Inductive implementation of Segmental HMMs as CS-HMMs

S. M. Houghton and C. J. Champion

School of Electronic, Electrical and Systems Engineering,
Gisbert Kapp Building, University of Birmingham B15 2TT, UK.

Abstract

Segmental models have been used in speech recognition to reduce the effect of the counterfactual assumptions of statistical independence which are made in more conventional systems. They have achieved their aim at the cost of a large increase in computational load arising from making assumptions on entire segments rather than on individual frames. In this paper we show how segmental algorithms can be refactored as iterative calculations, removing most of additional computational burden they impose. We also show that the iterative implementation leads naturally to increased flexibility in the handling of timing, allowing an arbitrary timing model to be incorporated at no extra cost.

Index Terms: segmental HMM, trajectory model, continuous state HMM

10.21437/Interspeech.2015-222

1. Introduction

Conventional speech recognition models assume that speech can be represented as a sequence of discrete, independent events. This is in stark contrast to how speech is produced. Conceptually the speech production process is very simple. A small number of articulators move between positions determined by the phonetic inventory of the language. Their motion is continuous, and it follows that a large part of speech consists of the smooth variation of acoustic properties between one sound and the next. Acoustic feature vectors are therefore not time-independent and may be highly correlated from frame to frame. In conventional systems feature vectors are assumed independent from frame to frame given state, and to come from a fixed distribution during a state (where the HMM states are the substates of a phoneme). These are two poor modelling assumptions.

Standard hidden Markov models use Gaussian mixture output distributions (justifying the tag 'HMM-GMM') and model each phoneme in the context of its neighbours. Within each HMM state the statistics are assumed constant and time-independent, with Δ and Δ^2 features used to model local variation. There are two objections to be made to these assumptions: firstly the speech signal is seldom spectrally stationary (the exceptions being the centres of occasional sustained sounds), and secondly it is inconsistent to assume stationarity and then introduce gradient as a feature expecting it not to be zero. We refer to models of this sort as 'discrete state hidden Markov models' (DS-HMMs).

There has been a sustained history of researchers in the field seeking to relax the assumptions of statistical independence and static states, often motivated by the desire to construct models which are more faithful to the known properties of speech. The systems which have resulted have often sought to model sequences of feature vectors as trajectories and have frequently made direct recourse to models of speech dynamics which are most naturally stated in articulatory or formant-based representations. Examples of approaches of this sort include semi-Markov models [1], segmental HMMs [2] and trajectory models [3, 4, 5].

Our own work [6, 7] has sought to construct a framework for speech analysis and recognition taking as its starting point the Holmes-Mattingly-Shearme model of speech synthesis [8]. It lies in the strand of 'recognition by synthesis' which includes earlier published work by Paliwal and Rao [9] and unpublished work by Jordan Cohen and others.

The common feature of this strand is that it models speech as an alternation of stationary phases ('dwells') at phoneme targets and transitions between them. In order for the analysis to be tractable the transitions need to be assumed linear, and this imposes constraints on the feature set because linearity is only present if the features are of an articulatory nature. The cepstral features beloved by conventional HMMs obscure the dynamics of the signal, as is illustrated by the first Figure of [6].

We have concentrated on the strict alternation dwell-transition-dwell-transition-... but the model is more flexible. It allows for the phenomenon sometimes known as 'undershoot' (and more accurately designated as negative dwell times) in which the transition towards a target may be curtailed and immediately followed by a transition to its successor with no intervening dwell. It also allows a transition to be split into two parts with different slopes (as was stipulated in [8]).

The distinctive feature of our own work is the construction of a mathematical formalism which perfectly fits the speech model we have described. Our continuous-state hiden Markov model (CS-HMM) is a variant of hidden Gaussian Markov Models described in [10], and has some similarities with Kalman filter techniques. It is computationally efficient, probabilistic over the features and feature spaces with which it works, and minimises unwarranted independence assumptions.

It turns out that the segmental HMMs (SHMMs) of Russell and others can be expressed in the CS-HMM framework and can be made more efficient by being looked at in this way: explaining this relationship is the main purpose of the present paper.

We start by giving a short overview of SHMMs and we then provide a brief tutorial on the manipulation of Gaussian PDFs by completing the square (which we view as the key technique in the field). The equations we develop are used in Section 4 to set up the CS-HMM computational framework; the following section shows how Segmental HMMs can then be seen as a special case. We show that the CS-HMM implementation of SHMMs leads to an iterative update procedure which reduces their computational load, and that one of the mechanisms it relies on for this is the inclusion of 'time in state so far' as a state component. Once this component is present it is possible to make use of arbitrary timing models at no extra cost. (Conventional HMMs have a strong preference for exponential dis-

tributions, implying that the shortest durations are aways the likeliest; we have an equally strong preference for lognormal distributions.)

2. Segmental HMMs

Several attempts have been made to incorporate faithful models of speech dynamics into recognition algorithms. These have mostly been based on segment models [4, 11, 12, 13, 14] in which a set of observations — known as a segment — is assumed to be generated from a single underlying state.

Segment models were introduced to address the erroneous modelling assumptions we have mentioned in standard HMM systems, namely: piecewise stationarity, independence of observations within a state, and the exponential timing model. Once it had been decided to model segments rather than frames, a choice was seen between two ways of developing the formalism. It could either be assumed that the features input to the system had the dynamic properties required (in practice linearity), or else it could be assumed that the features were chosen as easily computable (i.e. representations of the power spectrum) and that the model contained a hidden layer which satisfied the desired dynamic properties and was related to the observed features by some recoverable mapping.

In the first case there is difficulty in feature extraction: if we attempt to use formant frequencies then we immediately run into the formant tracking problem which continues to be an open area of research [15]. An alternative would be to directly measure articulatory features, for example EMA [16], but there is little prospect of recovering those measurements from audio.

In the second case a mapping must be assumed between the hidden and visible spaces. This mapping is likely to be highly non-linear, but has been approximated by either a multilayer perceptron [17, 3] or a linear mapping [13, 14].

The core idea for a segmental HMM is that given a sequence of data $\boldsymbol{y}_0, \dots \boldsymbol{y}_{T-1}$, and proposed state label ϕ, the segment can be assigned a likelihood

$$\mathbb{P}[\boldsymbol{f}|\phi] \prod_{t=0}^{T-1} \mathbb{P}[\boldsymbol{y}_t|\boldsymbol{f}(t)]. \tag{1}$$

Here, $\boldsymbol{f}$ is the trajectory, ϕ represents the state and $\mathbb{P}[\star]$ will typically be Gaussian probability density functions. The assumption here is that while the individual observations are mutually independent, they depend on the underlying trajectory: the term $\boldsymbol{f}(t)$ may yield a different distribution on observations for each t; meanwhile any offset or trend which affects the entire segment is taken into the term $\mathbb{P}[\boldsymbol{f}|\phi]$ rather than reappearing separately for each observation.

Provided that the trajectory is a good model for the observations, then the intra-segment variance will be much smaller than the overall variance. The trajectory can be thought of as removing any trend from the observations. This is an advantage over standard HMMs where all observations from the same state are treated as coming from the same statistical distribution.

There are three ways of constructing a decoder incorporating the expression (1) for a segment likelihood

1. *Fixed trajectory method*: $\mathbb{P}[\boldsymbol{f}|\phi]$ is taken to a fixed by the phoneme ϕ, i.e. only one trajectory $\boldsymbol{f}(t)$ is permitted;
2. *Maximum probability trajectory*: equation (1) is maximised over all permitted trajectories $\boldsymbol{f}$ and this maximum is recorded as the score,
3. *Integrated trajectory model*: equation (1) is integrated over all trajectories $\boldsymbol{f}$ to calculate the score.

Of these, the integrated trajectory method has been found to give the best results [12]. The fixed trajectory method is barely an SHMM at all since it fails to separate out the effects common to the observations within a segment; we will ignore it for the rest of this paper.

The maximum probability and integrated trajectory methods can be incorporated into a dynamic program in the same way. Each new observation is considered both as extending all existing hypotheses by one frame and as a candidate for the first frame in a new segment. If the observation extends an existing hypothesis, the expression (1) needs to be recomputed for the segment now that it contains an extra frame. The result is that data points are used in computation over and over again as every decision to extend a segment is considered. This leads to a large computational burden, making the systems difficult to work with.

3. Products of Gaussians

In eq. (1) above we saw that the likelihood of a segment within an SHMM is a product of probabilities. In this section we outline results for the product of Gaussian probability density functions (PDFs). These results become important when we reinterpret the SHMM as a CS-HMM.

We parametrise the β-dimensional Gaussian PDF, centred at $\boldsymbol{\mu} \in \mathbb{R}^\beta$ and with precision P as

$$n_\beta(\boldsymbol{x} - \boldsymbol{\mu}, P) = \sqrt{\frac{|P|}{(2\pi)^\beta}} \exp -\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^T P(\boldsymbol{x} - \boldsymbol{\mu})$$

where $\boldsymbol{x} \in \mathbb{R}^\beta$ and precision $P = \Sigma^{-1}$ where Σ is the $\beta \times \beta$ covariance matrix. It is a well established result that the product of two Gaussian PDFs, of the same dimension, is a scaled Gaussian PDF [18]. To be exact

$$n_\beta(\boldsymbol{x} - \boldsymbol{\mu}, P) n_\beta(\boldsymbol{x} - \boldsymbol{\nu}, Q) = k n_\beta(\boldsymbol{x} - \boldsymbol{m}, R) \tag{2}$$

where

$$\begin{aligned} R &= P + Q, \\ R\boldsymbol{m} &= P\boldsymbol{\mu} + Q\boldsymbol{\nu}, \\ k &= n_\beta(\boldsymbol{\mu} - \boldsymbol{\nu}, PR^{-1}Q). \end{aligned}$$

This result can be found either by expanding eq. (2) and completing the square, or equating coefficients.

Less well known is the fact that formulae similar to eq. (2) apply even when the dimensionality of the Gaussians being multiplied differ, provided there is a linear relationship between the spaces. Here the relationship is given by matrix D. To be explicit, consider

$$n_\alpha(\boldsymbol{x} - \boldsymbol{\mu}, P) n_\beta(D\boldsymbol{x} - \boldsymbol{\nu}, Q) = k n_\alpha(\boldsymbol{x} - \boldsymbol{m}, R) \tag{3}$$

where D is an $\beta \times \alpha$ matrix ($\beta \leq \alpha$). Just as above, we can expand these formulae and collect terms to show that

$$\begin{aligned} R &= P + D^T Q D, \\ R\boldsymbol{m} &= P\boldsymbol{\mu} + D^T Q\boldsymbol{\nu}, \\ k &= \sqrt{\frac{|P||Q|}{(2\pi)^\beta |R|}} \exp -\frac{1}{2}(\boldsymbol{\mu}^T P\boldsymbol{\mu} + \boldsymbol{\nu}^T Q\boldsymbol{\nu} - \boldsymbol{m}^T R\boldsymbol{m}). \end{aligned}$$

In the case when $\alpha = \beta$ and D is the identity matrix then these formulae reduce to (3). Written in this form, parallels can be seen between this approach and Kalman filters [19].

These formulae can be generalised further to the case of a Gaussian PDF raised to a power.

$$n_\alpha(\boldsymbol{x} - \boldsymbol{\mu}, P) n_\beta(D\boldsymbol{x} - \boldsymbol{\nu}, Q)^\gamma = k n_\alpha(\boldsymbol{x} - \boldsymbol{m}, R), \quad (4)$$

where

$$R = P + \gamma D^T Q D,$$
$$R\boldsymbol{m} = P\boldsymbol{\mu} + \gamma D^T Q \boldsymbol{\nu},$$
$$k = \sqrt{\frac{|P||Q|^\gamma}{(2\pi)^{\gamma\beta}|R|}} \exp -\frac{1}{2}\left(\boldsymbol{\mu}^T P \boldsymbol{\mu} + \gamma \boldsymbol{\nu}^T Q \boldsymbol{\nu} - \boldsymbol{m}^T R \boldsymbol{m}\right).$$

Care must be taken when applying this result to ensure that R remains positive definite, which is not guaranteed if $\gamma < 0$. The main use of this formula with negative γ is to divide out a distribution which has previously been multiplied in, in which case positive definiteness is assured. We perform precisely this operation in [6], in which an approximation to the true Gaussian PDF is multiplied into a formula at one point and the true PDF is learnt a little later: so the approximation needs to be divided back out before the true PDF is introduced.

4. CS-HMMs

In a DS-HMM system there is a one-to-one correspondence between hypotheses and states, and consequently it is not necessary to insist on the distinction between them. In a CS-HMM there is an infinite number of states while only a finite number of hypotheses can be stored in memory. Each hypothesis is therefore required to contain information about an infinite number of states, which will be a set of states agreeing in their discrete components such as the identity of the current phoneme, the phonetic history, and the time spent so far in the current phoneme. Information about the continuous components will take a parametric form (in practice Gaussian). In a linear trajectory model the continuous state components will be the startpoint and gradient of a vector trajectory.

Information about the continuous state components takes the form of a distribution on Baum-Welch alpha values, written $\alpha_t(\boldsymbol{x})$ where $\boldsymbol{x} \in \mathbb{R}^\alpha$ is a vector of continuous state components and t is time. This value is the sum of path probabilities over all paths arriving in state $\boldsymbol{x}$ at time t, where the paths are limited to those consistent with the assumed discrete state components. Here, a path probability is the product over previous times of the state probability, conditioned on its predecessor, and the observation probability. We are using the term 'probability' rather loosely to denote the value of the PDF.

It is important to understand why Baum-Welch alphas are the right numbers to consider. If our aim was to recover the likeliest sequence of trajectories, we would look at the probability of the likeliest path ending in any continuous state; this too could be assumed to be Gaussian and computed inductively. But we are not interested in recovering the path — our interest is solely in recovering the phonetic sequence; and the optimum criterion for ranking phonetic sequences is their probabilities, not their joint probabilities with some other item of no independent interest. The probability of a phonetic sequence needs to be obtained by marginalising (i.e. integrating over) nuisance parameters such as the continuous state components. And this integration is just what the Baum-Welch alphas give us.

Thus our use of Baum-Welch alphas corresponds to the integrated trajectory model in the same way as dynamic programming probabilities would correspond to a maximum probability model.

It follows from what we have said that we let the α_t take the parametric form

$$\alpha_t(\boldsymbol{x}) = K_t n_\alpha(\boldsymbol{x} - \boldsymbol{\mu}_t, P_t). \quad (5)$$

The scale factor K_t is the sum of probabilities of all paths consistent with the given hypothesis. This is the correct quantity to use when ranking hypotheses, so we define $\log K_t$ to be the *score* of a particular hypothesis.

The assumption we need to make about observations is that — given the discrete state components — they are Gaussianly distributed about a linear function of the continuous state components. (For example if the continuous component is a phoneme-dependent formant frequency target θ, the formant measurements will be Gaussianly distributed about θ). On this assumption, if we have α_t as in (5), we can update the hypothesis by multiplying in the observation PDF to give us the α_{t+1}:

$$\alpha_{t+1}(\boldsymbol{x}) = \alpha_t(\boldsymbol{x}) n_\beta(D\boldsymbol{x} - \boldsymbol{y}_{t+1}, E), \quad (6)$$
$$= K_t n_\alpha(\boldsymbol{x} - \boldsymbol{\mu}_t, P_t) n_\beta(D\boldsymbol{x} - \boldsymbol{y}_{t+1}, E), \quad (7)$$
$$= K_{t+1} n_\alpha(\boldsymbol{x} - \boldsymbol{\mu}_{t+1}, P_{t+1}), \quad (8)$$

(using the results in §3). α_{t+1} is itself a scaled Gaussian PDF, and by induction the alphas take this form for all time. (Recall that the subscripts α, β denote the dimensions of the Gaussian PDFs (which may differ), with D a mapping from the large space to the small one.)

A CS-HMM is therefore implemented as a simple induction. At each step we have a list of hypotheses, each of which can be extended either by taking the new observation $\boldsymbol{y}_{t+1}$ into the current discrete state, updating the parameters according to (8), or by assuming that the new observation is the first in a changed state, in which case possibilities will be generated for each new phoneme.

When we assign the new observation to the current state we have no need to reevaluate the effect of earlier observations belonging to it. When we start off a new phonetic state we multiply the current K by

$$L(\varphi) n_\alpha(\boldsymbol{y}_{t+1} - \boldsymbol{\theta}_\varphi, (E^{-1} + R_\varphi^{-1})^{-1}) \quad (9)$$

where $L(\varphi)$ is the language-model probability of the new phoneme φ, and $\boldsymbol{\theta}_\varphi$ and R_φ are the mean and precision of its realised targets. Equations for $\boldsymbol{\mu}_{t+1}$ and P_{t+1} can likewise be given in terms of the same parameters: see our earlier papers [6, 7] for full details.

5. Viewing Segmental HMMs as CS-HMMs

With the preliminaries complete, we can now show how the calculation at the heart of a SHMM can be cast as an example of a CS-HMM. The big benefit here is that it allows for an iterative update procedure where each observation is used just once. This reduces the computational overhead of a SHMM, making them practical to use in more situations.

If we unpack eq. (1) then how to cast an SHMM as a CS-HMM can be made more explicit. Taking the startpoint $\boldsymbol{z}$ and gradient $\boldsymbol{s}$ as our parametrisation of a segment ($\boldsymbol{z}, \boldsymbol{s} \in \mathbb{R}^\beta$), then $\mathbb{P}[\boldsymbol{f}|\phi]$ gives an initialisation,

$$\alpha_0(\boldsymbol{x}) = K_0 n_\alpha(\boldsymbol{x} - \boldsymbol{\mu}, P), \quad (10)$$

where $K_0 = 1$, $\alpha = 2\beta$, $\boldsymbol{x}^T = \begin{pmatrix}\boldsymbol{z}^T & \boldsymbol{s}^T\end{pmatrix}$ and $\boldsymbol{\mu}$, P have been learned by some training procedure. The expected value of the startpoint and gradient are concatenated together into vector $\boldsymbol{\mu}$ and P is the precision with which these are known. Another way to see this is that based on the proposed phonetic state ϕ, α_0 is encoding the distribution of startpoints and gradients as a probability density function.

Assume we have a sequence of observations $\boldsymbol{y}_t$, each made with an independent Gaussianly distributed error (zero mean and precision E). As seen in eq. (1), we must multiply in the probability of each of these observations — we do this iteratively. Suppose we have $\alpha_{t-1}(\boldsymbol{x})$ and wish to account for observation $\boldsymbol{y}_t$. Given a linear trajectory model, the probability density function for the observation is

$$n_\beta(\boldsymbol{y}_t - D_h\boldsymbol{x}, E) \tag{11}$$

where $D_h = \begin{pmatrix}\mathbb{I}_\beta & h\mathbb{I}_\beta\end{pmatrix}$, $\mathbb{I}_\beta$ being the $\beta \times \beta$ identity matrix and h the time spent in the current phonetic state.

When written in this way, the segmental HMM fits exactly into the framework of CS-HMMs. Observations are accounted for iteratively by repeated application of eq. (3). Interpreting the SHMM in this way gives us access to an iterative update procedure — computationally this is a big advantage.

As mentioned earlier, section 2, there are three variants of SHMMs. Each can be reinterpreted in terms of the CS-HMM.

1. *Fixed trajectory*: take $P \to \infty\mathbb{I}_\alpha$. By assuming that the startpoint and gradient of each segment are known with infinite precision (zero variance), these values will not be re-estimated as observations made.
2. *Maximum probability trajectory*: update startpoint and gradient as observations are made. The score of a segment is taken as $K_t|P|^{1/2}/(2\pi)^{\alpha/2}$. This is the maximum value taken by the (scaled) probability density function.
3. *Integrated trajectory model*: update startpoint and gradient as observations are made. The score of a segment is K_t, since the PDF part always integrates to one.

The difference in score between the maximum probability trajectory and the integrated trajectory models was noted by Holmes and Russell [12]. For both the maximum probability trajectory and the integrated trajectory model, the final estimate of startpoint and gradient will be a weighted sum of the estimate based on the phoneme state itself (giving a prior) and a least-squares fit to the data. The integrated trajectory model was found to give best performance [12], this is the only version of the segmental HMM which uses K_t as the score. We believe K_t is the correct quantity to use when comparing different explanations of the data.

6. Timing models

In typical HMM formulation, at each time iteration there is a fixed probability p of remaining in the present state, and therefore probability $1 - p$ of moving to the next state. The probability p, is the self-loop probability. This has the result that state durations are exponentially distributed. By including the current duration of a state as part of the hypothesis, arbitrary timing models can be incorporated in the CS-HMM with ease.

Denote the state duration by τ and suppose we have the probabilities $\mathbb{P}[\tau \geq h]$ for all h. These probabilities could come from any distribution, or from a histogram of measurements from data. Then, the probability of remaining in the state given that the state has already persisted for h time units is

$$p(h) = \mathbb{P}[\tau \geq h+1 | \tau \geq h] = \frac{\mathbb{P}[\tau \geq h+1]}{\mathbb{P}[\tau \geq h]}, \tag{12}$$

where the term $\mathbb{P}[\tau \geq h | \tau \geq h+1] = 1$ and has been omitted. The probability of leaving the state, given that the state has persisted for h time units, is $1 - p(h)$.

To incorporate this into the iterative process, after updating a hypothesis to account for an observation we must consider the possibility of remaining in the state or leaving, i.e. starting a new state. This is achieved by branching the hypothesis, the two possibilities being

- remain in state — update K_t to $K_t p(h)$,
- leave the state — update K_t to $K_t(1 - p(h))$ and update the state components as necessary for starting a new state under the model being considered.

The number of observations is t and should not be confused with τ which has been used as the parameter in the timing distribution.

One possible weakness of this system is the potential for unlimited growth in the number of hypotheses. In reality this is not an issue, we can apply thresholding on K_t which as discussed early is a suitable score for each hypothesis. Also, we have not considered the reduction of hypotheses possible through some form of state coalescence as found in dynamic programming algorithms.

In the same manner, an arbitrary language model can be imposed on the system. If we think about branching the *leave the state* option this can result in multiple hypotheses, one for each possible phonetic continuation, and they can be weighted by likelihood. The likelihood can depend on an arbitrarily long sequence of phonetic history, and can include dependence on the duration of the previous state, all without complication. These are examples of changes to the discrete state component within a CS-HMM.

7. Summary

We have shown how a SHMM can be implemented in the CS-HMM framework. This retains the benefit of improving the statistical independence assumptions, but also allows for an iterative calculation which is much more computationally efficient than the backward-looking dynamic program used for SHMMs.

The CS-HMM is very flexible. Arbitrary timing models, which may differ by phoneme, can be incorporated easily, as can language models based on phonetic history and/or phoneme duration. Additionally, observations may be made in any space which is a linear mapping from the hidden space. In itself, this opens up a wide range of possibilities, for example, online vocal tract length normalisation or adaptation to overall loudness of the signal.

8. Acknoweldgements

This work derives from discussions within the *Speech Recognition by Synthesis* project at the University of Birmingham between ourselves and Martin Russell (one of the founding fathers of Segmental HMMs), Peter Jančovič, Phil Weber and Mike Carey. It represents the interpretation we put forward in discussion with the other project members. We would like to thank them, and Martin Russell in particular, for prompting us to put our ideas into a coherent form.

9. References

[1] M. J. Russell and R. K. Moore, "Explicit modelling of state occupancy in hidden Markov models for automatic speech recognition," in *ICASSP*, 1985, pp. 5–8.

[2] M. Ostendorf, V. Digalakis, and O. Kimball, "From HMMs to segment models: a unified view of stochastic modeling for speech recognition," *IEEE Trans. Speech Audio Proc.*, vol. 4, no. 5, pp. 360–378, 1996.

[3] L. Deng and J. Ma, "Spontaneous speech recognition using a statistical coarticulatory model for the vocal-tract-resonance dynamics," *J. Acou. Soc. Am.*, vol. 108, no. 6, pp. 3036–3048, 2000.

[4] M. J. F. Gales and S. J. Young, "Segmental hidden Markov models," in *EUROSPEECH*, 1993.

[5] H. B. Richards and J. S. Bridle, "The HDM: a segmental hidden dynamic model of coarticulation," in *ICASSP*, vol. 1, 1999, pp. 357–360.

[6] C. Champion and S. M. Houghton, "Application of continuous state hidden Markov models to a classical problem in speech recognition," *Computer Speech and Language*, in press, 2015. [Online]. Available: http://dx.doi.org/10.1016/j.csl.2015.05.001

[7] P. Weber, S. M. Houghton, C. J. Champion, M. J. Russell, and P. Jančovič, "Trajectory analysis of speech using continuous state hidden Markov models," in *ICASSP*, 2014.

[8] J. Holmes, I. Mattingly, and J. Shearme, "Speech synthesis by rule," *Language and Speech*, vol. 7, pp. 127–143, 1964.

[9] K. K. Paliwal and P. V. S. Rao, "Synthesis-based recogntiion of continuous speech," *J. Acou. Soc. Am.*, vol. 71, no. 4, pp. 1016–1024, 1982.

[10] P. L. Ainsleigh, N. Kehtarnavaz, and R. L. Streit, "Hidden Gauss-Markov models for signal classification," *IEEE Trans. Signal Process.*, vol. 50, no. 6, pp. 1355–1367, 2002.

[11] M. J. Russell, "A segmental model for speech pattern processing," in *ICASSP*, vol. 2, 1993, pp. 499–502.

[12] W. J. Holmes and M. J. Russell, "Probabilistic trajectory segmental HMMs," *Computer Speech and Language*, vol. 13, no. 1, pp. 3–37, 1999.

[13] M. J. Russell and P. J. B. Jackson, "A multiple-level linear/linear segmental HMM with formant-based intermediate layer," *Computer Speech and Language*, vol. 19, no. 2, pp. 205–225, 2005.

[14] M. J. Russell, X. Zheng, and P. J. B. Jackson, "Modelling speech signals using formant frequencies as an intermediate representation," *IET Signal Processing*, vol. 1, pp. 43–50, 2007.

[15] L. Deng, X. Cui, R. Pruvenok, J. Huang, S. Momen, Y. Chen, and A. Alwan, "A database of vocal tract resonance trajectories for research in speech processing," in *ICASSP*, 2006.

[16] K. Richmond, S. King, and P. Taylor, "Modelling the uncertainty in recovering articulation from acoustics," *Computer Speech and Language*, vol. 17, pp. 153–172, 2003.

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

[18] L. Wasserman, *All of Statistics, A concise course in statistical inference*. Springer, 2004.

[19] R. E. Kalman, "A new approach to linear filtering and prediction problems," *J. of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.