N|y_1^T)$ in (19)) have been explored with some success [90], [91], although experiments in [79] suggest that further work is needed. In addition, an appropriate choice of the segmentation probability model might address the conditioning event mismatch problem for segmental features so that $p(a|z)p(z|y)$ is compared with $p(a'|w)p(w|y)$, and the recognition problem is a joint maximization over labels and features.

Context modeling is difficult in general since it requires robust estimation techniques to handle the practical problem of a large number of free parameters, which is an issue explored in [92] and [93] for posterior distributions. However, the bigger problem relates to conditional independence assumptions, which are theoretically inconsistent in some of the currently proposed segmental posterior distribution models. It is reasonable to simplify (19) by assuming

$$\mathrm{p}(a_1^N|l_1^N, y_1^T) = \mathrm{p}(\gamma_1^N|l_1^N, y_1^T) \equiv \prod_{i=1}^{N} \mathrm{p}(\gamma_i|l_1^N, y_1^T, \gamma_{i-1})$$

where $\gamma_i$ is a triphone label. Since successive triphones necessarily depend on each other, it is not appropriate to drop the $\gamma_{i-1}$ conditioning here. Further simplification, however, is, at best, a reasonable approximation, as in

$$\mathrm{p}(\gamma_i|l_1^N, y_1^T, \gamma_{i-1}) \approx \mathrm{p}(\gamma_i|\gamma_{i-1}, \{(l_j, Y_j), j = i-K, \cdots, i+K\})$$

for some $K$, assuming that observations sufficiently distant in time do not affect the current state. It is not reasonable to assume that $\gamma_i$ is independent of all $Y_j \neq Y_i$ given $Y_i$. This was shown experimentally in [79] and can be seen intuitively by considering an analogous problem for HMM's: The approximation

$$\mathrm{p}(s_t|y_1^T, s_{t-1}) = \mathrm{p}(s_t|s_{t-1}, y_t^T) \approx \mathrm{p}(s_t|s_{t-1}, y_t)$$

is roughly equivalent to ignoring the backward pass of the forward-backward algorithm.

This is not to say, however, that either posterior distribution modeling or the use of segmental features has been completely unsuccessful. Despite theoretical problems, fixed-length and segmental features have been used with practical success in a variety of systems. They facilitate the use of segmental neural networks [94], [90], [92] and allow for joint correlation modeling of the entire segment [95], [57], [80]. The question raised here is whether these results might be even more successful in a slightly revised framework. Since the area of posterior distribution modeling has received less attention than models based on class-conditional distributions, many of the questions of interest are not yet fully answered, and problems raised here will undoubtedly be addressed with further work.

## V. Discussion

In summary, segment models can be thought of as a higher dimensional version of a HMM, where Markov states generate random sequences rather than a single random vector observation. The basic segment model includes an explicit segment-level duration distribution and a family of length-dependent joint distributions (which are specified via region-dependent distributions and a time mapping to those regions). Since segment models are a generalization of HMM's, the standard HMM training and recognition algorithms can be easily extended to handle segment models, but with a higher computational cost due to the expanded state space.

The advantage of segment models is that there are many alternatives for representing a family of distributions, allowing for explicit trajectory and/or correlation modeling. Several distribution assumptions that have been proposed in the literature have been described here. Looking at the group of options as a whole, the key modeling assumptions include the following:

1) whether the trajectory model is hidden (as for the dynamical system and various segmental mixture models) versus observed (as for the constrained-mean trajectory and Gauss–Markov models)
2) whether correlation is modeled explicitly through Gauss–Markov assumptions or a mixture mode versus implicitly through the distribution mapping constraints
3) whether the trajectory (hidden or not) is represented parametrically or nonparametrically.

In addition, the use of a deterministic mapping to distribution regions raises questions about what is the best model of intrasegmental timing. Aspects of these alternatives have been explored in isolated experiments, but much more work is needed to assess the relative benefits of the different modeling assumptions as a whole. Of course, the answer to questions about structure will depend on the particular feature vectors used and the units represented, which raises further questions about the problems for which segmental models are best suited.

In addition to better understanding the behavior of the different segmental models through empirical studies, further algorithmic and theoretical development is needed on several fronts. For example, distribution clustering has proved to be very useful for HMM's, as well as for clustering region-dependent distributions in segment models. However, because subphonetic distribution clustering has been more successful than phone-level clustering, new techniques may be needed for robust estimation of context-dependent parametric trajectory models and segmental mixture models. A related problem

is adaptation. Both speaker and incremental adaptation have proved to be powerful tools for improving HMM performance, but there are many more parameters in a segment model than in an HMM. Therefore, segment models (even more than HMM's) will require adaptation techniques that can generalize from a small amount of data to a large number of parameters. Finally, the use of posterior distributions in segment modeling is still in its early stages, and much can be done to advance these models.

In conclusion, we note that much of the theoretical framework of the segment model can also be applied to other time series modeling problems. For example, phrase-structured language modeling (e.g. [96] and [97]) can be formulated as a variable-length state (segment) process, where a "segment" corresponds to a phrase, and the "observations" are words. In this case, the observation distribution assumptions would need to reflect the discrete nature of word-based observations. SM's can also be used in synthesis applications, as in [69], where trajectory modeling and structural constraints make the DS model more useful than HMM's. Thus, a further development of segment models will have implications beyond acoustic modeling for speech recognition.

## APPENDIX
## PARAMETER ESTIMATES FOR DIFFERENT DISTRIBUTION ASSUMPTIONS

Here, we provide the re-estimation formulae for the different distribution assumptions described in Section III, specifically for step 2) of the MLSS training algorithm. (The analogous solutions using the general forward-backward algorithm are simple extensions that use sums of all observations weighted by the likelihood of the hypothesized segment.) The full derivations of the results are omitted, but references are included in each case. In all cases, we simplify the notation by using the index $r$ to indicate segment model label and region index combined. Accordingly, we assume that training for model $a$ is based on the set of segment observations $\mathcal{Y}_a = \{Y_i : a_i = a\}$, where $Y_i = [y_{t_i}, \cdots, y_{t_i+l_i}]$, $y_t$ is a $D$-dimensional vector, and training for region $r$ in model $a$ is based on $\mathcal{Y}_a(r)$, which are the observation frames assigned to that region. Finally, define $|A|$ as the number of observations in the set $A$.

*Nonparametric Constrained Mean Trajectory:* For the nonparametric constrained trajectory model, the parameter estimates are simply the standard Gaussian mean and covariance estimates:

$$\hat{\mu}_r = \frac{1}{|\mathcal{Y}_a(r)|} \sum_{y_r \in \mathcal{Y}_a(r)} y_t \quad \text{and}$$

$$\hat{\Sigma}_r = \frac{1}{|\mathcal{Y}_a(r)|} \sum_{y_t \in \mathcal{Y}_a(r)} (y_t - \hat{\mu}_r)(y_t - \hat{\mu}_r)^{\mathrm{T}}.$$

*Parametric Constrained Mean Trajectory:* The Gaussian distributions for the parametric trajectory model are characterized by a $D \times (m+1)$ matrix $B$ for describing the vector mean trajectory with an $m$th-order polynomial and a single $D \times D$ covariance matrix $\Sigma$ used for all frames. The particular parameter estimation solution depends on the trajectory sampling assumption; therefore, different solutions are given in [56] and [15]. Here, we present the Gish–Ng solution [56] since their model is most similar to the other examples included in the Appendix. (These equations differ slightly from those in [56] since we use a transposed definition of $Y_i$ and have not included the segmental mixture terms.) Define the ML estimates of the trajectory parameter matrix for a single segment observation $Y_i$ as

$$\beta_i = Y_i Z_{l_i}^{\mathrm{T}} [Z_{l_i} Z_{l_i}^{\mathrm{T}}]^{-1}$$

where $Z_l$ is the $(m+1) \times l$ time sampling matrix described in Section III-A. Taking this statistic for all segments $Y_i$ that map to the specific model of interest, the new model parameters are

$$\hat{B} = \left[ \sum_{Y_i \in \mathcal{Y}_a} \beta_i Z_{l_i} Z_{l_i}^{\mathrm{T}} \right] \left[ \sum_{Y_i \in \mathcal{Y}_a} Z_{l_i} Z_{l_i}^{\mathrm{T}} \right]^{-1} \quad \text{and}$$

$$\hat{\Sigma} = \frac{1}{\sum_{Y_i \in \mathcal{Y}_a} l_i} \sum_{Y_i \in \mathcal{Y}_a} (Y_i - \hat{B} Z_{l_i})(Y_i - \hat{B} Z_{l_i})^{\mathrm{T}} \tag{20}$$

where $\hat{\mu}_j$ is the $j$th column of $\hat{B} Z_{l_i}$, which is the estimated mean for sample $j$ in an $l_i$ length segment. (In this case, the region $r$ is indexed by normalized time, which may be continuous valued.)

*Dynamical System and Gauss–Markov Models:* Using the EM algorithm for estimating the parameters of the dynamical system model, as proposed in [66], involves computing the conditional expectations of the sufficient statistics for the hidden state during the E-step, using these to re-estimate the parameters during the M-step, and iterating until convergence. In this discussion, we assume $t$ is such that $y_t \in \mathcal{Y}_a(r)$ and $|\mathcal{Y}_a(r)| = N$. At each iteration $p$, assuming that the observations $y_t$ are complete, the **E-step** involves computation of the expected first- and second-order statistics given the observation set $\mathcal{Y}_a(r)$ and current model parameters $\theta_r(p) = \{H_r, \mu_{Vr}, Q_{Vr}, F_r, \mu_{Wr}, Q_{Wr}, \mu_{0r}, \Sigma_{0r}\}$

$$E_{\theta_r(p)}\{x_t | \mathcal{Y}_a(r)\} = \hat{x}_{t|N}$$

$$E_{\theta_r(p)}\{x_t x_t^T | \mathcal{Y}_a(r)\} = \hat{x}_{t|N} \hat{x}_{t|N}^{\mathrm{T}} + \Sigma_{t|N}$$

$$E_{\theta_r(p)}\{x_t x_{t-1}^T | \mathcal{Y}_a(r)\} = \hat{x}_{t|N} \hat{x}_{t-1|N}^{\mathrm{T}} + \Sigma_{t,t-1|N}$$

(To simplify notation in further discussion, we drop the $r$ from $\mathcal{Y}_a$ and the specific parameters.) These statistics are calculated using the fixed-interval smoothing form of the Kalman filter, including forward and backward recursions as shown below, augmented with cross-covariance recursions to get second-order statistics. This solution is analogous to the HMM forward-backward algorithm, but state expectations are computed rather than likelihoods because the state is continuous (see the box at the bottom of the next page).

During the **M-step**, new parameter estimates are found based on the estimated state statistics. To further simplify the equations in this section, we define the operators

$$\langle \circ \rangle_{\mathcal{T}_o} = \frac{1}{|\mathcal{T}_o|} \sum_{t \in \mathcal{T}_o} \circ \quad \langle \circ \rangle_{\mathcal{T}_s} = \frac{1}{|\mathcal{T}_s|} \sum_{t \in \mathcal{T}_s} \circ$$

where $\mathcal{T}_o = \{t: y_t \in \mathcal{Y}_a(r)\}$ includes all observations mapping to $a$ and $r$, and $\mathcal{T}_s = \{t: y_{t+1} \in \mathcal{Y}_a(r)\}$ excludes the last observation time. The formulae for fully observable $y_t$ are then

$$[\hat{H} \quad \hat{\mu}_V] = \langle [y_t E\{x_t^{\mathrm{T}}|\mathcal{Y}_a\} \quad y_t] \rangle_{\mathcal{T}_o} \cdot \left( \left\langle \begin{bmatrix} E\{x_t x_t^{\mathrm{T}}|\mathcal{Y}_a\} & E\{x_t|\mathcal{Y}_a\} \\ E\{x_t^{\mathrm{T}}|\mathcal{Y}_a\} & 1 \end{bmatrix} \right\rangle_{\mathcal{T}_o} \right)^{-1} \tag{21}$$

$$\hat{Q}_V = \langle y_t y_t^{\mathrm{T}} \rangle_{\mathcal{T}_o} - \langle [y_t E\{x_t^{\mathrm{T}}|\mathcal{Y}_a\} \quad y_t] \rangle_{\mathcal{T}_o} \cdot [\hat{H} \quad \hat{\mu}_V]^{\mathrm{T}} \tag{22}$$

$$[\hat{F} \quad \hat{\mu}_W] = \langle [E\{x_{t+1} x_t^{\mathrm{T}}|\mathcal{Y}_a\} \quad E\{x_{t+1}|\mathcal{Y}_a\}] \rangle_{\mathcal{T}_s} \cdot \left( \left\langle \begin{bmatrix} E\{x_t x_t^{\mathrm{T}}|\mathcal{Y}_a\} & E\{x_t|\mathcal{Y}_a\} \\ E\{x_t^{\mathrm{T}}|\mathcal{Y}_a\} & 1 \end{bmatrix} \right\rangle_{\mathcal{T}_s} \right)^{-1} \tag{23}$$

$$\hat{Q}_W = \langle E\{x_{t+1} x_{t+1}^{\mathrm{T}}|\mathcal{Y}_a\} \rangle_{\mathcal{T}_s} - \langle [E\{x_{t+1} x_t^{\mathrm{T}}|\mathcal{Y}_a\} \quad E\{x_{t+1}|\mathcal{Y}_a\}] \rangle_{\mathcal{T}_s} \cdot [\hat{F} \quad \hat{\mu}_W]^{\mathrm{T}}. \tag{24}$$

When the means are assumed to be zero, these equations simplify as shown below for $\mu_V = 0$.

$$\hat{H} = \langle y_t E\{x_t^{\mathrm{T}}|\mathcal{Y}_a\} \rangle_{\mathcal{T}_o} (\langle E\{x_t x_t^T|\mathcal{Y}_a\} \rangle_{\mathcal{T}_o})^{-1}$$
$$\hat{Q}_V = \langle y_t y_t^{\mathrm{T}} \rangle_{\mathcal{T}_o} - \hat{H} \langle E\{x_t|\mathcal{Y}_a\} y_t^{\mathrm{T}} \rangle_{\mathcal{T}_o}$$

The estimates for $\mu_0$ and $\Sigma_0$ are simply the standard Gaussian mean and covariance equations using the estimated first- and second-order moments $E_{\theta(p)}\{x_0|\mathcal{Y}_a\}$ and $E_{\theta(p)}\{x_0 x_0^T|\mathcal{Y}_a\}$ for all instances of at least one observation mapping to region $r$. Since the Gauss–Markov model corresponds to the special case where there is no observation noise, the solution in this case is simply given by (23) and (24), substituting $y_t$ for $x_t$ and omitting the E-step.

This solution is based on the derivation in [66], [43] and assumes that $y_t$ is fully observable. These references also include solutions for the case when some $y_t$ are completely [66] or partially [26] missing and problems of nonuniform parameter tying [43]. (Observations $y_t$ might be missing when correlation is defined in terms of a hidden trajectory, i.e., the "sampled time" example in Fig. 3.)

*Discrete-Mode Segmental Mixture Model:* The parameters to be estimated in training the segmental mixture model include the weights for components of the segmental mixtures $\mathrm{p}(c_j)$ and the means and covariances of the frame-level Gaussians (assuming a constrained mean trajectory model). Training involves the EM algorithm, embedded in step 2 of the MLSS algorithm, which involves iteratively 1) computing the likelihood ("counts") of the different mixture modes and 2) re-estimating the parameters based on these counts.

In the **E-step** of the EM algorithm, we accumulate "counts" for each component of the segmental mixture distribution. At iteration $p$, the segmental "count" $\eta_{ij}$ gives the estimate of the probability of segmental component $c_j$ given segment observation $Y_i$

$$\eta_{ij} = \mathrm{p}^p(c_j|Y_i) = \frac{\mathrm{p}^p(Y_i|c_j)\mathrm{p}^p(c_j)}{\mathrm{p}^p(Y_i)},$$

where

$$\mathrm{p}^p(Y_i) = \sum_k \mathrm{p}^p(Y_i|c_j)\mathrm{p}^p(c_j).$$

The **M-step** requires estimation of mixture weights and component means and covariances based on the "counts" from the E-step. The update formulae for the parameters in, for

**E-Step: Forward recursions**

$$\hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t e_t$$
$$\hat{x}_{t+1|t} = F\hat{x}_{t|t} + \mu_W$$
$$e_t = y_t - \mu_V - H\hat{x}_{t|t-1}$$
$$K_t = \Sigma_{t|t-1} H^{\mathrm{T}} \Sigma_{e_t}^{-1}$$
$$\Sigma_{e_t} = H\Sigma_{t|t-1} H^T + Q_V$$
$$\Sigma_{t|t} = \Sigma_{t|t-1} - K_t \Sigma_{e_t} K_t^{\mathrm{T}}$$
$$\Sigma_{t,t-1|t} = (I - K_t H) F \Sigma_{t-1|t-1}$$
$$\Sigma_{t+1|t} = F\Sigma_{t|t} F^{\mathrm{T}} + Q_W$$

**E-Step: Backward Recursions**

$$\hat{x}_{t-1|N} = \hat{x}_{t-1|t-1} + A_t[\hat{x}_{t|N} - \hat{x}_{t|t-1}]$$
$$\Sigma_{t-1|N} = \Sigma_{t-1|t-1} + A_t[\Sigma_{t|N} - \Sigma_{t|t-1}]A_t^{\mathrm{T}}$$
$$A_t = \Sigma_{t-1|t-1} F_{t-1}^{\mathrm{T}} \Sigma_{t|t-1}^{-1}$$
$$\Sigma_{t,t-1|N} = \Sigma_{t,t-1|t} + [\Sigma_{t|N} - \Sigma_{t|t}]\Sigma_{t|t}^{-1}\Sigma_{t,t-1|t}$$

example, the nonparametric trajectory model are

$$\hat{\mathrm{p}}(c_j) = \frac{\sum_{Y_i \in \mathcal{Y}_a} \eta_{ij}}{\sum_{j'} \sum_{Y_i \in \mathcal{Y}_a} \eta_{ij'}}, \quad \hat{\mu}_{jr} = \frac{\sum_{Y_i \in \mathcal{Y}_a} \eta_{ij} \sum_{t \in \mathcal{T}(r)} y_t}{\sum_{Y_i \in \mathcal{Y}_a} \eta_{ij} |\mathcal{T}(r)|}, \quad \text{and}$$

$$\hat{\Sigma}_{jr} = \frac{\sum_{Y_i \in \mathcal{Y}_a} \eta_{ij} \sum_{t \in \mathcal{T}(r)} (y_t - \hat{\mu}_{jr})(y_t - \hat{\mu}_{jr})^{\mathrm{T}}}{\sum_{Y_i \in \mathcal{Y}_a} \eta_{ij} |\mathcal{T}(r)|}$$

where $\mathcal{T}(r)$ represents the subset of times in $1, \cdots, l_i$, that map to the region $r$ under consideration. The complete derivation of these equations is given in [79], together with the solution for the case where the model has both segment and frame-level mixtures. (For the parametric trajectory model solution, see [56].) Although no tying is assumed here, in practice, there may be situations where parameter tying is advantageous. In this case, the equations above change only slightly, essentially summing together counts of tied parameters.

*Continuous-Mode Segmental Mixture:* For the assumption of a constant hidden mean with a Gaussian prior, three views of the model led to three proposed parameter estimation algorithms. Gales and Young [9] find a closed form solution for the mean $\mu_0$, which for the MLSS training algorithm is

$$\hat{\mu}_0 = \frac{\sum_{Y_i \in \mathcal{Y}_a} [l_i \hat{\Sigma}_0 + \Sigma]^{-1} \sum_{t=1}^{l_i} y_{i,t}}{\sum_{Y_i \in \mathcal{Y}_a} [l_i \hat{\Sigma}_0 + \Sigma]^{-1}}$$

but no simple solution for the covariance terms. They address this problem by using the approximation that $l_i|\Sigma_0| \gg |\Sigma|$, to get parameters estimates in one step, avoiding the use of embedded iterations in the maximization step. Russell [8] gives the same solution for the mean and deals with the problem of the covariance estimates by using covariance values from a previous iteration in the update equations, giving another approximate one-step solution. Digalakis and Ostendorf [81], [13] treat the segment means as hidden variables and use the EM algorithm to get a ML solution, but embedded iteration is required. The EM solution is based on the dynamical system model, but the recursive E-Step equations are not needed here since there is only a single hidden state to estimate. The expected statistics of the hidden state of segment $Y_i$ computed in the E-Step for all $Y_i \in \mathcal{Y}_a$ are

$$E\{x_{i,0}|Y_i\} = \hat{x}_{i,0} = \left[\Sigma\mu_0 + \Sigma_0 \sum_{t=1}^{l_i} y_{i,t}\right] [\Sigma + l_i\Sigma_0]^{-1}$$

$$E\{x_{i,0}x_{i,0}^{\mathrm{T}}|Y_i\} = [l_i\Sigma^{-1} + \Sigma_0^{-1}]^{-1} + \mu_0\mu_0^{\mathrm{T}}$$

where $\hat{x}_{i,0}$ is the MAP estimate. Then M-Step update equations are then

$$\hat{\mu}_0 = \frac{1}{|\mathcal{Y}_a|} \sum_{Y_i \in \mathcal{Y}_a} \hat{x}_{i,0},$$

$$\hat{\Sigma}_0 = \frac{1}{|\mathcal{Y}_a|} \sum_{Y_i \in \mathcal{Y}_a} E\{x_{i,0}x_{i,0}^{\mathrm{T}}|Y_i\} - \hat{\mu}_0\hat{\mu}_0^{\mathrm{T}},$$

$$\hat{\Sigma} = \frac{1}{\sum_{Y_i \in \mathcal{Y}_a} l_i} \sum_{Y_i \in \mathcal{Y}_a} \sum_{t=1}^{l_i} (y_{i,t} - \hat{x}_{i,0}) y_{i,t}^{\mathrm{T}}.$$

The embedded estimation solution should give better results for the MLSS training approach, but the one-step solutions are better suited to training with the generalized forward-backward algorithm.

## ACKNOWLEDGMENT

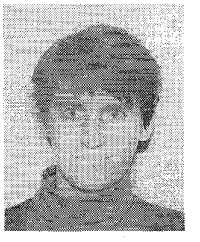
The authors wish to thank colleagues who provided comments on this manuscript, particularly A. Kannan, M. Bacchiani, and Y. Sagisaka. We also thank J. R. Rohlicek for many valuable discussions on segment modeling.

## REFERENCES

[1] L. R. Bahl, F. Jelinek, and R. L. Mercer, "A maximum likelihood approach to continuous speech recognition," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-5, no. 2, pp. 179–190, 1983.
[2] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, 1989.
[3] M. Russell and R. Moore, "Explicit modeling of state occupancy in hidden Markov models for automatic speech recognition," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, 1985, pp. 2376–2379.
[4] S. Levinson, "Continuously variable duration hidden Markov models for automatic speech recognition," *Comput. Speech Language*, vol. 1, pp. 29–45, 1986.
[5] C. J. Wellekens, "Explicit time correlation in hidden Markov models for speech recognition," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, 1987, pp. 384–386.
[6] P. F. Brown, "The acoustic modeling problem in automatic speech recognition," Ph.D. Thesis, Comput. Sci. Dept., Carnegie Mellon Univ., May 1987.
[7] P. Kenny, M. Lennig, and P. Mermelstein, "A linear predictive HMM for vector-valued observations with applications to speech recognition," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, no. 2, pp. 220–225, 1990.
[8] M. Russell, "A segmental HMM for speech pattern matching," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, vol. II, 1993, pp. 499–502.
[9] M. Gales and S. Young, "Segmental HMM's for speech recognition," in *Proc. Euro. Conf. Speech Commun. Technol.*, 1993, pp. 1579–1582.
[10] M. A. Bush and G. E. Kopec, "Network-based connected digit recognition," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-35, no. 10, pp. 1401–1413, 1987.
[11] V. Zue, J. Glass, M. Philips, and S. Seneff, "Acoustic segmentation and phonetic classification in the SUMMIT system," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, 1989, pp. 389–392.
[12] H. Meng and V. Zue, "Signal representation comparison for phonetic classification," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, May 1991, pp. 285–288.
[13] V. Digalakis, "Segment-based stochastic models of spectral dynamics for continuous speech recognition," Ph.D. Thesis, Elect. Comput. Syst. Eng. Dept., Boston Univ., Jan. 1992.
[14] A. Kannan and M. Ostendorf, "A comparison of trajectory and mixture modeling in segment-based word recognition," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, vol. II, Apr. 1993, pp. 327–330.
[15] L. Deng, M. Aksmanovic, D. Sun, and J. Wu, "Speech recognition using hidden Markov models with polynomial regression functions as nonstationary states," *IEEE Trans. Speech Audio Processing.*, vol. 2, no. 4, pp. 507–520, 1994.
[16] O. Ghitza and M. M. Sondhi, "Hidden Markov models with templates as nonstationary states: an application to speech recognition," *Comput. Speech Language*, vol. 2, pp. 101–119, 1993.
[17] J. He and H. Leich, "A unified way in incorporating segmental feature and segmental model into HMM," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, 1995, pp. 532–535.
[18] M. Ostendorf and K. Ross, "Recognition of intonation labels using a dynamical system model," in *Computing Prosody*, Y. Sagisaka, W. N. Campbell, and N. Higuchi, Eds. New York: Springer Verlag, in press.

[19] A. Ljolje and S. Levinson, "Development of an acoustic-phonetic hidden Markov model for continuous speech recognition," *IEEE Trans. Signal Processing.*, vol. 39, no. 1, pp. 29–39, 1991.
[20] H. Gish, K. Ng, and J. R. Rohlicek, "Secondary processing using speech segments for an HMM word spotting system," in *Proc. Int. Conf. Spoken Language*, vol. I, 1992, pp. 17–20.
[21] S. Roucos, M. Ostendorf, H. Gish, and A. Derr, "Stochastic segment modeling using the estimate-maximize algorithm," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, 1988, pp. 127–130.
[22] M. Ostendorf, A. Kannan, O. Kimball, and J. R. Rohlicek, "Continuous word recognition based on the stochastic segment model," in *Proc. DARPA Workshop CSR*, 1992.
[23] M. Ostendorf *et al.*, "Stochastic segment modeling for CSR: The BU WSJ benchmark system," in *Proc. ARPA Workshop Spoken Language Technol.*, 1994, pp. 9–14.
[24] C. Fong, "Statistical models of duration for synthesis and recognition," M.S. Thesis, Elect. Comput. Syst. Eng. Dept., Boston Univ., 1993.
[25] C.-H. Lee, F. K. Soong, and B.-H. Juang, "A segment model based approach to speech recognition," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, 1988, pp. 501–504.
[26] J. van Santen, "Segmental duration and speech timing," in *Computing Prosody*, Y. Sagisaka, W. N. Campbell, and N. Higuchi, Eds. New York: Springer-Verlag, in press.
[27] M. Ostendorf and S. Roukos, "A stochastic segment model for phoneme-based continuous speech recognition," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, no. 12, pp. 1857–1869, 1989.
[28] V. Digalakis, M. Ostendorf, and J. R. Rohlicek, "Fast search algorithms for phone classification and recognition using segment-based models," *IEEE Trans. Signal Processing.*, vol. 40, no. 12, pp. 2885–2896, 1992.
[29] F. Richardson, M. Ostendorf, and J. R. Rohlicek, "Lattice-based search strategies for large vocabulary speech recognition," in *Proc. Int'l Conf. Acoust., Speech Signal Processing*, 1995.
[30] M. Ostendorf *et al.*, "Integration of diverse recognition methodologies through reevaluation of N-best sentence hypotheses," in *Proc. DARPA Workshop Speech Natural Language*, 1991, pp. 83–87.
[31] L. Nguyen, R. Schwartz, Y. Zhao, and G. Zavaliagkos, "Is N-best dead," in *Proc. ARPA Workshop Human Language Technol.*, 1994, pp. 411–414.
[32] M. Rayner, D. Carter, V. Digalakis, and P. Price, "Combining knowledge sources to reorder N-best speech hypothesis lists," in *Proc. ARPA Workshop Human Language Technol.*, 1994, pp. 217–221.
[33] A. Kannan, M. Ostendorf, and J. R. Rohlicek, "Weight estimation for N-best rescoring," in *Proc. DARPA Workshop Speech Natural Language*, 1992, pp. 455–456.
[34] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood estimation from incomplete data," *J. Royal Statist. Soc. B*, vol. 39, no. 1, pp. 1–38, 1977.
[35] L. R. Rabiner, J. G. Wilpon, and B.-H. Juang, "A segmental $k$-means training procedure for connected word recognition," *AT&T Techn. J.*, vol. 65, no. 3, pp. 21–40, 1986.
[36] L. E. Baum, T. Petrie, G. Soules, and N. Weiss, "A maximization technique in the statistical analysis of probabilistic functions of finite state Markov chains," *Ann. Math. Stat.*, vol. 41, pp. 164–171, 1970.
[37] M. Gales and S. Young, "The theory of segmental hidden Markov models," Cambridge Univ. Eng. Dept., Tech. Rep., CUED/F-INFENG/TR.133, 1993.
[38] N. Merhav and Y. Ephraim, "Hidden Markov modeling using the most likely state sequence," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, 1991, pp. 469–472.
[39] R. Schwartz *et al.*, "Robust smoothing methods for discrete hidden Markov models," in *Proc. Int'l Conf. Acoust., Speech Signal Processing*, 1989, pp. 548–551.
[40] C. Lee., L. Rabiner, R. Pieraccini, and J. Wilpon, "Acoustic modeling for large vocabulary speech recognition," *Comput. Speech Language*, vol. 4, pp. 127–165, 1990.
[41] J. Gauvain and C. Lee, "MAP estimation of continuous density HMM: theory and applications," in *Proc. DARPA Workshop Speech Natural Language*, 1992, pp. 185–190.
[42] O. Kimball, M. Ostendorf, and I. Bechwati, "Context modeling with the stochastic segment model," *IEEE Trans. Signal Processing*, vol. 40, no. 6, pp. 1584–1587, 1992.
[43] K. Ross, "Computational models of intonation for speech synthesis," Ph.D. Thesis, Elect. Comput. Syst. Eng. Dept., Boston Univ., Apr. 1995.
[44] A. Kannan, M. Ostendorf, and J. R. Rohlicek, "Maximum likelihood clustering of Gaussians for speech recognition," *IEEE Trans. Speech Audio Processing*, vol. 2, no. 3, pp. 453–455, 1994.
[45] L. R. Bahl, P. V. de Souza, P. S. Gopalakrishnan, D. Nahamoo, and M. A. Picheny, "Decision trees for phonological rules in continuous speech," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, 1991, pp. 185–188.
[46] M.-Y. Hwang and X. Huang, "Subphonetic modeling for speech recognition," in *Proc. DARPA Workshop Speech Natural Language*, 1992, pp. 174–179.
[47] P. C. Woodland and S. J. Young, "The HTK tied-state continuous speech recognizer," in *Proc. Euro. Conf. Speech Commun. Technol.*, 1993, pp. 2207–2219.
[48] V. Digalakis and H. Murveit, "Genones: Optimizing the degree of tying in a large vocabulary HMM-based speech recognizer," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, vol. I, 1994, pp. 537–540.
[49] L. Breiman, J. Friedman, R. Olshen, and C. Stone, *Classification and Regression Trees*. Wadsworth and Brooks, 1984.
[50] J. Takami and S. Sagayama, "A successive state splitting algorithm for efficient allophone modeling," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, vol. I, 1992, pp. 573–576.
[51] M. Afify, Y. Gong, and J.-P. Haton, "Non-linear time alignment in stochastic trajectory models for speech recognition," in *Proc. Int. Conf. Spoken Language Processing*, 1994, pp. 291–293.
[52] V. Digalakis, M. Ostendorf, and J. R. Rohlicek, "Improvements in the stochastic segment model for phoneme recognition," in *Proc. DARPA Workshop Speech Natural Language*, 1989, pp. 332–338.
[53] W. Goldenthal and J. Glass, "Modeling spectral dynamics for vowel classification," in *Proc. Euro. Conf. Speech Commun. Technology*, 1993, pp. 289–292.
[54] O. Kimball and M. Ostendorf, "On the use of tied-mixture distributions," in *Proc. DARPA Workshop Speech Natural Language*, 1993, pp. 102–107.
[55] M. Ostendorf *et al.*, "The 1994 BU NAB news benchmark system," in *Proc. ARPA Workshop Spoken Language Technol.*, 1995, pp. 139–142 (see also the review by Pallett *et al.* in these proceedings.)
[56] H. Gish and K. Ng, "A segmental speech model with applications to word spotting," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, vol. II, 1993, pp. 447–450.
[57] S. Krishnan and P. V. S. Rao, "Segmental phoneme recognition using piecewise linear regression," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, vol. I, 1994, pp. 49–52.
[58] D. Sun, L. Deng, and C. Wu, "State-dependent time warping in the trended hidden Markov model," *Signal Processing*, vol. 39, no. 1, pp. 263–275, 1994.
[59] L. Deng and H. Sameti, "Speech recognition using dynamically defined speech units," in *Proc. Int. Conf. Spoken Language Processing*, vol. 4, 1994, pp. 2167–2170.
[60] K. K. Paliwal, "Use of temporal correlation between successive frames in a hidden Markov model based speech recognizer," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, vol. II, 1993, pp. 215–218.
[61] A. B. Poritz, "Linear predictive hidden Markov models and the speech signal," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, 1982, pp. 1291–1294.
[62] B.-H. Juang and L. R. Rabiner, "Mixture autoregressive hidden Markov models for speech signals," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, no. 6, pp. 1404–1413, 1985.
[63] L. Deng and C. Rathinavalu, "A Markov model containing state-conditioned second-order nonstationarity: Application to speech recognition," *Comput. Speech Language*, vol. 9, no. 1, pp. 63–86, 1995.
[64] P. Woodland, "Hidden Markov models using vector linear prediction and discriminative output distributions," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, 1992, pp. 509–512.
[65] S. Takahashi, T. Matsuoka, Y. Minami, and K. Shikano, "Phoneme HMM's constrained by frame correlations," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, vol. II, 1993, pp. 219–222.
[66] V. Digalakis, J. R. Rohlicek, and M. Ostendorf, "A dynamical system approach to continuous speech recognition," *IEEE Trans. Speech Audio Processing*, vol. 1, no. 4, pp. 431–442, 1993.
[67] R. Bakis, "An articulatory-like speech production model with controlled use of prior knowledge," notes from *Frontiers in Speech Processing: Robust Speech Recognition*, CD-ROM, 1993.
[68] W. Holmes and M. Russell, "Experimental evaluation of segmental HMM's," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, vol. 1, 1995, pp. 536–540.
[69] K. Ross and M. Ostendorf, "A dynamical system model for generating $F_0$ for synthesis," in *Proc. ESCA/IEEE Workshop Speech Synthesis*, 1994, pp. 131–134.
[70] M. Saerens and H. Bourlard, "Linear and nonlinear prediction for speech recognition with hidden Markov models," in *Proc. Euro. Conf. Speech Commun. Technology*, 1993, pp. 807–810.
[71] E. Levin, "Word recognition using hidden control neural architecture," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, 1990, pp. 433–436.

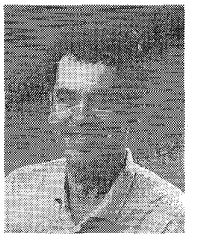
[72] J. Tebelskis and A. Waibel, "Large vocabulary recognition using linked predictive neural networks," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, 1990, pp. 437–440.
[73] K. Iso and T. Watanabe, "Speaker-independent word recognition using a neural prediction model," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, 1990, pp. 441–444.
[74] E. Tsuboka, Y. Takada, and H. Wakita, "Neural predictive hidden Markov model," in *Proc. Int. Conf. Spoken Language Processing*, 1990, pp. 1341–1344.
[75] L. Breiman and J. Friedman, "Estimating optimal transformations for multiple regression and correlation," *J. Amer. Stat. Assoc.*, vol. 80, pp. 580–607, 1985.
[76] C. Blackburn and S. Young, "Toward improved speech recognition using a speech production model," in *Proc. Euro. Conf. Speech Commun. Technol.*, 1995, pp. 1623–1626.
[77] L. Deng, K. Hassanein, and M. Elmasry, "Analysis of correlation structure for a neural predictive model with application to speech recognition," *Neural Networks*, vol. 7, no. 2, pp. 331–339, 1994.
[78] S. J. Young, J. J. Odell, and P. C. Woodland, "Tree-based state tying for high accuracy acoustic modeling," in *Proc. ARPA Workshop Human Language Technol.*, 1994, pp. 307–312.
[79] O. Kimball, "Segment modeling alternatives for continuous speech recognition," Ph.D. Thesis, Elect. Comput. Syst. Eng. Dept., Boston Univ., Sept. 1994.
[80] Y. Gong and J.-P. Haton, "Stochastic trajectory modeling for speech recognition," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, vol. I, 1994, pp. 57–60.
[81] M. Ostendorf and V. Digalakis, "The stochastic segment model for continuous speech recognition," in *Proc. 25th Asilomar Conf. Signals, Syst. Comput.*, 1991, pp. 964–968.
[82] W. Holmes and M. Russell, "Speech recognition using a linear dynamic segmental HMM," in *Proc. Euro. Conf. Speech Commun. Technol.*, 1995, pp. 1611–1614.
[83] H. Bourlard and C. Wellekens, "Links between Markov models and multilayer perceptrons," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 12, no. 12, pp. 1167–1178, 1990.
[84] M. D. Richard and R. P. Lippman, "Neural network classifiers estimate Bayesian a posteriori probabilities," *Neural Comput.*, vol. 3, no. 4, pp. 461–488, 1991.
[85] N. Morgan and H. Bourlard, "Neural networks for statistical recognition of continuous speech," *Proc. IEEE*, vol. 83, no. 5, pp. 742–770, 1995.
[86] S. Renals, N. Morgan, H. Bourlard, M. Cohen, and H. Franco, "Connectionist probability estimators in HMM speech recognition," *IEEE Trans. Speech Audio Processing*, vol. 2, no. 1, pt. II, pp. 161–174, 1994.
[87] C. W. Wightman and M. Ostendorf, "Automatic labeling of prosodic patterns," *IEEE Trans. Speech Audio Processing*, vol. 2, no. 4, pp. 469–481, 1994.
[88] M. Hochberg, G. Cook, S. J. Renals, A. J. Robinson, and R. S. Schechtman, "The 1994 Abbot hybrid connectionist-HMM large-vocabulary recognition system," in *Proc. ARPA Workship Spoken Language Technol.*, 1995, pp. 170–175.
[89] M. Ostendorf and J. R. Rohlicek, "Joint quantizer design and parameter estimation for discrete hidden Markov models," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, 1990, pp. 705–708.
[90] H. C. Leung, I. L. Hetherington, and V. Zue, "Speech recognition using stochastic explicit-segment modeling," in *Proc. Euro. Conf. Speech Commun. Technology*, vol. 2, 1991, pp. 931–934.
[91] O. Kimball, M. Ostendorf, and J. R. Rohlicek, "Recognition using classification and segmentation scoring," in *Proc. DARPA Workshop Speech Natural Language*, 1992, pp. 197–201.
[92] H. C. Leung, I. L. Hetherington, and V. Zue, "Speech recognition using stochastic segmental neural networks," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, vol. I, 1992, pp. 613–616.
[93] H. Bourlard, N. Morgan, C. Wooters, and S. Renals, "CDNN: a context dependent neural network for continuous speech recognition," in *Proc. Int. Conf. Acoust., Speech Signal Processing*, 1992, pp. 349–352.
[94] G. Zavaliagkos, Y. Zhao, R. Schwartz, and J. Makhoul, "A hybrid segmental neural net/hidden Markov model system for continuous speech recognition," *IEEE Trans. Speech Audio Processing*, vol. 2, no. 1, pt. II, pp. 151–160, 1994.
[95] W. Goldenthal and J. Glass, "Statistical trajectory models for phonetic recognition," in *Proc. Int. Conf. Spoken Language Processing*, 1994, pp. 1871–1874.
[96] M. McCandless and J. Glass, "Empirical acquisition of language models for speech recognition," in *Proc. Int. Conf. Spoken Language Processing*, 1994, pp. 835–838.
[97] B. Suhn and A. Waibel, "Toward better language models for spontaneous speech," in *Proc. Int. Conf. Spoken Language Processing*, 1994, pp. 831–834.



**Mari Ostendorf** (M'85) received the B.S., M.S., and Ph.D. degrees in 1980, 1981, and 1985, respectively, all in electrical engineering from Stanford University, Stanford, CA.

In 1985, she joined the Speech Signal Processing Group at BBN Laboratories, where she worked on low-rate coding and acoustic modeling for continuous speech recognition. She is currently an Associate Professor in the Department of Electrical and Computer Engineering at Boston University, Boston, MA, which she joined in 1987. Her research interests include data compression and statistical pattern recognition, particularly in speech processing applications. Her recent work involves investigation of segment-based and higher order models for continuous speech recognition, language modeling, and stochastic models of prosody for both recognition and synthesis.

Dr. Ostendorf has served on the Speech Processing Committee of the IEEE Signal Processing Society and is a member of Sigma Xi.



**Vassilios V. Digalakis** was born in Hania, Greece, in 1963. He received the Diploma in electrical engineering from the National Technical University of Athens, Greece, in 1986, the M.S. degree in electrical engineering from Northeastern University, Boston, MA, in 1988, and the Ph.D. degree in electrical and systems engineering from Boston University, Boston, MA, in 1992.

From January 1992 to February 1995, he was with the Speech Technology and Research Laboratory of SRI International, Menlo Park, CA. At SRI, he was a principal investigator for SRI/ARPA research contracts, and he developed speech recognition, speaker adaptation, and language education algorithms using speech recognition techniques. He is currently with the department of Electronic and Computer engineering of the Technical University of Crete, Hania, Greece, where he holds an assistant professor position. His research interests are in pattern and speech recognition, information theory, and digital communications.

**Owen A. Kimball** (M'84) received the B.A. degree in mathematics from the University of Rochester, Rochester, NY, in 1982, the M.S. degree in computer science from Northeastern University, Boston, MA, in 1988, and the Ph.D. degree in electrical engineering from Boston University, Boston, MA, in 1994. He was with BBN Corporation from 1982 to 1989, working in speech processing research. His principal focus was large-vocabulary speech recognition, including work on aids to the handicapped, and parallel processing speech algorithms. After completing the Ph.D. degree, he rejoined BBN in 1994, where he has been working on research for commercial applications of speech recognition.

Dr. Kimball is a member of Phi Beta Kappa and the signal processing society of the IEEE.