A NONLINEAR MIXTURE AUTOREGRESSIVE MODEL

FOR SPEAKER VERIFICATION

By

Sundararajan Srinivasan

A Dissertation Proposal

Submitted to the Faculty of

Mississippi State University

in Partial Fulfillment of the Requirements

for the Degree of Doctor of Philosophy

in Electrical Engineering

in the Department of Electrical and Computer Engineering

Mississippi State, Mississippi

January 2010

Copyright by

Sundararajan Srinivasan

2010

TABLE OF CONTENTS

CHAPTER

# INTRODUCTION 1

# Speaker Verification Problem 1

# Speech Signal Features – The Frontend 3

# Statistical Speaker Modeling Methods 4

# Proposal Organization and Contribution 4

# NONLINEARITIES IN THE SPEECH SIGNAL – A BRIEF SURVEY 6

# Nonlinearity in Speech 6

# Drawbacks of Using Nonlinear Invariant Features in Speech Processing 8

# GAUSSIAN MIXTURE MODEL - CURRENT TECHNOLOGY 9

# Basic Definition 9

# Statistical Modeling of Speech Using GMMs 11

# Limitations of GMMs in Speech Processing 12

# MIXTURE AUTOREGRESSIVE MODEL – A NONLINEAR APPROACH 14

# Why Use Nonlinear Models for Speech? 14

# MixAR: Basic Definition and Properties 15

# Comparison of MixAR to Other Models 16

# MixAR Parameter Estimation using the EM algorithm 18

# PRELIMINARY EXPERIMENTS 21

# Two-Way Classification with Synthetic Data 21

# Two-way Classification with Speech-like Data 23

# Speaker Verification Experiments with Synthetic Data 24

# Speaker Verification Experiments with NIST 2001 Database 26

# Speaker Verification Experiments with TIMIT Database 29

# PROPOSED WORK AND EXPERIMENTS 31

# Speaker Verification Experiments with TIMIT under Noisy Conditions 31

# Speaker Verification Experiments with Variable Amounts of Training Data 33

# Other Important Issues 34

# REFERENCES 36

## 

INTRODUCTION

In this chapter, the problem of speaker verification is introduced. Speech signal features used to represent speaker information are briefly explained and some popular approaches to the speaker modeling problem found in the literature are surveyed. This sets the tone for the following chapters.

## Speaker Verification Problem

Speaker-independent speech and text-independent speaker recognition are complementary viewpoints of the same problem. Given a speech signal, the goal of speaker-independent speech recognition is to decipher the transcription underlying the utterance, irrespective of the identity of the speaker . On the other hand, text-independent speaker recognition problem attempts to find or confirm the identity of the speaker without regard to what was spoken . illustrates the basic problem of speaker recognition.



Figure . An overview of the speaker recognition problem.

When the goal is to find the identity of an unknown speaker the problem is called speaker identification. Very often in practice, the goal in speaker recognition is to accept or reject the identity claim made by a speaker. This is called speaker verification. This is widely used in a variety of applications ranging from secured access and surveillance to multimodal verification.

In a speaker verification setup, there are several registered speakers in the database, and associated with each speaker is a trained model. During the evaluation phase, a person, X, claims an identity and speaks a preselected or randomized phrase into a microphone. The objective of speaker verification system is to determine if the speech signal is sufficiently close to the stored model associated with the claimed identity.

Speaker verification, like other problems involving binary outcomes (“accept” or “reject”) is plagued by two kinds of errors: false alarms and misses. When an imposter is accepted the error is called false alarm, while a true speaker getting rejected constitutes a miss. In all speaker verification problems, a threshold value determining the operating point is set. By varying this threshold we can decrease one error at the expense of an increase in the other. A graph depicting this relationship with the false alarm probability on the x-axis and the miss probability on the y-axis is called a Detection Error Tradeoff (DET) curve [3]. One model is said to be better than another if its DET curve lies closer to the origin than that of the other. In practice, this may not happen consistently for all values on the x-axis (or equivalently, the y-axis). This makes comparison between models difficult. Moreover, it is more convenient to use a single measure of performance to compare models. For these two reasons, scalar performance measures are more commonly used when stating and comparing speaker verification performance. One measure that applies equal importance to the reduction of both kinds of errors – miss and false alarm – is the Equal Error Rate (EER) [3]. This is the point at which the line having a slope of 1 and passing through the origin intersects the DET curve. At this point, the miss probability equals the false alarm probability, and hence the name. Since this work is aimed at a generic speaker verification application, there is no reason to weigh one kind of error more than the other. Hence, EER is used as the performance measure of a model for speaker verification.

## Speech Signal Features – The Frontend

It is evident from the discussion above that speaker verification is a pattern classification problem with data from each speaker’s utterance forming a distinct class [4]. A set of features to represent the characteristics of each speaker adequately, and also succinctly, is necessary for successful speaker classification. For speech as well as speaker recognition tasks, the most popular features are the Mel-Frequency Cepstral Coefficients (MFCCs), sometimes also referred to as cepstral features . It is not important here to delve into details of the algorithm used to compute these features, but we can simply note that the MFCCs are physically motivated based on auditory perception properties of the human ear. MFCCs have been the most successful features in speech and speaker recognition applications.

## Statistical Speaker Modeling Methods

The goal of statistical modeling in speaker recognition is to accurately and efficiently represent the probability distribution of speaker features so that even similar sounding speakers can be distinguished and can be done so with as few parameters and as little computational requirement as possible.

There are several statistical models that have been proposed in the literature. Non-parametric models are one important class of such models, and have been typically implemented using variants of vector quantization (VQ) . More common though, because of their robustness, are parametric models. The most popular approach to speaker modeling is the Gaussian Mixture Modeling (GMM) . Almost all the work on speaker verification utilizes GMMs entirely or in the form of hybrids with other modeling techniques. The commonly used statistical model for speech recognition – a Hidden Markov Model (HMM) – has also been applied to speaker modeling . Several kernel-based approaches, including Support Vector Machines (SVMs), have also been applied to this problem -[10]. More recently, discriminatory modeling techniques are finding increasing application in speaker recognition [11][12]. However, the GMM still remains an important baseline model because its performance has been studied across a variety of databases. The work presented here attempts to address the drawbacks of GMM modeling by using a nonlinear mixture autoregressive model for speaker modeling.

## Proposal Organization and Contribution

The structure of this proposal is outlined below. The current chapter dealt with an introduction to the problem of speaker verification, the standard features used to represent the speech signal information, and also briefly surveyed the speaker modeling schemes found in literature.

Chapter II examines the evidence from primary literature for the presence of nonlinearities in the speech signal and its consequences to speech processing. It also addresses the problems associated with the use of nonlinear invariants as features to represent the nonlinear information in the speech signal.

Chapter III is devoted to discussing the basics of Gaussian Mixture Modeling (GMM) and its application to speaker recognition. It is argued here that GMMs are incapable of modeling nonlinear evolution information in the features and that the use of differential features is only a linear approximation to the actual nonlinear dynamics.

Chapter IV motivates the need for a nonlinear model in speech processing. The mixture autoregressive (MixAR) model is borrowed from statistical literature as a novel nonlinear statistical model in speech processing. Comparisons of the MixAR model to the GMM model and other autoregressive models in speech literature are made. The problem of parameter estimation is discussed.

Chapter VI discusses the set of experiments that were run to evaluate the performance of MixAR in relation to GMM models. After two classification experiments on synthetic data, speaker verification experiments on both synthetic and standard speech data are presented.

Chapter VI discusses proposed future work on evaluating the robustness of the MixAR model. Furthermore, it outlines the plan for evaluating effect of training and evaluation utterance duration on performance.

## 

NONLINEARITIES IN THE SPEECH SIGNAL – A BRIEF SURVEY

In this chapter, the nonlinear nature of speech signals and its consequences in speech and speaker recognition are examined. Then the drawbacks of using nonlinear dynamical invariants as features in speech processing is exposed, motivating alternative approaches to representing the nonlinear dynamic information contained in speech.

## Nonlinearity in Speech

Prior to the 1960s, speech production in the vocal tract was considered a passive linear process. Later, Teager [13] showed that nonlinear mechanisms underlie the speech production process. However, until fairly recently, most speech modeling was based on linear representation of signals – particularly Linear Prediction and its variants [14][15]. This was primarily because of the ease of dealing with linear models, and also the limited computational power available at that time.

Over the past decade or so, there has been a resurgence of interest in accounting for the nonlinear nature of speech signals. Recent work suggests that speech signals have nonlinearities that could contain relevant information in speech processing [16][24]. The majority of this type of research relies on extracting novel speech features known as nonlinear dynamic invariants, and then using these nonlinear features along with conventional features in conventional pattern recognition or machine learning systems. The goal of these approaches is to supplement nonlinear dynamic information that the conventional feature set does not possess.

Nonlinear dynamic invariants quantify the degree of nonlinearity in a signal. These do not vary with transformations of the signal as long as they are smooth and invertible. Hence, they are called invariants. The three most commonly used nonlinear invariants are Lyapunov exponents, fractal dimension, and correlation entropy [17]-[19]. Lyapunov exponents characterize the rate of divergence between nearby trajectories in the phase-space of the signal, while correlation entropy quantifies the rate of information gain or loss. Both these signify the sensitivity of the system to initial conditions. Fractal dimensions capture the geometry of self-similar systems. All three signify the presence or absence of chaos in the system dynamics and can aid in detecting presence of nonlinearity in a signal.

The most common application of nonlinear invariants in speech and speaker recognition systems is to consider them as features and concatenate them with conventional features like MFCCs. In [18] it was shown that the additional information in the nonlinear invariant features extracted from speech could be beneficial in a phone classification task. May [17] demonstrated improvements when adding invariant features with MFCCs on a continuous speech recognition task involving noise-free recording conditions, but found that the combined performance worsened when noise was present. For speaker identification, Petry, *et* *al* [25] showed an improvement in identification accuracy of about 1% relative by adding nonlinear features to cepstral features. This improvement is only marginal considering that the database was composed of only isolated digits.

## Drawbacks of Using Nonlinear Invariant Features in Speech Processing

Judging by the results of the representative examples cited above, it is clear that adding nonlinear invariants as features has not improved the robustness of speech and speaker recognition technologies in harsh or mismatched environments. This failure can be attributed to two reasons. First, it is difficult to estimate invariants reliably from speech. In addition to the extensive tuning required by the parameter estimation algorithms, there is also the problem of a requirement of large durations of the acoustic event [22]. This gravely undermines the applicability of invariant features for a short-time stationary signal like speech. Even if it was somehow possible to estimate the invariants accurately, there is the second and more fundamental problem that invariants only quantify the degree of nonlinearity and do not characterize the nature of the dynamics completely.

This lack of success of nonlinear invariants in improving robustness in speech processing does not imply that nonlinear information is not present or that it is not so useful in speech. Rather, the current evidence from nonlinear invariants provides an almost unequivocal support to the presence of nonlinear dynamics in speech, and we should explore other ways of exploiting this information to advantage. The current work attempts to do this at the modeling level.

## 

THE GAUSSIAN MIXTURE MODEL

In the introductory chapter, it was stated that the majority of speaker recognition systems utilize Gaussian Mixture Models (GMMs) either entirely or as part of a hybrid model. In this chapter, the concept of a GMM is introduced. The motivation for these models and why such models currently form the basis of current speech and speaker recognition systems is addressed. Next, the main drawbacks of using GMMs are exposed, with particular emphasis placed on its inefficiency at modeling nonlinearly evolving feature streams. Recent findings on the presence of significant nonlinearities in speech signals were summarized in Chapter 2, and provided motivation for the use of nonlinear time-series models for better speech representation – the subject of the next chapter.

## Basic Definition

A random variable ***x*** drawn from a Gaussian Mixture Model has a probability density function defined by :

|  |  |  |
| --- | --- | --- |
|  |  |  |

Here, *m* is called the order of the GMM and the Gaussian distribution with mean vector  and covariance matrix  is defined by:

|  |  |  |
| --- | --- | --- |
|  |  |  |

It is apparent from this definition that a GMM can be thought of as a linear functional decomposition using Gaussian distribution functions as basis functions. This interpretation immediately leads us to ask if any function can be represented using a GMM. For our current purposes, it is sufficient to state that almost any probability distribution arising in the real world can be approximated with a GMM to any desired degree of accuracy provided the order *m* is large enough.



Figure : An overview of the GMM approach.

Another equivalent way of representing a GMM is the following:

|  |  |  |
| --- | --- | --- |
|  |  |  |

where is a normal random variable with mean 0 and covariance , and *w.p.* denotes “with probability”. According to this interpretation of GMM, a sample is drawn from any one of the *m* possible distinct modes (or single Gaussian densities), and the probability that mode *i* is chosen is determined by its weight *Wi*.

Such multimodal distributions are extremely useful in speech processing. For example, we can decompose a population of speakers based on the fundamental frequency of their voice [x] into three unimodal Gaussians -- one to represent male, another to represent females and a third to represent children. The overall distribution is the weighted sum of the three unimodal distributions.

## Statistical Modeling of Speech Using GMMs

GMM has been the primary statistical representation for speech signals for over two decades. In speech recognition, GMMs are predominantly used within the framework of Hidden Markov Models (HMMs) to model the probabilities of state observations . The success of this model is the result of the short-term stationarity of speech signals. Each state in an HMM can be made to represent the distribution during stationary segments of speech for a given phoneme reasonably well using a GMM. The transitions between these stationary segments can be represented by state transition probabilities. In this paradigm, the individual Gaussian modes of the GMMs can be understood to represent the idiosyncratic ways in which a particular sound can be generated by one or more speakers.

In speaker recognition, it is established that a 1-state HMM, or equivalently, a GMM, would suffice . Here, each Gaussian mode in the GMM represents a different broad class of sounds produced by the speaker; for example, one Gaussian distribution can model the vowel sounds, another the fricatives, and so on. Since the same phoneme varies across a population of speakers due to a number of linguistic and physiological phenomena, it is expected that their respective GMMs will be dissimilar. This idea can be used to recognize the speakers correctly.

GMMs are widely used in speech processing systems for several reasons . First, it is somewhat straightforward to estimate the parameters of a GMM from training data. The popular iterative Expectation-Maximization approach guarantees convergence to the maximum likelihood estimate of the parameters [26][27]. In practice, this convergence is achieved fairly quickly. Both Expectation (E) and Maximization (M) steps have closed-form expressions computed every iteration. Moreover, it is sufficient to approximate full covariance matrices for each Gaussian mixture component with a diagonal covariance matrix, thus greatly decreasing the computation requirements.

## Limitations of GMMs in Speech Processing

Application of GMMs in speech processing is not without its drawbacks. GMMs can only represent static distributions and hence cannot represent a random process that is evolving over time. In applications where we model MFCC feature streams representing speech data, the dynamic information in their time-evolution is lost. The most commonly used strategy to circumvent this limitation is to append velocity (first time-derivative) and acceleration (second time-derivative) coefficients of MFCCs to the absolute or static MFCCs [1].

However, this has two main drawbacks. The first drawback involves redundancy – there is obviously statistical dependence between absolute, static, and acceleration coefficients, but building GMMs over the complete concatenated vector does not take this redundancy into account. Hence, we tend to use more parameters than might be necessary. The second more serious drawback, which is the focus of this dissertation, is the implicit assumption of linearity in the MFCC dynamics. The derivatives of the cepstral features are only a linear approximation of the actual dynamics of the static features. However, as we saw in Chapter II, the speech signal contains significant nonlinear information, and using only derivative features to represent speech MFCC dynamics with GMM modeling is tantamount to discarding any nonlinear information present in the signal.

An obvious fix to this problem is to add features that can represent the nonlinear dynamic information. But as was seen in the previous chapter, this approach is fraught with difficulties. The primary goal of this dissertation is to approach the information representation problem at the modeling level, thereby accounting for the nonlinear dynamics of speech in the base model and minimizing the dimensionality of the feature space.

## 

THE MIXTURE AUTOREGRESSIVE MODEL – A NONLINEAR APROACH

In this chapter, a nonlinear model called the mixture autoregressive model (MixAR) is introduced. First, the basic definition and a few relevant properties of the model are stated. Connections and comparisons of MixAR are made to GMMs as well as to other autoregressive models found in the speech literature. The problem of parameter estimation is discussed in a framework of maximum likelihood estimation using the popular Expectation-Maximization (EM) approach.

## Why Use Nonlinear Models for Speech?



Figure : An overview of the MixAR approach.

It is evident from discussions in Chapter II that there are significant nonlinearities in the speech signal, that including nonlinear information can improve robustness of speech systems, and that nonlinear dynamic invariants are ineffective at capturing this information for short-term stationary speech signals. It then follows that we should explore capturing nonlinear information at the modeling level. In Chapter III we explained that the popular GMM approach can at best model linear dynamics using static and differential features. This motivates a search for a model that can capture the nonlinear information in speech from its MFCCs.

## MixAR: Basic Definition and Properties

A mixture autoregressive process (MixAR) of order *p*with *m* components, *X*={*x*[*n*]}, is defined as [28]-:

|  |  |  |
| --- | --- | --- |
|  |  |  |

where εi is a zero-mean Gaussian random process with a variance of σj2, “w.p.” denotes “with probability” and the gating weights, *W*i sum to 1. The linear prediction coefficients, {*a*i}, represent the dynamic model, where *a*i,0 are the component means, while {*wi* ,*gi*} are called gating coefficients. It is apparent that an *m*-mixture MixAR process is the weighted sum of *m* Gaussian autoregressive processes, with the time-dependent weights depending on previous data and the gating coefficients.

The basics of a 2-component MixAR model is illustrated in . One convenient way of viewing this model is as a process in which each data sample at any one point in time is generated from one of the component AR mixture processes chosen randomly according to its weight *W*i.

One property of MixAR that is of particular relevance here is the ability of MixAR to model nonlinearity in time series [28]. Though the individual component AR processes are linear, the probabilistic mixing of these AR processes constitutes a nonlinear model. Even when the mixture weights are fixed, the model reduces to MAR, which is still nonlinear. The addition of a gating system layer for weight generation increases the flexibility of the model even further, allowing us to model distributions as a function of past data.

## Comparison of MixAR to Other Models

It is easy to find parallels between the MixAR and GMM models. In particular, MixAR can be viewed as a generalization of GMM that models each component as a sum of the output of an autoregressive filter with a specified mean, and with mixture weights determined by a gating system similar to a mixture of experts. It should be noted that with the component orders and *gi* set to zero, MixAR, reduces to the familiar GMM. This similarity between the two makes it straightforward to replace GMM with MixAR for speaker recognition.

In a GMM, the distribution remains invariant to the past samples due to the static nature of the model. For MixAR, the conditional distribution given past data varies with time. This model is capable of modeling both the conditional means and variances. Thus, MixAR can model time series that evolve nonlinearly. This property becomes important in speech processing in the light of recent work on nonlinear processing of speech, the subject matter of Chapter II.

Some other properties of MixAR, including a mathematically rigorous proof of the ability of MixARs to arbitrarily closely model stochastic processes are derived in [28]. Note that in the original formulation, both the gate and prediction orders were constrained to be equal. In this paper, we restrict our use of MixAR order to one to avoid difficulties during parameter estimation.

Previous work on mixture autoregressive modeling for speech has been in the context of hidden Markov models for speech recognition . One of the earliest applications of autoregressive HMMs (AR-HMMs) considered an autoregressive filter to model state observations in a 5-state HMM for speaker verification . A more recent investigation of AR-HMMs  used a switching autoregressive process to capture signal correlations during state transitions. Results on speech recognition showed that at best their model was only comparable to an MFCC-based HMM using a GMM observation model. Another model considered speech features as a GMM white noise process filtered through an autoregressive signal for speaker identification .

A more sophisticated model introduced in  considers a mixture of autoregressive filters (MAR) for the observation model. Our earlier work [30] considered this model for phone classification. MixAR is a generalization of MAR, where the mixture weights are allowed to be time-varying and data-dependent. Applications of models somewhat related to MixAR were under the context of mixture of experts for time-series prediction . In this work, we apply the MixAR model to feature vectors in a speaker recognition task.

## MixAR Parameter Estimation using the EM algorithm

Similar to the well-known training procedure for GMM, maximum likelihood estimates for MixAR prediction and variance parameters can be calculated using the Expectation-Maximization (EM) algorithm -. Given the order, *p*, the parameter set for each of the *m* components of a MAR model consists of *p*+1 predictor coefficients (including the mean), the error variance, and mixing weight:

|  |  |  |
| --- | --- | --- |
|  |  |  |

To estimate these parameters, we first need an initial guess for these parameters and then we iterate with EM to successively refine the estimates. An initialization strategy that we found to work reasonably well was to first train a GMM with the same number of mixtures and then set each component of the MixAR to have the same mean, variance, and weight as the GMM model. We initialize the predictor coefficients and the data-dependency gating coefficients, {*Ai*} of MixAR to zero.

These initial parameters can be then refined recursively using an E-step [8]:

|  |  |  |
| --- | --- | --- |
|  |  |  |

where

|  |  |  |
| --- | --- | --- |
|  |  |  |

is the probability a sample was generated from component *l* at time instant *n*. The corresponding M-step is given by:

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

where

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |

and

|  |  |  |
| --- | --- | --- |
|  |  |  |

Refer to comments on estimation of predictor coefficients and variances for MAR in for further details.

However, a complication arises with respect to the estimation of gating coefficients. There is no closed-form solution for these, and hence a Newton gradient-ascent approach must be used:

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

where *Q* denotes the likelihood of the MixAR model for the training data. β and Δ are design parameters to be chosen empirically. In our experiments, we found that fixing Δ = 0.01 and running 10 iterations each with β = 0.9, β = 0.5, and β = 0.2 in succession provided a reasonably smooth and quick convergence.

## 

PRELIMINARY EXPERIMENTS

This chapter describes the experiments that were run to study the performance of MixAR model in relation to that of GMM. To better understand the efficacy of the MixAR model, first its performances on two pattern classification tasks are evaluated. The first task represents generic data with known nonlinearities. The second task is a simple classification task with data for the two classes synthesized from models trained on true speaker data. Next a speaker verification experiment with synthetic data that simulate combinations of noise and nonlinearity wss conducted. Finally, speaker verification experiments done on two standard speech databases, NIST [35] and TIMIT [36], are described.

Throughout, the performance of both MixAR and GMM models are studied in relation to their model parameter complexities. For all speaker verification experiments, results are reported either in the form of a detection-error-tradeoff (DET) curve, or equal error rate (EER), both of which are standard for this purpose [3].

Finally, all training and evaluation were conducted using ISIP’s Production System, a public-domain speech recognition system [37].

## Two-Way Classification with Synthetic Data

A simple 2-way classification experiment was designed to study the performance of MixAR and GMM. Two-dimensional data for the first class was generated using a linear dynamic system:

|  |  |  |
| --- | --- | --- |
|  |  |  |

Data for the second class was generated using the simple nonlinear equation:

Table : Classification (% error) results for synthetic data(the numbers of parameters are shown in parentheses).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # mix. | GMM Static | MixAR Static | GMM Static+∆ | MixAR Static+∆ |
| 0 | 36.0 (12) | 6.5 (20) | 10.0 (24) | 5.5 (40) |
| 4 | 35.5 (24) | 6.0(40) | 11.5 (48) | 4.5 (80) |

|  |  |  |
| --- | --- | --- |
|  |  |  |

In both cases, ***E*** denotes an uncorrelated 2-D Gaussian (normal) random variable with a zero mean and unit variance.

For each class, the training data consisted of a sequence of 10,000 vectors, and evaluation data consisted of 100 segments of 200 feature vectors each (the log-likelihood of the entire segment was used to assign a segment to a class). The classification error results are stated in . Clearly, when using only static features, MixAR does much better than GMM if nonlinearities are present. The use of dynamic features enhances GMM performance considerably but still falls far short of MixAR's performance.

## Two-way Classification with Speech-like Data

In order to evaluate how well MixAR does as compared to GMM for speech-like signals, two speakers from the 2001 NIST SRE Corpus  were selected. A 3-state HMM with 4 Gaussian mixtures per state and a MixAR model with 4 mixtures were trained over 12 static MFCC coefficients for each speaker. For each class (e.g, a speaker), two speech-like signals of 40,000 vectors were generated – a linear speech-like signal (***X1***) was synthesized from the HMM model, and a nonlinear speech-like signal (***X2***) was generated from the MixAR model. To simulate a range of signals with varying degrees of nonlinearity, the two signals were mixed with a mixing coefficient alpha:

|  |  |  |
| --- | --- | --- |
|  |  |  |

The first 20,000 vectors from each ***Xα*** were used as a training set while the remaining vectors were split into 200 segments of 100 vectors each for evaluation. The results are shown in .

Table : Classification Error Rate (%) with 12 speech MFCC-like synthetic features for GMM and MixAR Number of parameters in each case is in paranthesis. (\*: For this case, GMM performed better with only static features, and this value is stated).

|  |  |  |
| --- | --- | --- |
| *α* | GMM-8mix. Static+∆ | MixAR-4-mix. Static |
| 0.0\* | 1.5 (288) | 1.5 (240) |
| 0.25 | 3.25 (576) | 3.5 (240) |
| 0.50 | 10.25 (576) | 6.25 (240) |
| 0.75 | 24.75 (576) | 9.75 (240) |
| 1.0 | 26.75 (576) | 13.75 (240) |

From the table we can see that when the amount of nonlinearity is insignificant, GMM performs as well as MixAR. However, as the amount of nonlinearity in the signal increases, MixAR performs significantly better with just static features as compared to GMM with static+∆ features. This clearly demonstrates the superiority of MixAR when dynamics in the data are nonlinear.

## Speaker Verification Experiments with Synthetic Data

Now that it has been demonstrated that MixAR performs better than GMMs as a pattern classifier for signals that have significant nonlinearities in them, the next step is to find whether this holds true even for speaker verification. It is instructive to study the performance of MixAR and GMM when noise is present in addition to nonlinearity. To have control both on the presence of nonlinearity and noise, synthetic data is again used.

Since our goal is to study speaker verification, we used the development database in the 1-speaker detection task of the 2001 NIST SRE Corpus . This database is a standard for demonstrating speaker verification performance. The development database is small enough to make it manageable and yet large enough to provide a reliable estimate of the performance. All 60 speakers in the training set were used. Each training utterance was about 2 minutes long. Static (13 MFCCs), delta (26 MFCCs) and delta-delta (39 MFCCs) features were extracted.

Two kinds of clean data were synthesized. For the first type, a 10-state HMM with 4-Gaussians per state was trained for each utterance for each MFCC. For the second type, a 32-mixture MixAR model of prediction order 1 was trained for each utterance and for each MFCC. For each of the models trained, new training data of about 30,000 frames per speaker and evaluation data of 20 utterances with about 200 frames for eac utterace per speaker were generated according to that model.

Similarly, two kinds of noisy data were generated. For this purpose, first the clean training utterances from the development data were corrupted with car noise from to have an SNR of 5 dB using the FANT software . This was also the methodology followed when the TIDIGIT database was corrupted to generate the AURORA database [39]. The remainder of the steps to yield the two types of noisy data is exactly the same as those for the clean case.



Figure : Speaker Verification DET curves for MixAR-generated nonlinear data with 5B car noise

It is to be noted here that the motivation for generation of these two types of data and under noise conditions is to simulate 4 different test conditions: clean and linear, clean and nonlinear, noisy and linear, and, noisy and nonlinear.

Using the synthesized training data, both GMMs and prediction order-1 MixARs are trained for each speaker under each condition. Then the corresponding synthesized evaluation data are used for evaluating speaker verification performance.

For the clean case, there was little difference in performance between GMM and MixAR. For evaluation data containing 5 dB noise, again there was not much variation in performance between GMM and MixAR for HMM-generated data. However, for the data generated from the nonlinear MixAR model and with the addition of noise, MixAR model showed a significant improvement in performance using far fewer parameters. This is evident from the DET plot in Figure 4. These results provide support to the hypothesis that when there are significant nonlinearities in the signal, using this information makes the nonlinear model much more robust to the presence of noise.

## Speaker Verification Experiments with NIST 2001 Database

Next we applied the MixAR model to the 1-speaker detection task in the 2001 NIST SRE Corpus . Only the development database was used. All 60 speakers were used for training and all 78 utterances were used for evaluation. Each training utterance was about 2 minutes long, while the test utterances were of varying length not exceeding 60 seconds. Static (13 MFCCs), velocity (13 ∆-MFCCs), and acceleration (13 ∆∆-MFCCs) features were extracted.

First performance is evaluated with and without delta features and energy for a fixed number of mixtures. The results are tabulated in Table 4. For GMM, substantial improvement is obtained using the delta features and marginal improvements were obtained using delta-delta features. For MixAR, the use of any delta features provides no measurable improvements. This clearly indicates that MixAR can extract all necessary information from only the static features.

Table : Speaker recognition EER with NIST for MixAR and GMM as a function of #mix. (the numbers of parameters are shown in parentheses).

|  |  |  |
| --- | --- | --- |
| # mix. | GMM Static+∆+∆∆ | MixAR Static |
| 2 | 23.1(216) | 24.1(120) |
| 4 | 21.7(432) | 19.2(240) |
| 8 | 20.5(864) | 19.1(480) |
| 16 | 20.5(1728) | 19.2(960) |

Table : Speaker recognition EER with NIST for MixAR and GMM for different feature combinations.

|  |  |  |
| --- | --- | --- |
| Features | GMM-16-mix. | MixAR-8-mix. |
| Static(12) | 22.1 | 19.1 |
| Static+E(13) | 33.1 | 41.1 |
| Static+Δ(24) | 20.6 | 20.4 |
| Static+Δ+ΔΔ(36) | 20.5 | 20.5 |

MixAR and GMM performance was then evaluated as a function of the number mixtures. The detection error trade-off (DET) curves are shown in Figure 5. The EER results are shown in Table 3. Also indicated in parenthesis is the number of parameters for each case. From this table it is clear that MixAR can achieve about the same performance using almost 4x fewer parameters than GMM. This reduction in the number of parameters points to the efficiency of MixAR in capturing the dynamic information.



Figure : Speaker verification DET curves with NIST.

Moreover, even when considering the best case scenario for GMM with a large number of parameters (8 mixtures with static as well as velocity and acceleration coefficients), there is a 10.6% relative reduction in EER with MixAR. These results appear to strongly indicate that there is nonlinear evolution information in speech features that the GMM model cannot capture using linear derivatives alone and that MixAR can effectively employ this information for achieving better speaker recognition.

## Speaker Verification Experiments with TIMIT Database

Next, a real-world speaker verification performance of MixAR in comparison to GMM is studied using the standard TIMIT database [36]. The core test set in this database consists of 168 speakers, with 10 utterances each. For each speaker, 5 utterances were used for training while the remaining 5 were used for evaluation. Static (13 MFCCs), velocity (13 ∆-MFCCs), and acceleration (13 ∆∆-MFCCs) features were extracted from each utterance. GMM and MixAR models with 4-, 8-, and 16- mixtures were were trained for each speaker using training data and then speaker verification evaluation was conducted.

The EER results are shown in Table 5. Also indicated in parenthesis is the number of parameters for each case. From this table, it is clear that the MixAR model, using fewer parameters, outperforms GMM. In addition, only static MFCC features are sufficient for modeling speakers with MixAR. These results are similar to those reported in the previous section for the NIST 2001 database. This lends further support to the hypothesis that MixAR utilizes nonlinear information in speech to represent speaker characteristics better than what GMM modeling can achieve.

Table : Speaker Verification Performance (EER) on TIMIT core set (the numbers of parameters are shown in parentheses).

|  |  |  |
| --- | --- | --- |
| # mix. | GMM Static+∆+∆∆ | MixAR Static |
| 4 | 3.6 (432) | 3.0 (240) |
| 8 | 2.4 (864) | 1.8 (480) |
| 16 | 2.4 (1728) | 1.7 (960) |

## 

PROPOSED WORK AND EXPERIMENTS

Results from preliminary experiments have thus far supported the view that MixAR can achieve better speaker verification performance with fewer parameters than what GMM can achieve. The superior performance is perhaps because MixAR uses nonlinear dynamic information in speech that GMM cannot and the fewer parameters used points to better efficiency and less redundancy in MixAR compared to the GMM representation.

However, for a successful demonstration of MixAR over GMM for speaker verification, it is necessary to study the performance in noise more extensively, and also to study the variation in performance with varying training and evaluation data lengths.

## Speaker Verification Experiments with TIMIT under Noisy Conditions

In the preliminary experiments it was shown that MixAR does better than GMMs for speaker verification on the TIMIT database. The TIMIT data was collected under noise-free recording conditions. To study the how well MixAR performs compared to GMM under noisy conditions thoroughly, a suite of experiments must be conducted. Similar to the way AURORA database was generated from TIDIGITS in [39]for studying noise performance, several noise conditions will be simulated by adding synthesized noise from different noise sources and at different SNRs. The matrix of the proposed noise conditions to be tested is shown in .

The three types of noise sources were chosen to represent the most commonly occurring types of noise. Also, the four noise levels represent varying degrees of degradation that can occur in the real world. Speech processing systems tend to perform well when the SNR is above 10 dB, so we are primarily interested in studying situations where performance degrades severely. Depending on the trend in these results the actual final matrix may be restricted or expanded. For instance, if noise levels play little role in relative performance of MixAR over GMM, it may not be necessary to conduct experiments for 0 dB. If the type of noise has a large effect, a wider variety of noise types will need to be studied.

Table : Speaker Verification Performance (EER) with varying training and evaluation utterance durations

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GMM | SNR (dB) | White Noise | Car Noise | Babble Noise |
| Clean |  |  |  |
| 10 dB |  |  |  |
| 5 dB |  |  |  |
| 0 dB |  |  |  |
| MixAR | Clean |  |  |  |
| 10 dB |  |  |  |
| 5 dB |  |  |  |
| 0dB |  |  |  |

With these experiments, we hope that the trend observed with preliminary experiments continues to hold, and that MixAR consistently outperforms GMM.

## Speaker Verification Experiments with Variable Amounts of Training Data

Even if MixAR could do better under the conditions we have tested so far, it is possible that MixAR requires more training data than GMM for reliable parameter estimation. This could be a particular concern considering that MixAR attempts to learn nonlinear dynamic information, and nonlinear dynamics are notoriously difficult to characterize from short lengths of data. For example, it is known that estimates of Lyapunov exponents can be unreliable when the length of data is short [22]. Similar concerns also arise when utterance durations are short.

Table **:** Speaker Verification Performance (EER) with varying training and evaluation utterance durations

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GMM | Training Utterance Duration | EER | Evaluation Utterance Duration | EER |
| 60s |  | 30s |  |
| 45s |  | 10s |  |
| 30s |  | 3s |  |
| MixAR | 60s |  | 30s |  |
| 45s |  | 10s |  |
| 30s |  | 3s |  |

It is therefore necessary to study performance as a function of the amount of training data. Towards this end, we propose the set of experiments described in .

## Other Important Issues

Computational complexity is an important component to be studied to ensure that recognition is carried out in real time. It is of little utility in speaker recognition if a new model achieves near 100% accuracy, but takes days to verify the identity of a speaker from an utterance. In the present context, it is necessary to compare the computational requirements of MixAR and GMM. Training is mostly done offline and it does not matter much if this takes more time, but it is especially important that the evaluation has as little overhead as possible.

While we have used an ML-based approach for training both GMM and MixAR speaker models, several GMM systems now use an adapted GMM approach to train speaker models from a universal background model (UBM) . To achieve this for MixAR, an adaptation approach similar to that of GMM needs to be developed.

Discriminative approaches to classification are gaining importance in speech and speaker recognition. Already there exist speaker recognition systems based on GMMs that rely on discriminative training for estimating maximally-separated speaker models [12][10]. The key here is to define what separation between two models means, and to come up with an appropriate procedure to train models to maximize this distance for any two speakers. It would be interesting to design a discriminative approach to MixAR modeling for speaker verification and to note whether this improves over the ML approach to training models.

Another very important extension would be to generalize the applicability of MixAR to other speech processing tasks such as speech recognition. It would be instructive to demonstrate this first with simple isolated digit recognition tasks, and then extend this to large-vocabulary continuous speech recognition tasks.

REFERENCES

X. Huang, A. Acero, and H. Hon, *Spoken Language Processing: A Guide to Theory, Algorithm, and System Development*, Prentice-Hall, 2001.

D. A. Reynolds, “Speaker Identification and Verification using Gaussian Mixture Speaker Models”, *Speech Communication*, vol. 17, no. 1-2, pp. 91-108, 1995.

A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki, “The DET Curve in the Assessment of Detection Task Performance,” *Proceedings of the uropean Conference on Speech Communication and Technology*, pp. 1895-1898, 1997.

R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd edition, Wiley-Interscience, USA, October 2000.

F. K. Soong, E. Rosenburg, B. Juang, and L. Rabiner, “A Vector Quantization Approach to Speaker Recognition,” *AT&T Technical Journal*, vol. 66, pp. 14-26. 1987.

S. Kuroiwa , Y. Umeda , S. Tsuge , and F. Ren, “Nonparametric Speaker Recognition Method using Earth Mover's Distance,” *IEICE - Transactions on Information and Systems*, vol. E89-D, no. 3, pp. 1074-1081, March 2006.

M. Forsyth, “Discriminating Observation Probability HMM for Speaker Verification,” Speech Communication, vol. 17,nos. 1-2, pp. 117-129, August 1995.

W. M. Campbell, D. E. Sturim, and D. A. Reynolds, “Support Vector Machines using GMMM Supervectors for Speaker Verification,” *IEEE Signal Processing Letters*, vol. 13, no. 5, pp. 308–311, May 2006.

V. Wan and S. Renals, "Speaker Verification using Sequence Discriminant Support Vector Machines," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 2, pp. 203-210, March 2005.

M. Liu, Y. Xie, Z. Yao, and B. Dai, “A New Hybrid GMM/SVM for Speaker Verification,” *Proceedings of The 18th International Conference on Pattern Recognition*, vol. 4, pp. 314-317, Hong Kong, China, August 2006.

Q. Y. Kong and S. Kwong, “A Discriminative Training Approach for Text-Independent Speaker Recognition,” Signal Processing, vol. 85 no. 7, pp. 1449-1463, July 2005.

C. Ma, E. Chang, “Comparison of Discriminative Training Methods for Speaker Verification,” *Proceedings of the ICASSP 2003*, pp 192-195, 2003.

H. M. Teager and S. M. Teager, “Evidence for Nonlinear Production Mechanisms in the Vocal Tract,” *NATO Advanced Study Institute on Speech Production and Speech Modeling*, Bonas, France, pp. 241-261, July 1989.

J. Markel, and A. Gray, *Linear Prediction of Speech*, Springer-Verlag, New York, 1976.

R. W. Schafer, L. R. Rabiner, “Digital Representations of Speech Signals,” *Proceeding of the IEEE*, vol. 63, no. 4, April 1975.

I. Kokkinos, and P. Maragos, “Nonlinear Speech Analysis using Models for Chaotic Systems,” *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 6, pp. 1098-1109, November 2005.

D. May, *Nonlinear Dynamic Invariants For Continuous Speech Recognition*, M.S. Thesis, Department of Electrical and Computer Engineering, Mississippi State University, USA, May 2008.

S. Prasad, S. Srinivasan, M. Pannuri, G. Lazarou and J. Picone, “Nonlinear Dynamical Invariants for Speech Recognition,” *Proceedings of the International Conference on Spoken Language Processing*, pp. 2518-2521, Pittsburgh, Pennsylvania, USA, September 2006.

P. Maragos, A. G. Dimakis and I. Kokkinos, “Some Advances in Nonlinear Speech Modeling using Modulations, Fractals, and Chaos,” *Proceedings of the IEEE International Conference on Digital Signal Processing*, pp. 325-332, Santorini, Greece, July 2002.

M. Banbrook, S. McLaughlin, “Is Speech Chaotic?: Invariant Geometrical Measures for Speech Data,” *IEE Colloquium on Exploiting Chaos in Signal Processin*g, Digest No. 1994/193, pp. 8/1-8/10, London, U.K., June 1994.

P. Maragos, “Fractal Aspects of Speech Signals: Dimension and Interpolation,” *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 417-420, Toronto, Canada, May 1991.

M. Banbrook, G. Ushaw, and S. McLaughland, “How to Extract Lyapunov Exponents from Short and Noisy Time Series,” *IEEE Transactions on Signal Processing*, vol. 45, no. 5, pp. 1378-1382, May 1997.

H. F. V. Boshoff and M. Grotepass, “The Fractal Dimension of Fricative Speech Sounds,” *Proceedings of the South African Symposium on Communication and Signal Processing*, pp. 12-16, Pretoria, South Africa, August 1991.

A. C. Lindgren, M. T. Johnson and J. Povinelli, “Speech Recognition using Reconstructed Phase Space Features,” *Proceedings of the International Conference on Acoustics, Speech and Signal Processin*g, vol. 1, pp. I-60-63, Hong Kong, China, April 2003.

A. Petry, D. Augusto, and C. Barone, “Speaker Identification using Nonlinear Dynamical Features,” *Chaos, Solitons & Fractals*, vol. 13, Issue 2, pp. 221-231, February 200

A. Dempster, N. Laird, and D. Rubin, “Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of the Royal Statistical Society, Series B*, vol. 39, no. 1, pp. 1-38, February 1977.

J. Blimes, “A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation Problem for Gaussian Mixture and Hidden Markov Models,” *Technical Report TR-97-021*, University of California, Berkeley, , May 1997.

M. Zeevi, R. Meir, and R. Adler, “Nonlinear Models for Time Series using Mixtures of Autoregressive Models”, *Technical Report*, Technion University, Israel, available online at: *http://ie.technion.ac.il/~radler/mixar.pdf*, October 2000.

C. S. Wong, and W. K. Li, “On a Mixture Autoregressive Model,” *Journal of the Royal Statistical Society*, vol. 62, no. 1, pp. 95‑115, February 2000.

S. Srinivasan, T. Ma, D. May, G. Lazarou and J. Picone, "Nonlinear Mixture Autoregressive Hidden Markov Models for Speech Recognition," *Proceedings of the International Conference on Spoken Language Processing*, pp. 960-963, Brisbane, Australia, September 2008.

B. H. Juang, and L. R. Rabiner, “Mixture Autoregressive Hidden Markov Models for Speech Signals,” *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 33, no. 6, pp. 1404-1413, December 1985.

Y. Ephraim, and W. J. Roberts, “Revisiting Autoregressive Hidden Markov Modeling of Speech Signals,” *IEEE Signal Processing Letters*, vol. 12, no. 2, pp. 166-169, February 2005.

M. E. Ayadi, *Autoregressive Models for Text Independent Speaker Identification in Noisy Environments*, PhD Thesis, Department of Electrical and Computer Engineering, University of Waterloo, Canada, September 2008.

A. X. Carvalho and M. A Tanner, “Modeling Nonlinearities with Mixtures-of-Experts of Time Series Models,” *International Journal of Mathematics and Mathematical Sciences*, vol. 2006, no. 9, pp. 1-22, May 2006.

National Institute of Standards and Technology, “The 2001 NIST Speaker Recognition Evaluation,” *http://www.nist.gov/speech/tests/spk/2001*, 2001.

“TIMIT Acoustic-Phonetic Continuous Speech Corpus,” *http://www.ldc.upenn.edu /Catalog/CatalogEntry.jsp?catalogId=LDC93S1*, Linguistic Data Consortium, Philadelphia, 1993.

J. Picone, “ISIP Foundation Classes,” *http://www.isip.piconepress.com/projects/speech /software/downloads/production/*, Department of Electrical and Computer Engineering, Temple University, Philadelphia, Pennsylvania, USA, January 2010.

Filtering and Noise-adding Tool (FaNT), available online at *http://dnt.kr.hsnr.de/download.html.*

D. Pearce and H. Hirsch, The Aurora Experimental Framework for the Performance Evaluation of Speech Recognition Systems under Noisy Conditions, *ICSLP 2000*, pp. 29-32, October 2000.

Noise data from Signal Processing Information Base (SPIB) of Rice University, available online at *http://spib.rice.edu/spib/select\_noise.html*.