3-1996

Generalized Hidden Filter Markov Models Applied to Speaker Recognition

John M. Colombi
Air Force Institute of Technology, john.colombi@afit.edu

GENERALIZED HIDDEN FILTER MARKOV MODELS APPLIED TO SPEAKER RECOGNITION

DISSERTATION

John M. Colombi
Captain, USAF

AFIT/DS/ENG/96-01

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/DS/ENG/96-01

GENERALIZED HIDDEN FILTER MARKOV MODELS APPLIED TO SPEAKER RECOGNITION

DISSERTATION

John M. Colombi
Captain, USAF

AFIT/DS/ENG/96-01

The views expressed in this dissertation are those of the author and do not reflect the official policy or position of the Department of Defense or the U. S. Government.

AFIT/DS/ENG/96-01

GENERALIZED HIDDEN FILTER MARKOV MODELS APPLIED TO SPEAKER RECOGNITION

DISSERTATION

Presented to the Faculty of the Graduate School of Engineering

of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the

Requirements for the Degree of

Doctor of Philosophy

John M. Colombi, B.S.E.E.,M.S.E.E.

Captain, USAF

March, 1996

AFIT/DS/ENG/96-01

Generalized Hidden Filter Markov Models
Applied to Speaker Recognition

John M. Colombi, B.S.E.E., M.S.E.E.
Captain, USAF

Approved:

Dennis W. Ruck, Major, USAF — 12MAR96
Chairman, Advisory Committee — Date

Steven K. Rogers — 3/12/96
Member, Advisory Committee — Date

Timothy R. Anderson — 3/12/96
Member, Advisory Committee — Date

Mark Oxley — 3/12/96
Member, Advisory Committee — Date

Robert L. Hengehold — 3/12/96
Dean's Representative — Date

Accepted:

Robert A. Calico, Jr.
Dean, Graduate School of Engineering

Acknowledgements

Having completed the AFIT master's program then moving straight into the PhD program, the world, and the Air Force, has changed greatly. I am anxious to leave AFIT to experience both anew, yet with a deep longing for the world class professors and wonderful research environment I am leaving behind. The PhD program will no doubt be my best, and hardest, assignment.

I thank Dr. Dennis Ruck, my advisor, for giving me a lot of flexibility in this undertaking and guiding me to a successful completion. When, it is all ended, I know it will be my own - and I know it will be significantly better because of him. Dr. Tim Anderson has been my mentor providing a wealth of knowledge and experience. Unknowingly, he has also served, what I view, as the image of a world-class speech researcher. A special thanks goes to Dr. Steven Rogers, who made my Master's research so enjoyable and then played a big part in getting me into the doctoral program. I hope a little of his relaxed attitude, ceaseless drive and unique style has rubbed off on me. The computer support, administered by the (motherly) hands of Dan Zambon and David Doak, made my research easy, despite the gigabytes of disk space and multiple Sparc-20 jobs I would launch. I will miss the many faculty, staff and students who gave their insights, guidance, and advice along the way.

Most importantly, I thank my family - Cheryl, Andrew and Felicia. God has blessed me with forgiving children and a wonderful and understanding wife. Lastly, I thank all the New England family I have missed the past few years, receiving their support and encouragement from afar.

John M. Colombi

Table of Contents

List of Figures

List of Tables

List of Symbols

Abstract

Classification of time series has wide Air Force, DoD and commercial interest, from automatic target recognition systems on munitions to recognition of speakers in diverse environments. The ability to effectively model the temporal information contained in a sequence is of paramount importance. Toward this goal, this research develops theoretical extensions to a class of stochastic models and demonstrates their effectiveness on the problem of text-independent (language constrained) speaker recognition. Specifically within the hidden Markov model architecture, additional constraints are implemented which better incorporate observation correlations and context, where standard approaches fail. Two methods of modeling correlations are developed, and their mathematical properties of convergence and reestimation are analyzed. These differ in modeling correlation present in the time samples and those present in the processed features, such as Mel frequency cepstral coefficients. The system models speaker dependent phonemes, making use of word dictionary grammars, and recognition is based on normalized log-likelihood Viterbi decoding. Both closed set identification and speaker verification using cohorts are performed on the YOHO database. YOHO is the only large scale, multiple-session, high-quality speech database for speaker authentication and contains over one hundred speakers stating combination locks. Equal error rates of 0.21% for males and 0.31% for females are demonstrated. A critical error analysis using a hypothesis test formulation provides the maximum number of errors observable while still meeting the goal error rates of 1% False Reject and 0.1% False Accept. Our system achieves this goal. This research supports the many new electronic applications requiring speech-based biometric authentication such as secure access control, telephone-based recognition, transaction or credit account verification, forensic science, law enforcement and military intelligence.

GENERALIZED HIDDEN FILTER MARKOV MODELS APPLIED TO SPEAKER RECOGNITION

I. Introduction

1.1 Historical Overview

Classification of time series has wide Air Force, DoD and commercial interest, from automatic target recognition systems on munitions to recognition of speakers in diverse environments. The ability to effectively model the temporal information contained in a sequence is of paramount importance. Toward this goal, this research develops theoretical extensions to a class of stochastic models and demonstrates their effectiveness on the problem of text-independent (language constrained) speaker recognition. Specifically within the hidden Markov model architecture, additional constraints are implemented which better incorporate observation correlations and context, where standard approaches fail.

The speech signal contains a great deal of information more than just a sequence of words. It contains the acoustic environment (car, aircraft, machinery, office noise), gender, prosody (pitch changes, syllable stress, loud or soft speech, emotional state of the speaker), language, dialect or ethnic characteristics and speaker information. This latter information in the speech signal is desired and exploited for a speaker recognition system. Speaker recognition applications include closed-set identification, open-set identification and verification. With the electronic age, there comes many new applications for biometric authentication, in addition to forensic science [49, 50], security access and specific military requirements [129].

A speaker has two biological areas of uniqueness [85]. These include the vocal physiology and the learned neural control of the articulators which control the physiology. The first area includes such physical factors as length of vocal tract; size of mouth and nasal cavities; glottal size, shape and pulse patterns; and teeth and lip characteristics. The second area includes the learned habits of these facilities such as dialect or regional accents, pronunciation or ethnic traits, and speed and timing of the articulators. The latter

neural code may never be modeled directly, but the overall effect shows up eventually in the dynamics of acoustic signal, such as formant transitions and coarticulation effects. In fact, anatomical models attempting to estimate control of vocal articulators have been proposed for speech recognition [43, 44, 118]. For the receiver, biological acoustic phenomena also support the value of classification of speech and speakers using a temporal model. Auditory psychoacoustic studies provide a wealth of examples that relate specific temporal changes in the acoustic signal to a specific auditory event [82, 83] or measured electro-chemical response [117, 132]. Together, both the effects of physiology and the learned neural traits dynamically alter the acoustic spectrum through formant transitions and coarticulation effects; the ability to accurately model these spectra should be useful for speaker recognition.

Historically, speaker recognition has made use of techniques borrowed from speech recognition research. Distortion based methods were first chosen to compare speaker spectral representations. These methods used long term spectral averages as a representation [116]. Later, some form of dynamic time warping was used for text-dependent applications [130], allowing recognition of previously recorded utterances. Depending on the extracted features, certain distortions or metrics were proposed which were optimal for those features [55, 95]. Similarity of test speech to speaker models was based on overall distance or distortion. In the mid-1980's, Soong [119] proposed a clustering approach for text-independent applications. This classic approach will be referred to as vector quantization (VQ) since a clustering of a speaker's training features/ vectors becomes the model and classification is determined by minimum quantization error. Many successful applications and variations of this procedure have been accomplished [22, 23, 38, 61, 79, 120, 135]. Vector quantization assumes each observation is independent in time, clearly not true for speech signals.

Over the last decade, the predominant speech recognizers have been based on the hidden Markov model (HMM), first pioneered by Baum and his colleagues [7, 8, 9, 10] and soon thereafter applied to automatic speech recognition (ASR) [96]. This statistical model is complex enough to model the variability of the speech waveform, yet simple enough for its parameters to be estimated [16]. The HMM framework provides efficient Maximum Likelihood (ML) reestimation/ training algorithms with desirable properties and methods

to model and decode / recognize the many levels of speech - acoustic, phoneme, word and language. Speaker recognition, for instance when needing personal identification numbers (PIN) or passwords, may need to perform both speech and speaker recognition. The ability to remove the effects of the word sequence and extract speaker dependencies alone is an unsolved problem. With the increasing performance of hidden Markov models on speech recognition, several researchers started examining these statistical techniques for automatic speaker recognition.

Poritz [92] was one of the first to pioneer hidden Markov models for speaker identification as well as a hidden filter method, though his results were preliminary. In the early 1990's, Tishby [124] extended these hidden filters, complete with multiple mixtures [57, 60]. His results indicated that the transitions (temporal structure) of the hidden Markov chain was unnecessary. Furui [113] later compared vector quantization (VQ) codebooks to the ergodic HMM structure and also concluded that output density mixture numbers alone where responsible for performance. In effect, these researchers concluded that only modeling the spectrum of a speaker, and not the temporal patterns of the spectrum, alone was necessary for recognition. This appears to contradict the second well-known characteristic of voice differences, namely the speaking habits and learned patterns of speech. Levinson [72] has pointed historically to key experiments, including Markov himself, which demonstrated certain HMM architectures will learn the structure of the language itself. Thus, specific architectures of an HMM may not be well-suited to model speaker dependencies.

Another related approach making an observation independence assumption is the Gaussian Mixture Model (GMM), pioneered by Reynolds [101, 102, 103, 104]. In this model, a speaker's spectral vectors are represented by a mixture of multivariate normal densities, reestimated using the Expectation Maximization algorithm of Dempster [28]. The GMM assumes no temporal structure within the signal and can be considered a special case of the more general HMM, with a single state. Each of these researchers applied the hidden Markov models to unlabeled speech, where a single model represented all possible speech interactions and transitions. These methods sharply contrast to speech recognition where tens of phoneme models or thousands of context dependent tri-phone models are required.

Hidden Markov models make erroneous, simplifying assumptions in the dynamics of the speech observations. Existing models assume speech features are generated by a discrete state Markov process. Furthermore, the observations are the result of a probabilistic function of this hidden process and considered conditionally independent. It seems intuitive that past and/ or future observations provide extra information concerning the context of the current realization. In order to improve upon current statistical techniques, this independence assumption must be removed. Recently, several researchers have been relaxing these assumptions. Methods such as multi-layer perceptrons (MLP) and other neural network/ HMM hybrids [13, 26] have emerged for speech recognition though they have required specialized hardware for training. Others have proposed linear predictive densities [66, 131] or joint normal densities [16] for speech recognition though they have showed little improvement. Still others have tried polynomial representations [29] and Kalman filtering approaches [33]. An original contribution of this work includes modeling speaker dependent phonemes by the use of Markov modulated rational filters.

Speaker recognition continues to be a potential application area for better time series modeling, attracting entire workshops [113] and recent dissertations [18, 102]. This time series provides a challenge since channel and recording instrumentation, effects of particular text, prosody and speaker variability add to the classification difficulty. Recently at an international conference focusing on speaker recognition research, Furui supported this dissertation's approach stating [113],

> As fundamental research, it is important to pursue a method for extracting and representing the speaker characteristics that are commonly included in all the phonemes irrespective of the speech text.... It is expected that diversified research related to speaker-specific information in speech waves will become more active in the near future.

Lastly, the contributions of accurately modeling speakers may provide for better speech recognition. Speaker adaptation are the methods used to transform a speaker independent (SI) speech recognizer for a particular speaker. Large, accurate speech models require large amounts of training data, and it is often impractical and impossible to acquire enough training data for each speaker. Instead, speech from many speakers is used to train a speaker independent recognizer, then these models are adapted to become speaker

dependent (SD). Research in speaker modeling provides valuable insight to solutions into this adaptation.

1.2 Problem Statement and Scope

A complete framework which encompasses many older and newly developed models of discrete state dynamic systems will be created. New analysis and reestimation of several classes of linear functions within a hidden Markov model will be accomplished. Specifically, probabilistic linear functions of a hidden Markov process will account for context and correlation in the observations. These new models will then be applied to the difficult problem of modeling speaker dependencies within language-constrained (digits) speech.

1.2.1 Scope. Existing automatic speaker recognition methods do not model the spectral phoneme-level dynamics, since the current models assume observations are statistically independent. Past methods have attempted modeling speakers by either assuming 1) independent observations, 2) models assuming state-conditional independent observations or 3) architectures which grossly estimated language and grammar dynamics. This has left a large window of opportunity for extensions of the current statistical models. Whether the goal is to classify a sequence of observations, predict a time series, or uncover the hidden "state" of a system, this research has great relevance. This research addresses the reestimation of generalized statistical models for eventual classification of time series, and in particular applying these to speaker dependent phoneme modeling.

1.2.2 Research Contributions. Toward successful accomplishment of these problems, a number of original research contributions have been completed. These include:

Generalized Hidden Filter Architecture. A complete framework including many existing linear and nonlinear systems used for classification, as well as prediction, is developed for discrete state Markov models. The existing hidden Markov model independence assumptions are reviewed and removed, thus defining a new, more generalized, hidden filter Markov model.

New reestimation methods are provided for autoregressive (AR) and autoregressive moving average (ARMA) as well as an optimal initialization strategy. This models are allowed nonzero biases and either state-conditioned or common noise statistics. The ability to reestimate these filters adequately for the difficult ergodic case is novel and shown by example. This new class of ARMA Markov modulated hidden filters is applicable to specific broad classes of phonemes, with a spectral zero component. Lastly, filters operating on frames of speech have been extended from simple architectures to multi-state phoneme models.

Vector Autoregressive Hidden Filters. The extension from sample or frame based filters to full vector autoregressive hidden filters is developed with an emit-on-state notation. Several variations of the model include the regression characteristics of each vector element on past elements and noise correlation. The choice of spectral features, the Mel frequency cepstral coefficients, dictate a diagonal matrix filter, with a least-squared solution developed within. A procedure of *a posteriori* mean removal is developed to separate the state mean estimation from the filter coefficients for numerical stability.

HMM and Hidden Filter Analysis. A new proof of monotonic convergence for Gaussian mixtures is presented using a new equivalence model paradigm. A new proof of monotonic convergence for hidden filter Markov models is then demonstrated. An application of the Markov property of the observations for hidden filter models is applied to the Fielding [42] information theoretic proof. Since pattern recognition methods seek ways which reduce entropy (to reduce classification errors), this new theorem justifies the hidden filter model over standard hidden Markov models.

Phonetic Modeling for Speaker Recognition. The extensive Linguistic Data Consortium (LDC) YOHO database is used for all experimentation. A speaker dependent phoneme-based hidden filter Markov model approach is accomplished for both speaker identification and verification. The most current speech recognition tools are incorporated such as phonetic labeling, word dictionaries, bi-word language models and Viterbi scoring constraints. The method of forced Viterbi decoding of phoneme based

temporal models for speaker verification is the first to be published. Likelihood ratio normalization using cohorts is accomplished and error rates shown using a newly developed second order cohort selection strategy. A unique critical error analysis is provided for YOHO at the mixed 5% and 25% significance levels for false acceptance and false rejection target error rates, respectively.

Many current techniques apply models which assume independent observations or do not target the dynamics present in speech or the processed speech vectors. Those techniques which do attempt to model the dynamical properties have not targeted individual phonemes. Our state-of-the art approach develops state-dependent dynamic systems within phoneme for speaker recognition, providing equal error rates of 0.21% for males and 0.31% for females. These error rates have also been shown to statistically satisfy the hypothesis that our system meets or beats the U.S. Government target error rates of 1% false rejection and 0.1% false acceptance.

1.3 Dissertation Organization

This document is organized into six main chapters. The following chapter provides background material concerning hidden Markov models theory and several recent developments. It provides a new architecture unifying many other techniques. Chapter III develops the reestimation equations for hidden filter Markov models, at the scalar (sample and frame) and vector (feature) levels. In Chapter IV, the analysis of the monotonic likelihood reestimation is demonstrated along with an information theoretical justification for the hidden filter model. Chapter V provides an in-depth analysis of phonetic hidden filter Markov modeling approach to speaker recognition. The final chapter offers several research-directed recommendations and conclusions with a brief review of contributions.

II. Background

2.1 Introduction

This chapter introduces the hidden Markov model and several extensions for use in modeling speech and speakers. Given the intra-speaker variability of speech over a set of words, a statistical model which attempts to estimate these variabilities presents the best solution. The HMM makes use of a hidden changing state, where the state may represent some particular spectrum of speech or some dynamics of this spectrum. The next section describes the theory underlying standard HMMs, and the assumptions often made. Next, the assumptions are relaxed to model the dynamics of speech, for both frames of speech and processed features. The last section exemplifies the typical processing of speech for extracting features and analyzes their independence. Lastly, a linear method to extract transitional information of the feature process is provided.

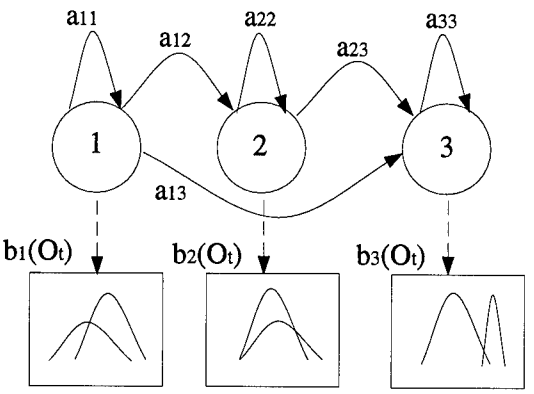
2.2 Statistical Hidden Markov Models

Consider a source system which traverses between N hidden states or characteristic modes, denoting this sequence as $q_1, q_2, \ldots, q_T$, where $q_t \in \{1, 2, \ldots, N\}$. This sequence is a Markov chain and will be assumed to be a discrete first order Markov process. As such its behavior can be described completely by a set of state transition probabilities $\mathbf{A}$ and initial state probabilities Π. Assuming stationarity of this process allows the transitions to be independent of time.

$$\mathbf{A} = (a_{ij}) = P\left(q_t = j | q_{t-1} = i, q_{t-2}, \ldots, q_1\right) = P\left(q_t = j | q_{t-1} = i\right) \tag{1}$$

An *ergodic* model is generally assumed to allow the full set of transitions between all states. Most often in using speech, a restricted set is used. A *left-to-right* model is composed of an upper diagonal $\mathbf{A}$ matrix, and occasionally further restricts skipping states. An example of the standard left-to-right model is shown in Figure 1.

$$\Pi = (\pi_i) = P\left(q_1 = i\right) \tag{2}$$



$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22} & a_{23} \\ 0 & 0 & a_{33} \end{bmatrix}$$

Figure 1. Standard three state left-to-right multivariate Gaussian mixture hidden Markov model. Shown with upper triangular transition matrix **A**. Each state is described by a parametric output density $b_i(O_t)$.

For the left-to-right model, $\pi_1 = 1$ and $\pi_j = 0$, $j > 1$. These will not have to be reestimated. The states of an ergodic model are also characterized by stationary distributions so that

$$\Pi^{\infty} = (\pi_i^{\infty}) = P(q_t = i) \tag{3}$$

At each time, the system generates an observation O_t based on some probabilistic function of the Markov chain. It is this function which is the most important component of the HMM [124]. The output distribution function for each state can be either discrete or continuous. In the discrete case, the distribution function is a set of probabilities associated with each output symbol. Often these symbols relate to a particular codeword of a codebook. Typically, the output function is continuous - a convex combination of multivariate Gaussian densities.

$$\begin{aligned} b_i(O_t) &= \sum_{k=1}^{M} c_{ik} \mathcal{N}(O_t; \bar{\mu}_{ik}, \Sigma_{ik}) \\ &= \sum_{k=1}^{M} c_{ik} \frac{1}{(2\pi)^{d/2} |\Sigma_{ik}|^{1/2}} \exp\left\{ -\frac{1}{2} (O_t - \bar{\mu}_{ik})^T \Sigma_{ik}^{-1} (O_t - \bar{\mu}_{ik}) \right\} \end{aligned} \tag{4}$$

where this density has parameters c_{ik}, $\bar{\mu}_{ik}$, and Σ_{ik}, denoting the mixture weights, mean and covariance for the i-th state and k-th mixture, respectively. This now enables a formal definition for a hidden Markov model.

Definition II.1 *(Hidden Markov Model) A Hidden Markov Model is a probabilistic function of a first order Markov state process, denoted by the triple* $\lambda = (\Pi, \mathbf{A}, \mathbf{B})$ *where* Π *is the* $N \times 1$ *vector of initial state probabilities,* $\mathbf{A}$ *is the* $N \times N$ *matrix of transitions and* $\mathbf{B}$ *is the set of all parameters describing the unconditional output state density for all states. These include*

- μ_{ik}*: mean for state* i*, mixture* k
- Σ_{ik}*: covariance for state* i*, mixture* k
- c_{ik}*: state* i*, mixture* k *weight*

The maximum likelihood estimation of all parameters will be examined in Chapter III. The trained Markov models can be compared to observation sequences, by a decoding process which attempts to uncover the hidden state sequence and provides a likelihood of the observation given the model parameters. Consider an observation sequence, $\mathcal{O} = \{O_1, O_2, \ldots, O_T\}$, with its corresponding hidden state sequence $\mathcal{Q} = \{q_1, q_2, \ldots, q_T\}$ [100]. Making use of the model assumptions, the likelihood of the observation sequence for this state sequence is

$$p(\mathcal{O}|\mathcal{Q}, \lambda) = \prod_{t=1}^{T} p(o_t|q_t, \lambda)$$

which can be expanded using the the Markov property.

$$p(\mathcal{Q}|\lambda) = \pi_{q_1} a_{q_1,q_2} a_{q_2,q_3} \ldots a_{q_{T-1},q_T}$$

Solve for the marginal likelihood as follows.

$$p(\mathcal{O}, \mathcal{Q}|\lambda) = p(\mathcal{O}|\mathcal{Q}, \lambda) p(\mathcal{Q}|\lambda)$$

$$p(\mathcal{O}|\lambda) = \sum_{\mathcal{Q}} p(\mathcal{O}|\mathcal{Q}, \lambda) p(\mathcal{Q}|\lambda) \tag{5}$$

Equation 5 provides the likelihood of a sequence and is used to score how similar a sequence $\mathcal{O}$ compares to a particular model λ. This exact calculation requires on the order of N^T summations. The Viterbi decoding algorithm approximates this quantity by the joint

likelihood of observation and hidden state sequence. This can be accomplished in only N^2T operations.

$$p(\mathcal{O}|\lambda) \approx \max_{\mathcal{Q}} p(\mathcal{O}, \mathcal{Q}|\lambda) \tag{6}$$

Ephraim [81] has shown that the difference between the two approaches is bounded. The logarithm of this last expression, Equation 6, will be used in all classification experiments to score a test observation for a particular speaker model λ. For our research, a speaker will actually be represented by 22 phoneme or subword models. Current speech recognition techniques would create 49 phoneme models or over 3000 context-dependent triphone models for unrestricted vocabulary. An efficient Viterbi decoding method using multiple models and allowing easy grammar constraints is the Token Passing algorithm [133].

2.2.1 Standard Assumptions. Hidden Markov models are providing the most successful methods for automatic speech recognition. Speech is ideally suited, in some respects, to HMM modeling since speech is "quasi-stationary," i.e., the statistics are unchanging over small frames of 30-70 msecs [90]. However, adequate speech recognition performance requires tripling the feature dimensions by concatenating first and second order regression features, indicating the basic HMM model with Gaussian mixtures may be lacking capabilities in capturing the dynamics of the observations. The need for these transitional features can be found in the inherent model assumptions. Many tutorial papers can be found for the standard hidden Markov model [93, 97, 100], where the following assumptions are required.

- **First Order Markov state process** Hidden state sequence conforms to a discrete Markov chain stationary process:

$$p(q_t = j|q_{t-1} = i, q_{t-2}, \ldots, q_1, O_{t-1}, O_{t-2}, \ldots, O_1) = p(q_t = j|q_{t-1} = i) = a_{ij}$$

- **Observation Independence**: Observations are independent of their past values:

$$p(O_t, q_t|q_{t-1}, \ldots, q_1, O_{t-1}, \ldots O_1) = p(O_t, q_t|q_{t-1})$$

- **Current State Dependence**: Observations independent of past observations and also of past states:

$$p(O_t, q_t|q_{t-1}, \ldots, q_1, O_{t-1}, \ldots, O_1) = p(O_t, q_t|q_{t-1}) = a_{ij}p(O_t|q_t)$$

- **Output Probability density family**: Output defined by a mixture of M normal densities:

2.2.2 Removal of Output Independence Assumption. Though hidden Markov models have been the model of choice for the past decade in speech recognition, the assumption of state-conditioned observation independence is not valid. This prompted the development of output densities produced by other stochastic functions of the observations. The earliest known is the Hidden Filter HMM by Poritz [92]. Instead of the simple discrete or continuous normal output density conditioned only on state, this likelihood is conditioned both on state and past observations. Observation frames are assumed generated by an autoregressive source, Equation 7. The general p-th order autoregressive AR(p) model [63, 94] bases the current output on p past outputs. Let an observations O_t be frame of K samples such that $O_t = (x_1, x_2, \ldots, x_K)$.

$$x_t = -\sum_{j=1}^{p} a_j x_{t-j} + e_t = \hat{x}_t + e_t \tag{7}$$

where a_j is the j-th predictor coefficient and the process e_t is typically a Gaussian white noise process with variance σ^2. The term autoregression implies x_t is a linear regression on itself with $\hat{x}_t$ representing the prediction of x_t at time t. This simple model works particularly well for voiced speech segments [27, 90]. Using this linear relation, it is easily seen the probability density function of a sample given past samples has the same density

of e_t, only shifted[1]

$$b_i(x_t|x_{t-1}, x_{t-2}, \ldots, x_{t-p}) = \frac{1}{(2\pi\sigma_i^2)^{1/2}} \exp\{-\frac{1}{2\sigma_i^2}(x_t + \sum_{j=1}^{p} a_j^i x_{t-j})^2\}$$

Surprisingly, the unconditional probability density function for the entire frame O_t has the same functional form as the conditional sample density, since the noise process is independent.

$$\begin{aligned} b_i(O_t) &= \prod_{t=1}^{K} b_i(x_t|x_{t-1}, \ldots, x_{t-p}) \\ &= \frac{1}{(2\pi\sigma_i^2)^{K/2}} \exp\{-\frac{1}{2\sigma_i^2} \sum_{t=1}^{K}(x_t + \sum_{j=1}^{p} a_j^i x_{t-j})^2\} \end{aligned} \tag{8}$$

where σ_i^2 is the noise variance over the K samples within a frame.

These models were further generalized to linear AR *mixture* models by Juang and Rabiner [60] and later used within ergodic structures by Tishby [124]. Equation 8 has an efficient form, first demonstrated by Juang [57]. The output density for an autoregressive frame $O_t = (x_1, x_2, \ldots, x_K)$, for state i described by predictor coefficients $\bar{a}_i = (a_1, a_2, \ldots, a_p)$ and noise variance σ_i is

$$b_i(O_t) = \frac{1}{(2\pi\sigma_i^2)^{K/2}} \exp\{-\frac{1}{2\sigma_i^2}\delta(O_t, \bar{a}_i)\} \tag{9}$$

and

$$\delta(O_t, \bar{a}_i) = r_a(0)r_x(0) + 2\sum_{j=1}^{p} r_a(j)r_x(j) \tag{10}$$

where $\delta(O_t, \bar{a}_i)$ can be considered a distortion or distance metric between a frame O_t and a hidden filter $\bar{a}_i$. The efficiency of this equation is that the frame samples need not be known - only the biased autocorrelation estimate, r_x, of the frame and the autocorrelation of the filter r_a. Equation 9 describes a single mixture of an HMM state. For a state i with

[1]The dilemma we are faced with is notation. All signal processing, statistical modeling uses "a_j" as a predictor, autoregressive or IIR filter coefficient. Also, the hidden Markov literature always uses "a_{ij}" as a transition probability. Since the latter has little significance in this research, it should be clear filters are often discussed.

M mixtures,

$$b_i(O_t) = \sum_{m=1}^{M} c_{im} b_{im}(O_t) \tag{11}$$

this architecture will attempt to model a p-th order filter $\bar{a}_{im}$ for each state i and each mixture m. This description defines the frame autoregressive hidden Markov model, graphically shown in Figure 2.

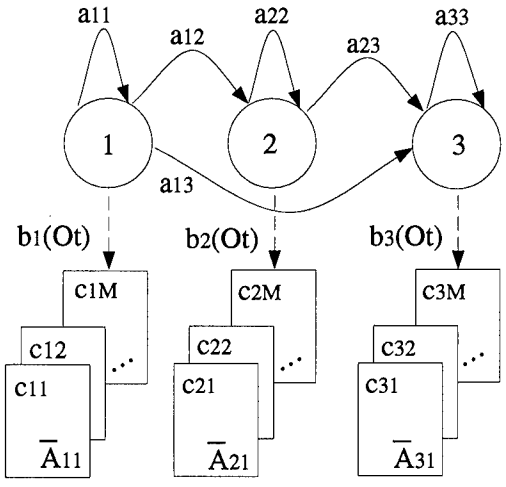


Figure 2. Juang's frame autoregressive mixture extensions to the Poritz hidden filter. While Poritz proposed single filter states, Juang extended to multiple mixtures.

Definition II.2 *(Frame Autoregressive Hidden Markov Model) A frame autoregressive hidden Markov model is a probabilistic function of a first order Markov state process, denoted by the triple* $\lambda = (\Pi, \mathbf{A}, \mathbf{B})$ *where* Π *is the* $N \times 1$ *vector of initial probabilities,* $\mathbf{A}$ *is the* $N \times N$ *matrix of transitions and* $\mathbf{B}$ *is the set of all parameters describing the conditional output state densities. These include:*

- $\bar{A}_{im} = (a_{im1}, a_{im2}, \ldots, a_{imp})$*: p-th order filter coefficients for state i, mixture m*
- σ^2_{im}*: residual error variance for state i, mixture m*
- c_{im}*: state i mixture m weight*

The approach taken by Kenny [66] and later Woodland [131] models the vector-valued features as an linear predictive source. Including a separate mean per state, the

vector observations are assumed generated by

$$O_t = \bar{\mu}(q_t, q_{t-1}) + A_1(q_t, q_{t-1})O_{t-1} + , \ldots , + A_l(q_t, q_{t-1})O_{t-l} + \bar{E}_t \tag{12}$$

where, $\bar{E}_t$ is a multivariate white Gaussian process with covariance $\Sigma(q_t, q_{t-1})$. Note this model uses the notation of "emission on state transition", where the quantities of interest are conditioned on the state pair, (q_t, q_{t-1}). Kenny applied this model to phoneme recognition and examined specific lags l. His results indicated no improvement over standard hidden Markov models. Woodland used the more common "emit on state" assumption with a state model of the form

$$O_t = \bar{\mu} + \sum_{j=1}^{p} A_j(O_{t-j} - \bar{\mu}_j) + \bar{E}_t. \tag{13}$$

This regression is similar to Kenny's model, but with the added offset mean parameters, $\bar{\mu}_j$. He also selects a portion of the residual space to enhance discrimination. The corresponding multivariate output density for state i is given by

$$b_i(O_t) = \frac{1}{(2\pi)^{d/2}|\Sigma_i|^{1/2}} \exp\left\{-\frac{1}{2}\bar{E}_t^T \Sigma_i^{-1} T \bar{E}_t\right\} \tag{14}$$

where the T transformation selects the most discriminating dimensions. Woodland was able to demonstrate better performance by reducing the feature sizes (using the T transform) when applied to a small "E-set"[2].

To date, the only application of the two previous models have focused on linear prediction using specific lags (forward or backward) and applied to phoneme recognition. Since knowledge of the most important lags is unavailable, either for speech or speaker recognition, a full autoregressive should will be examined. The multivariate conditional output density defined in Equation 14, without the mean offset $\bar{\mu}_j$ and transform matrix T, will be defined as the vector autoregressive hidden Markov model.

[2]The E-set typically consists of the small English alphabet (B,C,D,E,G,P,T,V).

Definition II.3 *(Vector Autoregressive Hidden Markov Model) A vector autoregressive hidden Markov model is a multivariate probabilistic function of a first order Markov state process, denoted by the triple* $\lambda = (\Pi, \mathbf{A}, \mathbf{B})$ *where* Π *is the* $N \times 1$ *vector of initial probabilities,* $\mathbf{A}$ *is the* $N \times N$ *matrix of transitions and* $\mathbf{B}$ *is the set of all parameters describing the conditional output state densities. These include*

- $\bar{\mu}_{im}$*: mean vector for state* i*, mixture* m
- $B_{im} = (A_{i1}, A_{i2}, \ldots, A_{ip})$*: p-th order filter matrices for state* i*, mixture* m
- Σ_{im}*: multivariate noise covariance for state* i*, mixture* m
- c_{im}*: state* i *mixture* m *weight*

This section described the research in linear dynamic systems, applied most often to speech recognition. The common philosophy to all these approaches examines the statistics of the observations within a state. Standard hidden Markov models assume features are generated as a constant state mean with any observation errors accounted by the covariance estimate. Hidden filters, on the other hand, account for the (prediction) error after a linear regression is applied. The next section examines the approach when linearity is removed from the state model.

2.2.3 Nonlinear Hybrid Markov Models. Several researchers have recently combined the pertinent features of HMMs and multilayer perceptrons or neural networks. The HMM provides an explicit discrete state model, including efficient optimization strategies of model parameters; the neural networks provide nonlinear input-output mappings, and discriminative class estimation.

The first complete treatment of HMM hybrids is the recent work by Bourlard and Morgan [13, 14]. Their presentation of the subject of HMMs is based on variations of "local contribution", which they define as the joint probability of the state and observation conditioned on all previous states and observations (and the current set of weights W).

$$p(q_t = i, O_t | q_1, \ldots, q_{t-1}, O_1, \ldots, O_{t-1}, W) \tag{15}$$

Hybrid techniques then make various simplifying assumptions or relaxations of this likelihood and attempt to approximate it through MLPs or recurrent architectures. Generalization of the local contribution in Equation 15 can use past and future observations. Bourlard and Morgan use a feedforward MLP to approximate this likelihood by training with state desired values.

Neural networks can also be trained to approximate both nonlinear autoregressive (NAR) and nonlinear autoregressive moving average (NARMA) through gradient descent learning. If the stochastic inputs are unknown, as is usually the case, they may be approximated by using the prediction residual of the previous prediction [24]. All these stochastic time series models can be extended, in theory, with a Markov structure. One such NAR/ HMM hybrid approach was developed by Levin [70] called the "Hidden Control Neural Network" and later detailed in [71]. A few enhancements and applications by other researchers have also been published [39, 121, 122] and shown successful.

During the past three years, the similarities of hidden Markov models and recurrent architectures have been studied. These interpretations have been accomplished by Bridle and Kehagias [15, 64, 65] and explicitly used for phoneme recognition by Robinson [105, 106, 107, 108]. The recurrent architecture can be shown as a non-linear state-space model. Robinson, for example, uses these networks to retain context in the hidden activation nodes. Standard HMM processing can then be integrated on the back-side for hierarchical word modeling, state-duration modeling and overall word likelihood calculation. Like Bourlard and Morgan, Robinson requires specialized hardware to calculate the error gradients during training, due to the extensive amounts of training data needed for reliable speaker independent subword modeling.

2.3 General Hidden Filter Framework

Extensions to the standard Gaussian mixture HMM have developed recently to add context and discriminative capabilities. Context has been attempted through the use of linear prediction, whereas discriminative learning is provided by feedforward MLPs or feedback recurrent networks. Other related Markov modulated sources have included noise corrupted polynomials [30] and mixed state-observation approaches [45]. Each method, to

date, fits into the general framework of Markov-modulated dynamic systems or *Generalized Hidden Filter Markov Models (GHFMM)*. The underlying state probabilistic functions may be linear or non-linear, conditional or non-conditional, and causal or noncausal. Figure 3 shows all HMM approaches in a new unified framework.

This research indicates a wide range of applicability to modeling general, possibly even chaotic, time series. Many applications requiring prediction, monitoring of dynamic systems or classification of noise corrupted observations potentially benefit through the use for GHFMMs. This research will specifically examine classification of acoustic signals, which can be considered noise corrupted observations from the a particularly personal dynamic system - human speech production.

Chapter III will demonstrate that hidden filter Markov models can be applied to raw speech samples, frames of speech or processed features. The following sections in this chapter examine the typical processing of raw speech into features which is often performed prior to speech or speaker recognition. The last section demonstrates the feature extraction procedure, then it will be shown these features are highly correlated.

2.4 Feature Analysis

Standard speech processing techniques were used to extract features from the raw samples. It should be noted that no "best" feature set has been determined for speaker recognition tasks. Since speaker modeling has such a rich history - one which parallels speech recognition - many popular features have been examined. [1, 2, 4, 11, 25, 47, 53, 61, 62, 77, 114, 120]. Recent studies for open set speaker identification, on both high quality TIMIT [56, 84] and tactical radio GREENFLAG [40, 41] databases indicate that no one feature may prove optimal in all cases [91].

2.4.1 Signal Processing of Speech. Features are extracted using many standard signal processing techniques [99, 98, 59, 134]. The speech signal traverses through many stationary points with specific spectral signatures. It is these short-time signatures which separate phones or phonemes. Accordingly, a *phone* is the smallest individual acoustic unit, in the field of phonetics [87, 90]. In the study of descriptive linguistics, the small-

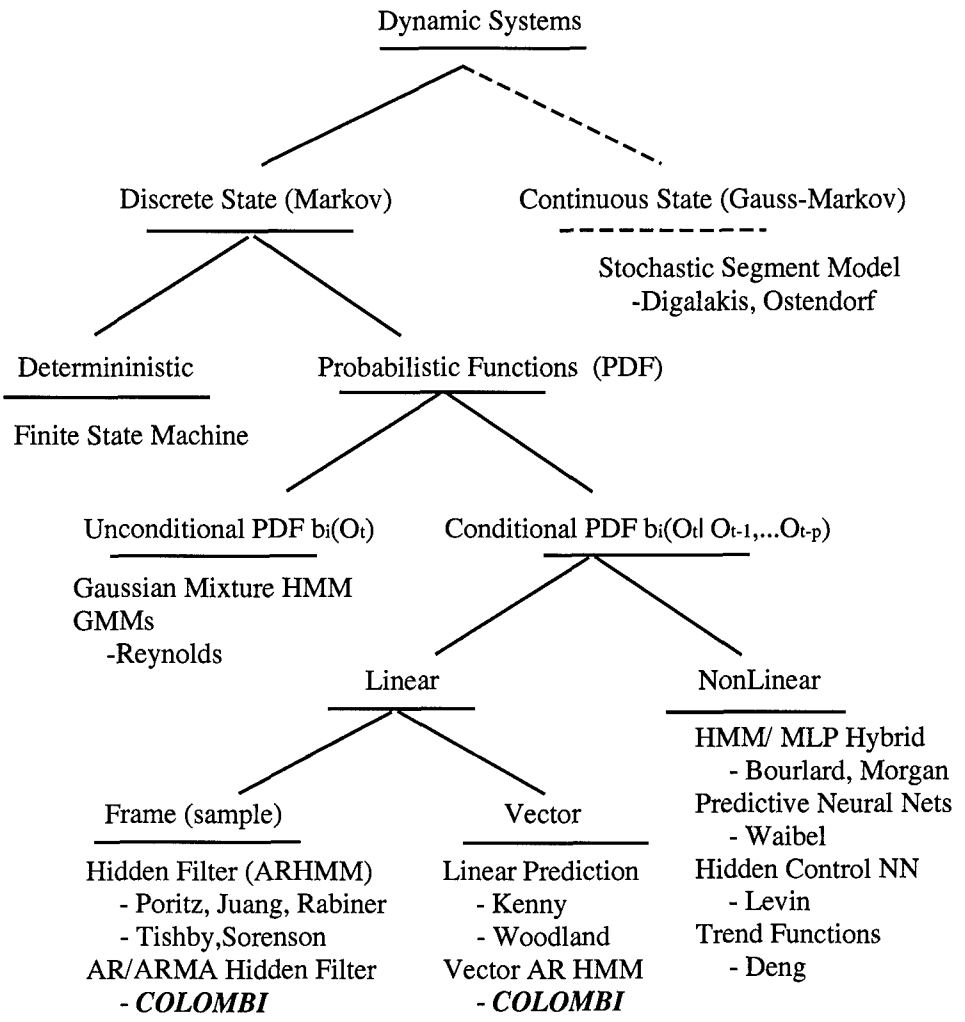


Figure 3. Architecture for Generalized Hidden Filter Markov Models (GFHMM).

est unit is the *phoneme.* A phoneme is that entity which must be altered to change word meaning, i.e., "bat" and "cat" differ only in the phoneme /b/. Since there is much overlap between the two fields of study, this research will use Parson's definition [90].

Definition II.4 *(Phoneme) A Phoneme is the smallest acoustic unit in a given language that is able to change word meaning. A model of this unit will be referred to as a phoneme or monophone model.*

Modeling of speaker dependencies within phoneme acoustics will be explored. As such, labeled phonetic data will be required for initial training and separate phoneme models will be created for all speakers. As will be discussed in Chapter V, testing will string together the correct phoneme models relating to the particular phrase prompted.

All raw data consist of 8 kHz sampled speech. The original signal conditioning and acquisition were designed by Campbell [67] to provide bandwidth and linear phase up to 3.8 kHz. The resulting bandpass filter response models the DoD's STU-III secure voice terminal's input characteristics very closely.

Analysis frames of 20 msec are first pre-emphasized to remove lip radiation effects by a simple high pass filter. Then, a Hamming window is applied to decrease frame edge effects in the Fourier transform. Frames are analyzed every 10 msec. If one stops at this point and displays the magnitude of the resulting short-term Fourier transform, a *spectrogram* results (See Figure 4 for an example YOHO database combination lock utterance).

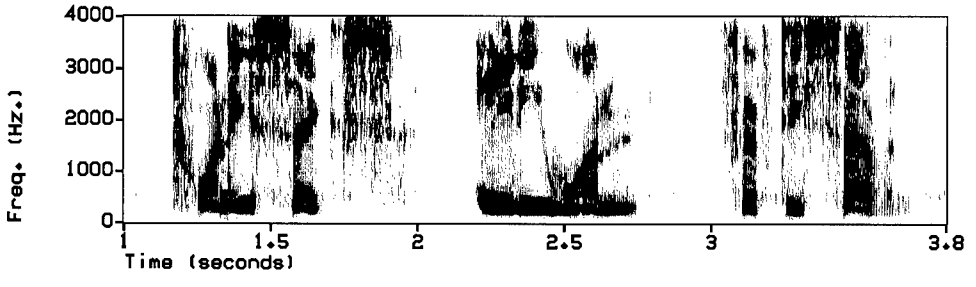


Figure 4. Spectrogram of YOHO combination lock phrase, "Forty One, Sixty Nine, Fifty Six".

The magnitude transform coefficients are correlated with each of 24 triangular filters spaced linearly up to 1 KHz and logarithmic thereafter, see Figure 5. On a Mel scale, the filters are spaced linearly,

$$Mel(f) = 2595 \log_{10}(1 + \frac{f}{700}).$$

This nonlinear frequency analysis models human perception [90] and empirically improves speech recognition performance [1, 2, 3]. The logarithm of the energy outputs from these filters, denoted m_j, are the Mel frequency spectral coefficients. To reduce and decorrelate

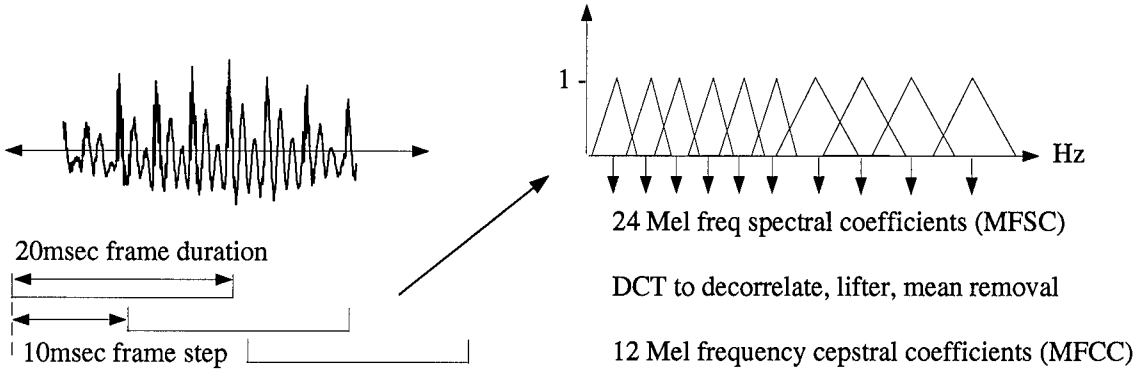


Figure 5. Typical Speech Processing/ Feature extraction.

these $N = 24$ coefficients, a Discrete Cosine Transform is applied which reduces the features to 12 Mel frequency cepstral coefficients, c_i.

$$c_i = \sqrt{\frac{2}{N}} \sum_{j=1}^{N} m_j \cos(\frac{\pi i}{N}(j - 0.5))$$

A raised cosine is applied to account for noisy low and high order coefficients. This filtering process, called *liftering*, uses the following weighting for $L = 20$.

$$c_i' = (1 + \frac{L}{2} \sin \frac{\pi i}{L}) c_i$$

Lastly, to remove channel effects, removal of the Mel frequency cepstral time average is performed. This homomorphic deconvolution [88] compensates for microphone and other long-term recording effects present in the signal. The logarithm of the frame energy is appended to all cepstral vectors. This value is normalized by the maximum energy present in the utterance. Thus, the baseline feature contains 13 coefficients.

2.4.2 Cepstral Characteristics. Digalakis [32] recently examined linear and non-linear regression of the cepstral coefficients within and between phoneme segments. His conclusions were that within phoneme segments, a linear regression (model) can explain up to 88% of the variance in predicting the next cepstral vector for most frames. However, between phonemes the linear model breaks down.

The following graph show the scatterplots for the first cepstral coefficient at various lags l, see Figures 6. Each subplot presents $c_1(t)$ against $c_1(t+l)$. The inset is the calculated correlation coefficient for this data. These indicate that close frames, separated up to 70 msec, are not statistically independent in time. Also, the scatterplots appear Gaussian through the seventh lag. The statistical independence assumption of standard hidden Markov models is obviously not valid and must be removed. Digalakis suggests a linear model will be appropriate and relevant for phoneme modeling. For larger subword models (syllables, diphones, etc.) possibly the hybrid HMM/ neural approaches are more suitable.

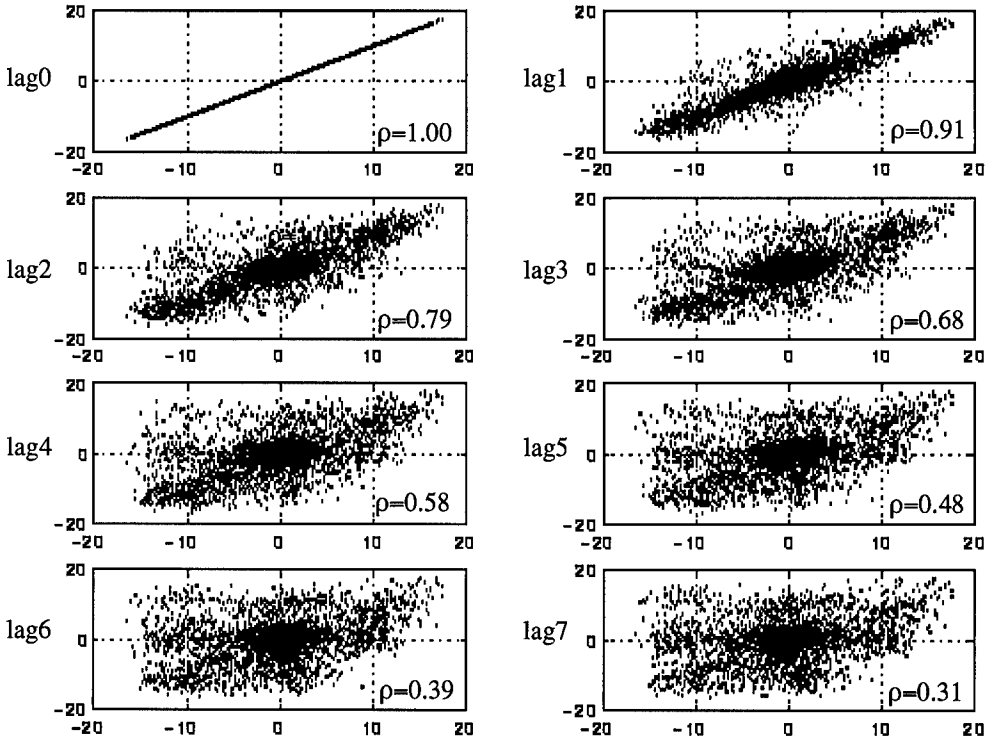


Figure 6. Scatterplot of first cepstral coefficient $c_1(t)$ for lags 0-7. Each point within a subplot is the order pair $(c_1(t), c_1(t+l))$ where l is lag. Inset within each subplot is the correlation coefficient over all data.

2.4.3 Transitional Coefficients. One method of modeling transitional effects in the observations is through the use of regression coefficients often denoted by Δ_t. These

coefficients are found by fitting the best linear line through a set of observations

$$(O_{t-W}, O_{t-W+1}, \ldots, O_{t-1}, O_t, O_{t+1}, \ldots, O_{t+W-1}, O_{t+W})$$

and passing through the point O_t, using a window of width $\pm W$. The equation for this linear line is $y = a \cdot k + O_t$ which is a shift of the origin to the place t with an unknown slope, a. Define the squared error cost criterion to be

$$J = \sum_{k=-W}^{W} (O_{t+k} - y)^2 = \sum_{k=-W}^{W} (O_{t+k} - ak - O_t)^2.$$

The value of the slope a which minimizes this quadratic occurs at a $\partial J/\partial a = 0$. Thus

$$\begin{aligned} 0 &= \sum_{k=-W}^{W} 2(O_{t+k} - ak - O_t)(-k) \\ &= \sum_{0} k(O_{t+k} - ak - O_t) + \sum_{k=-1}^{-W} k(O_{t+k} - ak - O_t) + \sum_{k=1}^{W} k(O_{t+k} - ak - O_t) \end{aligned}$$

and letting $l = -k$,

$$\begin{aligned} 0 &= 0 - \sum_{l=1}^{W} (-l)(O_{t-l} + al - O_t) + \sum_{k=1}^{W} k(O_{t+k} - ak - O_t) \\ &= \sum_{l=1}^{W} (l)(-O_{t-l} - al + O_t) + \sum_{k=1}^{W} k(O_{t+k} - ak - O_t) \end{aligned}$$

and combining summations,

$$\begin{aligned} 0 &= \sum_{k=1}^{W} k(O_{t+k} - O_{t-k} - 2ak) = \sum_{k=1}^{W} k(O_{t+k} - O_{t-k}) - 2a \sum_{k=1}^{W} k^2 \\ \Delta_t &\equiv a = \frac{\sum_{k=1}^{W} k(O_{t+k} - O_{t-k})}{2 \sum_{k=1}^{W} k^2}. \end{aligned} \tag{16}$$

This linear least squared error solution to the slope is the standard regression coefficient found in calculating "Delta", and subsequently "Delta-Delta", coefficients in speech recognition. The approximation to this regression, called the *differenced* coefficient is sometimes also used, Equation 17.

$$\delta_t \equiv O_{t+W} - O_{t-W} \tag{17}$$

2.5 Conclusion

This chapter has developed a general hidden Markov model framework, and reviewed the necessary linear submodels within each state. The motivation to extend the current techniques is better acoustic modeling of phoneme context and correlation. The full potential of temporal stochastic models has not yet been applied to the speaker modeling problem. To date, published material for speaker modeling has used frame based linear prediction within an ergodic HMM structure. It will be demonstrated that ergodic models greatly account for the effects due to language, rather than the speaker. To circumvent language modeling, speaker dependent phoneme hidden filter modeling is proposed.

The phoneme continues to be the popular subword unit for speech. The acoustics within a phoneme segment are relatively stationary and as such, this research will focus on their speaker dependent modeling. This approach provides an inherent text-independent application since the set of all phonemes can be modeled. For experimentation, the YOHO database will be used which constrains utterances to combination lock phrases, which only need a subset of the full phoneme acoustic space. The next chapter proposes new extensions and develops the reestimation of these extensions to the baseline hidden Markov model. Methods will be shown applicable to raw samples of a signal, frames of samples or a sequence of processed feature vectors.

III. Model Reestimation

3.1 Introduction

In their book, "Connectionist Speech Recognition", Bourlard and Morgan write,

> For speech recognition in particular, it is important to improve our models to better take into account the dynamical properties of the speech process. In this framework, methods should be developed to use more contextual information for classification.

This chapter first presents the theory behind hidden Markov model reestimation, then provides the reestimation of hidden filter models. The choice of hidden filters comes naturally from the long accepted speech production model [5, 95, 99]. Speech can be grossly viewed as source signal (either noise-like or periodic) convolved with a rational filter describing the vocal tract. While rational filter models such as autoregressive (AR) and autoregressive-moving average (ARMA) have a rich history in spectral estimation, signal prediction, speech processing and economics, their effectiveness within a Markov modulated structure for modeling speakers is yet unknown.

This chapter develops three levels of hidden filters and provides their reestimation procedures. The first level models the sample or raw observations. While this may be the most efficient [46], it also requires extensive calculations for both reestimation and decoding, due to the amount and frequency of the data. The next level combines observations into frames. Efficiency is gained since the actual raw samples are not needed in reestimation - only the autocorrelation of the samples. Also the frequency of reestimation has been reduced substantially. The last level of modeling occurs on some processed spectral representation of these frames. Methods such as the mean-subtracted, liftered, Mel frequency cepstral vectors have been researched extensively to provide a compact, decorrelated representation of the log-spectrum. This last level of modeling reduces the occurrence, number and complexity of the Baum-Welch algorithm and for this reason, it will be the primary technique applied to large speaker recognition experiments.

3.2 Hidden Markov Model Reestimation

The hidden Markov model reestimation provides maximum likelihood parameter estimates given a set of training sequences. The reestimation will be solved for a single observation sequence and is easily extended for multiple training sequences. Two probabilistic quantities, which are often used throughout standard HMM model reestimation, are the forward and backward variables. These take on great significance in deciding which observations get used to reestimate a particular states' parameters. These initial calculations (*Forward-Backward* algorithm), along with the final parameter updates, are collectively called the *Baum-Welch* algorithm.

3.2.1 Forward-Backward Variables. From [96, 112], define

$$\alpha_t(i) = p(O_1, \ldots, O_t, q_t = i|\lambda)$$

as the joint likelihood of the observation $O_1, \ldots, O_t$ and state $q_t = i$ given the model λ. The derivation for $\alpha_t(i)$ is inductive (see Appendix A). Two important points are that $\alpha_{t+1}(i)$ is a function of the previous α_t

$$\alpha_{t+1}(j) = b_j(O_{t+1}) \sum_{i=1}^{N} a_{ij}\alpha_t(i)$$

and the forward variable evaluated at the last time sample provides the total likelihood of the observation sequence given this current model

$$\begin{aligned} p(O_1 \ldots O_T|\lambda) &= \sum_{i=1}^{N} p(O_1 \ldots O_T, q_T = i|\lambda) \\ &= \sum_{i=1}^{N} \alpha_T(i). \end{aligned}$$

The backward algorithm is also inductive, derived in a similar manner. Let

$$\beta_t(i) = p(O_{t+1} \ldots O_T|q_t = i, \lambda)$$

where the backward variable, $\beta_t(i)$ is the likelihood of observing the partial sequence $O_{t+1} \ldots O_T$ given the current state $q_t = i$ and the model λ. The inductive calculation of $\beta_t(i)$ (see Appendix A) becomes

$$\beta_t(i) = \sum_{i=1}^{N} \pi_i b_i(O_1)\beta_1(i).$$

The total likelihood can be evaluated at $t = 1$.

$$p(O_1 \ldots O_T|\lambda) = \sum_{i=1}^{N} \pi_i b_i(O_1)\beta_1(i)$$

3.2.2 State Likelihood. Lastly, a related quantity is denoted by [95, 96]

$$\xi_t(i,j) = p(q_{t+1} = j, q_t = i|O_1 \ldots O_T, \lambda) = \frac{p(q_{t+1} = j, q_t = i, O_1 \ldots O_T|\lambda)}{p(O_1 \ldots O_T|, \lambda)}$$

which can be expressed in the forward and backward quantities as

$$\xi_t(i,j) \quad = \quad \frac{\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{p(O_1 \ldots O_T|, \lambda)}.$$

The following single state likelihood is most useful in practice, often denoted by $\gamma_t(i)$.

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i,j) = p(q_t = i|O_1 \ldots O_T)$$

3.2.3 Baum Auxiliary Function. The goal of the training phase for HMMs is to model a set of observations with a maximum likelihood set of parameters representing the underlying Markov process and probabilistic function of that process. Denote the model by $\lambda = (\Pi, A, B)$. Given a set of observations, $(O_1, O_2, \ldots, O_T)$, search over all $\bar{\lambda} \in \bar{\Lambda}$ to maximize the likelihood, $p(O_1, O_2, \ldots, O_T|\bar{\lambda})$. Brute force approaches would search for critical points of this likelihood such that various probabilistic constraints of $\bar{\lambda}$ are satisfied, often using Lagrange techniques [100].

A better approach is the Dempster [28] Expectation Maximization (EM) algorithm, developed for maximum likelihood estimation with missing data [28]. The missing data for

the HMM problem are the unknown (hidden) state sequence. The EM algorithm solves for the maximum likelihood model by first defining the *Auxiliary Function*, $Q(\lambda, \bar{\lambda})$, which is a function of both the current model λ and a re-estimated model $\bar{\lambda} = (\bar{\Pi}, \bar{A}, \bar{B})$.

$$Q(\lambda, \bar{\lambda}) = \sum_{q_0, q_1, \ldots, q_T} p(O_1 \ldots O_T, q_1, \ldots, q_T|\lambda) \log p(O_1 \ldots O_T, q_1, \ldots, q_T|\bar{\lambda}) \qquad (18)$$

The properties which make this optimization procedure so attractive are the following:

- If $Q(\lambda, \bar{\lambda}) \geq Q(\lambda, \lambda)$ then $p(O_1 \ldots O_T|\bar{\lambda}) \geq p(O_1 \ldots O_T|\lambda)$
- For a broad class of models, Q has a single global maximum - true for a single normal density.

First, $p(O_1 \ldots O_T|\lambda)$ is usually written as [100]

$$\begin{aligned} p(O_1 \ldots O_T|\lambda) &= \sum_{q_1, q_2 \cdots, q_T} p(O_1 \ldots O_T|q_1, \ldots, q_T, \lambda) p(q_1, \ldots, q_T|\lambda) \\ &= \pi_{q_1} b_{q_1}(O_1) a_{q_1 q_2} b_{q_2}(O_2) \cdots a_{q_{T-1} q_T} b_{q_T}(O_T). \end{aligned}$$

Expanding the joint likelihood from Equation 18,

$$\begin{aligned} \log p(O_1 \ldots O_T, q_1, \ldots, q_T|\bar{\lambda}) &= \log \left[\pi_{q_1} b_{q_1}(O_1) a_{q_1 q_2} b_{q_2}(O_2) \cdots a_{q_{T-1} q_T} b_{q_T}(O_T)\right] \\ &= \log \pi_{q_1} + \sum_{t=1}^{T} \log a_{q_{t-1} q_t} + \sum_{t=1}^{T} \log b_{q_{t-1}}(O_t) \end{aligned}$$

then,

$$\begin{aligned} Q(\lambda, \bar{\lambda}) = & \sum_{q_1, q_2, \ldots, q_T} p(O_1 \ldots O_T, q_1, \ldots, q_T|\lambda) \log \pi_{q_1} \\ & + \sum_{q_1, q_2, \ldots, q_T} p(O_1 \ldots O_T, q_1, \ldots, q_T|\lambda) \sum_{t=1}^{T-1} \log a_{q_t q_{t+1}} \\ & + \sum_{q_1, q_2, \ldots, q_T} p(O_1 \ldots O_T, q_1, \ldots, q_T|\lambda) \sum_{t=1}^{T} \log b_{q_t}(O_t) \end{aligned}$$

and by defining subfunctions,

$$Q(\lambda, \bar{\lambda}) \;=\; Q_{\Pi}(\lambda, \bar{\Pi}) + Q_A(\lambda, \bar{A}) + \sum_{i=1}^{N} Q_b(\lambda, \bar{B}_i)$$

where the use of Kronecker delta function, $\delta(\ )$, can be used to sample a particular state.

$$\begin{aligned}
Q_{\Pi}(\lambda, \bar{\Pi}) &= \sum_{q_1, q_2, \ldots, q_T} p(O_1 \ldots O_T, q_1, \ldots, q_T|\lambda) \log \pi_{q_1} \\
&= \sum_{q_1, q_2, \ldots, q_T} p(O_1 \ldots O_T, q_1, \ldots, q_T|\lambda) \sum_{i=1}^{N} \log \pi_i \delta(q_1 - i) \\
&= \sum_{i=1}^{N} p(O_1 \ldots O_T, q_1 = i|\lambda) \log \pi_i \\
Q_A(\lambda, \bar{A}) &= \sum_{q_1, q_2, \ldots, q_T} p(O_1 \ldots O_T, q_1, \ldots, q_T|\lambda) \sum_{t=1}^{T-1} \log a_{q_t q_{t+1}} \\
&= \sum_{q_1, q_2, \ldots, q_T} p(O_1 \ldots O_T, q_1, \ldots, q_T|\lambda) \sum_{t=1}^{T-1} \sum_{i=1}^{N} \sum_{j=1}^{N} \log a_{ij} \delta(q_t - i) \delta(q_{t+1} - j) \\
&= \sum_{t=1}^{T-1} \sum_{i=1}^{N} \sum_{j=1}^{N} p(O_1 \ldots O_T, q_{t+1} = j, q_t = i|\lambda) \log a_{ij} \\
Q_b(\lambda, \bar{B}_i) &= \sum_{q_1, q_2, \ldots, q_T} p(O_1 \ldots O_T, q_1, \ldots, q_T|\lambda) \sum_{t=1}^{T} \log b_i(O_t) \delta(q_t - i) \\
&= \sum_{t=1}^{T} p(O_1 \ldots O_T, q_t = i|\lambda) \log b_i(O_t)
\end{aligned}$$

It is readily noted that the auxiliary function can be maximized individually for $\bar{A}, \bar{\Pi}$ and the output density parameters contained in $\bar{B}$. Scaling the Q-function by $p(O_1 \ldots O_T|\lambda)$ results in the following (shown for Q_b only), where $\gamma_t(i)$ is a product of the Forward-Backward algorithm.

$$\begin{aligned}
Q_b(\lambda, \bar{B}_i) &= \sum_{t=1}^{T} p(O_1 \ldots O_T, q_t = i|\lambda) \log b_i(O_t) / p(O_1 \ldots O_T|\lambda) \\
&= \sum_{t=1}^{T} p(q_t = i|O_1 \ldots O_T, \lambda) \log b_i(O_t) = \sum_{t=1}^{T} \gamma_t(i) \log b_i(O_t) \qquad (19)
\end{aligned}$$

The update equations are found by examining critical points of this scaled Q-function. The solution for the output density parameters of a single normal yields

$$\begin{aligned} \bar{\mu}_i &= \frac{\sum_{t=1}^{T}\sum_{j=1}^{N}\alpha_{t-1}(i)a_{ij}b_j(O_t)\beta_t(j)O_t}{\sum_{t=1}^{T}\sum_{i=1}^{N}\alpha_{t-1}(i)a_{ij}b_i(O_t)\beta_t(j)} \\ &= \frac{\sum_{t=1}^{T}\gamma_t(i)O_t}{\sum_{t=1}^{T}\gamma_t(i)} \end{aligned} \tag{20}$$

and similarly,

$$\begin{aligned} \bar{\sigma^2}_j &= \frac{\sum_{t=1}^{T}\sum_{j=1}^{N}\alpha_{t-1}(i)a_{ij}b_j(O_t)(O_t-\bar{\mu}_j)(O_t-\bar{\mu}_j)\beta_t(j)}{\sum_{t=1}^{T}\sum_{j=1}^{N}\alpha_{t-1}(i)a_{ij}b_j\beta_t(j)} \\ &= \frac{\sum_{t=1}^{T}\gamma_t(i)(O_t-\bar{\mu}_j)^2}{\sum_{t=1}^{T}\gamma_t(i)}. \end{aligned} \tag{21}$$

Note that when the number of states equals one, then $\gamma_t(1) = 1$ for all t, and Equations 20 and 21 are the maximum likelihood estimates for a mean and covariance of a random sample. Note also these equations are all functions of the Forward-Backward variables, which in turn are derived from the current model λ. While this holds for single Gaussian densities, multiple mixtures may be estimated providing a richer, statistical model for each state.

This section has presented the standard Baum-Welch algorithm for the output density parameters, μ_i, σ_i^2 and their mixture extensions. We have purposely not examined the transition matrix or the initial state probabilities, because all further techniques and models will not change their reestimation. The derivation can be found in Rabiner [95, 96]. The scope of the remaining sections within this chapter full examines the assumptions of the output density functions, $b_i(O_t)$.

3.3 Hidden Filter Markov Model Reestimation

Standard Markov models describe observations as noisy realizations of a constant signal for each state. This research examines models describing linear dynamic systems for each state. These hidden filters may be applied at various levels, based on the nature of the dynamics. The first level applies to the actual samples themselves. For voiced and

some unvoiced speech signals, based on rational polynomial source models, this appears quite appropriate. Also, some glottal-stop consonants last for only a few microseconds, shorter than the typical frame length.

3.3.1 Yule-Walker Approach. The Yule-Walker equations, known also as the Wiener-Hopf or normal equations, provide the maximum likelihood estimate (MLE) for the predictor coefficients assuming a random scalar process generated by

$$O_t = -\sum_{k=1}^{p} a_k O_{t-k} + e_t \tag{22}$$

where the a_k is the kth autoregressive or predictor coefficient and e_t is assumed to be an innovations sequence assumed a white noise process with zero mean and variance σ^2. The solution to the filter coefficients is a set of linear equations given by,

$$-\begin{bmatrix} r_o(1) \\ r_o(2) \\ \vdots \\ r_o(p) \end{bmatrix} = \begin{bmatrix} r_o(0) & r_o(1) & \cdots & r_o(p-1) \\ r_o(-1) & r_o(0) & \cdots & r_o(p-2) \\ \vdots & & \ddots & \vdots \\ r_o(-p+1) & r_o(-p+2) & \cdots & r_o(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} \tag{23}$$

which uses the biased autocorrelation estimate for a frame of K samples,

$$r_o(i) = \frac{1}{K-i} \sum_{j=1}^{K-i} O_j O_{j+i}.$$

The maximum likely noise variance is obtained by using the MLE filter coefficients.

$$\sigma^2 = r_o(0) + \sum_{k=1}^{p} r_o(k) a_k$$

Several variations of these equations exist (such as covariance, modified covariance, or Burg) which make assumptions concerning data outside the frame boundary or use of data within the frame [27, 63, 123]. This set of equations will have a similar counterpart for each individual state of an HMM.

3.3.2 Reestimation of Zero Mean AR Filters. This section details the reestimation of hidden filter Markov models using zero-mean observations, with no framing. The reestimation assumes each state is described by a zero mean autoregression of the form

$$O_t = -\sum_{k=1}^{p} a_{ik} O_{t-k} + e_t,$$

where $e_t \sim \mathcal{N}(0, \sigma_i^2)$. The logarithm of the output density for each Markov state i is given by

$$\log b_i(O_t) = -(1/2)\log 2\pi - (1/2)\log \sigma_i^2 - \frac{1}{2\sigma_i^2}(O_t + \sum_{k=1}^{p} a_{ik} O_{t-k})^2. \tag{24}$$

Solving for the gradient of the auxiliary function, which equals zero at a critical point (see Equation 19)

$$\partial Q_b(\lambda, \bar{B}_i)/\partial a_{il} = \sum_{t=1}^{T-1} \gamma_t(i)[\frac{1}{2\sigma_i^2}(O_t + \sum_{k=1}^{p} a_{ik} O_{t-k}) 2 O_{t-l}] = 0.$$

Typical of linear systems, we solve a set of p simultaneous equations for a_{ik},

$$-\sum_{t=1}^{T-1} \gamma_t(i) O_t O_{t-l} = \sum_{t=1}^{T-1} \gamma_t(i) \sum_{k=1}^{p} a_{ik} O_{t-k} O_{t-l}, \qquad \forall l = (1, 2, \ldots, p) \tag{25}$$

which is reminiscent of the autocorrelation method, weighted by the state likelihood $\gamma_t(i)$. Solving these equations provides the maximum likelihood estimate of the a_{ik} filter coefficients for each state. The noise variance σ_i^2 is then solved using these values of $\hat{a}_{ik}$.

$$\sigma_i^2 = \sum_{t=1}^{T-1} \gamma_t(i)(O_t + \sum_{k=1}^{p} \hat{a}_{ik} O_{t-k})^2 / \sum_{t=1}^{T-1} \gamma_t(i) \tag{26}$$

If the same noise is present, or assumed present across all states, then

$$\begin{aligned} \sigma^2 &= \sum_{i=1}^{N} \sum_{t=1}^{T-1} \gamma_t(i)(O_t + \sum_{k=1}^{p} \hat{a}_{ik} O_{t-k})^2 / \sum_{t=1}^{T-1} \gamma_t(i) \\ &= \frac{1}{T-1} \sum_{i=1}^{N} \sum_{t=1}^{T-1} \gamma_t(i)(O_t + \sum_{k=1}^{p} \hat{a}_{ik} O_{t-k})^2. \end{aligned}$$

3.3.3 Reestimation of non-Zero Mean AR Filters. Applying a similar approach used by Kenny [66] on a vector process, instead let the sample observations have state dependent bias, μ_i. The logarithm of the output density becomes,

$$\log b_i(O_t) = -(1/2)\log 2\pi - (1/2)\log \sigma_i^2 - \frac{1}{2\sigma_i^2}(O_t - \mu_i + \sum_{k=1}^{p} a_{ik} O_{t-k})^2. \qquad (27)$$

Solving for the gradients of Q with respect to both a_{il} and μ_i, and critical values yields

$$\begin{aligned} \partial Q_b(\lambda, \bar{B}_i)/\partial a_{il} &= \sum_{t=1}^{T-1} \gamma_t(i)[(O_t - \mu_i + \sum_{k=1}^{p} a_{ik} O_{t-k}) O_{t-l}] = 0 \\ \partial Q_b(\lambda, \bar{B}_i)/\partial \mu_i &= \sum_{t=1}^{T-1} \gamma_t(i)(O_t - \mu_i + \sum_{k=1}^{p} a_{ik} O_{t-k}) = 0. \end{aligned}$$

or shown in vector-matrix notation,

$$\sum_{t=1}^{T-1} \gamma_t(i) \begin{bmatrix} O_t O_{t-1} \\ O_t O_{t-2} \\ \vdots \\ O_t O_{t-p} \\ O_t \end{bmatrix} = \sum_{t=1}^{T-1} \gamma_t(i) \begin{bmatrix} O_{t-1}O_{t-1} & O_{t-2}O_{t-1} & \cdots & O_{t-p}O_{t-1} & O_{t-1} \\ O_{t-1}O_{t-2} & O_{t-2}O_{t-2} & \cdots & O_{t-p}O_{t-2} & O_{t-2} \\ \vdots & & \ddots & \vdots & \vdots \\ O_{t-1}O_{t-p} & O_{t-2}O_{t-p} & \cdots & O_{t-p}O_{t-p} & O_{t-p} \\ O_{t-1} & O_{t-2} & \cdots & O_{t-p} & 1 \end{bmatrix} \begin{bmatrix} a_{i1} \\ a_{i2} \\ \vdots \\ a_{ip} \\ \mu_i \end{bmatrix}.$$

The noise variance σ_i^2 is then estimated using the maximum likelihood values of $\bar{\mu}_i$ and $\bar{a}_{ik}$.

$$\partial Q_b(\lambda, \bar{B}_i)/\partial \sigma_i^2|_{\bar{a}_i, \bar{\mu}_i} = 0$$

$$\sigma_i^2 = \sum_{t=1}^{T-1} \gamma_t(i)(O_t - \bar{\mu}_i + \sum_{k=1}^{p} \bar{a}_{ik} O_{t-k})^2 / \sum_{t=1}^{T-1} \gamma_t(i) \qquad (28)$$

For a model which used the same driving statistics across all states [31],

$$\begin{aligned} \partial Q_b(\lambda, \bar{B}_i)/\partial \sigma^2|_{\bar{a}_i, \bar{\mu}_i} &= 0 \\ \sigma^2 &= \sum_{i=1}^{N} \sum_{t=1}^{T-1} \gamma_t(i)(O_t - \bar{\mu}_i + \sum_{k=1}^{p} \bar{a}_{ik} O_{t-k})^2 / \sum_{t=1}^{T-1} \gamma_t(i) \end{aligned}$$

$$= \frac{1}{T-1} \sum_{i=1}^{N} \sum_{t=1}^{T-1} \gamma_t(i)(O_t - \bar{\mu}_i + \sum_{k=1}^{p} \bar{a}_{ik} O_{t-k})^2.$$

3.3.4 AR Proof of Concept Trial. This subsection examines the ability to estimate two autoregressive filters which switch ergodically according to a Markov process. The forward-backward procedure determines the likelihood, $\gamma_t(i)$, that each observation was generated by a particular hidden state (i.e. filter). The new filters are reestimated by the weighted autocorrelation given in Equation 25 with the noise variance given by Equation 26. Figure 8 demonstrates the ability to recover the underlying, hidden state sequence by using applying the maximum operator to the process γ_t.

The test sequence contains 500 samples shown in Figure 7. The ergodic Markov transition matrix has $A(1,1) = A(2,2) = 0.9$. The original model parameters are

$$A_1(z) = (1, .05, .80), \sigma_1 = 3.00, \qquad A_2(z) = (1, .20, -.50), \sigma_2 = 2.00$$

with the final estimates given after eight Baum-Welch iterations.

$$\hat{A}_1(z) = (1, -.02, .78), \hat{\sigma_1} = 3.00, \qquad \hat{A}_2(z) = (1, .21, -.53), \hat{\sigma_2} = 2.10.$$

This is the first known application of uncovering a hidden state sequence for a hidden filter Markov model, as well as the ability to estimate filters with state dependent noise variances. The next section extends autoregressive sample-based hidden filter modeling to a more general, robust, autoregressive moving-average (pole-zero) filter.

3.3.5 Reestimation of MA and ARMA Filters. Other filters, besides the all-pole, autoregressive can also be Markov-modulated. Linear prediction on speech samples has long been an effective representation for voiced speech sounds [27, 63, 78]. However, for many phonemes, especially nasals and other unvoiced fricatives, a moving average (MA) component is more appropriate [37, 90]. Bourlard and Morgan strongly justify the use of autoregressive models, which are suited well for dynamic systems. While very applicable

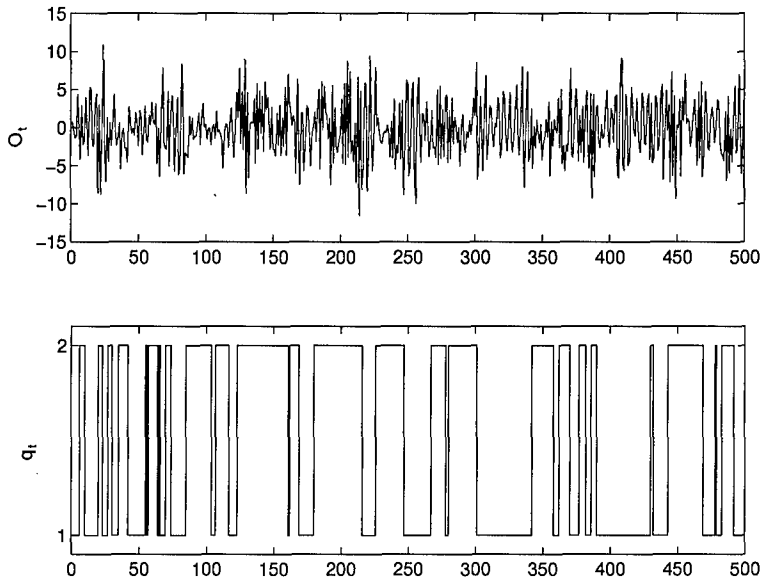


Figure 7. Sample AR(2) Markov-Modulated Source with Actual State Sequence.

to speech, a better model would be autoregressive-moving average (ARMA) [94]. Kay [63] observes,

> Since nearly all data are corrupted by some amount of observation noise, the ARMA model is nearly always the appropriate one.

A linear autoregressive moving average ARMA(p,q) model is defined as

$$O_t = -\sum_{i=1}^{p} a_i O_{t-i} + \sum_{j=1}^{q} b_i e_{t-j}$$

where the p and q represent the order of the Moving Average (MA) and Autoregressive (AR) processes and e_t is often assumed white, Gaussian noise. This model reflects a white noise input to a *pole-zero* filter, with transform

$$H(z) = \frac{B(z)}{A(z)} = \frac{\sigma^2 \cdot (1 + b_1 z^{-1} + b_2 z^{-2} + \ldots b_q z^{-q})}{1 + a_1 z^{-1} + a_2 z^{-2} + \ldots a_p z^{-p}}. \tag{29}$$

The estimation of ARMA(p,q) models involves solving a set of highly nonlinear equations, thus only efficient suboptimal techniques exist. Durbin's approach [63, 123] models the $A(z)$ and $B(z)$ filters separately, first solving for the maximum likelihood estimate of

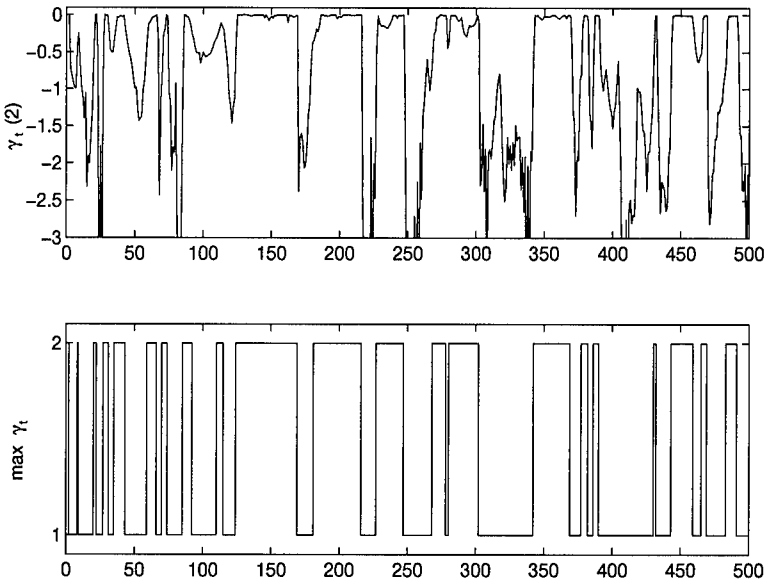


Figure 8. Uncovering the AR Hidden State Sequence. Shown are the actual state log likelihood $\gamma_t(2)$ and the most likely state sequence $\max_i \gamma_t(i)$ for the process described in Figure 7.

the AR(p) process then applying the filter to create an approximate MA approximation. The method is considered an approximate maximum likely estimator (MLE) for the ARMA coefficients.

First the $A(z)$ filter from Equation 29 is estimated using Equations 25 and 26. Then, a new approximate MA process O_t^* is created by filtering the original signal O_t with the maximum likelihood state q_t filters

$$O_t^* = -\sum_{j=1}^{p} a_{j,q_t} O_{t-j}. \tag{30}$$

Durbin approximation for MA filter estimation involves the following assumption, which uses a large AR model or order L to approximate the MA coefficients.

$$B(z) = \sum_{j=0}^{q} b_j z^{-j} = \frac{1}{A_\infty(z)} = \frac{1}{\sum_{j=0}^{\infty} a_j z^{-j}} \approx \frac{1}{\sum_{j=0}^{L} a_j z^{-j}} = \frac{1}{A_L(z)}$$

All previous Markov-modulated AR reestimation (see Section 3.3.2) then applies to this L-th order AR approximation. Approximate MLE estimates of $B(z)$ use the autocorrelation method of model order q where the $A_L(z)$ coefficients $(1, a_{i1}, a_{i2}, \ldots a_{iL})$ are treated as "data" [63].

3.3.6 Proof of Concept Trial. An examination of an ARMA Markov modulated process shows the ability to estimate rational hidden filters. The 1000 samples were generated by two ARMA(2,2) filters with Markov transition probabilities of $A(1,1) = A(2,2) = 0.9995$ (Figure 9). Following an initial uniform segmentation, eight Baum-Welch iterations produced the following state likelihoods (Figure 10). Various large AR approximations ($L = 10, 20$ and 30) to the MA filter were successful. Note for this example, the AR process was not Markov modulated and could be estimated directly. The original rational

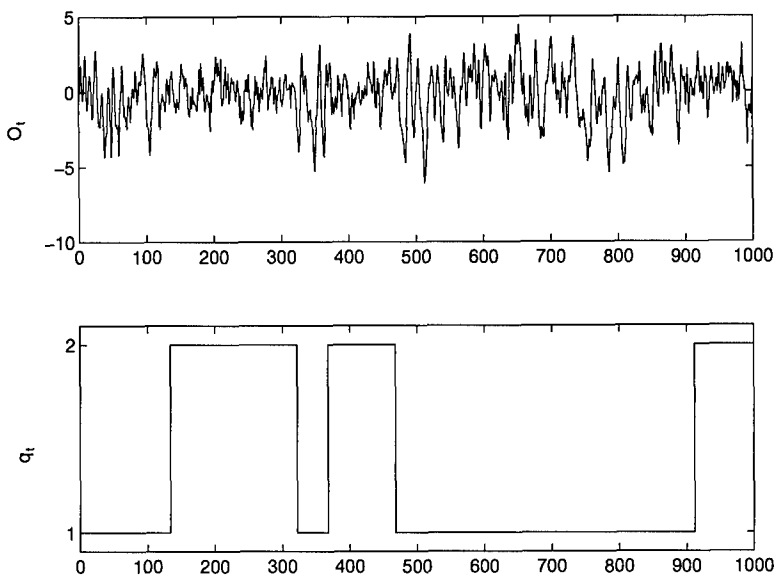


Figure 9. Markov-Modulated ARMA(2,2) Process

filters were

$$H_1(z) = \frac{.5(1.00 + 0.50z^{-1} + 0.30z^{-2})}{1.00 - 1.00z^{-1} + 0.30z^{-2}}$$

$$H_2(z) = \frac{.5(1.00 - 0.40z^{-1} + 0.20z^{-2})}{1.00 - 1.00z^{-1} + 0.30z^{-2}}.$$

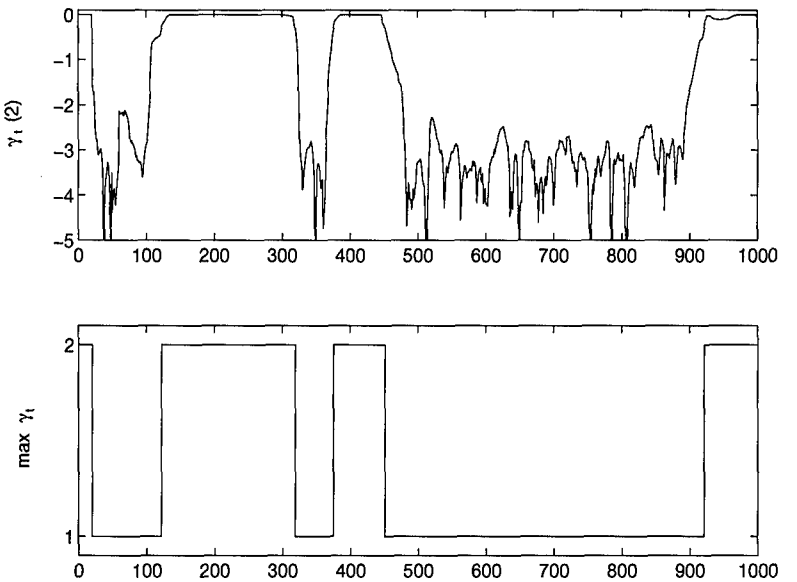


Figure 10. Uncovering the ARMA Hidden State Sequence. Shown are the actual likelihoods $\gamma_t(2)$ and the most likely state sequence $\max_i \gamma_t(i)$ for the process described in Figure 9.

Table 1 shows final ARMA filters and noise variance estimates. $A(z)$ was estimated to be

$$A(z) = 1.00 - 1.06z^{-1} + 0.30z^{-2}.$$

Table 1. Estimated Markov-modulated ARMA Filters.

$B(z) \approx 1/A_L(z)$	$B_2(z)$ filter	$B_1(z)$ filter	σ_2^2	σ_2^2
L=10	1.00, -0.43, 0.13	1.00, 0.50, 0.21	0.6975	0.5623
L=20	1.00, -0.47, 0.15	1.00, 0.50, 0.17	0.6876	0.5492
L=30	1.00, -0.49, 0.13	1.00, 0.55, 0.21	0.6783	0.5298

This example provided the ability to find and estimate pole-zero filters which are generated by a hidden Markov process. It has been demonstrated that for certain speech phonemes, ARMA is the model of choice. However, only approximate MLE methods exist for their solutions and their methods involve filtering the sequence with estimated filters. The next section returns to autoregressive model, but this time on frames of observations.

3.4 Frame Autoregressive Hidden Filter Reestimation

This section examines the reestimation of filters when applied to frames of observations. We believe this technique has much merit, especially when trained and applied in a new architecture to model speaker dependent phonemes. The following derivations are expansions and clarifications from [29, 57, 58, 60, 72, 92, 93, 96]. First, assume a single hidden filter for each state modeling frames of observations. For any autoregressive observation , Juang [57, 60] defines the output density of the frame, $O_t = (x_1, x_2, \ldots, x_K)$ when the observation sequence length K is much greater than the autoregressive order, as

$$p(O_t|\bar{\sigma}_i, \bar{a}_i) = \frac{1}{(2\pi)^{K/2}(\sigma^2)^{K/2}} \exp\left\{-\frac{1}{2}\alpha(x_1, \ldots, x_K; \bar{a}_i)\right\}. \tag{31}$$

The gain-independent density, where $(s_1, \ldots, s_t) = (x_1/\sigma \ldots x_T/\sigma)$ is simply

$$p(s_1, \ldots, s_K|\bar{a}_i) = \frac{1}{(2\pi)^{K/2}} \exp\left\{-\frac{1}{2}\alpha(s_1, \ldots, s_K; \bar{a}_i)\right\}. \tag{32}$$

Juang uses a total prediction error in the form expressed by

$$\alpha(x_1, \ldots, x_K; \bar{a}) = r_a(0)r_x(0) + 2\sum_{i=1}^{p} r_a(i)r_x(i) \tag{33}$$

and the autocorrelations are further defined as

$$\begin{aligned} r_a(i) &= \sum_{j=1}^{p-i} a_j a_{j+i} \\ r_x(i) &= \sum_{j=1}^{K-i} x_j x_{j+i}. \end{aligned}$$

This derivation assumes the driving error was a zero mean white process, normally distributed.

Kay [63] defines a similar density (after the first p samples) as

$$\begin{aligned} p(x_{p+1}, \ldots, x_K|\sigma, \bar{a}_i) &= \frac{1}{(2\pi\sigma_i^2)^{(K-p)/2}} \exp\left\{-\frac{1}{2\sigma_i^2}\sum_{t=p+1}^{K}(x_t + \sum_{k=1}^{p} a_{ik}x_{t-k})^2\right\} \\ &= \frac{1}{(2\pi\sigma_i^2)^{(K-p)/2}} \exp\left\{\alpha^*(x_1, \ldots, x_t; \bar{a})\right\} \end{aligned} \tag{34}$$

The squared prediction residual for Kay's density $\alpha^*(x_1, \ldots, x_t; \bar{a})$ can be shown to be identical to Juang's, under the assumption of $K \gg p$, and the fact that $a_0 = 1$, and $x_t = 0,\ t \leq 0, t > T$

$$\begin{aligned} \alpha^*(x_1, \ldots, x_K; \bar{a}) &= \sum_{t=p+1}^{K} (x_t + \sum_{i=1}^{p} a_i x_{t-i})^2 \\ &\approx r_a(0) r_x(0) + 2 \sum_{i=1}^{p} r_a(i) r_x(i) = \alpha(x_1, \ldots, x_K; \bar{a}). \end{aligned}$$

The original method by Poritz [92] noted that another, simpler expression for the prediction error, realized through a matrix product.

$$\begin{aligned} \hat{\alpha}(x_1, \ldots, x_K; \bar{a}) &= [1\ a_1\ a_2\ \cdots\ a_p] \begin{bmatrix} r_x(0) & r_x(1) & \cdots & r_x(p) \\ r_x(-1) & r_x(0) & \cdots & r_x(p-1) \\ \vdots & & \ddots & \vdots \\ r_x(-p) & r_x(-(p+1)) & \cdots & r_x(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_1 \\ \vdots \\ a_p \end{bmatrix} \\ &= \bar{a}^T R_x \bar{a} \end{aligned} \tag{35}$$

Thus, the three methods contained in Equations 33, 34 and 35 provide a method to evaluate the output density of the current frame with respect to the state filter coefficients. For reestimation, the critical points of the auxiliary function with respect to the filter coefficients and residual energy is examined, now using the frame-based density function given in Equation 31. The results are expressible in terms of the autocorrelation coefficients of the frame. Using Juang's notation of the autocorrelation function of the t-th frame having length K,

$$r_t(j) = \sum_{k=1}^{K-j} O_{t,k} O_{t,k+j}.$$

it can be shown [57, 60] the MLE predictor coefficients can be solved through the normal equations, Equation 23,

$$\sum_{t=1}^{T-1}\gamma_t(i)\begin{bmatrix} r_t(1) \\ r_t(2) \\ \vdots \\ r_t(p) \end{bmatrix} = \sum_{t=1}^{T-1}\gamma_t(i)\begin{bmatrix} r_t(0) & r_t(-1) & \cdots & r_t(-p+1) \\ r_t(1) & r_t(0) & \cdots & r_t(-p+2) \\ \vdots & & \ddots & \vdots \\ r_t(p-1) & r_t(p-2) & \cdots & r_t(0) \end{bmatrix}\begin{bmatrix} a_i(1) \\ a_i(2) \\ \vdots \\ a_i(p) \end{bmatrix}$$

where $r_t(j)$ is the average state autocorrelation function expressed by

$$r_i(j) = \frac{\sum_{t=1}^{T}\gamma_t(i)r_t(j)}{\sum_{t=1}^{T}\gamma_t(i)}. \tag{36}$$

Denote the linear equations as

$$\sum_t \gamma_t(i)\bar{r}_t = \sum_t \gamma_t(i)R_t\bar{a}_i \tag{37}$$

or simply

$$\bar{r}_i = R_i\bar{a}_i.$$

Similarly, the noise variance estimate uses the maximum likelihood $\bar{a}_i$, which solves the equation

$$\sigma_i^2 = \frac{\sum_{t=1}^{T}\gamma_t(i)\bar{a}_i^T R_i\bar{a}_i}{K\sum_{t=1}^{T}\gamma_t(i)}.$$

In summary, this section demonstrated the procedure when hidden state changes occur at frame boundaries and hidden filters represent a linear dynamic system describing the entire frame.

3.4.1 Initialization By Clustering. Since all of the reestimation schemes for HMMs are both iterative and without theoretical convergence to global extrema, the need for good initial models exists. Often, a uniform segmentation process is used to cluster data into the number of HMM states; these cluster centroids are then mapped to probabilistic distributions. Depending on the feature representation, some expectation is used within

this uniform segmentation process. It is demonstrated that for autoregressive features, also known as linear predictive coding (LPC), this sample mean produces non-optimal initialization when using an appropriate distortion optimality criterion.

3.4.1.1 Spectral Distortion Measures. Each frame of data can be represented by autoregressive filter coefficients, the autocorrelation function or by some other spectral feature, such as Mel frequency cepstral coefficients. In order to measure "closeness" amongst frames of data, a suitable distortion must be defined. Distortions in spectral shape or overall spectrum can make use of mathematical metrics. For example, the L_2 metric between two log spectra results in

$$d_2^2(s_1, s_2) = \|S_1(w), S_2(w)\| = \frac{1}{2\pi}\int_{-\pi}^{\pi} |\log S_1(w) - \log S_2(w)|^2 dw$$

Applying this metric to two unity gain LPC spectra[1] results in the Itakura measure [95]. Using the density of a linear prediction coefficient, the definition of the Itakura-Saito distance [27] is a form of the Mahalanobis distance, defined as

$$d_{IS}(\bar{a}_1, \bar{a}_2) = \frac{(\bar{a}_2 - \bar{a}_1)^T R_{\bar{a}_1}(\bar{a}_2 - \bar{a}_1)}{a_1^T R_{a_1}\ a_1}. \tag{38}$$

When clustering cepstral coefficients for initial state model, it turns out that the L_2 norm on the log spectra results in the typical Euclidean norm of the cepstral coefficients. Thus, the sample mean is the optimal cluster center,

$$\begin{aligned} d_2^2(s_1, s_2) &= \frac{1}{2\pi}\int_{-\pi}^{\pi} |\log S_1(w) - \log S_2(w)|^2 dw \\ &= \int_{-\pi}^{\pi} \sum_{n=-\infty}^{\infty} |c_{1,n} - c_{2,n}|^2 \frac{dw}{2\pi} = \sum_{n=-\infty}^{\infty} |c_{1,n} - c_{2,n}|^2 \end{aligned}$$

[1]The unity gain LPC spectrum is denoted as

$$S(w) = \frac{1}{|A(e^{jw})|^2}$$

where a finite approximation using L coefficients is often used.

$$d_2^2(s_1, s_2) = \sum_{n=0}^{L} |c_{1,n} - c_{2,n}|^2$$

This will be true even if perceptual weightings are performed, such as a Mel frequency analysis.

3.4.1.2 Uniform Segmentation and Clustering. In order to create an initial model for the reestimation process, it will be necessary to cluster the features into initial "states." Assume the frames are represented by autoregressive coefficients. Then, define the sample expectation over L autoregressive or LPC vectors as

$$\bar{\mu}_{IS} = E[\bar{a}_1, \bar{a}_2, \ldots, \bar{a}_L] = \min_{\bar{a}} \sum_{i=1}^{N} d_{IS}(\bar{a}, \bar{a}_i)$$

which is simply the LPC representation with minimum Itakura-Saito distortion to all L LPC vectors. Without consideration to feature representation, the arithmetic mean is often used [95, 134], denoted by $\bar{\mu}_A$

$$\bar{\mu}_A = \frac{1}{L} \sum_{i=1}^{L} \bar{a}_i.$$

Solving for the minimum of the $\bar{\mu}_{IS}$ and using Equation 38, one seeks $\bar{a}$ which solves the necessary optimality condition

$$\nabla_{\bar{a}} \sum_{i=1}^{L} d_{IS}(\bar{a}, \bar{a}_i) = \bar{0}^T$$

which occurs, for gain-normalized frames, as the solution of

$$\sum_{i=1}^{L} R_i \bar{a} = \sum_{i=1}^{L} \bar{r}_i \tag{39}$$

where $\bar{r}_i$ denotes the autocorrelation function for frame i and R_i denotes corresponding matrix.

3.4.1.3 Relationship to AR HMMs. During the reestimation for new state filter coefficients, the Baum-Welch procedure applied to the frame AR HMM problem resulted in Equation 36. This described the new state autocorrelation function, interpreted as a weighted sum of individual frame autocorrelations, using the weight, $\gamma_t(i)$. The maximum likely state sequence would find those frames belonging to state i. This would result in the new estimate for the ith state autocorrelation function as

$$r_i(l) = \frac{\sum_{t=1}^{T_i} r_t(l)}{T_i}$$

or the sample mean of the frame autocorrelation functions. Thus, minimizing the Baum auxiliary function for a frame autoregressive hidden Markov model with respect to the state filters, Equation 36 and 37, results in minimization of the Itakura-Saito distortion across those frames.

3.4.2 Proof of Concept Trial. Poritz applied this frame-based hidden filter reestimation using a 5-state ergodic architecture with simple third order filters. The reulsts are shown for a female speaker of the YOHO database in Figure 11, which demonstrates in the inherent language modeling by this method. The five states naturally form five phonetically-similar broad classes [92, 72]. These include silence (S), vowels (V), nasals (N), liquid-glides (L), and consonants (C) as evidenced by the spectral characteristics.

Another contribution of this research is the extension of this technique to model the sample correlations within each phoneme separately, shown in Figure 12. In order to extract temporal information within a phoneme, a 3-state left-to-right model for each phone has been created, each with a more appropriate 12-th order predictor. Not only does this architecture better model the overall spectrum of each phoneme, but the 3-state left-to-right architecture models the transitions within a phoneme. Another useful result of this method is the ability to provide state-of-the-art speech recognition based on these type of sub-word models.

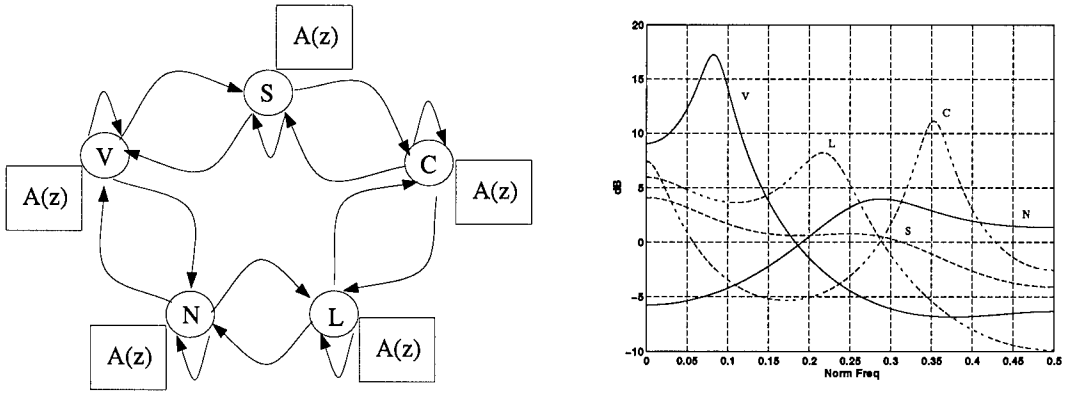


Figure 11. Poritz Method Applied to a YOHO Speaker - Showing Language Broad Class Modeling. Right: The architecture provides an ergodic 5-state hidden filter Markov model. Note: not all transitions shown for clarity. Left: The five filters attempt to model a broad phonetic category. These include silence (S), vowels (V), nasals (N), liquid-glides (L), and consonants (C) as evidenced by the power spectral densities of the resulting filter estimates.

3.5 Vector Hidden Filter Markov Model Reestimation

Thus far, the reestimation of hidden filters operating on samples have been developed. Options have included samples with and without a bias, autoregressive, autoregressive moving average and frame based techniques. The third and final level where hidden filter Markov models may prove extremely useful is the feature space. This level of modeling first attempts to optimize the feature extraction, where relatively small-sized vectors are analyzed at efficient rates. Then, the dynamics within each state, assumed generated by a vector autoregressive process, is estimated. Begin with the definition of a vector autoregressive hidden Markov model. For each state i,

$$\bar{O}_t = -\sum_{j=1}^{p} A_{ij}\bar{O}_{t-j} + \bar{W}_t$$

where the last expression $\bar{W}_t$ can be a non-zero mean multivariate white Gaussian noise source and the predictor matrices given for state i are denoted by A_{ij}. This equation is

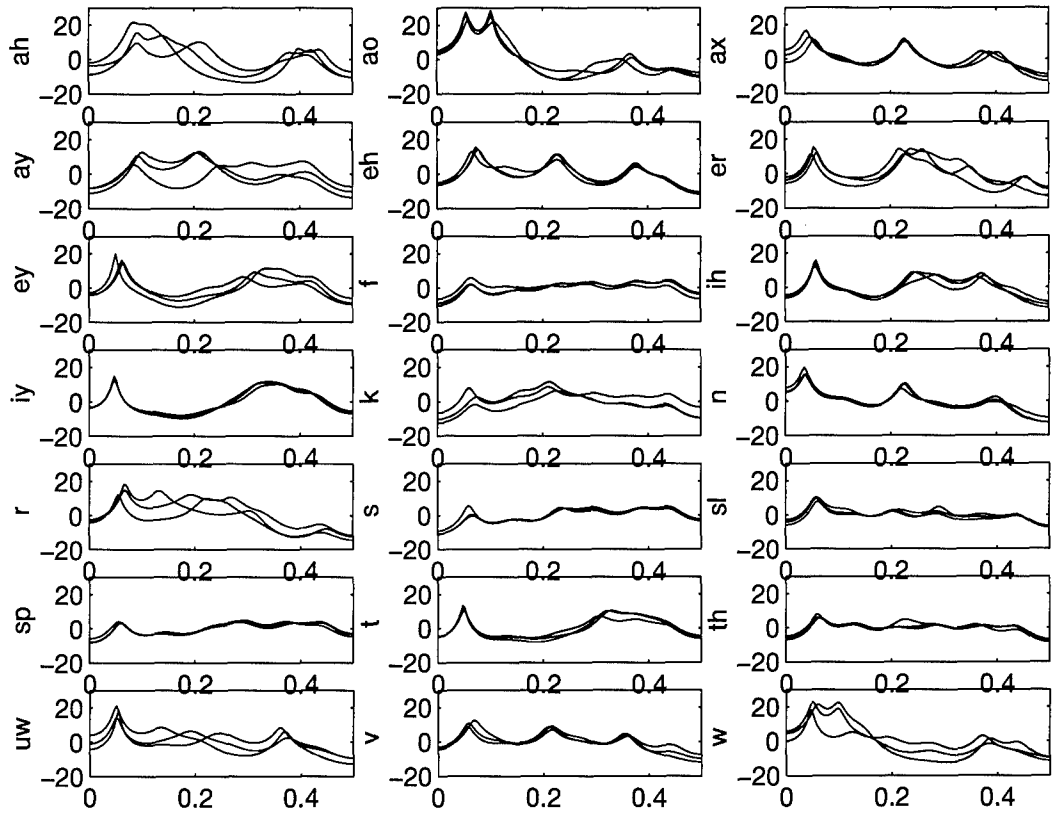


Figure 12. Extended Poritz Method for Temporal Phoneme Modeling of YOHO Speaker. Each speaker is represented by 21 3-state left-to-right monophone models. Shown for one speaker, the power spectrum of the resulting filter estimates for all models and all states.

expressible with a zero mean Gaussian input, $\bar{E}_t$, as

$$\bar{O}_t = \bar{\mu}_i - \sum_{j=1}^{p} A_{ij} \bar{O}_{t-j} + \bar{E}_t \tag{40}$$

where we seek to estimate the A_{ij} matrices and the state mean, $\bar{\mu}_i$. First, the relation to the standard multivariate Linear Prediction is established.

3.5.1 Multivariate LPC Appoach. For zero mean multivariate noise, Kay [63] analyzes the multidimensional spectral estimation of vector Linear Predictive Coding (LPC) processes. The solution is a matrix equivalent of the Yule-Walker equations, using the

biased autocorrelation function estimator.

$$-\begin{bmatrix} R_x(1) \\ R_X(2) \\ \vdots \\ R_x(p) \end{bmatrix} = \begin{bmatrix} R_x(0) & R_x(-1) & \cdots & R_x(-(p-1)) \\ R_x(1) & R_x(0) & \cdots & R_x(-(p-2)) \\ \vdots & & \ddots & \vdots \\ R_x(p-1) & R_x(p-2) & \cdots & R_x(0) \end{bmatrix} \begin{bmatrix} A_{i1}^T \\ A_{i2}^T \\ \vdots \\ A_{ip}^T \end{bmatrix} \tag{41}$$

where each $R_x(j)$ is a (d x d) matrix corresponding to lag j of the vector process.

For the non-zero mean case, there exists the less-known relation of the covariance function satisfying the Yule-Walker equations [20, 21]. Let the estimated matrix covariance function be substituted for the autocorrelation function in Equation 41.

$$C_x(j) = R_x(j) - \bar{\mu}\bar{\mu}^T \tag{42}$$

Equation 42 will be shown identical to the technique of maximizing the Baum auxiliary function $Q(\lambda, \bar{\lambda})$ with respect to the vector and matrix quantities for each state.

3.5.2 Special Cases. Four cases can be developed based on vector autoregressive modeling:

- Diagonal A_i, Diagonal Σ: Each current observation dimension d separately is regressed on past observations, but same dimensions. The current observation has independent dimensions (uncorrelated) as expressed by its covariance.

$$\bar{O}_t^d = \mu^d - \sum_i A_i(d,d)\bar{O}_{t-i}^d + E_t^d, \qquad \Sigma_{jk} = 0, \qquad A_i(j,k) = 0, j \neq k$$

- Full A_i, Diagonal Σ: Each dimension, in turn, is regressed on past observations, all dimensions. The current observation still has independent dimensions (uncorrelated).

$$\bar{O}_t = \mu - \sum_i A_i\bar{O}_{t-i} + E_t, \qquad \Sigma_{jk} = 0, j \neq k,$$

- Diagonal A_i, Full Σ: Each current observation dimension d separately is regressed on past observations, same dimensions. The current observation has full covariance

and dimensions may be correlated.

$$\bar{O}_t^d = \mu^d - \sum_i A_i(d,d)\bar{O}_{t-i}^d + E_t, \qquad A_i(j,k) = 0, j \neq k$$

- Full A_i, Full Σ: Each dimension, in turn, is regressed on past observations, all dimensions. The current observation has full covariance - dimensions may be correlated.

$$\bar{O}_t = \mu - \sum_i A_i \bar{O}_{t-i} + E_t,$$

3.5.3 Full predictor, Full Covariance Reestimation. Without any a prior information concerning the vector process, it would be safe to apply the full predictor with full covariance equations to the problem. The solution of the new estimates, begins by taking the partial derivative of the Baum auxiliary function with respect to each states' predictor and covariance matrix. Using simplify notation [66], the logarithm of the output density of the multivariate model in Equation 40 is given by,

$$\log b_i(\bar{O}_t) = C - \frac{1}{2}|\Sigma_i| - \frac{1}{2}(\bar{O}_t - \bar{\mu}_i + \sum_{j=1}^{p} A_{ij}\bar{O}_{t-j})^T \Sigma_i^{-1}(\bar{O}_t - \bar{\mu}_i + \sum_{j=1}^{p} A_{ij}\bar{O}_{t-j}). \quad (43)$$

Make the following matrix substitutions.

$$B_i = \begin{bmatrix} A_{i1} & A_{i2} & \dots & A_{ip} \end{bmatrix}, \qquad \bar{Y}_t = \bar{O}_t, \qquad \bar{X}_t = (\bar{O}_{t-1}\bar{O}_{t-2}\dots\bar{O}_{t-p})^T$$

This allows certain summations to appear as matrix multiplications. The matrix equations which satisfy the critical point of the Baum auxiliary function, Equation 18, using the density function of Equation 43, become

$$\begin{aligned} \partial Q_b/\partial B_i &= \sum_{j=1}^{N}\sum_{t=1}^{T-1} \frac{1}{2}\gamma_t(i,j)(\bar{Y}_t - \bar{\mu}_i + B_i\bar{X}_t)\bar{X}_t^T = 0 \\ &= \sum_{t=1}^{T-1} \gamma_t(i)(\bar{Y}_t\bar{X}_t^T - \bar{\mu}_i\bar{X}_t^T + B_i\bar{X}_t\bar{X}_t^T) = 0. \end{aligned} \quad (44)$$

Applying the gradient for the state mean provides

$$\begin{aligned} \partial Q_b/\partial \bar{\mu}_i &= \sum_{j=1}^{N}\sum_{t=1}^{T-1}\frac{1}{2}\gamma_t(i,j)[\bar{Y}_t - \bar{\mu}_i + B_i\bar{X}_t] = 0 \\ &= \sum_{t=1}^{T-1}\gamma_t(i)[\bar{Y}_t - \bar{\mu}_i + B_i\bar{X}_t] = 0 \end{aligned} \tag{45}$$

Making the following matrix substitutions,

$$\begin{aligned} S_y &= \textstyle\sum_t \gamma_t(i)\bar{Y}_t & S_x &= \textstyle\sum_t \gamma_t(i)\bar{X}_t \\ S_{yx} &= \textstyle\sum_t \gamma_t(i)\bar{Y}_t\bar{X}_t^T & S_{yy} &= \textstyle\sum_t \gamma_t(i)\bar{Y}_t\bar{Y}_t^T \\ S_{xx} &= \textstyle\sum_t \gamma_t(i)\bar{X}_t\bar{X}_t^T & N &= \textstyle\sum_t \gamma_t(i) \end{aligned}$$

then dropping state notation, Equations 44 and 45 simply become the following.

$$S_{yx} = \bar{\mu}S_x - BS_{xx}, \qquad S_y = N\bar{\mu} - BS_x \tag{46}$$

Lastly, the covariance of the noise source Σ is estimated by

$$\Sigma_i = \sum_{t=1}^{T-1}\gamma_t(i)(Y_tY_t^T + BY_tX_t^T + BX_tY_t^T + BX_tX_t^TB^T - N\bar{\mu}_i\bar{\mu}_i^T)/\sum_{t=1}^{T-1}\gamma_t(i)$$

and dropping state notation

$$\Sigma = \frac{1}{N}\left[S_{yy} + BS_{yx}^T + S_{yx}B^T + BS_{xx}B^T - N\bar{\mu}\bar{\mu}^T\right]. \tag{47}$$

The solution of the Equations 46, the joint vector-matrix simultaneous equations is

$$\begin{aligned} S_yS_x^T - S_{yx} &= B(NS_{xx} - S_xS_x^T) \\ B &= (S_yS_x^T - S_{yx})(NS_{xx} - S_xS_x^T)^{-1} \end{aligned} \tag{48}$$

and

$$\bar{\mu} = \frac{1}{N}(S_y + BS_x). \tag{49}$$

3.5.4 Diagonal Predictor, Diagonal Covariance Reestimation. The choice of speech spectral features, Mel frequency cepstral, support a diagonal predictor structure.

This decision is based on the Discrete Cosine Transform uncorrelating the elements within each vector. Thus, one seeks the best diagonal A_{ij} matrices, such that B_i takes the form

$$B_i = \begin{bmatrix} A_{i1} & A_{i2} & \dots & A_{ip} \end{bmatrix}$$

and each submatrix A_{ij} now resembles,

$$A_{ij} = \begin{bmatrix} a_{i11} & 0 & 0 & 0 \\ 0 & a_{i12} & 0 & 0 \\ \vdots & \dots & \ddots & \vdots \\ 0 & 0 & 0 & a_{i1d} \end{bmatrix}.$$

We observe that each dimension of Equation 48 can be solved separately using least squares. This occurs since there is still $d * p$ linear equations (for each dimension) but only p unknowns. Solving the filter coefficients reduces to the familiar Yule-Walker equations, substituted with the covariance quantities instead of the autocorrelation ones.

3.5.5 Numerical Stability. Noting in Equation 49, any imprecision in the current filter affects both the new covariance and mean estimates. For this reason, a similar model which uses the same mean estimate vector of a standard HMM, namely the *a posteriori* mean or probabilistically weighted mean, is proposed. Using the vector autoregressive model

$$\bar{O}_t = \bar{\mu}_i^A - \sum_{j=1}^{p} A_{ij}\bar{O}_{t-j} + \bar{E}_t$$

where the original $\bar{\mu}_i^A$ has been identified as dependent on the filter. Let the new observations ($\bar{O}_t^*$) be reduced by the current state mean estimate, $\bar{O}_t^* = \bar{O}_t - \bar{\mu}_i$. Note that $\bar{\mu}_i^A$ is not the *a posteriori* mean of a standard HMM, which shall be denoted by $\bar{\mu}$.

$$\bar{\mu}_i^A = \bar{\mu}_i + \sum_{j=1}^{p} A_{ij}\bar{\mu}_i$$

The vector process O_t^* is zero mean within each state or mixture and the model becomes,

$$\bar{O}_t^* = -\sum_{j=1}^{p} A_{ij}\bar{O}_{t-j}^* + \bar{E}_t$$

with the estimation of A_{ij} matrices and the *a posteriori* state mean, $\bar{\mu}_i$ proceeding accordingly to the standard hidden Markov model mean update. Naturally, in both cases, when $p = 0$ or $A_{ij} = 0$, the reestimation reduces to the standard Gaussian HMM model. Also note that in the single state $N = 1$ case, the reestimation is the direct multivariate LPC model [63] from Section 3.5.1.

3.5.6 Proof of Concept Trial. To demonstrate the ability of this model to extract low-pass, high-pass and bandpass filters across different dimension from a vector Markov state source, ten sequences of 200 observations (2 dimensional) where created using the following multivariate filters. The left-to-right transition matrix has $a_{11} = 0.99$.

$$B_1 = [I|A_{11}|A_{12}] = \left[\begin{array}{cc|cc|cc} 1.00 & 0.00 & -1.20 & 0.00 & 0.429 & 0.00 \\ 0.00 & 1.00 & 0.00 & 0.24 & 0.00 & 0.795 \end{array}\right]$$

$$B_2 = [I|A_{21}|A_{22}] = \left[\begin{array}{cc|cc|cc} 1.00 & 0.00 & 1.124 & 0.00 & 0.39 & 0.00 \\ 0.00 & 1.00 & 0.00 & -1.237 & 0.00 & 0.775 \end{array}\right]$$

$$\mu_1 = \begin{bmatrix} 2.0 \\ 4.0 \end{bmatrix}, \Sigma_1 = \begin{bmatrix} 0.05 & 0.00 \\ 0.00 & 0.02 \end{bmatrix} \quad \mu_2 = \begin{bmatrix} 1.0 \\ -3.0 \end{bmatrix}, \Sigma_2 = \begin{bmatrix} 0.10 & 0.00 \\ 0.00 & 0.05 \end{bmatrix}$$

For the diagonal model, the estimated output density parameters were as follows:

$$\tilde{B}_1 = [I|A_{11}|A_{12}] = \left[\begin{array}{cc|cc|cc} 1.00 & 0.00 & -0.878 & 0.00 & 0.084 & 0.00 \\ 0.00 & 1.00 & 0.00 & 0.176 & 0.00 & 0.819 \end{array}\right]$$

$$\tilde{B}_2 = [I|A_{21}|A_{22}] = \left[\begin{array}{cc|cc|cc} 1.00 & 0.00 & 1.136 & 0.00 & 0.398 & 0.00 \\ 0.00 & 1.00 & 0.00 & -1.224 & 0.00 & 0.770 \end{array}\right]$$

$$\tilde{\mu}_1 = \begin{bmatrix} 1.80 \\ 3.93 \end{bmatrix}, \tilde{\Sigma}_1 = \begin{bmatrix} 0.131 & 0.00 \\ 0.00 & 0.059 \end{bmatrix} \quad \tilde{\mu}_2 = \begin{bmatrix} 0.99 \\ -3.04 \end{bmatrix}, \tilde{\Sigma}_2 = \begin{bmatrix} 0.10 & 0.00 \\ 0.00 & 0.05 \end{bmatrix}$$

The power spectrum of the generator for each state and each dimension is shown in Figure 13. When using a known feature, such as cepstral coefficients, the predictor should be

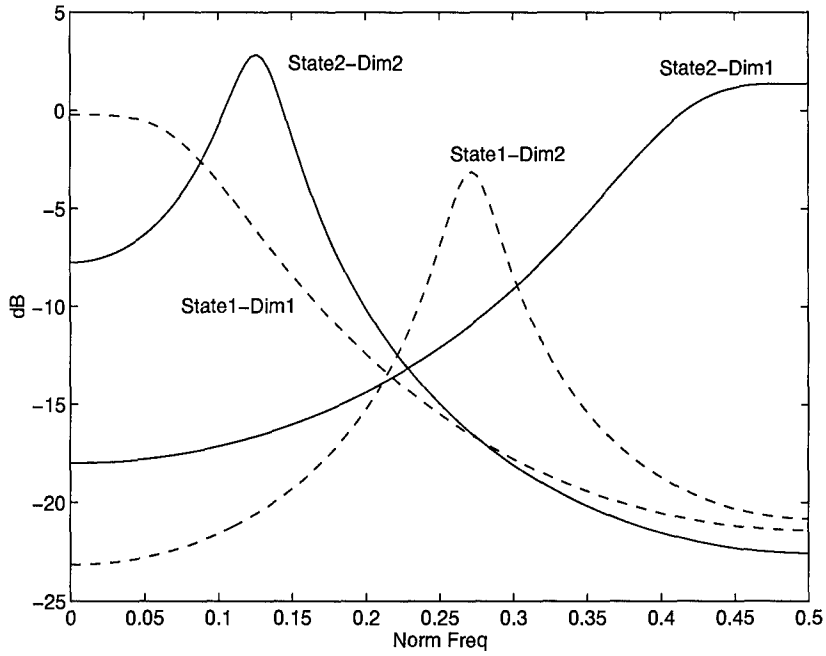


Figure 13. Markov-Modulated Vector AR(2) process.

diagonal. For this test signal, both full and diagonal predictor types were applied and the spectrums of the estimated filters shown in Figure 14.

3.6 Conclusion

The methods developed in this chapter now allow for modeling Markov-modulated linear dynamic system, at the sample level, the frame level and the processed features level. Key examples have shown their ability to find ergodic AR filters, ergodic ARMA models, phoneme-based frame level left-to-right AR filters and vector autoregressive models.

In summary, the Baum-Welch reestimation procedure follows a prescribed sequence:

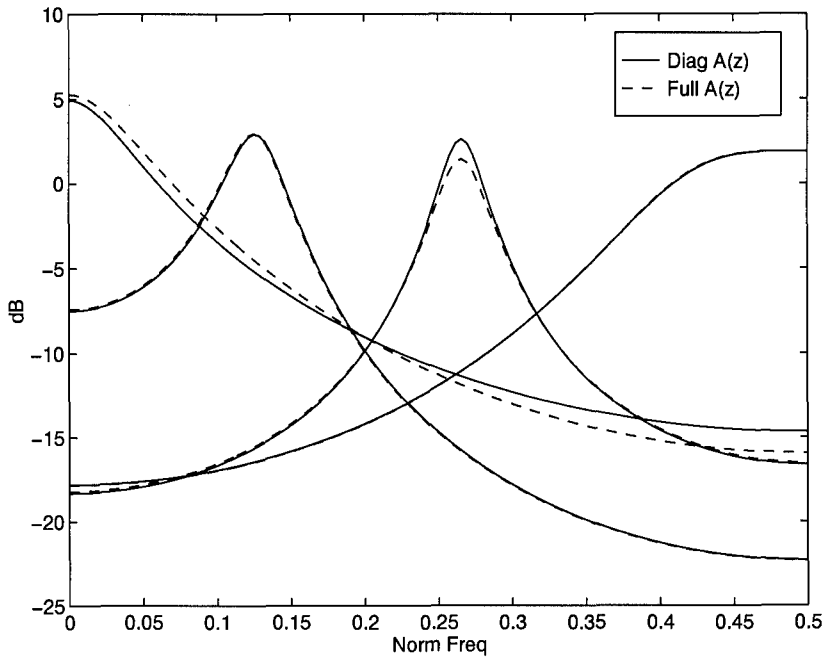


Figure 14. Estimated Markov-modulated vector AR(2) spectrum using both a diagonal and full predictors.

- Initial model λ_o, often solved with some clustering or segmental k-means procedure [Sec 3.4.1]
- Forward-Backward algorithm solves for $\gamma_t(i)$ [Sec 3.2.1 and 3.2.2]
- Solve ML estimates of Π, $\mathbf{A}$ using standard hidden Markov model procedures [96]
- Solve simultaneous equations for the ML $\mathbf{B}$ output density parameters including:
  - Sample AR: $\bar{a}_i, \sigma_i^2$ and/or μ_i [Sec 3.3.2, 3.3.3]
  - Sample ARMA: $\bar{a}_i, \bar{b}_i, \sigma_i^2$ [Sec 3.3.5]
  - Frame: $\bar{a}_i, \sigma_i^2$ [Sec 3.4]
  - Vector: $B_i = [A_{i1}, \ldots, A_{ip}], \Sigma_i, \bar{\mu}_i$ [Sec 3.5]
- Repeat until convergence.

Naturally, each can be extended to multiple mixtures, with state mixture weighting similar to the standard hidden Markov model approach.

Before exploring their effectiveness experimentally, several theoretic properties concerning the a priori classification and convergence of the Baum-Welch learning must be resolved. These are proven in Chapter IV. Then in Chapter V, particular versions of these models will be applied to the challenging problem of large population, speaker identification and recognition.

IV. Hidden Filter Analysis

4.1 Introduction

Several key properties of hidden filter Markov models will be demonstrated analytically and experimentally in this chapter. The first justifies their use over other methods for pattern classifications problems. Based on a theorem by Fielding [42], it will be shown an assumed hidden filter Markov source reduces the joint entropy over an assumed Gaussian mixture Markov model. Secondly, it will be shown that construction of an equivalent single mixture structure can be generated for any finite mixture. If all output densities have the property of negative log concavity for this equivalence model, then each step of the Baum-Welch algorithm will find the global maximum for that iteration, as well as the overall convergence will be monotonic. Lastly, the hidden filter output densities will also demonstrate the property of negative log concavity.

4.2 Entropy Analysis of Markov Sources

Fielding [42] recently provided a relation between information theory and pattern classification. Entropy, the average measure of information over a set of observations, provides a useful tool for comparing classifiers. A classification system which reduces uncertainty in a set of observations by using useful assumptions of the source model, will, reduce the probability of error [72]. It is therefore desirable to find models which reduce joint entropy, $H(O_1, O_2, \ldots, O_n)$ defined over a set of observations as

$$H(O_1, O_2, \ldots, O_n) = -\sum_{N} p(O_1, O_2, \ldots, O_n) \log p(O_1, O_2, \ldots, O_n).$$

The summation over N accounts for all possible orderings of the sequence of observations. Two key facts concern the joint entropy properties of sequences. The first, attributed to Blahut [12], is

$$H(O_1, O_2, \ldots, O_n) \leq \sum_{i=1}^{n} H(O_i)$$

with equality holding if the random variables are independent. Thus, the entropy of a sequence will also be less than or equal to the entropy of an individual observation. The

second fact provides insight to Markov processes. Let the observation sequence be a p-th order Markov process. Then,

$$H_p(O_1, O_2, \ldots, O_n) \leq \sum_{i=1}^{n} H(O_i)$$

where $H_p(O_1, O_2, \ldots, O_n)$ denotes the entropy of a p-th order Markov process and equality holds for independence. Fielding's final results demonstrate [42]

$$H_p(O_1, O_2, \ldots, O_n) \leq H_1(O_1, O_2, \ldots, O_n) \leq H(O_1, O_2, \ldots, O_n) \leq \sum_{i=1}^{n} H(O_i) \qquad (50)$$

where $H_1(O_1, O_2, \ldots, O_n)$ as the entropy of a first order Markov process. An increasing Markov dependency in the sequence results in a decreasing joint entropy. A pattern recognizer which models this dependency should have better classification. While Fielding chose the hidden Markov model as the source model, this dissertation examines hidden filter Markov models. It will be shown that the observations produced by a hidden Markov models are not a Markov process. However, if the assumed source is a p-th order hidden filter Markov model, then the observations will be a p-th order Markov process and Fielding's theorem applies directly.

Lemma IV.1 *A hidden Markov model λ generates observations which are not Markov, but independent. Hence,*

$$p(O_t|O_{t-1}, \ldots, O_1) = p(O_t).$$

Proof: Using the hidden Markov model standard assumptions (Section 2.2.1), the conditional likelihood is shown to be unconditioned on any past observations:

$$\begin{aligned} p(O_t|O_{t-1}, \ldots, O_1) &= \sum_{q_t} p(O_t|q_t, O_{t-1}, \ldots, O_1) p(q_t) \\ &= \sum_{q_t} p(O_t|q_t) p(q_t) \\ &= p(O_t) \qquad \square \end{aligned}$$

Lemma IV.2 *A p-th order hidden filter Markov model λ_p generates observations which are a p-th order Markov process, having the property,*

$$p(O_t|O_{t-1},\ldots,O_1) = p(O_t|O_{t-1},\ldots,O_{t-p})$$

Proof: Disprove independence by using the source output density, (Equations 24 or 27). For state q_t,

$$p(O_t|O_{t-1},\ldots,O_1,q_t(i)) = \mathcal{N}(O_t + \sum_{j=1}^{p} a_{ij}O_{t-j}, \sigma_j^2)$$

which clearly demonstrates past observation dependence. Therefore,

$$\begin{aligned} p(O_t|O_{t-1},\ldots,O_1) &= \sum_{q_t} p(O_t|O_{t-1},\ldots,O_1,q_t)p(q_t) \\ &= \sum_{q_t} p(O_t|O_{t-1},\ldots,O_{t-p},q_t)p(q_t) \\ &= p(O_t|O_{t-1},\ldots,O_{t-p}) \qquad \square \end{aligned}$$

These two Lemmas provide insight to the following theorem.

Theorem IV.1 *Let λ_p denote a p-th order Markov model. Let λ denote a standard Gaussian mixture Markov model. Then, given an observation sequence $(O_1 \ldots O_T)$, the joint entropy of this observation assuming a hidden filter source will have less entropy than a hidden Markov model source. That is,*

$$H_{\lambda_p}(O_1 \ldots O_T) \leq H_{\lambda}(O_1 \ldots O_T)$$

Proof: The hidden filter model λ_p generates a p-th order Markov process by Lemma IV.1 with joint entropy $H_p(O_1 \ldots O_T)$. The standard hidden Markov model λ produces observations with joint entropy $H(O_1 \ldots O_T) = \sum_{t=1}^{T} H(O_t)$ by Lemma IV.2. Direct application of Fielding's theorem given by Equation 50 completes this proof. $\square$

This theorem provides justification for hidden filter Markov models in pattern recognition problems. Similar to arguments made by Le Chevalier [68] and Libby [74], if a classifier uses an algorithm to account for this Markov dependency within a sequence, recognition will increase. By using maximum likelihood parameter estimates, we inherently assume the working model is the same as the source model which generated the observations. For example, when using hidden Markov modeling, it is assumed the source is a hidden Markov model. For observation sequences which appear correlated over particular changing blocks of the sequence, it should be assumed the source is some hidden filter Markov model. Now that the model is justified, the next sections analyze some important properties of the learning algorithm.

4.3 Monotonic Reestimation

One property of the Expectation Maximization (EM) algorithm guarantees the likelihood of the observations given the model is increased whenever the auxiliary function is increased. For hidden Markov models with unimodal log concave output densities, Baum and Petre [10] demonstrated that each EM iteration steps to the global maximum of the auxiliary function. This is shown true for a single Gaussian output density and was extended to the more general elliptically symmetrical density function by Liporace [76]. Extending an architectural concept introduced by Rabiner [96], it is demonstrated that HMMs with mixture components can be recast into an equivalent model with only unimodal state densities and a particular transformed probability transition matrix. Thus, Gaussian *mixture* models are now guaranteed to step to the global maximum of the auxiliary function each iteration of the Baum-Welch algorithm. Lastly, an examination of conditional densities, such as hidden filter models with and without mean, demonstrates they also maintain log concavity and results in optimal global maximum steps.

4.3.1 Single Mixture Gaussian HMM. First, the properties of the Baum-Welch (or Expectation Maximization) algorithm, specifically when the output densities are negative log concave in the parameters, will be reviewed. Recall, the scaled auxiliary function $Q(\lambda, \bar{\lambda})$ can be maximized for each of the main parameter sets $\bar{\lambda} = (\bar{\Pi}, \bar{A}, \bar{B})$ separately

for each state. Specifically for the new output densities,

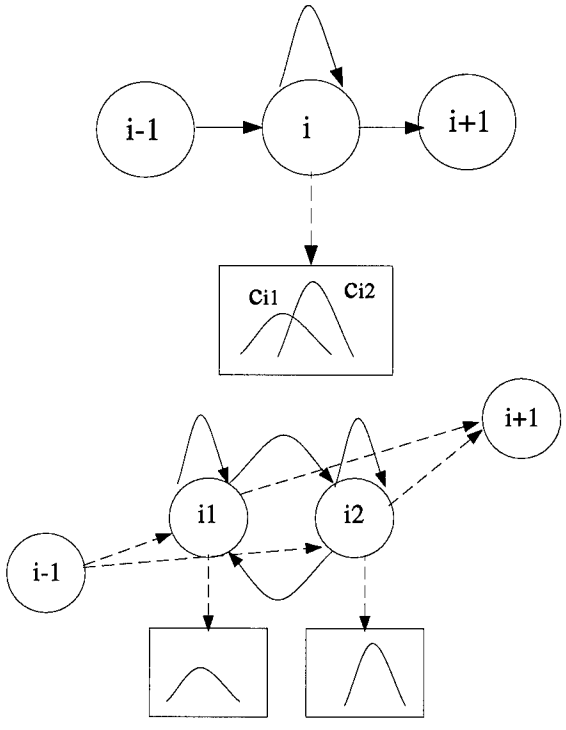
$$\begin{aligned} Q_b(\lambda, \bar{B}_i) &= \sum_{t=1}^{T} p(q_t = i | O_1 \ldots O_T, \lambda) \log b_i(O_t) \\ &= \sum_{t=1}^{T} \gamma_t(i) \log b_i(O_t). \end{aligned} \tag{51}$$

Baum examines the conditions on $b_i(O_t)$ to insure a critical point is also a global maximum over all new models, $\bar{\lambda}$. A proof using transformed observations and a log concave $(b_i(O_t)'' < 0)$ property was used by Baum and colleagues [10] showing Q has a negative second derivative at a critical point. Liporace [76] then provides a more general proof for elliptically symmetrical densities. So for any HMM with single Gaussian density functions, each step of the EM algorithm is guaranteed to increase the likelihood function monotonically by stepping to the maximum of the Q function.

4.3.2 Multiple Mixture Gaussian HMM. In practice, multiple mixtures are used to model more complex distributions of data within each state. However, since $\log b_i(O_t)$ no longer satisfies negative log concavity, Baum's Theorem [10] no longer holds. His proof used a centered process to attain a unit normal with zero mean. Since a mixture density does not satisfy this structure, his theorem no longer applies to multiple mixtures.

4.3.2.1 Rabiner Model. Rabiner presents a similarity between Gaussian mixtures and models with extra states [95]. However, his analysis required special non-emitting entry and exit states for each mixture. Also, this theoretical architecture is not easily verified with existing implementations due to these non-emitting states. Though these special states could be analyzed as being a trivial zero "probabilistic function" of a Markov state sequence, they would not lend themselves to theoretical convergence proofs. Another similarity transformation needs to be defined.

4.3.2.2 Equivalence Model. These non-emitting states can be transformed in a special structure entirely defined within the Markov state transition matrix. Let a constructive example show this fact (Figure 15).



$$A_\lambda = \begin{bmatrix} - & a_{i-1,i} & - \\ - & a_{i,i} & a_{i,i+1} \\ - & - & - \end{bmatrix}$$

$$A_{\lambda_E} = \begin{bmatrix} - & a_{i-1,k} \cdot c_{ik} & a_{i-1,k} \cdot c_{ik} & - \\ - & a_{i,i} \cdot c_{ik} & a_{i-1,k} \cdot c_{ik} & a_{i,i+1} \\ - & a_{i,i} \cdot c_{ik} & a_{i-1,k} \cdot c_{ik} & a_{i,i+1} \\ - & - & - & - \end{bmatrix}$$

Figure 15. Functional equivalence of HMM λ and Equivalence model λ_E. Top model (multiple mixture) can be recast as bottom (single mixture) with a particular transition structure. (-) denotes uninvolved transitions.

Each state can be expanded into substates with the transitions being products of the original transitions and the mixture weights. The following 2 matrices show an original 1 state - 2 mixture HMM generator λ and the theoretical equivalent model λ_E [1].

$$A_\lambda = \begin{bmatrix} - & 1.00 & - \\ - & 0.96 & 0.04 \end{bmatrix} \qquad A_{\lambda_E} = \begin{bmatrix} - & 0.30 & 0.70 & - \\ - & 0.29 & 0.67 & 0.04 \\ - & 0.29 & 0.67 & 0.04 \end{bmatrix}$$

[1] When not directly applicable to the state transformation, unaffected values have been shown as "-".

The original model generated 100 variable length sequences. Both models were trained using the Baum-Welch algorithm of Chapter III, initialize to equivalent random parameters. Table 2 provides the original output density parameters and those learned for both architectures.

Table 2. Actual and Learned (Baum-Welch) Output Densities.

Parameter	Actual λ	Learned $\tilde{\lambda}$	Learned $\tilde{\lambda}_E$
Mean	3.53, -1.98	3.60, -2.00	3.60, -2.00
Variance	0.74, 0.22	0.68, 0.24	0.68, 0.24
Mixtures	0.30, 0.70	0.29, 0.71	- , -

The final estimates of the transitions matrices are as follows, denoted by $A_{\tilde{\lambda}}$ and $A_{\tilde{\lambda_E}}$. Note the similarity to the the original and the theoretical equivalent. Figure 16 shows the monotonically increasing log-likelihoods for each iteration of the Baum-Welch algorithm. Though the two models have different architectures, both converge to the equivalent overall model having -29.45 log-likelihood.

$$A_{\tilde{\lambda}} = \begin{bmatrix} - & 1.00 & - \\ - & 0.94 & 0.06 \\ - & 0.000 & 0.00 \end{bmatrix} \qquad A_{\tilde{\lambda_E}} = \begin{bmatrix} - & 0.32 & 0.68 & - \\ - & 0.28 & 0.67 & 0.05 \\ - & 0.27 & 0.67 & 0.06 \end{bmatrix}$$

Having constructed the model and shown equivalence by example, the formal definition for an Equivalence Model is as follows.

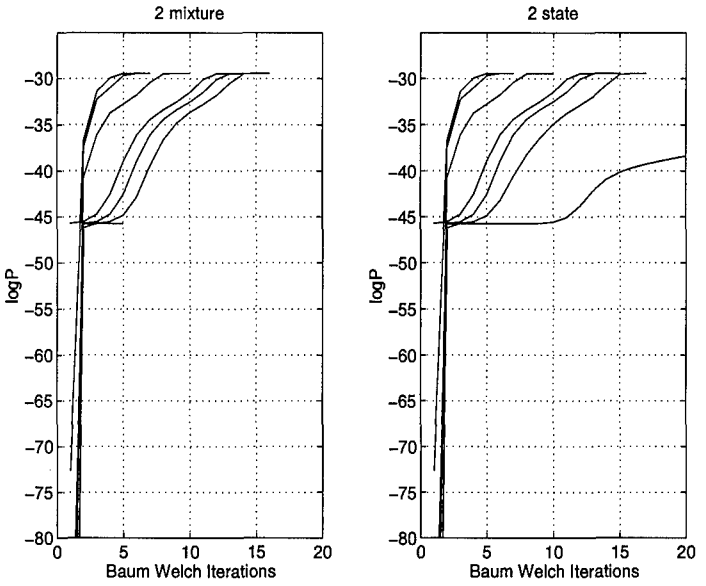


Figure 16. Learning the maximum likelihood models from 10 random starts based on the architectures of the original HMM λ (2 mixture) and theoretical equivalence model λ_E (2 state). Almost all models converged to the same equivalent log-likelihood value of -29.45.

Theorem IV.2 *(Equivalence Model) Given a hidden Markov model λ such that $b_i(O)$ is a state density function consisting of a finite convex combination of negative log concave densities*

$$b_i(O) = \sum_{k=1}^{M} c_{ik} b_{ik}(O), \qquad \textit{such that} \sum_{k=1}^{M} c_{ik} = 1, \quad c_{ik} \geq 0$$

an Equivalence Model λ_E *exists which is functionally equivalent to λ, such that each original state i is expanded into M substates, with the following properties:*

- *Each substate of λ_E is described by one of $b_{ik}(O)$;*
- *The state transition matrix entries of λ_E, for the original state i, new substate k, are given by,*

$$A_{\lambda_E} \quad = \quad (a_{i,k}) = \begin{cases} a_{i,k} = a_{i-1,i} \cdot c_{ik} \\ a_{k,k} = a_{i,i} \cdot c_{ik} \\ a_{k,i+1} = a_{i,i+1} \end{cases}$$

Proof: Proof by construction.

Furthermore, if $f_{i,m}(u)$ satisfies the properties of Liporace Theorem 2 [76] or Baum [10] then $Q(\lambda_E, \bar{\lambda_E})$ has a unique global maximum as a function of $\bar{\lambda_E}$, for fixed λ_E. The results of this Theorem IV.2 insure that each step of the Baum-Welch algorithm for *mixture* densities will increase the likelihood of the model - demonstrated in Figure 16.

4.4 Monotonic Reestimation of Hidden Filters

While the previous analysis was presented for standard hidden Markov models, similar results will be extremely beneficial for hidden filter model. It was discussed that a desirable property of the output density function was either 1) negative log concavity or 2) elliptically symmetric. This section demonstrates that hidden filters also demonstrate this property. The approach of Baum and later Liporace examined the negative definite property of the second derivative of the auxiliary function with respect to the space of new models $\bar{\lambda}$. The following proof takes a similar approach.

Again, the scaled auxiliary function $Q(\lambda, \bar{\lambda})$ can be maximized for each of the main parameter sets $\bar{\lambda} = (\bar{\Pi}, \bar{\mathbf{A}}, \bar{\mathbf{B}})$ separately for each state, Equation 52. It will be shown that the auxiliary function is negative definite for the space of reestimated filter models $\bar{\Lambda}$. This is most easily demonstrated using a similar approach to Liporace [76]. This method first chooses two arbitrary models, λ_1 and λ_2 and defines a new model $\bar{\lambda}$ which is a linear (convex) combination of these two. It can then be shown that any linear convex combination of these models is negative definite. Since λ_1 and λ_2 are arbitrary, it suffices that the entire space of new models is negative definite. Intuitively, the space must have only one global maximum at the single critical point of the auxiliary function which is where the Baum-Welch algorithm steps. Concisely,

Theorem IV.3 *(Hidden Filter Auxiliary Global Maximum) Given a hidden filter Markov model such that the state i conditional density, potentially with mean, is given by*

$$\log b_i(O_t) = -(1/2)\log 2\pi - (1/2)\log \sigma_i^2 - \frac{1}{2\sigma_i^2}(O_t - \mu_i + \sum_{k=1}^{p} a_{ik} O_{t-k})^2$$

then the Baum auxiliary function, pertaining to the output densities,

$$Q_b(\lambda, \bar{\lambda}) = \sum_t \gamma_t \left[-(1/2)\log 2\pi - (1/2)\log \sigma_i^2 - \frac{1}{2\sigma_i^2}(O_t - \mu_i + \sum_{k=1}^{p} a_{ik} O_{t-k})^2 \right] \quad (52)$$

has a single global maximum for fixed λ.

Proof: For the unidimensional, single order case, $p = 1$, let the predictor coefficient be denoted by b. The Baum auxiliary function defined in Equation 18 can be maximized separately for each state (Equation 52). Drop the state notation and define the reestimated model as

$$\bar{\lambda} = \theta\lambda_1 + (1-\theta)\lambda_2 \quad (53)$$

for $0 \leq \theta \leq 1$ where the new model $\bar{\lambda}$ is a linear combination of two arbitrary ones. Now examine the partial derivative of the auxiliary function with respect to θ, still updating to a critical point. Equation 53 implies the following is true.

$$\bar{\mu} = \theta\mu_1 + (1-\theta)\mu_2$$
$$\bar{\sigma} = \theta\sigma_1 + (1-\theta)\sigma_2$$
$$\bar{b} = \theta b_1 + (1-\theta)b_2$$

Letting $c \equiv 1/\sigma^2 > 0$,

$$\begin{aligned}
\partial^2 Q_b(\lambda, \bar{\lambda})/\partial\theta^2 &= \sum_{t=1}^{T} \gamma_t \left[-\frac{1}{2}\frac{(c_1 - c_2)^2}{\bar{c}^2} - \bar{c}((b_1 - b_2)O_{t-1} - (\mu_1 - \mu_2))^2 \right. \\
&\quad \left. -2((b_1 - b_2)O_{t-1} - (\mu_1 - \mu_2))(O_t - \bar{\mu} + \bar{b}O_{t-1})(c_1 - c_2) \right] \\
&= \sum_{t=1}^{T} \gamma_t \left[-\frac{1}{2}\frac{(c_1 - c_2)^2}{\bar{c}^2} - \bar{c}((b_1 - b_2)O_{t-1} - (\mu_1 - \mu_2))^2 \right. \\
&\quad \left. -2(b_1 - b_2)O_{t-1}(O_t - \bar{\mu} + \bar{b}O_{t-1})(c_1 - c_2) \right]
\end{aligned}$$

$$+2(\mu_1 - \mu_2)(O_t - \bar{\mu} + \bar{b}O_{t-1})(c_1 - c_2)] \tag{54}$$

Now, seek the second partial only at a critical point, implying both $\partial Q/\partial\bar{\mu} = 0$ and $\partial Q/\partial\bar{b} = 0$. Expanding,

$$\partial Q_b(\lambda, \bar{\lambda})/\partial\bar{b} = -\sum_{t=1}^{T} \gamma_t \left[-\bar{c}O_{t-1}(O_t - \bar{\mu} + \bar{b}O_{t-1})\right] = 0$$

implying

$$0 = \sum_{t=1}^{T} \gamma_t \left[-2(c_1 - c_2)(b_1 - b_2)O_{t-1}(O_t - \bar{\mu} + \bar{b})O_{t-1})\right] \tag{55}$$

$$\partial Q_b(\lambda, \bar{\lambda})/\partial\bar{\mu} = \sum_{t=1}^{T} \gamma_t \left[\bar{c}(O_t - \bar{\mu})\right] = 0$$

implying

$$0 = \sum_{t=1}^{T} \gamma_t \left[2(\mu_1 - \mu_2)(c_1 - c_2)(O_t - \bar{\mu} + \bar{b})O_{t-1})\right] \tag{56}$$

These last two expressions (Equations 55 and 56) cancel the last two terms in Equation 54 leaving a negative sum of squared positive terms. Since the sum is negative for all choices of λ_1 and λ_2, the auxiliary function is negative definite at the critical point. Also, if there were two critical points, the auxiliary function would have to switch positive for some pairs of λ_1 and λ_2. Since this was not evident, then only one critical point must exist and it is the global maximum. □

Naturally this result for a single conditional density applies to mixtures of conditional densities whereby the previous section constructively demonstrated a simpler *equivalence* model exists. Applying both results of this section and the last concludes that multiple mixtures of hidden filters can be transformed into an equivalent model with single filters per state, and each Baum-Welch iteration will step to the global maximum of the auxiliary function. Multiple iterations of Baum-Welch will monotonically increase the likelihood of the reestimated parameters.

4.5 Conclusion

Based on the assumed source model, it was first proven that a hidden filter Markov model sequence has less joint entropy than a sequence generated from a standard hidden Markov model. Pattern recognizers based on these correct models should exhibit lower classification errors. Gaussian mixture hidden Markov models have been demonstrated to be equivalent to single mixture larger models, with increased states. This allows currently known theorems relating to the convergence properties of the algorithm to be satisfied. Likewise, for conditional state density functions, the Baum auxiliary function is guaranteed to have a global maximum at the single critical point which is achieved for each iteration of the Baum-Welch algorithm. In summary, the direct application of the reestimation equations in Chapter III guarantees better models each iteration they are applied, and further implying they monotonically converge in likelihood. The next chapter uses the reestimation outlined in Chapter III, with the insight of the algorithmic properties outlined in this chapter, for the application of modeling speaker dependent phonemes for speaker recognition.

V. Speaker Recognition

5.1 Introduction

This chapter describes the extensive experimentation and evaluation of the hidden filter Markov modeling approach. The next section reminds the reader of why this application requires better techniques. A systems level description is first provided, shown in Figure 17. The YOHO database is described and used for all experiments, with initial experiments applied to speaker identification. Where appropriate, all methods compare results to vector quantization, a well-proven technique for text-independent speaker modeling. Speaker verification, an extremely difficult problem, compares log-likelihood ratios to *a posteriori* globally determined thresholds. Three methods of normalization, using close *cohort* speakers as a reference, are examined, with a second order approach being developed in this research. Lastly, an important general pattern recognition concern is analyzed, which answers the question, "Does my system meet requirements?" It will be shown that a particular configuration of our system meets the stringent U.S. Government requirement of 1% false reject and 0.1% false acceptance rates.

5.2 Why Better Speaker Recognition?

The National Institute of Standards and Technology (NIST) recently provided a set of guidelines [86] to Federal agencies and departments for verifying the identities of computer system users. They describe biometric-based authentication as the measurement of a unique biological feature used to verify the claimed identity of an individual through automated means. Biometric authentication mechanisms will attempt to measure a unique biological feature to the degree that only one person may be authenticated as a specific user. The biological feature may be based on a physiological or behavioral characteristic as remarked in Chapter I. The physiological characteristics measure vocal tract and other speech production physiology while the behavioral characteristics measure all other voice habits and patterns. This chapter examines hidden filter Markov modeling of phonemes for this identification and verification process.

Campbell writes [18]

> The LA Times recently reported that $1.2 billion is lost annually from telephone calling card fraud and the accounting firm of Ernst and Young estimates that high-tech computer thieves in the U.S. steal $3 to $5 billion annually.

The use of automatic speaker recognition could reduce these thefts substantially. In addition to these problems, legislation is being considered to automate, nationwide, the electronic distribution of welfare benefits using voice verification [36] among other techniques.

As introduced in Chapter I, speaker recognition includes speaker identification and speaker verification. When performing verification or authentication, the errors can be categorized by two measures, the False Acceptance Rate (FAR) and the False Rejection Rate (FRR). The FAR (*Type 2* errors) represents the percentage of unauthorized users who are incorrectly identified as valid users. The FRR (*Type 1* errors) represents the percentage of authorized users who are incorrectly rejected.

All experiments were performed on the Linguistic Data Consortium's (LDC) YOHO database, with initial identification results providing insight to the more extensive verification experiments. Following these experiments, a hypothesis analysis will provide the maximum *critical errors* allowed while still meeting the goal levels specified of 1% FR and 0.1% FA.

5.3 YOHO Database

The YOHO Speaker Verification database is the only large scale[1], scientifically controlled and collected, high-quality speech database for speaker authentication testing at high confidence levels. This corpus has been designed to test speaker verification at U.S. Government required error rates of 1% false rejection and 0.1% false acceptance [17, 67], with a goal level of one magnitude better. (0.1% False-Reject and 0.01% False-Accept). The 138 subjects, 106 males and 32 females, were asked to participate in 14 sessions over a 3-month interval. These included 4 enrollment sessions of 24 utterances each and 10 verification sessions of four utterances each.

[1]When uncompressed the raw speech consists of 1.2 gigabytes of data [67].

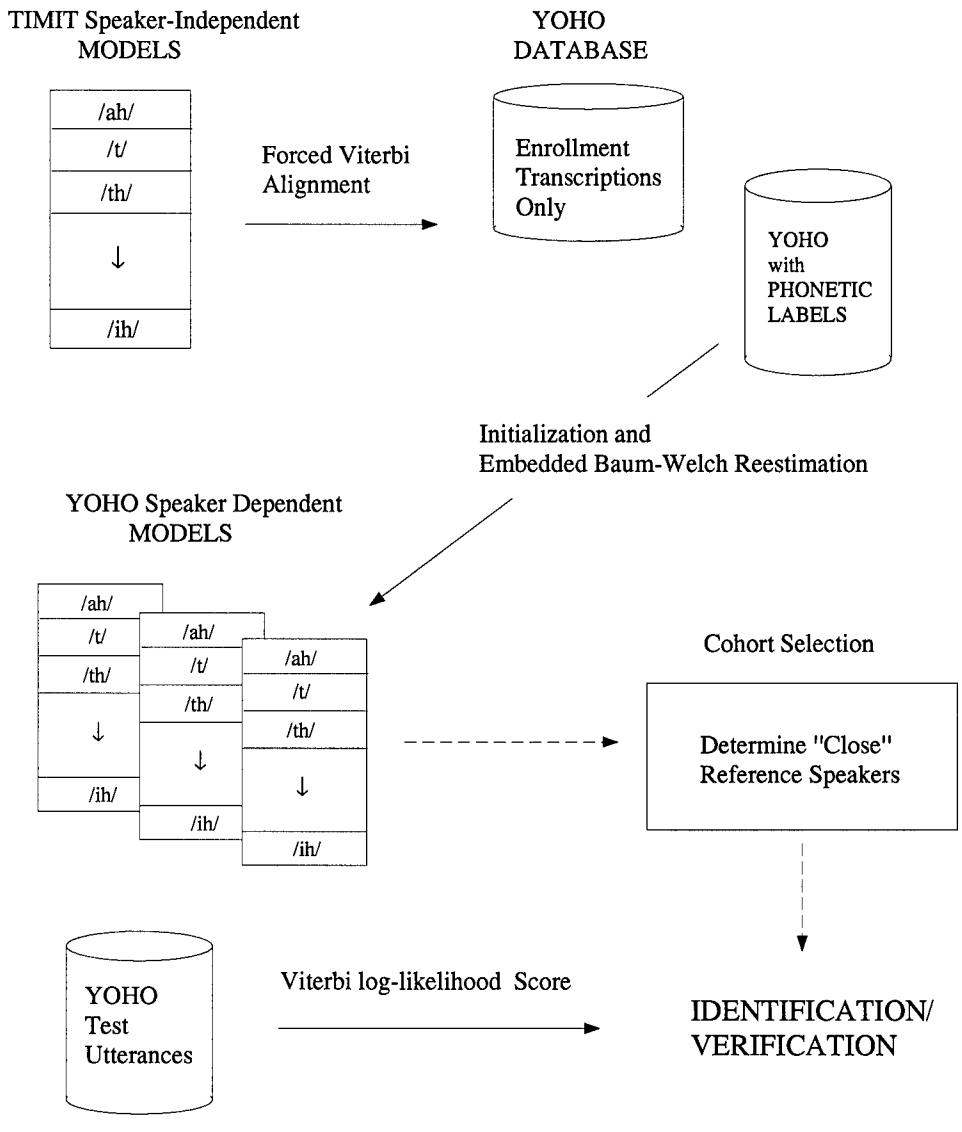


Figure 17. Speaker recognition system overview.

The speech material consists of "combination-lock" phrases. An example prompt is: "57 - 26 - 64", pronounced "fifty-seven, twenty-six, sixty-four". Each phrase consists of three number doublets. The doublets are chosen from a list which includes all the doublets from 21 to 99 with the following exceptions: (1) no exact decades (30, 40, etc.), (2) no double digits (22, 33, etc.), and (3) no numbers ending in "8" (28, 38, etc.). Pausing between the doublets is optional, but not encouraged [67]. The total number of words is sixteen producing 56 possible doublets and a list of 166,320 phrases.

5.4 Phonetic Labeling and Training

Using the full TIMIT database [56], single mixture - 3 state models were previously trained by Anderson [3] based on 12 MFCC, 12 Δ MFCC and 12 $\Delta\Delta$ MFCC, including log energy, Δ log energy and $\Delta\Delta$ log energy. The full set of Kai-Fu Lee's 49 phoneme models [69] allowed segmentation and labeling of the (as yet unlabeled, but transcribed) YOHO database.

5.4.1 Forced Viterbi Alignment. Viterbi decoding [48], for a single hidden Markov model, provides the most probable state sequence given an observation sequence. The algorithm also provides overall likelihood of the sequence. Since the transcriptions are provided for each enrollment utterance, a network of phoneme models which must be traversed from beginning to end is known. Consider building a very large, single hidden Markov model from the individual phoneme models. The Viterbi algorithm can uncover the most likely state sequence which in turn provides a phoneme label for each analysis frame. Thus, the *forced Viterbi* procedure constrains the decoding of an input observation sequence to a ordered list of word and phoneme transcriptions. While there is not yet a substitute for hand segmentation by a phonetician, the overall process is fast, efficient and remarkably reliable with a good set of trained models.

Table 3 provides the initial TIMIT phoneme list for monophone models constrained to the YOHO vocabulary. See also Appendix 5.A for example words using these phonemes and the actual language grammar. The YOHO vocabulary has 19 monophones, with an additional /sil/ (leading and trailing silence) and /sp/ (interword space). The /DX/ is not used in the TIMIT grammar.

Table 3. YOHO phoneme model List, with silence (sil) and interword space (sp).

AH	AX	(DX)	ER	F	IY	N	S	TH	V	sil
AO	AY	EH	EY	IH	K	R	T	UW	W	sp

The relative proportions for each phoneme, over the entire YOHO database, is provided in Figure 18. As evident from the graph, the enrollment data follows the identical

distribution as the test data[2]. These results indicate there is adequate coverage of the phoneme space within the enrollment data [19].

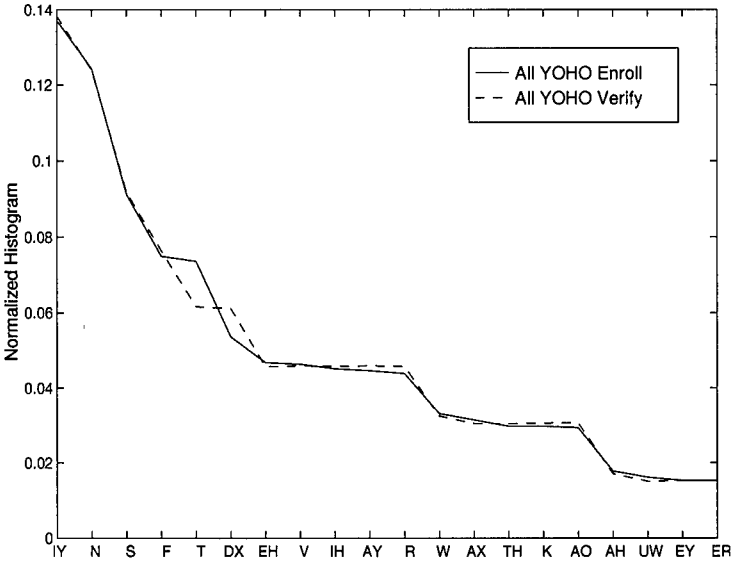


Figure 18. Histogram of all YOHO enrollment and verification utterances, after a forced Viterbi segmentation bootstrapped from TIMIT.

5.4.2 Embedded Reestimation. Once the entire YOHO database was phonetically marked, all four sessions of enrollment data were used to train speaker dependent models. The phoneme models were reestimated individually using the Baum Welch algorithm, with the initial model being the speaker independent TIMIT trained models, when possible. When not possible due to the architecture involved, an initialization procedure consisted, for each monophone separately, as follows: 1) Uniform segmentation into states 2) Segmental k-means based on Viterbi's most likely state sequence. Then, an embedded reestimation of all speaker models was accomplished by concatenating the individual monophones for the utterances and updating all models simultaneously using the Baum-Welch algorithm [134].

[2]Enrollment data accounts for only 0.057% of the total phrases possible.

5.5 *Speaker Identification Results on YOHO*

Speaker identification uses a Bayesian classifier, assuming equal priors, choosing speaker model i from the normalized Viterbi log likelihoods for an utterance, or set of utterances, $\mathcal{U}$. These results provide a reference for speaker separation, model and feature choice trade-offs. Speaker identification is provides an upper bound on verification error [34], based on entropy.

$$i = \arg\max_{k} \{ \log p(\mathcal{U}|\lambda_k) \} \tag{57}$$

First, an examination of vector quantization (VQ) provides a baseline on the YOHO males and females separately. The VQ procedure assumes independent observations and clusters speech without any temporal assumptions. Next, the frame and vector autoregressive techniques are applied to the YOHO database. These identification results select the alternative techniques which will be further investigated for verification.

5.5.1 Vector Quantization. The classic approach to modeling speakers creates a representation of their spectral vectors [4, 6, 47, 119], in this case the Mel frequency cepstral vectors. Codebooks were derived using the Linde-Buzo-Gray (LBG) clustering algorithm [75] over all enrollment sessions until convergence. Test results are derived from the Euclidean minimum distortion over all test utterance frames. Table 4 shows the closed set speaker identification for both 32 and 64 codeword models testing with 1, 2 and 4 combinations phrases. These results serve as the baseline performance for a non-temporal model applied to YOHO.

Table 4. Closed-set speaker identification error Rates(%) for 1,2 and 4 combination lock phrases applied to both 32 and 64 VQ codeword models. Features consisted of the 12 dimensional Mel frequency cepstral coefficients (MFCC) only.

Method	Males(Females)		
	1	2	4
VQ - 32 codewords	6.86 (6.17)	2.50 (2.50)	1.41 (0.94)
VQ - 64 codewords	4.27 (2.97)	1.65 (1.25)	1.04 (0.63)

5.5.2 Phonemic Frame AR Hidden Filters. Poritz [92] proposed the fundamental hidden filter model for speaker identification. His architecture consisted of a 5-state ergodic model using third order filters. This section extends the training to all individual phoneme models and reestimates all simultaneously using the Embedded Baum-Welch algorithm. The following approach (see Figure 19) uses labeled phoneme enrollment data to initialize the hidden filters and builds networks for embedded Baum-Welch reestimation based on transcriptions and word dictionaries. The result is a set of left-to-right speaker-dependent models, corresponding to specific phones. A forced Viterbi alignment using utterance transcription and word dictionaries provides the overall log likelihood score.

Unlike the VQ case which requires some spectral representation processing, the frame AR hidden Markov model only requires the autocorrelation of the raw samples. The frames must be gain normalized, since raw autocorrelation features vary greatly with signal energy. It was also demonstrated in Section 3.4 that by using the Poritz method on frames, only the autocorrelation coefficients are required in the reestimation. The resulting p-th order phoneme hidden filters are those which minimize the Itakura-Saito distortion to all frames assigned to a hidden state. Table 5 shows closed set speaker identification error rates for various p-th order filters and architectures. While these results are competitive to vector quantization, we further examine models using the vector Mel frequency cepstral process.

Table 5. Closed-set Speaker Error Rates (%) using Poritz Phoneme Models on 1, 2 and 4 combination phrases. Monophones consist of either 1 or 3-state left-to-right models with filter order p.

Method	Females		
	1	2	4
1-state, p=8	11.09	5.00	2.19
1-state, p=10	8.91	2.34	0.94
1-state, p=12	8.20	2.81	0.63
3-state, p=12	4.84	1.72	0.94

5.5.3 Vector Hidden Filters. The vector autoregressive hidden filters can be easily related to many existing statistical speaker recognition approaches. Denoting the number of states as N, the number of mixtures per state, M, and the predictor matrices,

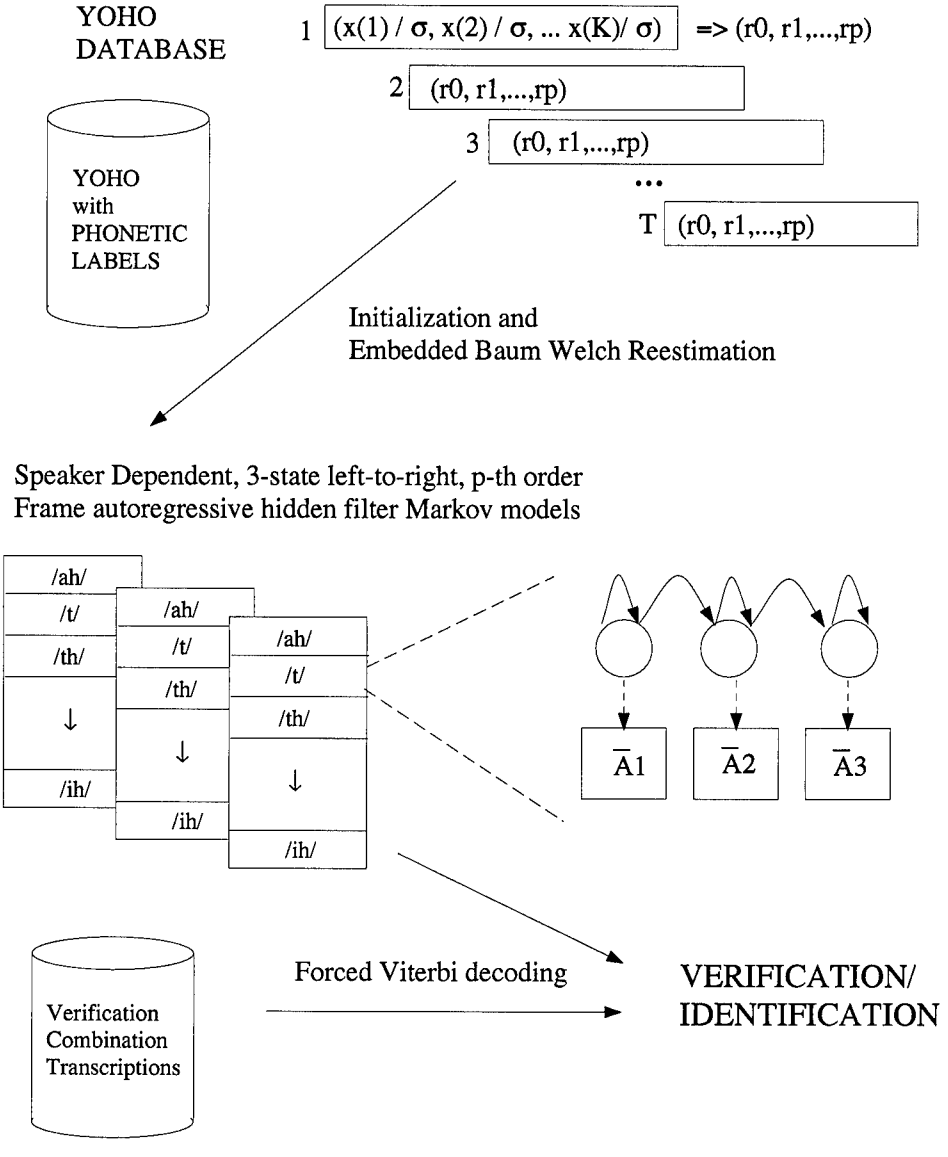


Figure 19. The phoneme frame autoregressive hidden filter approach models individual phonemes as 3-state left-to-right hidden filters. The $\bar{A}$ denotes a p-order hidden filter, as reestimated in Section 3.4.

$B_i = [A_{i1}, \ldots, A_{ip}]$, then the following models in Table 6 are attainable. Note that the hidden Markov model is attained when the hidden filter predictor matrices are all set to zero. A hidden Markov model based on Δ coefficients (Section 2.4.3) is related by a particular choice of matrices being set to I and $-I$ respectively.

Table 6. Relationship of Vector AR Hidden Filters to Other Models. Note: W denotes the size of the Δ window.

Vector Quantization	$B_i = 0, \Sigma_i = I, \forall i, M = 1,$
	All Transitions a_{ij} equiprobable.
Gaussian	$B_1 = 0$,N = 1, M = 1
Gaussian Mixture Model	$B_1 = 0$,N = 1
Hidden Markov Model (baseline)	$B_i = 0, \forall i$
Hidden Markov Model, (Δ coeffs)	$A_1 = 1, A_{2W} = -I, A_i = 0$, all other i
Vector AR Hidden Filter Markov model	Unconstrained reestimation

However, the correct choice of model filter order remains a difficult procedure, for any linear system [63, 94]. Several single state models have been examined, with increasing filter order. Appendix 5.B examines penalty function methods for correct model order selection. For hidden filter Markov models, this analysis is unique. By increasing the filter order, the residual variance, or prediction error, decreased, but all vector hidden filter models lacked the ability to distinguish between speakers or phonemes. Others have paradoxically noted better likelihood scores, yet decreased recognition. We propose an explanation for this phenomena, detailed in Appendix 5.C, based on the strict stationarity of the original speech samples. All further results will be based on a zero-th order vector hidden filter Markov model, i.e. HMM. We continue to extract and model context information by examining both first and second order regressive coefficients within this zero-th order architecture.

Table 7 shows error rates for various numbers of combination lock phrases. For each gender, two different Viterbi constraints were examined, Forced Viterbi alignment and Word-Pair Grammar[3]. The latter can be used to check if the prompted text matched the most likely Viterbi label hypothesis. The Word-Pair grammar also catches many confused doublets over a simple word dictionary grammar. For example, for the prompt "75-29-47", Viterbi with word grammar only may hypothesize a transcription of "seventy-five-one-nine-forty-seven" where this label is not valid under a word pair grammar, nor is it a valid

[3]In addition to forced Viterbi and Word-Pair, one easily could perform Word-Only grammar or No-grammar Phoneme decoding. It will be shown that allowing impostor's greater decoding flexibility decreases the separation between true user and impostor log-likelihood scores.

YOHO transcription. Given the superiority of the forced Viterbi alignment procedure, all remaining results chose forced Viterbi alignment based on the prompted transcription. An analysis concerning the entropy of the language, induced by the choice of grammar, dictates forced Viterbi is most suitable for the speaker recognition problem. See Appendix 5.D for full details.

Table 7. Closed-Set Speaker Error Rates(%) with Viterbi Constraints for 1,2 and 4 combination phrases.

Method	Males(Females)		
	1	2	4
Forced Viterbi	1.70(1.72)	0.47(0.78)	0.19(0.31)
Word Pair	1.75 (2.19)	0.47 (0.63)	0.38(0.31)

A practical pattern recognition concern is the amount of training data for model reestimation. With first and second order regression coefficients, each speaker is represented by 21 three-state monophone models, resulting in 4914 output density parameters per speaker. Based on an average of 38,000 enrollment observations, the ratio of training patterns to model parameters is 7.7. To increase this ratio, feature reduction and covariance sharing were performed. Table 8 shows that reducing the model size by removing transitional features increases error rates, and sharing covariance matrices among individual monophone states shows the opposite effect.

The best identification results are shown in Table 9 when the architecture includes two-mixtures per state, single shared diagonal covariance for each monophone and 21 monophones per speaker. Features include transitional information by incorporating Δ and $\Delta\Delta$ Mel frequency and energy coefficients. Experimentally, all male tests were correctly classified when prompting four combinations as a test trial. The inability to correctly identify all females can be explained by Campbell [17], where speaker #240 used a "false voice." See Appendix 5.E for the typical log-likelihood of these four test utterances. If this session were removed, all females would correctly classified, as well.

5.5.4 False Voice Effects. It has been noted by Campbell [17] that Speaker #240, in test session #969 used a "false" voice for all four utterances. This effected identification

Table 8. Closed-Set Speaker Error Rates (%) with/without Shared Covariance using forced Viterbi decoding for 1,2 and 4 combination phrases. Base feature is MFCC + Energy.

Feature	Males(Females), Σ/state		
	1	2	4
Base+Δ + $\Delta\Delta$	1.70(1.72)	0.47(0.78)	0.19(0.31)
Base+Δ	2.52(2.34)	0.99(0.94)	0.57 (0.31)
Base	5.83(5.55)	2.55(2.34)	1.60(0.94)
Feature	Males(Females), Σ/monophone		
	1	2	4
Base+Δ + $\Delta\Delta$	1.06(1.48)	0.47(0.78)	0.19(0.31)
Base+Δ	1.37(1.25)	0.57(0.47)	0.28(0.31)
Base	3.56(2.19)	1.46(1.41)	0.85(0.31)

Table 9. Closed-Set Speaker Error Rates(%) Using Decreasing Transitional Features and 2 Mixtures for 1,2 and 4 combination phrases. Base feature is MFCC + Energy.

Feature	Males(Females), Σ/monophone		
	1	2	4
Base+Δ + $\Delta\Delta$	0.92(1.48)	0.28(0.78)	0.00(0.31)
Base+Δ	1.16(1.02)	0.52(0.47)	0.19(0.31)
Base	3.09(2.03)	1.04(0.78)	0.66(0.31)

(and verification) results continually by misclassification of these particular trials. Shown in Figure 20 is the drop by several orders of magnitude in the normalized log-likelihood scores for these utterances using speaker #240's model. Note in identification (and verification) results for females, the 0.31% result is a consequence of these utterances.

5.6 Verification Methodology

The procedure of speaker verification involves some method of comparing the test utterance to "relative proximity" of the claimed speaker model, instead of simply choosing the maximum score for speaker identification (Equation 57). Often some threshold needs to be specified, either globally for all speakers or individual thresholds can be used. Recently, a proposal to use cohort speakers provides a method to use likelihood ratios as a basis for verification where a a global *a posteriori* threshold will be examined for equal error rate

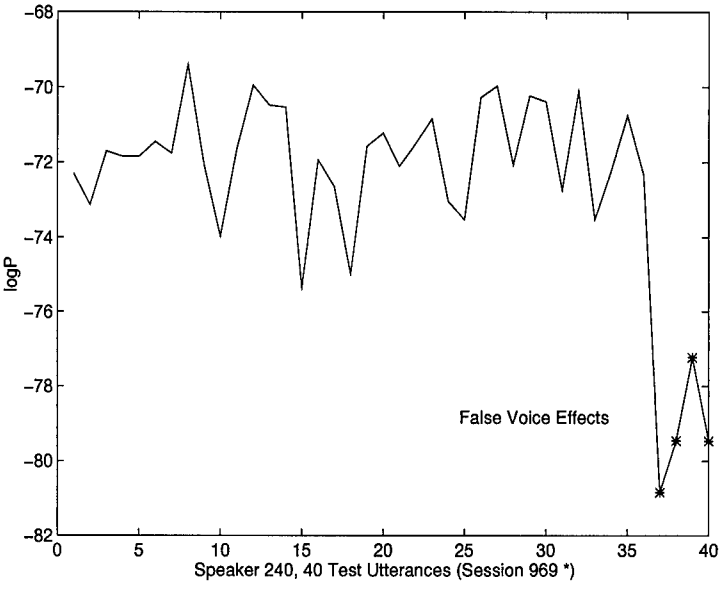


Figure 20. False voice effects of speaker 240, session 969, shown in the forced Viterbi normalized log-likelihood scores (*).

analysis. Examine Figure 21 to understand the reason for a relative threshold. For several test utterances of a male speaker, the forced Viterbi log likelihoods are plotted using the true speaker's model and several impostor models. The log ratios vary greatly across test utterances, yet each models appear to track with all others. Obviously, poor results would be observed using a single, fixed threshold.

5.6.1 Likelihood Ratios. The likelihood ratio test is a useful tool based on Bayesian analysis for performing speaker verification. The Bayes error rate, a statistical upper bound on performance of any pattern classifier [35, 110], is achieved by applying the Bayes decision rule. This maximum *a posteriori* (MAP) approach, given utterance $\mathcal{U}$, will

$$\begin{aligned} &\text{choose } \lambda_1 \quad \text{if, } p(\lambda_1|\mathcal{U}) \geq p(\lambda_2|\mathcal{U}) \\ &\text{choose } \lambda_2 \quad \text{otherwise} \end{aligned}$$

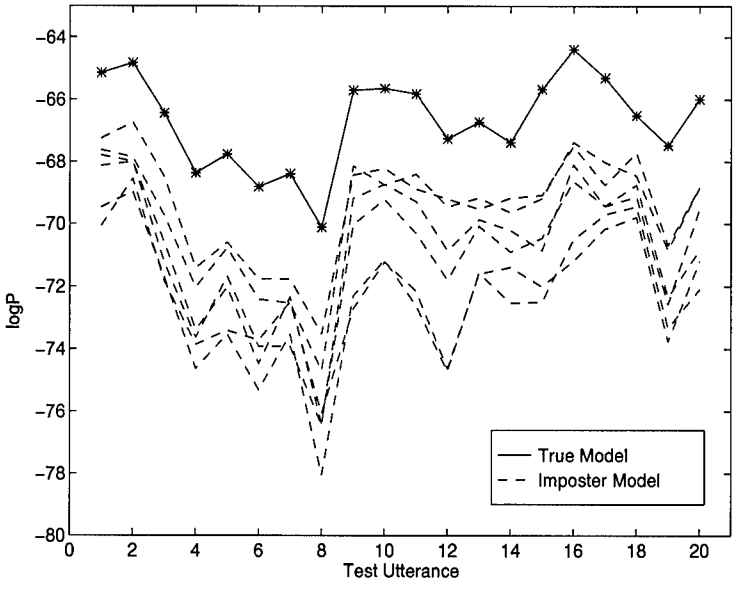


Figure 21. Typical log-likelihood of true model and impostor models shows the variability based on the transcription (prompted words) which forces some non-fixed thresholding scheme.

or by using Bayes rule, the probability density functions, either known or approximated, can be used. Taking the logarithm,

$$\log \frac{p(\mathcal{U}|\lambda_1)}{p(\mathcal{U}|\lambda_2)} \geq T, \qquad \text{where } T \equiv \log \frac{p(\lambda_2)}{p(\lambda_1)}$$

Speaker verification systems are then based on this log-likelihood ratio $\mathcal{L}$ of the utterance (or set of utterances) by applying the concept to a claimed model (λ_1) against not the claimant (λ_2).

$$\begin{aligned} \mathcal{L}(\mathcal{U}) &\equiv \log \frac{p(\mathcal{U}|\lambda = \lambda_{claim})}{p(\mathcal{U}|\lambda \neq \lambda_{claim})} \\ &= logp(\mathcal{U}|\lambda = \lambda_{claim}) - \log p(\mathcal{U}|\lambda \neq \lambda_{claim}) \end{aligned} \tag{58}$$

If the above quantity is greater than the threshold T, which accounts for the unknown speaker prior probabilities, the maximum likelihood decision is to accept the utterance $\mathcal{U}$ as the claimed speaker. We seek to approximate this last quantity using a set of "close" reference speakers, as suggested by Higgins [51]. Campbell establishes methods for testing

on YOHO by calling these reference speakers "cohorts". To determine "close", we examine training utterances through the set of reference models. This procedure will be referred to as cohort normalization of the log-likelihood ratio.

Furui [113] discusses several measures for cohort normalization, each a potentional approximation to the last expression of the log likelihood (Equation 58). Some of these approximations include the logarithm of the summation of cohort likelihoods or the summation (average) of log likelihoods [80]. This latter geometric mean cohort normalization method was used for these experiments. Specifically, define a set of cohort speakers $\mathcal{C}$ of size $|\mathcal{C}|$. Then, using the joint likelihood of the set of cohort speakers, the log-likelihood ratio is given by

$$\begin{aligned} \mathcal{L}(\mathcal{U}) &\approx \log \frac{p(\mathcal{U}|\lambda = \lambda_{claim})}{\frac{1}{|\mathcal{C}|}\sum_{j\in\mathcal{C}} p(\mathcal{U}|\lambda_j)} \\ &= \log p(\mathcal{U}|\lambda_{claim}) - \log \frac{1}{|\mathcal{C}|}\sum_{j\in\mathcal{C}} p(\mathcal{U}|\lambda_j). \end{aligned} \tag{59}$$

In practice, it has been reported this can be further approximated by

$$\mathcal{L}(\mathcal{U}) \approx \log p(\mathcal{U}|\lambda_{claim}) - \frac{1}{|C|}\sum_{j\in\mathcal{C}} \log p(\mathcal{U}|\lambda_j). \tag{60}$$

If the last expression is assumed to be dominated by the single closest reference speaker, then the maximum operator can also be used for normalization.

5.6.2 Measure of HMM Similarity. Each speaker is represented by 21 speaker dependent phoneme models, including silence and interword space. All 96 enrollment utterances are used to establish first and second order statistics for the Viterbi log likelihoods. Creating cohort sets is accomplished in one of three ways, each a sorted list of "close" speakers. Define the *Difference of Means* log ratio as

$$\begin{aligned} d_{DOM}(\lambda_i, \lambda_j) &\equiv \log \frac{p(\mathcal{U}|\lambda_i)}{p(\mathcal{U}|\lambda_j)} \\ &= \log p(\mathcal{U}|\lambda_i) - \log p(\mathcal{U}|\lambda_j) \end{aligned}$$

where $\mathcal{U}$ are all enrollment utterances for speaker i. Reynolds [104] provides a *Symmetric* distortion measure between two models using enrollment utterances from both the target and the potential cohort to determine similarity. If $\mathcal{U}_i, \lambda_i$ represent speaker i training observations and model respectively, then a symmetric distortion measure can be defined as

$$d_{SYM}(\lambda_i, \lambda_j)(\lambda_i, \lambda_j) \equiv \log \frac{p(\mathcal{U}_i|\lambda_i)}{p(\mathcal{U}_i|\lambda_j)} + \log \frac{p(\mathcal{U}_j|\lambda_i)}{p(\mathcal{U}_i|\lambda_j)}.$$

These approaches are examples of a first order statistical analysis of the output distributions. Several researcher's have examined the issue of measuring "distances" between HMMs [59] for measuring model similarity. The goal then is to search for the set of cohort HMMs which are close to the claimant's HMM in some probabilistic distance. If enough training sequences from each speaker are evaluated against each HMM, a distribution of log likelihoods begins to form, where a sample mean and variance can be extracted [42].

Higher order statistics can be used in conjunction with the *Bhattacharyya* distance for measuring the separability between the output distributions of a pair of HMMs [46]. The Bhattacharyya distance is derived from an analysis of determining an upper bound on the Bayes error rate of a two class problem. The form of this distance, for the 1-dimensional log likelihoods, is

$$d_B(\lambda_i, \lambda_j) \equiv \frac{1}{4}\frac{(m_i - m_j)^2}{\sigma_i^2 + \sigma_j^2} + \frac{1}{2}\log\left(\frac{\frac{\sigma_i^2+\sigma_j^2}{2}}{(\sigma_i^2\sigma_j^2)^{\frac{1}{2}}}\right)$$

where m_i represents the enrollment likelihood mean and σ_i^2 represents the enrollment likelihood variance. The first term is a measure of the class separability due to the difference in the means while the second term is a measure of separability due to the variance difference. Fielding [42] has shown how the use of second order statistics can be useful for HMM model comparisons. The Bhattacharyya distance applied to the log probability statistics provides a unique approach to log-likelihood ratio normalization.

5.7 Speaker Verification Results on YOHO

To avoid statistical dependence between phrases within each verification session, all four combination phrases are taken as a test sample, as outlined in [17]. Results are also shown when this dependence assumption is not made and all utterances (or pairs) are each taken as a sample. The standard procedure is not performing inter-gender tests or testing with cohort speakers.

5.7.1 Vector Quantization. Verification using the VQ approach makes use of a similar method to log likelihood ratios. If each speaker cluster is assumed the mean of a unity variance normal density, and all frames are independent, then the negative log-likelihood of a test utterance is proportional to the overall test utterance VQ distortion. A rank ordering of close speakers or "cohorts" is accomplished by the simple Difference of Means, the second order Bhattacharyya and Symmetric selection strategies using negative distortion for the log-likelihoods.

These cohorts provide a reference for the verification system. The claimed speaker model distortion is normalized by the average distortion of his or her closest cohorts. Equal Error Rates (EER) for each of the three cohort normalization methods are shown in Tables 10 and 11 for codebook sizes of 32 and 64 and cohort sizes of 5 and 10. Note, very little difference in cohort selection methods is evident.

Table 10. Speaker Verification Equal Error Rates (%) using vector quantization overall distortion for 1 and 4 combination phrases. Cohort normalization methods on the negative distortion include Difference of Means, Bhattacharyya, and Symmetric using 5 cohorts.

Cohort Normalization	Males (Females), VQ 32		Males (Females), VQ 64	
	1	4	1	4
Difference Of Means	4.88(6.64)	2.08(3.44)	3.56(3.76)	1.69(1.25)
Bhattacharyya	5.09(6.42)	2.15(3.12)	3.68(3.76)	1.60(1.28)
Symmetric	4.90(5.54)	1.96(2.18)	3.65(3.81)	1.60(1.32)

5.7.2 Phonemic Frame AR Hidden Filters. While identification results using frame autoregressive hidden filters performed adequately, the method applied to verifica-

Table 11. Speaker Verification Equal Error Rates (%) using Vector Quantization overall distortion for 1 and 4 combination phrases. Cohort normalization methods on the negative distortion include Difference of Means, Bhattacharyya, and Symmetric using 10 cohorts.

Cohort Normalization	Males (Females), VQ 32		Males (Females), VQ 64	
	1	4	1	4
Difference of Means	3.96 (4.92)	1.51 (2.17)	2.83 (2.87)	1.23 (0.96)
Bhattacharyya	4.09(4.86)	1.51 (1.93)	2.90(3.12)	1.13 (0.94)
Symmetric	4.03 (4.22)	1.59 (1.56)	2.90(3.12)	1.13 (0.94)

tion did not perform as well as VQ. This can obviously be attributed to the closeness of impostor and true claimant log-likelihood scores. For example, using four combination lock phrases and five cohorts, the best verification equal error rates were 4.68%, 4.68% and 4.33% for DOM, Bhattacharyya, and Symmetric cohort selection strategies, respectively. These results were based the best frame AR hidden filter model in the identification tests - 3-state left-to-right monophones with 12-th order filters.

5.7.3 Vector Hidden Filters. Tables 12 and 13 summarize the extensive verification test undertaken for this research. For each of the features, (MFCC+E, MFCC+E+Δ and MFCC+E+$\Delta\Delta$), a complete set of hidden filter Markov models (0-th order) were trained for each speaker. This set include 21 monophones per speaker, with each monophone model consisting of a 3-state left-to-right two-mixture hidden Markov model, sharing a single diagonal covariance.

All possible male (female) test utterances were applied to all male (female) models, respectively. The tables were generated using the approximation to the log-likelihood $\mathcal{L}$ defined by Equation 60 using a specific ordered set of cohorts $\mathcal{C}$. This ordered set was previously determined by passing all enrollment data through all the trained models and sorting by the three cohort similarity measures - d_{DOM}, d_{SYM} and d_B. The final equal error rate (EER) is calculated by stepping a global threshold until the average difference between false accepts error rates and false reject errors converges. Note the best equal error rate is found using the full 39-dimensional features with the Bhattacharyya cohort normalization and prompting 4 combination locks as a single test trial.

The extensiveness of these tests is further clarified. When one combination lock phrase is tested for verification, the number of potential false reject tests (true speaker claiming him/herself) is 4240 for males and 1280 for females. The potential false accept tests (impostors) for the five cohort table is 424,000 for males and 33,280 for females.

Table 12. Speaker Verification Equal Error Rates (%) using 5 cohorts based on 2 mixture 3-state monophones. Base is MFCC+E.

Feature	DOM Males(Females)		
	1	2	4
BASE+Δ + $\Delta\Delta$	1.39 (1.89)	0.89 (1.95)	0.66 (0.93)
BASE +Δ	1.58 (2.16)	0.99 (1.25)	0.74 (0.71)
BASE	2.38 (3.14)	1.50 (2.03)	1.22 (1.25)
Feature	Bhattacharyya Males(Females)		
	1	2	4
BASE +Δ + $\Delta\Delta$	1.53 (1.89)	0.89 (0.92)	0.68 (0.63)
BASE +Δ	1.70 (1.95)	1.07 (1.09)	0.83 (0.63)
BASE	2.57 (2.98)	1.55 (2.03)	1.13 (0.94)
Feature	Symmetric Males(Females)		
	1	2	4
BASE +Δ + $\Delta\Delta$	1.37 (1.72)	0.85 (0.78)	0.57 (0.63)
BASE +Δ	1.53 (1.79)	0.90 (0.94)	0.566 (0.60)
BASE	2.38 (2.82)	1.51 (1.89)	1.03 (1.25)

Figure 22 demonstrates the effectiveness of the Bhattacharyya distance when used in conjunction with the log-ratio normalization compared to other cohort selection methods.

5.8 Critical Error Analysis

Higgins [51], and more recently Campbell [17], has examined the statistical significance of the YOHO experiments based on confidence intervals. This section presents an alternative method using hypothesis test analysis at the highest significance levels for the amount of YOHO data. This presentation using significance levels provides a straightforward approach in accepting a potential speaker verification system. The technique easily generalizes to any pattern recognition problem where a target level of acceptability is provided. We place much greater emphasis on rejecting potentially unacceptable systems

Table 13. Speaker Verification Equal Error Rates (%) using 10 cohorts based on 2 mixture 3-state monophones. Base is MFCC+E.

Feature	DOM normalization Males(Females)		
	1	2	4
BASE+$\Delta+\Delta$	0.94 (1.41)	0.51 (0.63)	0.38 (0.35)
BASE +Δ	1.04 (1.41)	0.56 (0.94)	0.47 (0.31)
BASE	1.72 (2.17)	0.95 (1.12)	0.75 (0.66)
Feature	Bhattacharyya normalization Males(Females)		
	1	2	4
BASE +$\Delta+\Delta$	0.92 (1.56)	0.47 (0.63)	0.21 (0.55)
BASE +Δ	1.01 (1.56)	0.66 (0.63)	0.47 (0.31)
BASE	1.79 (2.50)	1.03 (1.41)	0.56 (0.94)
Feature	Symmetric normalization Males(Females)		
	1	2	4
BASE +$\Delta+\Delta$	0.97 (1.25)	0.52 (0.51)	0.38 (0.32)
BASE +Δ	1.06 (1.39)	0.56 (0.47)	0.47 (0.31)
BASE	1.84 (1.95)	1.08 (0.97)	0.68 (0.56)

than on accepting potentially acceptable ones. For the speaker verification problem, the consequences of a wrong decision dictates this approach.

Define the null hypothesis, H_0, to be the System Error Rate, Ser, does not meet the Target Error Rate Ter,

$$\begin{aligned} H_0 &: Ser > Ter \quad \text{UNACCEPTABLE} \\ H_1 &: Ser \leq Ter \quad \text{ACCEPTABLE} \end{aligned} \tag{61}$$

Previously, results have been reported at the 75% confidence level for False Acceptance and False Reject target values. However, this method would pass a large percentage of systems that are in reality unacceptable.

The main concern should not be the probability of meeting the Target Error Rate, which a confidence level analysis provides; the main concern should be in the decision to reject potential candidates taking into account the consequences of a wrong decision. Conjecture all systems are unacceptable and allow the experimental evidence (observed

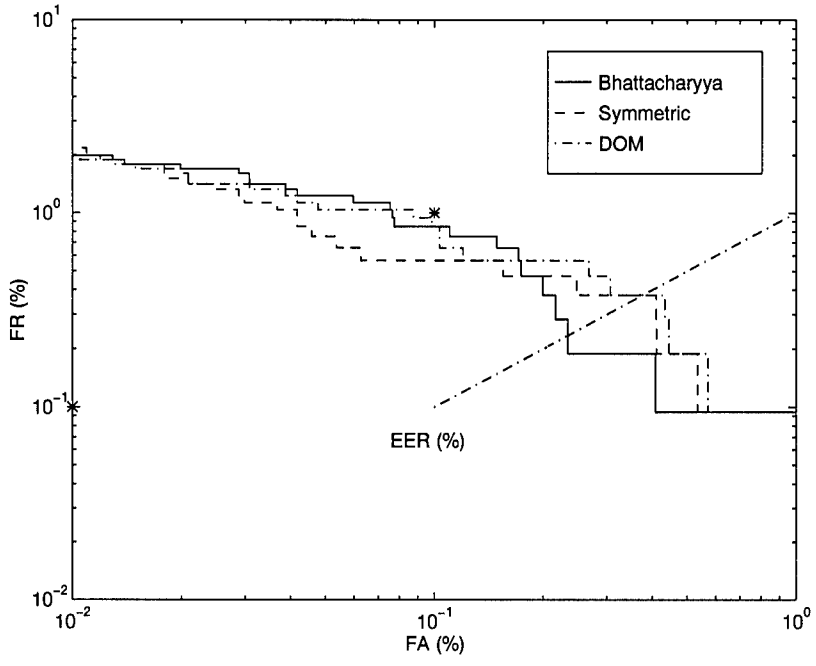


Figure 22. Speaker Verification False Accept and False Reject Error Rates (%) using DOM, Bhattacharyya and Symmetric cohort selection strategies. Results show the effect of an unseen threshold. Data only used male speakers, when prompted with 4 combination lock phrases and normalized with 10 cohorts using full 39-dimensional features (MFCC+E+Δ + $\Delta\Delta$). Best Equal Error Rate shown is 0.21% using the Bhattacharyya normalization. Note: (*) denotes U.S. Government requirement of 1% FR and 0.1% FA [18, 17].

errors) to reject this conjecture [89]. One can also examine the probability of failing acceptable systems, but this is a secondary concern.

5.8.1 Statistical Assumptions. Many times, we perform a set of tests and report average results, typically with confidence intervals. Tests can average over several random initial experiment setups (Monte Carlo Confidence Interval) or averages can be based on the number of total test observations (Classifier Confidence Interval) [111]. However, if a target error rate is specified, then instead of bounding the results, one needs to specify how confident we are of meeting or exceeding this specification.

Ruck [111] reviews the approach and the procedures for both Monte Carlo and Classifier confidence intervals. Each independent recognition trial is a Bernoulli random variable, taking values 0 and 1 if the verification or identification was correct or incorrect, respec-

tively. From elementary probability, the sum of Bernoulli random variables takes on a binomial distribution, thus the total number correct (or incorrect) is a binomial random variable. Under certain conditions, a Poisson or normal random variable may be used to approximate the binomial and easily specify confidence intervals.

Let X be the total number of errors - a random variable. Given n independent tests with a p probability of error, then the binomial distribution is

$$p(X = x; n, p) = \sum_{i=0}^{x} \binom{n}{k} p^k (1-p)^{n-k}$$

which is the probability of observing x total errors. Suppose we observe x errors on the n tests performed. Our point estimate for p is x/n. However, a better method of specifying the true error probability p is to bound it at the $\gamma = 95\%$ or 99% confidence interval. The boundary values (random variables) we seek are p_L and p_H such that

$$P(p_L \leq p \leq p_H) = \gamma.$$

Since n is exceedingly large for YOHO experiments, an approximation to the binomial proves efficient. It has been noted that X is approximately normal when n is large with mean np and variance $pqn, q = 1 - p$. Hoel [52] provides some experimental insight, in that this approximation is valid when $np > 5$, $p \leq .5$, $nq > 5$ and $q > .5$. Small values of p with "moderately" large n would skew the distribution, and thus the following summary holds in practice:

n small	$\rightarrow$	Use Binomial,
n large, p large/small	$\rightarrow$	Use Poisson,
n large, p moderate	$\rightarrow$	Use Normal.

Using the Poisson approximation to the binomial (which is good for error rates less than 5% and number of trials greater than 100), critical error curves [51] are drawn based on the hypothesis test formulation in Equation 62 using the most stringent significance levels, Figures 23. Following Higgins [51],

Definition V.1 *(Critical Error) The Critical Error is the maximum number of errors able to be observed before rejecting the recognition system at a given significance level.*

The probabilities of accepting a system for various critical errors is given in Figure 24. Using these graphs allows recalculation of critical errors for YOHO in Table 14 and Table 15.

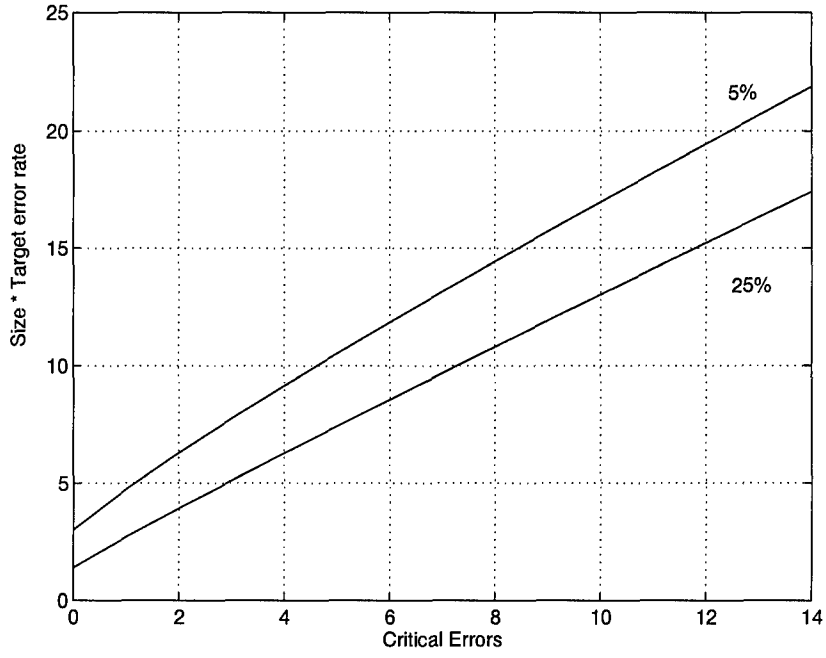


Figure 23. Critical Errors for Tests Designed at the 5% and 25%. The curve is generated by searching for the appropriate λ value given a particular (discrete) Critical Error. The curve is used by knowing $\lambda = N \cdot Ter$, the Target Error Rate (Ter) and the size of the database N, then reading over and down.

We review the creation of these graphs, since their full understanding can lead to applications elsewhere. For our application, these graphs provide the maximum number of critical errors able to be seen while still satisfying the target error rate. However, one could use this analysis for sizing a particular database by pre-specifying the critical errors. The Poisson distribution has been chosen in our case since: 1) n can be up to 4240 for identification and over 110,000 for verification and 2) the error rates are specified at 1%

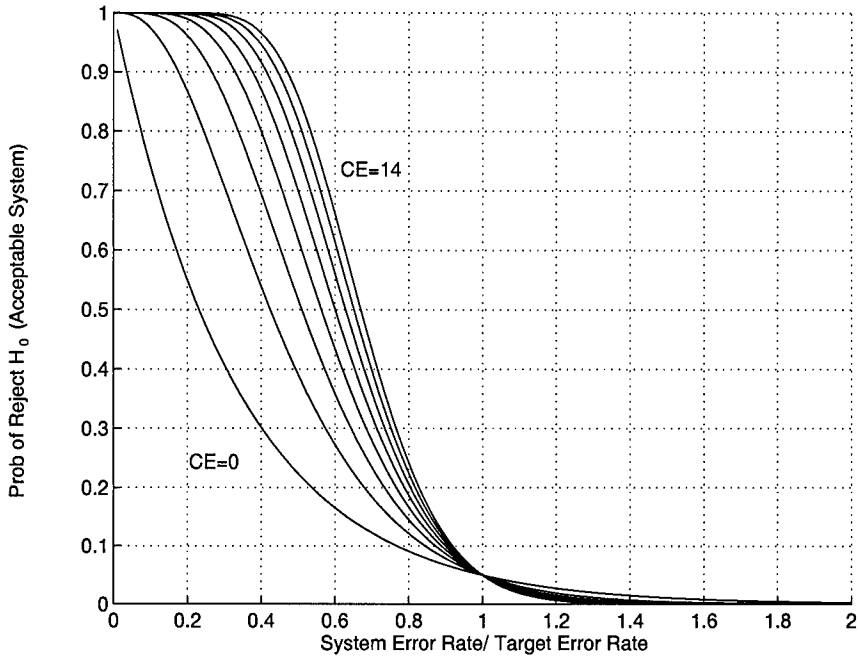


Figure 24. Probability of Rejecting H_o - Accepting the System Meets the Target Error Rate Ter, for number of critical errors (0,2,4,6,8,10,12,and 14) at the 5% significance level.

and 0.1%:

$$\text{Poisson distribution: } p(x;\lambda) = \sum_{k=0}^{x} \frac{\lambda^k e^{-\lambda}}{k!}$$

As can be seen, only the single parameter λ specifies this distribution and subsequently its mean and variance.

5.8.2 Application of Hypothesis Test. Table 14 provides critical errors (CE) for particular FA and FR target error rates. Since the number of false rejects is limited (4,240 for males and 1,280 for females) one cannot report results at the 5% significance level, and the entries for False Accept are provided at the 25% significance level [17]. Also, we chose to use all impostor tests available, counting each session as statistically independent. This amounts to total false acceptance (FA) tests of 106,000, and 100,700 based on the number of cohorts (5 and 10) respectively. The rationale for this decision is based on allowing more than one session for false reject testing and counting those as independent.

Table 14. Critical Errors (CE) at the 5% and 25% Significance level (Sigf) for the U.S government Required and Goal Target Error Rates (Target). Shown separately for False Accept (FA - based on 5 cohorts) and False Reject (FR) tests. Table 14 provides approximate readings from Figure 24, where the ratio of hypothesized $Ser/Ter = e$ and the Probability of Accepting the System is denoted Ppass. Sizes is attainable with the YOHO database.

Test	Target	Sigf	Ppass	e	Size	CE
FR	1.0%	25%	70%	2/3	1,080	8
FR	0.1%	25%	50%	1/2	1,386	0
FA	0.1%	5%	99%	2/3	105,065	88
FA	0.01%	5%	57%	1/2	105,131	5
FA	0.1%	25%	99%	2/3	105,517	98
FA	0.01%	25%	88%	1/2	96,845	7

Table 15. Critical Errors (CE) at the 5% for the U.S government Required and Goal Target Error Rates (Target). Shown for False Accept (FA - based on 10 cohorts). All other columns described in Table 14.

Test	Target	Sigf	Ppass	e	Size	CE
FA	0.1%	5%	98%	2/3	100,700	84
FA	0.01%	5%	99%	1/2	91,535	4
FA	0.1%	25%	99%	2/3	100,345	93
FA	0.01%	25%	88%	1/2	96,845	7

For example, in order to pass a system at the 5% significance level with a Target of 0.1% False Acceptance Rate, one must achieve less than or equal to 88 errors in 105,065 impostor tests. In addition, if we think our system is twice as good as the target, $e = 1/2$, then we have a 99% probability of accepting the system. An interesting conclusion to this analysis will be demonstrated through Figure 25. Similar to Figure 22, though instead of percent errors, actual counts are plotted as an unseen threshold is varied. The "*" denotes FA and FR critical errors at 5% and 25% significance level. While Figure 22 appears to indicate that all three cohort normalization methods passed within the specified U.S. Government target, in actually, when the hypothesis test (Equation 62) is used with a specified significance level, only one method would actually be accepted.

Due to the great imbalance of impostor tests, we can make a much stronger statement by mixing the significance levels as follows.

The speaker dependent phoneme system, based on cohort selection using the symmetric score passes the 0.1% False Acceptance target rate at the 5% significance level, while passing the 1% False Reject target rate at the 25% significance level.

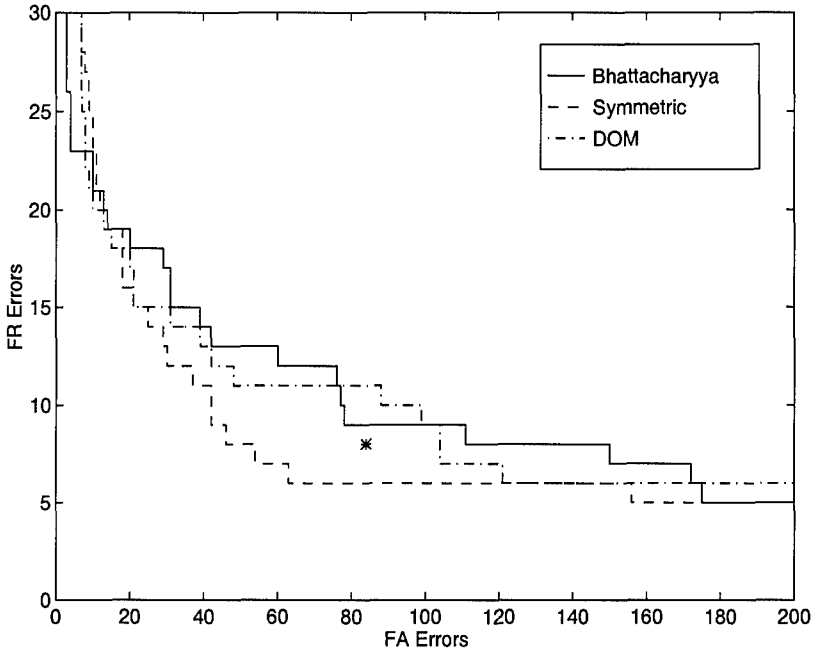


Figure 25. Speaker Verification False Accept and False Reject Error (#) using DOM, Bhattacharyya and Symmetric cohort selection strategies. Results show the effect of an unseen threshold. Data only used male speakers, when prompted with 4 combination lock phrases and normalized with 10 cohorts using full 39-dimensional features (MFCC+E+Δ + $\Delta\Delta$).

5.9 Conclusion

This chapter demonstrated several new findings concerning the ability to model and subsequently identify or verify speakers based on the acoustic signal. First, it was demonstrated that vector quantization, a reliable and proven method, provides similar performance to the Poritz, frame autoregressive model. Whereas both have about equal param-

eters, the hidden filter approach can also hypothesize the word-string spoken. Though the vector hidden filters method using the baseline Mel frequency cepstral representation showed better modeling with increased filter orders (Appendix 5.B), they did not provide any classification usefulness. One plausible explanation relates to the strict stationarity of the cepstral coefficients as hypothesized in Appendix 5.C. This manifests into a trivial filter across all phonemes and all speaker. Another explanation may relate to the dilemma related to classifying signals based on prediction. Often the better the prediction of training data, the less generalization occurs during test.

By using a 0-th order filter, which models states as noisy constant functions, statistically significant improvements were demonstrated over vector quantization. The addition of transitional coefficients and the addition of several prompted phrases monotonically decrease errors. A shared covariance used across the phoneme states also decreased errors for identification, probably due to the limitation on enrollment data. Best results of 100% identification on both male and female[4] were demonstrated using two mixture 0-th order filters.

Log ratios and log ratio normalization using cohorts were introduced. Three methods of selecting close speakers were examined, with the Bhattacharyya distance, a new approach developed within this research which includes second order statistics, was shown the optimal selection scheme when equal error rate (EER) is the benchmark. A more significant critical error analysis was developed to specify the maximum errors allowed while still achieving a specified target error rate. This analysis was applied to the YOHO database. The noteworthy conclusion of this section included the ability to easily make a claim of meeting requirements based on the maximum number of errors seen during testing.

Comparison to Campbell's synopsis [18] of recent known results demonstrates the effectiveness of these approaches, see Table 16. These tests were designed to model context and coarticulation at the subword (phoneme) level for speaker modeling. Historical insight dictated that both the physiology of a speaker and the neural habits and patterns together

[4]100% based on removing the false voice session of speaker 240.

Table 16. Recent LDC YOHO Database Results [17].

Reference	Verification EER (%)	Identification Error (%)
ITT NN	0.5	
ITT CSR	1.7	
MIT/LL GMM	0.51 (0.2m, 1.5f)	0.8 (0.3m, 2.2f)
Rutgers' NTN	0.65	
Rutgers' HMM		1.36
Rutgers' LVQ		0.36
COLOMBI	0.21m, 0.31f	0.0m, 0.31f

differentiate speakers. Both these unique traits alter the dynamics of the acoustic signal which we have successfully modeled with a hidden Markov architecture.

VI. Recommendations and Conclusions

6.1 Recommendations

During the course of good research, several avenues often arise which are not taken.

Two roads diverged in a wood, and I –
I took the one less traveled by,
And that has made all the difference.

Robert Frost, 1915

This section addresses those areas which will have great potential for robust dynamic time series modeling or applications in speaker recognition. We recommend the following research areas, in order of importance:

Speaker Verification Normalization. Normalization of the likelihood scores provides orders of magnitudes improvement over fixed thresholds. The fundamental reason for their requirement lies in the overall likelihood score containing much more than speaker contributions. As evident from Figure 21, the variability in the log-likelihood scores reflects word sequence and ordering, phonemic content and other language phenomena - all not related to speaker verification. Basic research in removing language and grammar effects, which are present in current speaker models, would be significant to future systems concerning speaker authentication and speaker adaptation.

NonCausal Filters. It has been observed that the human body appears to be a multichannel, noncausal processing machine [109]. Multichannel refers to the several human sensors, all coherently merged in our billions of neurons to form a single consistent "world model". Various noncausal capabilities have been observed in our perception of time and sound (see auditory illusion of phoneme restoration by Warren and Warren [127, 128]). This research has focused exclusively on causal filters. In the signal processing and modeling literature there has been recent interest in Two Sided Prediction (TSP) and other noncausal approaches. These models, for classification, should be examined for providing better forward and backward context, potentially within a hidden Markov model.

Prediction and Classification. The need for theoretic relations between the accurate ability to classify sequences and their prediction needs to be addressed. Under certain assumptions, Levin has shown for the nonlinear Markov-modulated dynamic systems, there exists a direct relation between reducing mean squared error in the training set and the overall likelihood of the training set. However, an investigation should be conducted relating optimal prediction to optimal classification.

Discriminant Models. The theory behind Maximal Mutual Information (MMI) [16] attempts to *not* model the maximum likelihood estimates of parameters, such as Baum-Welch achieves. Instead, since the correct source models will never be known, training methods should be discriminative and speaker models should be trained in conjunction with all others for optimal discrimination. While this may be optimal in terms of the least amount of needed assumptions, the amount of training data and the ability to add new classes make this technique inefficient. Methods should be researched which weight discrimination to the ability to add new classes (speakers) and to estimate parameters efficiently. Naturally, *discriminative hidden filter* models should also be investigated.

Another related research area lies the effective use of artificial neural network technologies. As presented in the framework of general hidden filters (Chapter II), nonlinear and discriminative techniques have recently been examined for output density estimation, within hidden Markov models. However, they have often involved large "stupid" neural networks and required specialized, fast hardware for training. Continued work in new neural architectures and their placement in the hidden Markov architecture should be performed.

6.2 Contributions

A number of original research contributions have been provided.

Generalized Hidden Filter Architecture. A complete framework for many existing linear and nonlinear systems used for classification, as well as prediction, was developed for discrete state Markov models. The existing hidden Markov model inde-

pendence assumptions were reviewed and removed, which defined a new, more generalized, hidden filter Markov model.

AR and ARMA hidden filters. New reestimation methods are provided for autoregressive (AR) and autoregressive moving average (ARMA) as well as an optimal initialization strategy. The ability to reestimate these filters adequately for the difficult ergodic case is novel and shown by example. The new ARMA Markov modulated hidden filters are applicable to specific broad classes of phonemes, with a spectral zero component. An extension to frame autoregressive hidden filters was proposed for accurate phoneme modeling and applied to speaker recognition.

Vector Autoregressive Hidden Filters. The extension from sample or frame based filters to full vector autoregressive hidden filters was developed using an emit-on-state notation. Full and diagonal regression variations were developed. The choice of spectral features used in this research, the Mel frequency cepstral coefficients, dictated a diagonal predictor and noise model. A procedure of *a posteriori* mean removal was developed to separate the state mean estimation from the filter coefficients for numerical stability.

HMM and Hidden Filter Convergence. A new proof of monotonic convergence for Gaussian mixtures was presented using an *equivalence* model paradigm. A new proof of monotonic convergence for hidden filter Markov models was then demonstrated. An application of the Markov property of the observations for hidden filter models was applied to the Fielding [42] information theoretic proof. Since pattern recognition methods seek ways which reduce entropy (and reduce classification errors), this proof justified the hidden filter model over standard hidden Markov models.

Phonetic Modeling for Speaker Recognition. A speaker dependent phoneme-based hidden Markov model system was accomplished for both speaker identification and verification using the extensive YOHO database. State-of-the-art speech recognition tools were incorporated into the system such as phonetic labeling, word dictionaries, bi-word language models and Viterbi scoring constraints. The left-to-right 3-state

phoneme models were analyzed exclusively. The use of ergodic structures was hypothesized and demonstrated as modeling language effects and thus dictated the choice of left-to-right monophone models. The method of forced Viterbi decoding of phoneme based temporal models for speaker verification was shown optimal, with a theoretic explanation demonstrated with language entropy. A novel approach to correct hidden filter model order using penalty functions was reported indicating a monophone dependent filter order. However, the optimal hidden filter used for all tests happened to be the 0-th order hidden Markov model - a hypothesis concerning the strict sense stationarity of speech was offered to explain this effect. A new second order metric for cohort selection was developed and shown to provide the best equal error rate of 0.21% on YOHO males and 0.31 on females. A critical error analysis is provided for YOHO using a hypothesis test technique which demonstrated the importance of comparing results to a test statistic.

6.3 Conclusions

A complete system framework for hidden filter Markov models has been developed and applied to the speaker recognition problem. This research proposed theoretical extensions to a class of stochastic models and demonstrated their effectiveness on the problem of text-independent (constrained) speaker recognition. Analysis concerning multiple mixtures and hidden filter models guarantee monotonically increasing likelihoods during learning. Using information theory, the hidden filter Markov models were demonstrated optimal over hidden Markov models for pattern recognition problems. Both closed set identification and normalized likelihood ratio verification using cohorts were performed on the extensive YOHO database. Perfect identification for males and females was possible prompting four combination lock phrases. Equal error rates of 0.21% males and 0.31%, females was accomplished using a forced Viterbi scoring and cohort normalization incorporating a newly developed Bhattacharyya distance metric. Where other researchers report equal error rates, this research demonstrated the importance of a critical error analysis, basing acceptance on the number of critical errors - found using a hypothesis test technique. We feel this document advances the state-of-the-art in areas of Markov modulated

dynamic systems, and their properties, log-likelihood normalization, and speaker authentication/ verification techniques.

Many new applications will require speech-based biometric recognition such as secure access control, telephone-based recognition, transaction and credit account verification, forensic science, law enforcement and military intelligence gathering. Successful methods, such as demonstrated by this research, provide excellent results of only 2 errors in 1000 attempts. The theoretic contributions clearly demonstrate the efficiency of training these models and their justifiable use over existing techniques for many pattern recognition problems, beyond speaker recognition. Insights from world-class researchers, suggesting the importance of dynamic modeling of phonemes, have contributed to make this research state-of-the-art in the challenging field of speaker recognition.

Appendix A. Induction Derivation of the Forward-Backward Variables

The following provides the inductive calculation of the forward and backward variables. Initial condition:

$$\begin{aligned}\alpha_1(i) &= p(O_1, q_1 = i|\lambda) \\ &= p(O_1|q_1 = i, \lambda)p(q_1 = i|\lambda) = b_i(O_1)\pi_i\end{aligned}$$

which is valid for $1 \leq i \leq N$. Given α_t, now find α_{t+1}:

$$\begin{aligned}\alpha_{t+1}(j) &= p(O_1 \cdots O_{t+1}, q_{t+1} = j|\lambda) \\ &= p(O_{t+1}|O_1, \ldots, O_t, q_{t+1} = j, \lambda)p(O_1, \ldots, O_t, q_{t+1} = j|\lambda) \\ &= b_j(O_{t+1})p(O_1, \ldots, O_t, q_{t+1} = j|\lambda)\end{aligned}$$

Expanding,

$$\begin{aligned}p(O_1, \ldots, O_t, q_{t+1} = j|\lambda) &= \sum_{i=1}^{N} p(O_1, \ldots, O_t, q_{t+1} = j, q_t = i|\lambda) \\ &= \sum_{i=1}^{N} p(q_{t+1} = j|O_1, \ldots, O_t, q_t = i, \lambda)p(O_1, \ldots, O_t, q_t = i|\lambda) \\ &= \sum_{i=1}^{N} a_{ij}\alpha_t(i)\end{aligned}$$

Hence,

$$\alpha_{t+1}(j) = b_j(O_{t+1}) \sum_{i=1}^{N} a_{ij}\alpha_t(i)$$

and this is valid for $1 \leq t \leq T-1$ and $1 \leq j \leq N$. For $t = T$ the total probability is given as

$$\begin{aligned}p(O_1 \ldots O_T|\lambda) &= \sum_{i=1}^{N} p(O_1 \ldots O_T, q_T = i|\lambda) \\ &= \sum_{i=1}^{N} \alpha_T(i)\end{aligned}$$

For the backward variable, let the initial condition be

$$\beta_T(i) = p(O_{T+1} \cdots O_T | q_t = i, \lambda) = 1$$

for $1 \leq i \leq N$. Given β_{t+1}, now find β_t

$$\begin{aligned}
\beta_t(i) &= p(O_{t+1} \ldots O_T | q_t = i, \lambda) \\
&= \sum_{j=1}^{N} p(O_{t+1} \ldots O_T, q_{t+1} = j | q_t = i, \lambda) \\
&= \sum_{j} p(O_{t+1} \ldots O_T | q_{t+1} = j, q_t = i, \lambda) p(q_{t+1} = j | q_t = i, \lambda) \\
&= \sum_{j} a_{ij} p(O_{t+1} | O_{t+2} \ldots O_T, q_{t+1} = j, q_t = i, \lambda) p(O_{t+2} \ldots O_T | q_{t+1} = j, q_t = i, \lambda) \\
&= \sum_{j} a_{ij} b_j(O_{t+1}) p(O_{t+2} \ldots O_T | q_{t+1} = j, q_t = i, \lambda)
\end{aligned}$$

Note $O_{t+2} \ldots O_T$ is independent of $q_t = i$ by Markov property

$$\begin{aligned}
\beta_t(i) &= \sum_{j} a_{ij} b_j(O_{t+1}) p(O_{t+2} \ldots O_T | q_{t+1} = j, \lambda) \\
&= \sum_{j} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)
\end{aligned}$$

Hence,

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)$$

for $t = T-1, T-2, \ldots, 1$ and $1 \leq i \leq N$. For $t = 1$ the total probability is calculated as:

$$\begin{aligned}
p(O_1 \ldots O_T | \lambda) &= \sum_{i=1}^{N} p(O_1 \ldots O_T, q_1 = i | \lambda) \\
&= \sum_{i} p(O_1 \ldots O_T | q_1 = i, \lambda) p(q_1 = i | \lambda) \\
&= \sum_{i} \pi_i p(O_1 | O_2 \cdots O_T, q_1 = i, \lambda) p(O_2 \cdots O_T | q_1 = i, \lambda) \\
&= \sum_{i=1}^{N} \pi_i b_i(O_1) \beta_1(i)
\end{aligned}$$

Appendix B. Phonetic Listing With Examples

The following Table 17 provides the list of phoneme with examples. Each phoneme will be represented by a 3-state left-to-right hidden filter Markov model trained separately for each speaker. Table 18 provides the TIMIT language grammar used in all experiments and two additional grammars - a grammar from the Resource Management (RM) database [134] and an optional mixture of RM and TIMIT. The examination of these grammars provide dictionaries in which clear read text, conversational speech or some combination of the two will be observed. After initial experiments, recognition results were not significantly different and the TIMIT grammar was used for the remaining experiments.

Table 17. Partial Phonetic List from Parsons [90] Applied to Digits.

Arpabet	Example	Digits
AH	bud	one
AX	*a*head	seven
AO	hawed	four
AY	hide	nine
EH	head	seven
EY	hayed	eight
ER	heard	thirty
IH	hid	six
IY	heed	three
UW	who'd	two

Arpabet	Example	Digits
W	wow	one
N	noon	nine
T	tug	two
K	kick	six
TH	thick	three
F	fife	four
S	cease	six
R	roar	four
V	verve	seven
(DX)	ba*tt*er	for*ty*

Table 18. YOHO word grammar. [ ] denotes optional monophone, and | denotes dual path through the word grammar. These grammars get used to expand a transcription into the network of subword models for forced Viterbi decoding or to provide standard Viterbi syntax for automatic speech recognition.

Word	YOHO Monophone Grammar	Source
one	W AH N [sp]	TIMIT,RM
two	T UW [sp]	TIMIT,RM
three	TH R IY [sp]	TIMIT
four	F AO R [sp]	TIMIT
five	F AY V [sp]	TIMIT
six	S IH K S [sp]	TIMIT
seven	S EH V AX N [sp]	TIMIT
nine	N AY N [sp]	TIMIT
twenty	T W EH N T IY [sp]	TIMIT
	T W EH N IY [sp]	RM
	T W EH N [T] IY [sp]	OPTION
thirty	TH ER T IY [sp]	TIMIT
	TH ER DX IY [sp]	RM
	TH ER DX |T IY [sp]	OPTION
forty	F AO R T IY [sp]	TIMIT
	F AO R DX IY [sp]	RM
	F AO R DX |T IY [sp]	OPTION
fifty	F IH F T IY [sp]	TIMIT,RM
sixty	S IH K X T IY [sp]	TIMIT,RM
seventy	S EH V AX N [T] IY [sp]	TIMIT
	S EH V AX N T IY [sp]	RM
	S EH V AX N [T] IY [sp]	OPTION
eighty	EY T IY [sp]	TIMIT
	EY DX IY [sp]	RM
	EY DX|T IY [sp]	OPTION
ninety	N AY N T IY [sp]	TIMIT
	N AY N IY [sp]	RM
	N AY N [T] IY [sp]	OPTION

Appendix C. Penalty Functions for Order Identification

For autoregressive models, both unidimensional and vector processes, several penalty function methods exist for determining proper model order. Several of these base their optimality criterion on some combination of the error variance and free parameters, yet are derived from the Kullback-Liebler distance between a model PDF and the true PDF of the data. Since it can be shown that error variance will monotonically decrease with model order, a penalty term is added to prohibit excessively large order models. This is the concept behind *parsimonious* models - ones with as few parameters as possible. Several methods include the Akaike Information Criterion (AIC), the Final Prediction Error (FPE), Parzen's Criterion of AR Transfer (CAT) function and the Bayesian Information Criterion (BIC) [20, 21, 63, 94, 54]. The AIC is often chosen for small data samples and both the AIC and FPE converge to the same solution as the number of samples increase [63]. The extension of these penalty functions has not been explored for Markov models, yet will be needed for further investigations of their usefulness. The Akaike Information Criterion is defined as

$$\mathrm{AIC}(p) = T \log \sigma_p^2 + 2p$$

where p is the model order, σ_p^2 is the MLE of the noise variance and T is the total number of observations. This penalty function has been extended to multidimensional vector autoregressive processes [21] and its properties continually evaluated [21, 126]. For a d-dimensional vector process the AIC is further defined as

$$\mathrm{AIC}(p) = T \log |\Sigma_p| + 2d^2p$$

where similarly, the Σ_p is the MLE covariance for a Vector AR(p) process. Given an N-state hidden filter Markov model, these penalty functions could be extended to sum the AIC associated with each state. This would imply the following functional forms:

- Full Predictor, Full Covariance: $\mathrm{AIC}(p, N) = \sum_i^N T \log |\Sigma_p(i)| + 2d^2p$
- Diag Predictor, Full Covariance: $\mathrm{AIC}(p, N) = \sum_i^N T \log |\Sigma_p(i)| + 2dp$
- Full Predictor, Diag Covariance: $\mathrm{AIC}(p, N) = \sum_i^N T \log trace(\Sigma_p(i)) + 2d^2p$

- Diag Predictor, Diag Covariance: $\text{AIC}(p, N) = \sum_i^N T \log trace(\Sigma_p(i)) + 2dp$.

For the simple case of single state hidden filter phoneme models, the original versions apply directly, demonstrated in Figure 26.

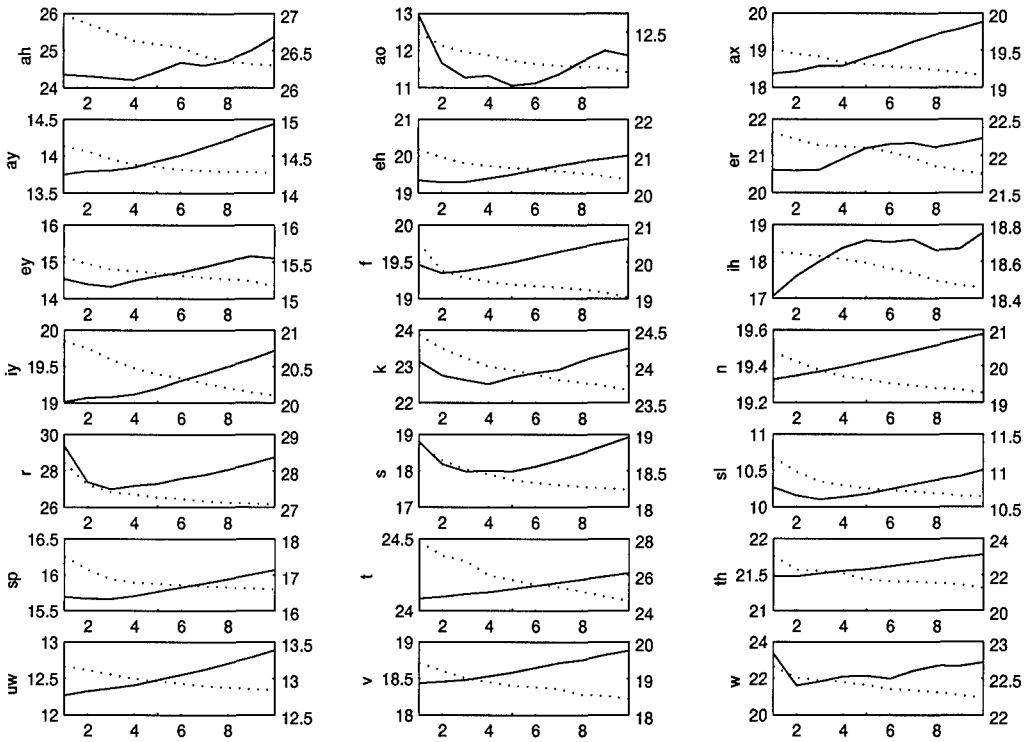


Figure 26. Akaike Information Criterion (AIC) for Vector Phoneme Models By Examining the Diagonal Covariance at Several Order Models. The subplots show each of the 21 monophone models' MLE of the average noise variance (dashed line). Note this MLE decreases with increasing model order and the AIC (solid line) acts accordingly, decreasing to a minimum, then increasing.

This presentation of model order selection for hidden filters is the first known treatment using a statistical penalty function methodology. Though the interactions between optimal model order based on residual variance and model order for best recognition is unclear, this appendix does suggest that different phonemes should have varying order models.

Appendix D. Vector AR Modeling of Strictly Stationary Speech

While other attempts to model spectral dynamics have not been overly successful for speech recognition [16, 66], the applicability to speaker recognition has not been examined. It is intuitive that the correlations and context of the changing phoneme vectors contains a source of untapped speaker dependent information. However, if certain standard assumptions are made of the speech signal within a phoneme, then the following two propositions explain the (negative) results of past researchers using conditional models.

Proposition D.1 *Speech cepstral coefficients within a phone are a Strict Sense Stationary (SSS) vector process.*

Speech is considered quasi-stationary, assumed stationary over 30-70 msec. This time relates to between 3 and 7 frames of data, using typical speech framing techniques. Strict sense stationarity of the speech samples, x_t, implies that frames of speech samples are also a SSS vector process, X_t, - easily shown since different frames have the same n-th order density. The SSS characteristic of the X_t process is maintained even after subjected to linear transforms, $\mathcal{L}$, (e.g. Fourier) and memoryless systems transformations (like square-law, mel and log). Thus, the cepstral vector process, $\bar{c}_t = \mathcal{L}[\log(\mathcal{L}[X_t]^2)]$ is strict sense stationary.

If a vector autoregressive model (for each hidden state) is used, then the following proposition results. Let the state dynamics of the cepstral vectors be defined by

$$\bar{c}_t = \bar{\mu} - \sum_{i=1}^{P} A_i \bar{c}_{t-i} + \bar{E}_t$$

with the noise ($\bar{E}_t$) being a white, normal vector process.

Proposition D.2 *During periods of stationarity (within a phone), the reestimation of a P-th order vector autoregressive model, given observations of cepstral coefficients, results in non-unique solutions. Two trivial solutions possible are:*

1. *(Trivial filter)* $\bar{\mu} = \bar{0}$ *and* $A_1 = -I, \quad A_i = \bar{0}, i = 2, \ldots P$
2. *(HMM)* $\bar{\mu} = \bar{\mu}^*$ *and* $A_i = 0, \quad i = 1, \ldots P$

For the i-th state and based on the assumed model, the state conditional density can be specified,

$$b_i(\bar{c}_t) = \frac{1}{(2\pi)^{d/2}|\Sigma_i|^{1/2}} \exp\left\{-\frac{1}{2}(\bar{c}_t - \bar{\mu} + \sum_{i=1}^{p} A_i \bar{c}_{t-i})^T \Sigma_i^{-1} (\bar{c}_t - \bar{\mu}_i + \sum_{i=1}^{p} A_i \bar{c}_{t-i})\right\}$$

and then,

$$E_t[\bar{c}_t | \bar{c}_{t-1}, \bar{c}_{t-2}, \ldots, \bar{c}_{t-p}] = \bar{\mu} + \sum_{i=1}^{p} A_i \bar{c}_{t-i}. \tag{62}$$

The stationary characteristic of the cepstral vector process implies the unconditional expectation, $E[\bar{c}(t)]$ is constant, denoted by $\bar{\mu}^*$. Now, given a sequence of a particular stationary phoneme process, the expectation of Equation 62 over all past observations, is

$$\begin{aligned} E\left[E_t[\bar{c}_t, |\bar{c}_{t-1}, \bar{c}_{t-2}, \ldots, \bar{c}_{t-p}]\right] &= E[\bar{\mu}] + \sum_{i=1}^{p} A_i E[\bar{c}_{t-i}] \\ &= \bar{\mu} + \sum_{j=1}^{p} A_i \bar{\mu}^* = \bar{\mu}^*. \end{aligned} \tag{63}$$

Hence,

$$\bar{\mu} = \bar{\mu}^* - \sum_{j=1}^{p} A_i \bar{\mu}^*$$

and when $A_i = 0$, then $\bar{\mu} = \bar{\mu}^*$ which is the standard Gaussian hidden Markov model. The reestimation will attempt to model this behavior for each state. In doing so, the two trivial solutions in the proposition are easily seen to be true based on Equation 63.

Appendix E. Syntactic Explanation for Forced Viterbi

The ability to model a sequence of symbols has been shown to reduce entropy [42] and guarantee a reduction in probability of error [73]. Statistical and syntactic pattern recognition provides a foundation for classifying targets which have an inherent stochastic grammar, $\mathcal{G}$. This grammar induces a set of possible observation sequences called a stochastic language, $\mathcal{L}(\mathcal{G})$ [125]. When given a set of hidden Markov models, a hierarchy of constraints can be placed on the Viterbi decoding process, in effect changing the grammar. The grammar, in turn, changes the size of the language. For speaker recognition, best results occur when a forced Viterbi decoding is used over alternative methods such as word grammar, word-pair grammar or simple phoneme decoding. This section provides a mathematical explanation.

For example, consider the following four grammars, each a language level constraint on the Viterbi decoding process. The first are two methods using phoneme based grammars.

- $\mathcal{L}_{FV}$, Constrain all phonemes to a transcription(Forced Viterbi)
- $\mathcal{L}_{NG}$, No constraints on phonemes (NoGrammar)

These next set are based on word models. First, a dictionary is created such that words are defined by a fixed sequence of phonemes, with optional silence.

- $\mathcal{L}_{WP}$, Constrained phonemes within words and constrained word pairs (WordPair)
- $\mathcal{L}_{WG}$, Constrained phonemes within words (WordGrammar)

For the statistical approach with several monophone models, let Λ represent the overall speaker model. Since, recognition scores change several orders of magnitude based on the word sequence along, explicitly show this variable into the Viterbi score. Denote the word sequence by W. Viterbi provides the joint likelihood score of the observation and the maximum likelihood word,

$$\begin{aligned} p(O_1 \ldots O_T|\Lambda_c, \mathcal{L}, W) &= \max_W p(O_1 \ldots O_T, W|\Lambda_c, \mathcal{L}) \\ &= \max_W p(O_1 \ldots O_T|W, \Lambda_c, \mathcal{L}) p(W|\Lambda_c, \mathcal{L}) \end{aligned} \tag{64}$$

In order to compare speakers using Viterbi decoding, this second term, must be the same. When using forced Viterbi decoding, the set of models is fixed and the size of the language $|\mathcal{L}| = 1$. However, any other grammatical approach will incur a different multiplicative expression based on the complexity of the grammar. Any other method such as phoneme decoding, word grammar or word-pair grammars, which subsequently induces a larger language $\mathcal{L}$, will be comparing two speakers on potentially different word and phoneme sequences.

We demonstrate that as the language increases by choice of grammar, the entropy (bits/phn, Equation 65) increases and this results in increased equal error rate, shown in Table 19 and Figure 27. Figure 27 further demonstrates the relationship by plotting entropy (dashed) against EER for male and female speakers separately. For this demonstration, cohorts were not used specifically to examine the overlap between true claimant and impostor scores without any normalization. Recall equal error rate occurs when the false acceptance error rate (impostor errors) equals the false rejection error rate (true claimant errors). Using Levinson's definition of entropy [72],

$$H(\mathcal{G}) = \frac{\log_2 |\mathcal{L}|}{E[n]} \tag{65}$$

which uses the size of the language $|\mathcal{L}|$ and the average number of words per utterance converted to bits per phoneme. Table 19 shows the size of the language with entropy for the various grammars. In the table, m is the number of phoneme/ word choices at each time. The $E[n]$ is the expected number of phonemes/ words during an utterance and $|\mathcal{L}|$ denote how many possible paths exist through the grammar. All quantities have been converted to phoneme units for calculation of entropy.

In summary, by changing the grammar or syntax allowed by Viterbi, different size stochastic languages are created. For automatic speech recognition, these language constraints insure the recognition fits semantically acceptable speech. However, these larger languages also allow impostors to find better paths through the language, which may not fit the semantics of the transcriptions. By using likelihoods of observations, we must insure

Table 19. YOHO Language Constraints, where the language allows an average of $E[n]$ symbols from a set of m. $|\mathcal{L}|$ denote how many possible paths exist through the grammar. All entries for entropy were converted to bits per phoneme using an average of 2.9 phonemes per word.

Grammar $\mathcal{G}$	m	$E[n]$	Language $\|\mathcal{L}\|$	$H(\mathcal{G})$
Transcription	1	1	1	0.0
WordPair	57	3	1.85e+5	0.60
Word	16	6	1.67e+7	0.83
NoGrammar	21	29	5.5e+38	3.5

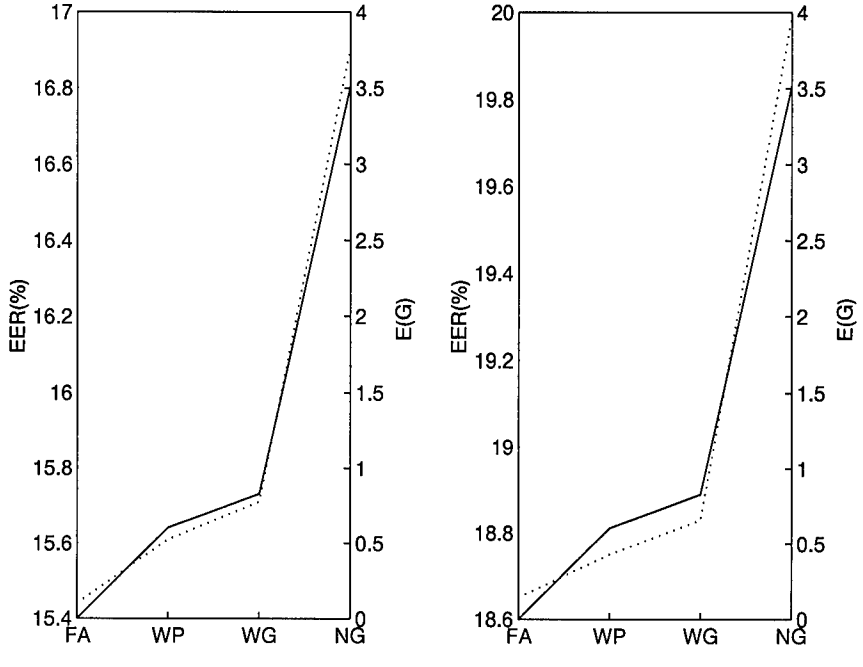


Figure 27. Equal error rates (Dashed) for males (left) and females (right) over the YOHO database with one combination per test trial. Also shown is Entropy (Solid) in bits/phoneme of the language induced by the grammar.

that the conditioning of word and phoneme sequences is identical for all Viterbi scores used in recognition.

Appendix F. Language Hypothesis

In this appendix, a hypothesis concerning ergodic hidden Markov model use for speaker recognition is proposed and demonstrated experimentally. Experimental results of Poritz [92, 113, 115, 124] and added interpretations by Levinson [73] also support this assumption. In detailing methods of speech recognition, Levinson interprets the experiments of Poritz as representing the structure found in the symbols of the language. He further substantiates this interpretation by reference to 1) English text modeling using an ergodic HMM framework by Cave and Neuwirth, and 2) originally by Markov, himself, for analyzing printed Russian text.

Proposition F.1 *An ergodic hidden Markov model λ trained with unlabeled speech to model a speaker will represent language model statistics in the Markov state transition matrix.*

Unless a predefined transition structure is provided, the state densities will model speaker dependent spectra, but the transitions between these spectra will be language dependent as evidenced by Levinson and references within. To demonstrate this experimentally, compare the steady state probabilities of a trained ergodic hidden Markov model to the statistics of the broad class transcriptions.

Beginning with phoneme labels (Table 20), transform the automatic phoneme segmentation to broad class labels and estimate the bi-class probabilities. Using unlabeled

Table 20. Broad Class/ Phoneme Relation

Broad Class	Phoneme
Vowel (V)	IY IH EH AX AH UX UH AO EY AY
Liquid/Glide (L)	R W ER
Nasal (N)	N
Consonant (C)	(DX) T K V F TH S
Silence (S)	sp sil

data, the Baum-Welch algorithm reestimated the parameters of two ergodic, five-state hidden Markov models. These systems included a HMM based on Mel frequency cepstral

features (with regression) and a third order hidden filter Markov model (similar to Poritz). The resulting transition matrices were extracted and stationarity, $p(q_t = i)$, probabilities were analyzed (Table 21).

Table 21. Steady State Language Statistics.

Vowel	Liquid-Glide	Nasal	Consonant	Silence
.19	.05	.10	.19	.47

The steady state probabilities from the five-state ergodic HMMs, using both Poritz and Mel frequency cepstral, can then be compared in Table 22.

Table 22. Learning the Language with Ergodic Models

	V	L	N	C	S
Ergodic Poritz Method (5 state)	.19	.01	.18	.26	.36
Ergodic Gaussian HMM (5 state)	.19	.01	.14	.25	.41

While the automatic phoneme transcriptions will not be precise, the similarity of the broad language statistics of the data to the learned model stationary state statistics is remarkable. Based on Poritz initial experiments, Levinson's clarification of these results with historical ergodic interpretations and these YOHO results, the explanation of past "failures" in useful transition modeling is complete. Recall the reference to Nolan [85] in Chapter I, who overviews several researchers claiming that in addition to the vocal anatomy, voice differences are the result of neural patterns and habits. These manifest themselves in the acoustic signal through coarticulation effects and formant dynamics. Effective speaker recognition strategies should monopolize on these dynamics.

Bibliography

1. Timothy R. Anderson. Speaker independent phoneme recognition with an auditory model and a neural network: A comparison with traditional techniques. In *Proc. of the 1991 ICASSP*, pages 149–152, 1991.

2. Timothy R. Anderson. A comparison of auditory models for speaker independent phoneme recognition. In *Proc. of the 1993 ICASSP*, 1993.

3. Timothy R. Anderson. Armstrong Laboratories, Personal interviews. Dayton, OH, 1994-1995.

4. B. S. Atal. Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification. *J. Acoust. Soc. Amer.*, pages 1304–12, June 1974.

5. B. S. Atal and Suzanne L. Hanaufer. Speech analysis and synthesis by linear prediction of the speech wave. *J. Acoust. Soc. Amer.*, 50(2):637–655, April 1971.

6. Bishnu S. Atal. Automatic recognition of speakers from their voices. *Proc. of the IEEE*, 64(4):460–75, April 1976.

7. L. E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Annals of Mathematical Statistics*, 41(1):164–171, 1970.

8. Leonard E. Baum. An inequality and associated maximization technique is statistical estimation for probabilistic functions of Markov process. *Inequalities*, 3:1–8, 1972.

9. Leonard E. Baum and J. A. Egon. An inequality with applications to statistical estimation for probabilistic function of a Markov process and to a model for ecology. *Bulletin of the Americal Meteorological Society*, 73:360–363, 1967.

10. Leonard E. Baum and Ted Petrie. Statistical inference for probabilistic functions of finite state Markov chains. *Annals of Mathematical Statistics*, 37:1554–1563, 1966.

11. S. W. Beet et al. Improved speech recognition using a reduced auditory representation. In *Proc. of the 1988 ICASSP*, pages 75–78, New York, 1988. IEEE Press.

12. Richard E. Blahut. *Principles and Practice of Information Theory.* Addison-Wesley, Reading MA, 1987.

13. Hervé Bourlard and Christian J. Wellekens. Links between Markov models and multilayer preceptrons. *IEEE Trans. on Pattern and Machine Intelligence*, 12(12):1167–1177, December 1990.

14. Hervé A. Bourlard and Nelson Morgan. *Connectionist Speech Recognition: A Hybrid Approach.* Kluwer Academic Publishers, Boston MA., 1994.

15. John S. Bridle. Alpha-nets: A recurrent neural network architecture with a hidden Markov model interpretation. *IEEE Trans on Neural Networks*, 9(1):83–92, February 1990.

16. Peter F. Brown. *The Acoustic-Modeling Problem in Automatic Speech Recognition.* PhD thesis, Dept of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, May 1987. CMU-CS-87-125.

17. Joseph P. Campbell, Jr. Testing with the YOHO CD-ROM voice verification corpus. In *Proc. of the 1995 ICASSP*, pages 541–545, 1995.

18. Joseph Paul Campbell, Jr. *Features and Measures for Speaker Recognition.* PhD thesis, Oklahoma State University, December 1992.

19. ChiWei Che and Qiquang Lin. Speaker recognition using HMM with experiments on the YOHO database. Submitted to the 1996 ICASSP, 1995.

20. Byoung Seon Choi. *ARMA model Identification.* Springer-Verlag, New York, NY, 1992.

21. ByoungSeon Choi. On the covariance matrix estimators of white noise process of a vector autoregressive model. *Commun. Statist.-Theory Meth.*, 23(1):249–256, 1994.

22. J. Colombi, T. Anderson, S. Rogers, D. Ruck, and G. Warhola. Auditory model representation for speaker recognition. In *Proc. of the 1993 ICASSP*, volume II, pages 700–703, 1993.

23. John M. Colombi. Cepstral and auditory model features for speaker recognition. Master's thesis, Air Force Institute of Technology, December 1992. AFIT/GE/ENG/92D-11.

24. Jerome T. Connor, R. Douglas Martin, and L.E. Atlas. Recurrent nerual networks and robust time series prediction. *IEEE Trans on Neural Networks*, 5(2):240–253, March 1994.

25. Steven B. Davis and Paul Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Trans. ASSP*, 28(4):357–366, August 1980.

26. Johan de Veth and Hervé Bourlard. Comparison of hidden Markov model techniques for automatic speaker verification. In *Proc. of the 1994 ECSA Workshop on Speaker Recog. Ident. and Ver.*, 1994.

27. John R. Jr. Deller, John G. Proakis, and John H.L. Hansen. *Discrete-Time Processing of Speech Signals.* Macmillan Publishing Co., Englewood Cliffs, NJ, 1993.

28. A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. Royal Stat. Society*, 39(1):1–38, 1977.

29. Li Deng. A generalized hidden Markov model with state-conditioned trend functions of time for the speech signal. *Signal Processing*, 27:65–78, 1992.

30. Li Deng and C. Daniel Geisler. A composite auditory model for processing speech sounds. *J. Acoust. Soc. Amer.*, 82(6):2001–2012, 1987.

31. Subhrakanti Dey, Vikram Krishnamurthy, and Thierry Salmon-Legagneur. Estimation of Markov-modulated time-series via EM algorithm. *IEEE Signal Processing Letters*, 1(10):153–155, October 1994.

32. V Digalakis, J. R. Rohlicek, and M. Ostendorf. ML estimation of a stochastic linear system with the EM algorithm and its application to speech recognition. *IEEE Trans. Speech Audio Proc.*, 4(1):431–442, October 1993.

33. Vassilios V. Digalakis. *Segment-Based Stochastic Models of Spectral Dynamics for Continuous Speech Recognition.* PhD thesis, Boston University, 1992.

34. George R. Doddington. Speaker recognition - identifying people by their voices. *Proceedings of the IEEE*, 73(11):1651–1664, November 1985.

35. D Duda and S. Hart. *Pattern Analysis and Scene Classification.* Academic Press, Inc., San Diego, CA, 1990.

36. Electronic benefits transfer: Use of biometrics to deter fraud in the nationwide EBT program. Letter Report, GAO/OSI-95-20, September 1995.

37. El-Jaroudi and Makhoul J. Discrete pole-zero modeling. In *Proc. of the 1989 ICASSP*, pages 2162–2166, 1989.

38. Nikos Fakotakis, Anastasios Tsopanoglou, and George Kookinakis. A text-independent speaker recognition system based on vowel spotting. *Speech Communications*, 12:57–68, 1993.

39. Falaschi et al. Ergodic hidden control neural networks. In *Proc. of the 1993 ICASSP*, pages I–605–I–608, 1993.

40. Laurie Fenstermacher. Multi-sensor fusion techniques for tactical speaker recognition. *SPIE: Applications and Science of Artificial Neural Networks*, 2492:717–729, April 1995.

41. Laurie Fenstermacher and Douglas Smith. Tactical speaker recognition using feature and classifier fusion. *SPIE: Applications of Artificial Neural Networks*, 2243:34–41, April 1994.

42. Kenneth Henry Fielding. *Spatio-temporal Pattern Recognition using Hidden Markov Models.* PhD Dissertation, Air Force Institute of Technology, June 1994. AFIT/DS/ENG/94J-02.

43. James L. Flanagan. *Speech Analysis, Synthesis and Perception.* Springer-Verlag, New York, 1972.

44. James L. Flanagan and Lawrence R. Rabiner, editors. *Speech Synthesis.* Dowden, Hutchington and Ross, Inc., Stroudsburg, Penn, 1973.

45. Andrew M. Frazer and Alexis Dimitriadis. Hidden Markov models with mixed states. In A. Weigend and N. Gershenfeld, editors, *Predicting the Future and Understanding the Past*, Santa Fe Institute, Studies in Sciences of Complexity. Addison-Wesley Publishing., Redwood City, CA., 1993.

46. Keinosuke Fukunaga. *Introduction to Statistical Pattern Recognition.* Academic Press, New York, second edition, 1990.

47. Sadaoki Furui. Cepstral analysis techniques for automatic speaker verification. *IEEE Trans. ASSP*, 29(2):254–272, April 1981.

48. Jr. G. David Forney. The viterbi algorithm. *Proc. of the IEEE*, 61(3):268–277, March 1973.

49. Paul C. Giannelli. Daubert: Interpreting the federal rules of evidence. *Cardozo Law Review*, 15(6/7):1999–2026, 1994.

50. Sharon E. Gregory. Voice spectrography evidence: Approaches to admissibility. *University of Richmond Law Review*, 20:357–376, Winter 1986.

51. A. Higgins, L. Bahler, and J. Porter. Speaker verification using randomized phrase prompting. *Digital Signal Processing*, 1:89–106, 1991.

52. Paul G. Hoel. *Introduction to Mathematical Statistics.* John Wiley & Sons, Inc., New York, NY, forth edition, 1971.

53. Melvyn J. Hunt and Claude Lefebvre. Speaker dependent and independent speech recognition experiments with an auditory model. In *Proc. of the 1988 ICASSP*, pages 215–218, New York, 1988.

54. Clifford M. Hurvich and Chih-Ling Tsai. A corrected Akaike information criterion for vector autoregressive model selection. *Journal of Time Series*, 14(3):271–279, 1993.

55. Fumitada Itakura. Minimum prediction residual principle applied to speech recognition. *IEEE Trans. ASSP*, 23(1):67–72, February 1975.

56. Charles Jankowski, Ashok Kalyanswamy, Sara Basson, and Judith Spitz. TIMIT: A phonetically balanced, continuous speech, telephone bandwidth speech database. In *Proceedings of the IEEE ICASSP*, pages 109–112, New York, 1990.

57. B. H.. Juang. On the hidden Makrov model and dynamic time warping for speech recognition - A unified view. *AT&T Bell Labs Tech Journal*, 63(7):12131243, September 1984.

58. B. H. Juang. Maximum-likelihood estimation for mixture multivariate stochastic observations of Markov chains. *AT&T Technical Journal*, 64(6):1235–1249, July-August 1985.

59. B. H. Juang and L. R. Rabiner. A probabilistic distance measure for hidden Markov models. *AT&T Technical Journal*, 64(2):391–408, February 1985.

60. Biing-Hwang Juang and Lawrence R. Rabiner. Mixture autoregressive hidden Markov models for speech signals. *IEEE Trans. on Acoustics Speech and Signal Processing*, ASSP-33(6):1404–1412, December 1985.

61. Yu-Hung Kao, P. K. Rajasekaran, and John S. Baras. Free-text identification over long distance telephone channel using hypothesized phonetic segmentation. In *Proc. of the 1992 ICASSP*, volume 2, pages 177–180, 1992.

62. Yu-Hung Kao, P. K. Rajasekaran, and John S. Baras. Robust free-text speaker identification over long distance telephone channels. Proc. of the 1993 ICASSP, 1993.

63. Steven M. Kay. *Modern Spectral Estimation: Theory and Application.* Prentice-Hall Signal Processing Series. Prentice-Hall, Englewook Cliffs, NJ, 1988.

64. Athanasios Kehagias. Stochastic recurrent networks: Prediction and classification of time series. Tech report, Division of Applied Mathematics, Brown University, Providence, RI, 1991.

65. Athanasios Kehagias. Stochastic recurrent networks training by the local backward-forward algorithm. Tech report, Division of Applied Mathematics, Brown University, Providence, RI, 1991.

66. Patrick Kenny, Matthew Lennig, and Paul Mermelstein. A linear predictive HMM for vector-valued observations with applications to speech recognition. *IEEE Trans. on Acoustics Speech and Signal Processing*, 38(2):220–225, February 1990.

67. Linquistic Data Consortium (LDC). YOHO speaker verification database. CD-ROM, 1994.

68. Francois Le Chevalier, Gerard Bobillot, and Cecile Fugier-Garrel. Radar target and aspect angle identification. In *Proceedings of the IEEE 1978 International Conference on Pattern Recognition*, pages 398–400, 1978.

69. Kai-Fu Lee. *Automatic Speech Recognition: The Development of the SPHINX System.* Kluwer Academic Publishers, Norwell, MA, 1989.

70. Esther Levin. Word recognition using hidden control neural networks. In *Proc. of the 1990 ICASSP*, pages 433–436, 1990.

71. Esther Levin. Hidden control neural architecture modeling of nonlinear time varying systems and its application. *I.E.E.E. Trans Neural Networks*, 4(1):109–116, January 1993.

72. S. E. Levinson, L. R. Rabiner, and M. M. Sondhi. An introduction to the applications of the theory of probabilistic functions of a Markov process to automatic speech recogniton. *The Bell System Technical Journal*, 62(4):1035–1074, 1983.

73. Stephen E. Levinson. Structual methods in automatic speech recognition. *Proc. of the IEEE*, 73(11):1625–1641, November 1985.

74. Edumnd W. Libby. *Application of Sequence Comparison Methods to Multisensor Data Fusion and Target Recognition.* Ph.D. Dissertation, Air Force Institute of Technology, Wright-Patterson AFB, OH 45433, July 1993.

75. Yoseph Linde, Andrés Buzo, and Robert M. Gray. An algorithm for vector quantizer design. *IEEE Trans. on Comm.*, COM-28(1):84–94, January 1980.

76. Louis A. Liporace. Maximum likelihood estimation for multivariate observations of Markov sources. *I.E.E.E. Trans on Info Theory*, IT-28(5):729–734, 1982.

77. Liu et al. Study of line spectrum pair frequencies for speaker recognition. In *Proc. of the 1990 ICASSP*, volume 1, pages 277–280, 1990.

78. John Makhoul. Linear prediction: A tutorial review. *Proc. of the IEEE*, 63(4):561–580, April 1975.

79. Tomoko Matsui and Sadaoki Furui. Comparison of text-independent speaker recognition methods using VQ distortion and discrete/continuous HMMs. In *Proc. of the 1992 ICASSP*, volume 2, pages 157–160, 1992.

80. Tomoko Matsui and Sadaoki Furui. Similarity normalization method for speaker verification based on a posteriori probability. In *Proc. of the 1994 ECSA Workshop on Speaker Recog. Ident. and Ver.*, 1994.

81. Neri Merhav and Yariv Ephraim. Hidden Markov modeling using the most likely state sequence. In *Proc. of the 1991 ICASSP*, pages 469–472, 1991.

82. Aage R. Moller. *Auditory Physiology.* Academic Press, New York, 1983.

83. Brian C. J. Moore. *An Introduction to the Psychology of Hearing.* Academic Press, New York, third edition, 1989.

84. NIST. *The DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus (TIMIT): Training and Test Data and Speech Header Software*, October 1990.

85. Francis Nolan. *The Phonetic Bases of Speaker Recognition.* Cambridge University Press, New York, NY, 1983.

86. National Institute of Standards and Technology (NIST). Guideline for the use of advanced authentication technology alternatives. Federal Information Processing Standards Publication 190, September 1994.

87. Joseph P. Olive, Alice Greenwood, and John Coleman. *Acoustics of American English Speech: A Dynamic Approach.* Springer-Verlag, Yew York, NY, 1993.

88. Alan V. Oppenheim and Ronald W. Schafer. *Discrete-Time Signal Processing.* Prentice Hall, Englewood Cliffs, New Jersey, 1989.

89. A. Papoulis. *Probability, Random Variables, and Stochastic Processes.* McGraw-Hill, Inc., New York N.Y., 3rd edition, 1991.

90. Thomas W. Parsons. *Voice and Speech Processing.* McGraw-Hill, New York, 1987.

91. Stephen V. Pellissier. Text-independent, open-set speaker recognition. Master's thesis, Air Force Institute of Technology, Wright-Patterson AFB OH, March 1996.

92. Alan B. Poritz. Linear prediction of hidden Markov models. In *Proc. of the 1982 ICASSP*, pages 1291–1294, 1982.

93. Alan B. Poritz. Hidden Markov models: A guided tour. In *Proc. of the 1988 ICASSP*, pages 7–13, 1988.

94. M. B. Priestley. *Non-linear and Non-stationary Time Series Analysis.* Academic Press, Inc., San Diego, CA, 1988.

95. L. Rabiner and B.-H. Juang. *Fundamentals of Speech Recognition.* Englewood Cliffs NJ: PTR Prentice Hall (Signal Processing Series), 1993. General Intro : ISBN 0-13-015157-2.

96. L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. of the IEEE*, 77(2), February 1989.

97. L. R. Rabiner and B. H. Juang. An introduction to hidden Markov models. *IEEE ASSP Magazine*, January 1986.

98. Lawrence Rabiner and Biing-Hwang Juang. *Fundamentals of Speech Recognition.* PTR Prentice-Hall inc, Englewood Cliffs, NJ, 1993.

99. Lawrence Rabiner and Ronald Schafer. *Digital Signal Processing of Speech Signals.* Prentice Hall, Inc., Englewood CLiffs, NJ, 1978.

100. Lawrence R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, February 1989.

101. D. A. Reynolds and R. C. Rose. An integrated speech-background model for robust speaker identification. In *Proc. of the 1992 ICASSP*, volume 2, pages 185–188, 1992.

102. Douglas A. Reynolds. *A Gaussian Mixture Modeling Approach to Text- Independent Speaker Identification.* PhD thesis, Georgia Institute of Technology, August 1992.

103. Douglas A. Reynolds. Large population speaker identification using clean and telephone speech. *IEEE Signal Processing Letters*, 2(3):46–48, March 1995.

104. Douglas A. Reynolds. Speaker identification and verification using Gaussian mixture speaker models. *Speech Communication*, 17(1-2):91–108, 1995.

105. Anthony J. Robinson. An application of recurrent nets to phone probability estimation. *IEEE Trans on Neural Networks*, 5(2):298–304, March 1994.

106. Anthony John Robinson. *Dynamic Error Propagation Networks.* PhD thesis, Cambridge University, February 1989.

107. Tony Robinson and Frank Fallside. Phoneme recognition from the TIMIT database using recurrent error probagation networks. Technical Report CUED/F-INFENG/TR-42, Cambridge University Engineering Dept., March 1990.

108. Tony Robinson and Frank Fallside. A recurrent error propagation network speech recognition system. *Computer Speech and Language*, 5(3), July 1991.

109. Steven K. Rogers. Personal interviews. AFIT, WPAFB OH, 1992-1995.

110. Steven K. Rogers, John M. Colombia, Curtis E. Martin, James C. Gainey, Ken H. Fielding, Tom J. Burns, Dennis W. Ruck, Matthew Kabrisky, and Mark Oxley. Neural networks for automatic target recognition. *Neural Networks*, 8(7/8):1153–1184, 1995.

111. Dennis W. Ruck. *Characterization of Multilayer Perceptrons and their Application to Mulitsensor Target Detection.* PhD thesis, AFIT, December 1990.

112. Dennis W. Ruck. Class Notes, EENG 621, Summer 1992.

113. Furui Sadaoki. An overview of speaker recognition technology. In *Proc. of the 1994 ECSA Workshop on Speaker Recog. Ident. and Ver.*, 1994.

114. Marvin R. Sambur. Selection of acoustic feature for speaker identification. *IEEE Trans. on Acoustics Speech and Signal Processing*, ASSP-23(2):176–182, April 1975.

115. M. Savic and J. Sorenson. Phoneme based speaker verification. In *Proc. of the 1992 ICASSP*, volume 2, pages 165–168, 1992.

116. R. Schwartz. The application of probability density estimation to text-independent speaker identification. In *Proc. of the 1982 ICASSP*, pages 1649–1652, 1982.

117. Hugh E. Secker-Walker and Campbell L Searle. Time-domain analysis of auditory-nerve firing rates. *J. Acoust. Soc. Amer.*, 88(3):1427–1436, September 1990.

118. Man Mohan Shondi. Direct estimation of the vocal tract shape by the inverse filtering of acoustic waveforms. *IEEE Trans. Acoust. Speech. Sig. Proc*, 27(3):268–273, 1979.

119. F.K. Soong, A.E. Rosenburg, L.R. Rabiner, and B.H. Juang. A vector quantization approach to speaker recognition. In *Proc. of the 1985 ICASSP*, volume 1, pages 387–390, 1985.

120. Frank K. Soong and Aaron E. Rosenburg. On the use of instantaneous and transitional spectral information in speaker recognition. *IEEE Trans. ASSP*, 36(6):871–79, June 1988.

121. Helge B.D. Sorensen and Uwe Hartmann. Self-structuring hidden control neural networks for speech recognition. In *Proc. of the 1992 ICASSP*, pages II–353–II–357, 1992.

122. Helge B.D. Sorensen and Uwe Hartmann. Pi-sigma and hidden control based self-structuring models for text-independent speaker recogntion. In *Proc. of the 1993 ICASSP*, pages II–537–II–540, 1993.

123. G Therrien. *Discrete Random Signals and Statistical Signal Processing.* McGraw Hill, 1992.

124. Naftali Z. Tishby. On the Application of Mixture AR Hidden Markov Models to Text-Independent Speaker Recognition. *I.E.E.E. Trans. Sig. Proc.*, 39(3):563–570, March 1991.

125. Julius T. Tou and Rafael C. Gonzalez. *Pattern Recognition Principles.* Addison-Wesley Pubishing Company, 1st edition, 1974.

126. Ah Chung Tsoi and Andrew D. Back. Locally recurrent globally feedforward networks: A critical review of architectures. *IEEE Trans on Neural Networks*, 5(2):229–238, March 1994.

127. Richard M. Warren. Perceptual restoration of missing speech sounds. *Science*, 164:392–393, 1970.

128. Richard M. Warren and Roslyn P. Warren. Auditory illusions and confusions. *Scientific American*, 223:30–36, December 1970.

129. Clifford J. Weinstein. Opportunities for advanced speech processing in military computer-based systems. *Proc. of the IEEE*, 79(11):1627–39, November 1991.

130. J. Wolf. Efficient acoustic parameters for speaker recognition. *Journal of the Acoustical Society of America*, 51(6):2044–2056, 1972.

131. Philip C. Woodland. Hidden Markov models using vector linear prediction and discriminative output distributions. In *Proc. of the 1992 ICASSP*, volume I, pages 509–513, 1992.

132. Eric D. Young and Murray B. Sachs. Representation of steady-state vowels in the temporal aspects of the discharge patterns of populations of auditory nerve fibers. *J. Acoust. Soc. Amer.*, 66(5):1381–1403, November 1979.

133. S.J. Young, N.H. Russell, and J.H.S. Thornton. Token passing: a conceptual model for connected speech recognition systems. Technical report, CUED Technical Report TR38, Cambridge University, 1989. Available by anonymous ftp from svr-ftp.eng.cam.ac.uk.

134. S.J. Young, P.C. Woodland, and W.J.Byrne. *HTK:Hidden Markov Model Toolkit.* Cambridge University Engineering Department Speech Group and Entropic Research Laboratories, Inc., v1.5 edition, December 1993.

135. Xiaoyuan Zhu. Text-Independent Speaker Recognition Using VQ, Mixture Gaussian VQ and Ergodic HMMs. In *Proc. of the 1994 ECSA Workshop on Speaker Recog. Ident. and Ver.*, 1994.

Vita

Captain John M. Colombi [redacted]s. He graduated from Weymouth South High School in June 1982, and later received a U.S. Air Force ROTC scholarship while attending the University of Lowell, Lowell Massachusetts. He graduated as ROTC Distinguished Graduate with a Bachelor of Science degree in Electrical Engineering Captain Colombi came on active duty to the Communication's Directorate of formerly Rome Air Development Center, Griffiss AFB, New York where he served as a Communication Systems Engineer until April 1991. In Rome, he developed advanced communications network management algorithms and protocols. In May of 1991 he entered the School of Engineering, Air Force Institute of Technology at Wright-Patterson Air Force Base, Ohio, to pursue a Master of Science degree in Electrical Engineering. He received his MSEE as a Distinguished Graduate in December of 1992, specializing in pattern recognition and biological information processing. After completing Squadron Officer School, John returned to AFIT to pursue hid Ph.D. His doctoral research focused on statistical time series models for pattern recognition, specifically speaker verification. Captain Colombi's next assignment will be with the National Security Agency, Ft. George Meade, MD. He has active memberships with IEEE, Tau Beta Pi, Eta Kappa Nu and AFA. He is married to Cheryl Anne (Gately) Colombi of Weymouth, Massachusetts and has two children - Andrew and Felicia.

[redacted]

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	12 March 1996	Ph.D. Dissertation

4. TITLE AND SUBTITLE
Generalized Hidden Filter Markov Models Applied to Speaker Recognition

5. FUNDING NUMBERS

6. AUTHOR(S)
John M. Colombi, Captain, USAF

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Institute of Technology
WPAFB OH 45433

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT/DS/ENG/96-01

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Lt. D. Smith
RL/IRA
Griffiss AFB NY 13441

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Distribution Unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

Classification of time series has wide Air Force, DoD and commercial interest, from automatic target recognition systems on munitions to recognition of speakers in diverse environments. The ability to effectively model the temporal information contained in a sequence is of paramount importance. Toward this goal, this research develops theoretical extensions to a class of stochastic models and demonstrates their effectiveness on the problem of text-independent (language constrained) speaker recognition. Specifically within the hidden Markov model architecture, additional constraints are implemented which condition the hidden output densities on past observations. The reestimation of these techniques for samples, frame or vectors is developed, and the mathematical properties of convergence are analyzed. The system models speaker dependent phonemes and recognition is based on normalized log-likelihood forced Viterbi decoding. Both closed set identification and speaker verification using cohort normalization are performed on the YOHO database - a large scale, multiple-session, speech database containing 138 speakers. A critical error analysis using a hypothesis test provides the maximum number of errors observable while still meeting the required error rates of 1% False Reject and 0.1% False Accept. Equal errors rate down to 0.21% for males and 0.31% females are obtained and shown able to meet these requirements.

14. SUBJECT TERMS
speaker recognition, speaker verification, hidden Markov models, hidden filter Markov models, log-ratio normalization, phoneme models, critical errors

15. NUMBER OF PAGES
135

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL