ACOUSTIC MODEL ADAPTATION FOR AUTOMATIC SPEECH RECOGNITION AND ANIMAL VOCALIZATION CLASSIFICATION

by

Jidong Tao, B.Eng., M.S.

A Dissertation submitted to the Faculty of the Graduate School,
Marquette University,
in Partial Fulfillment of the Requirements for
the Degree of Doctor of Philosophy

Milwaukee, Wisconsin

May, 2009

ABSTRACT
ACOUSTIC MODEL ADAPTATION FOR AUTOMATIC SPEECH RECOGNITION AND ANIMAL VOCALIZATION CLASSIFICATION

Jidong Tao, B.Eng., M.S.

Marquette University, 2009

Automatic speech recognition (ASR) converts human speech to readable text. Acoustic model adaptation, also called speaker adaptation, is one of the most promising techniques in ASR for improving recognition accuracy. Adaptation works by tuning a general purpose acoustic model to a specific one according to the person who is using it. Speaker adaptation can be categorized by Bayesian-based, transformation-based and model combination-based methods. Model combination-based speaker adaptation has been shown to have an advantage over the traditional Bayesian-based and transformation-based adaptation methods when the amount of adaptation speech is as small as a few seconds. However, model combination-based rapid speaker adaptation has not been widely used in practical applications since it requires large amounts of speaker-dependent (SD) training data from multiple speakers. This research proposes a new technique, ***eigen-clustering***, to eliminate the need for large quantities of speaker-labeled training utterances so that model combination-based adaptation can be started from much more inexpensive speaker-independent (SI) data. Based on ***principal component analysis*** (PCA), this technique constructs an eigenspace using each utterance in the training set. This proposed adaptation method can not only improve human speech recognition directly, but also contribute to animal vocalization analysis and behavior studies potentially. Application to the field of bioacoustics is especially meaningful because the amount of collected animal vocalization data is often limited and therefore fast adaptation methods are naturally suitable.

PREFACE

As I am finalizing my dissertation, I am considering the two most valuable experiences in my doctoral process. The first is that of identifying a specific research direction for my Ph.D. study, and the second is that of figuring out how to accomplish it. Generally, a perfect topic would allow a student to accomplish his or her program in less time with better quality, but it is very difficult to find this "right" direction at the early stages of study. The research topics I had been working on originally spanned many different areas in both human speech technologies and bioacoustics, including acoustic enhancement for improving audio quality, acoustic feature extraction at the front-end of recognition systems, and looking at the Lombard effect for investigating the auditory system. Eventually, I settled on acoustic model adaptation as my dissertation topic. I have gained much research experience and knowledge from all these areas.

Before I decided my research direction, thorough research in this direction was critical. This taught me what other researchers have done in this area, and which part of this direction was still open. However, one more practical point I often ignored was how those people implemented their methods in terms of experimental work and software programming, so I did not focus until almost the last year in my Ph.D. life on whether I could realistically implement the same experiments as what the people did in their works. I finally realized that it was nothing to be proud of to just understand complicated algorithms, such as the *expectation maximization* (EM), because the derivation of statistical equations is a fundamental skill to a Ph.D. candidate in electrical engineering. Having an earlier consciousness of programming implementation would have given me a better understanding for the time and effort I needed for this research direction, and help me make a wise

decision as to both theory and practice.

This Ph.D. work was funded by Dr. Dolittle project, which focuses on development of a broad framework for pattern analysis and classification of animal vocalizations by integrating successful models and ideas from the field of speech processing and recognition into bioacoustics (Johnson *et al.*, 2003). Therefore my work naturally consists both of a theoretical aspect for human speech and a practical aspect for bioacoustic application. Although the field of bioacoustics is challenging due to its multidisciplinary nature, speech technology is the original foundation. I am hopeful that my Ph.D. research will benefit both fields.

ACKNOWLEDGMENTS

Jidong Tao, B.Eng., M.S.

I could not have completed this dissertation without the support from my colleagues, family members, church saints, and friends. The first person I want to say thanks is my academic advisor, Dr. Michael T. Johnson. It was he who gave me the opportunity to work with him, brought me to this project, and encouraged me to work hard to overcome any difficulties. I also thank my committee for all of the valuable research ideas and comments, and all my collaborators in Dr. Dollitle project for discussion and data sharing. Thanks to my intern co-workers at Vlingo Corporation in summer 2008 for providing me such a golden opportunity to work with them in a fast paced environment, to learn working experience in a real world, and to improve my skills in speech technology to a totally different level.

Thanks to my parents and brothers for their support and understanding that I could not return home to visit during my studies. Thank you to my church saints in Boston, Los Angeles, and Milwaukee for their prayer, so I could keep a clean and peaceful mind during these studies. Thanks to all my friends in different locations of the world for their support and encouragement. Finally, I want to give all glory to God for everything He has created.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

C H A P T E R 1 INTRODUCTION

1.1 Problem Definition

Automatic speech recognition (ASR) is the process of converting human speech to readable text. The core of all speech recognition systems consists of a set of statistical acoustic models representing the various sounds of the language to be recognized. ASR can be classified as speaker-independent (SI) or speaker-dependent (SD), depending on whether the system's acoustic model is trained for a variety of speakers or targeted to a specific user. Each speaker has differences in a range of physical characteristics: age, sex, dialect, personal style, etc. Because of these differences, speaker-independent systems often show large performance fluctuations when recognizing the data of new speakers, due to mismatches between the new speaker's data and the training data. A well-trained speaker-dependent system generally performs significantly better than a speaker-independent one in recognizing speech from the target speaker because it has no such mismatch issue. However, speaker-dependent ASR requires a large amount of speech data for each specific user of the system, and all users need to train individual acoustic models in order to obtain acceptable performance.

Acoustic model adaptation, also called speaker adaptation, is a method for quickly decreasing sensitivity to speaker variability when the amount of data for a new user is limited. Speaker adaptation techniques take utterances from a specific user, called adaptation data, and tune the parameters of pre-trained background acoustic models to create a speaker-adapted (SA) system, close to an ideal SD system trained for that user. Background models can be a single SI model set or several SD models, depending on training data availability.

For example, the background models are SI in both Bayesian-based ***maximum a posterior*** (MAP) (Gauvain and Lee, 1994) and transformation-based ***maximum likelihood linear regression*** (MLLR) (Leggetter and Woodland, 1995), whereas they are SD in model combination-based methods such as ***reference speaker weighting*** (RSW) (Hazen, 2000) and ***eigenvoice*** (EV) (Kuhn *et al.*, 2000).

Speaker adaptation improves ASR performance effectively with a small amount of adaptation data. When adaptation data is only a few seconds, the task is called rapid speaker adaptation. Not all the techniques mentioned above are able to implement rapid adaptation. Model combination-based adaptation methods are commonly referred to as "rapid" techniques because they are more effective than both Bayesian-based and transformation-based methods when data is minimal (Kuhn *et al.*, 2000).

Rapid speaker adaptation is important to ASR research since large vocabulary continuous speech recognition (LVCSR) systems often have little time or data to adapt new speakers. However, current rapid adaptation techniques are based on SD rather than SI model and thus require large amounts of speaker-dependent training data. Because SD background models are much more expensive in terms of data collection, transcription, and training time than SI models, rapid adaptation has not yet been widely used in practical applications. An important open question to address therefore is whether we can accomplish rapid speaker adaptation without pre-trained SD models and training data.

In addition to being a useful tool for improving the accuracy of ASR systems, adaptation can be effectively implemented in other domains as well. In fact, the original motivation for pursuing this particular research problem arose out of animal vocalization classification, in the field of bioacoustic signal processing. In such tasks, it is possible to use model

adaptation to improve classification accuracy, which, as discussed in more detail in Section 4.3, the classification for ortolan bunting vocalizations was successfully demonstrated (Tao and Johnson, 2008). In the vast majority of bioacoustic domains, it is necessary to use fast adaptation because the length of individual vocalizations and the duration of caller (the animal making the vocalization) turns is short and irregular, but implementation is difficult because few datasets contain sufficient identity-labeled data to build the speaker-dependent models that are required to use existing rapid adaptation approaches.

1.2 Purpose

This dissertation introduces a new method in model combination-based rapid speaker adaptation using only SI training data. The new technique, ***eigen-clustering***, eliminates the need for large quantities of speaker-labeled training utterances, so that speaker adaptation with extremely small amount of adaptation data (e.g., a few seconds) can be accomplished from inexpensive SI data. The proposed method can not only improve human speech recognition, but also contribute to animal vocalization analysis and behavior studies. The ultimate purpose of this research is to benefit both human speech technology and bioacoustics applications.

1.3 Dissertation Overview

The first chapter has been a brief overview of the dissertation and the motivation behind the research. The second chapter discusses background knowledge and related works in the fields of speech recognition and bioacoustics. The third chapter proposes the Eigen-clustering framework and mathematical description.

The fourth chapter details the experiments and results of the new method for both human speech recognition and animal vocalization classification. The recognition results based on the existing adaptation techniques are also presented for comparison purpose.

The final chapter, five, gives a summary of the dissertation, discusses the contributions of the research, and suggests possibilities for future work.

C H A P T E R 2 BACKGROUND

2.1 Automatic Speech Recognition

Automatic speech recognition (ASR) aims to provide a human-machine interface for transferring spoken language to written text. The basic structure of an ASR system (Yu, 2006) is shown in Figure 2.1.

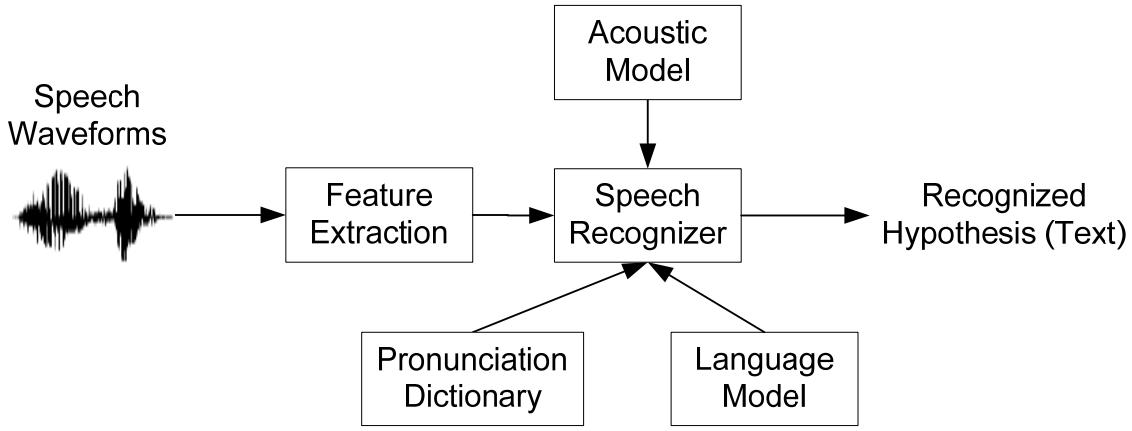

Figure 2.1 ASR System

The speech signal is processed in a feature extraction or front-end processing module that extracts salient feature vectors, referred to as observations. Given the extracted observation sequence $O = o_1, o_2, ..., o_T$, where T is the length of the sequence, generally three sources of information are required by the recognizer: an acoustic model, a language model, and a dictionary. The acoustic model is a statistical representation of knowledge about acoustics, phonetics, gender, and dialect differences among speakers. The language model incorporates knowledge of possible word sequence, semantics, and grammatical variation. The dictionary, or lexicon, maps pronunciation units such as phonemes, from which the acoustic model is constructed, to the word set present in language model.

The speech recognizer, sometimes referred to as decoder, uses all the above information to hypothesize the word sequence $\hat{W} = w_1 w_2 \ldots w_L$, where L is the number of words in the sequence, with the maximum a posterior probability $P(W|O)$ as expressed by

$$\hat{W} = \arg\max_{w} P(W|O) = \arg\max_{w} \frac{P(O|W)P(W)}{P(O)} = \arg\max_{w} P(O|W)P(W). \quad (2.1)$$

The likelihood $P(O|W)$ is determined by the acoustic model and the prior $P(W)$ is determined by the language model.

2.1.1 Hidden Markov Models

The most popular and successful acoustic model to date is the Hidden Markov Model (HMM) (Juang, 1984; Rabiner, 1989; Rabiner and Juang, 1993; Jelinek, 1999). This is a natural framework for modeling speech, which has temporal structure and features that can be encoded as a sequence of spectral vectors in the frequency domain. An HMM acoustic model based ASR system is emphasized in this dissertation.

A left-to-right HMM with N states is shown in Figure 2.2. At each time instance t that a state j is entered, $1 \le t \le T$, an observation feature vector O_t is generated one by one by the probability density $b_j(O_t)$ from the $N-2$ emitting states. The generation starts from the first non-emitting state and stops at the last, with no observations generated by the two start and end states. The transition probability, a_{ij}, indicates the probability of which state transits to either itself or the continuous right state. Therefore, the problem of estimating the likelihood $P(O|W)$ in equation (2.1) is replaced by estimating the HMM λ as

$$P(O|W) = P(O|\lambda) = \sum_{s} a_{s_0 s_1} \prod_{t=1}^{T} b_{s_t}(o_t) a_{s_t s_{t+1}}, \quad (2.2)$$

where $S = s_0, s_1, ..., s_{T+1}$ is a state sequence associated with the observation sequence through the HMM, and s_0 and s_{T+1} correspond to the non-emitting states shown in Figure 2.2.

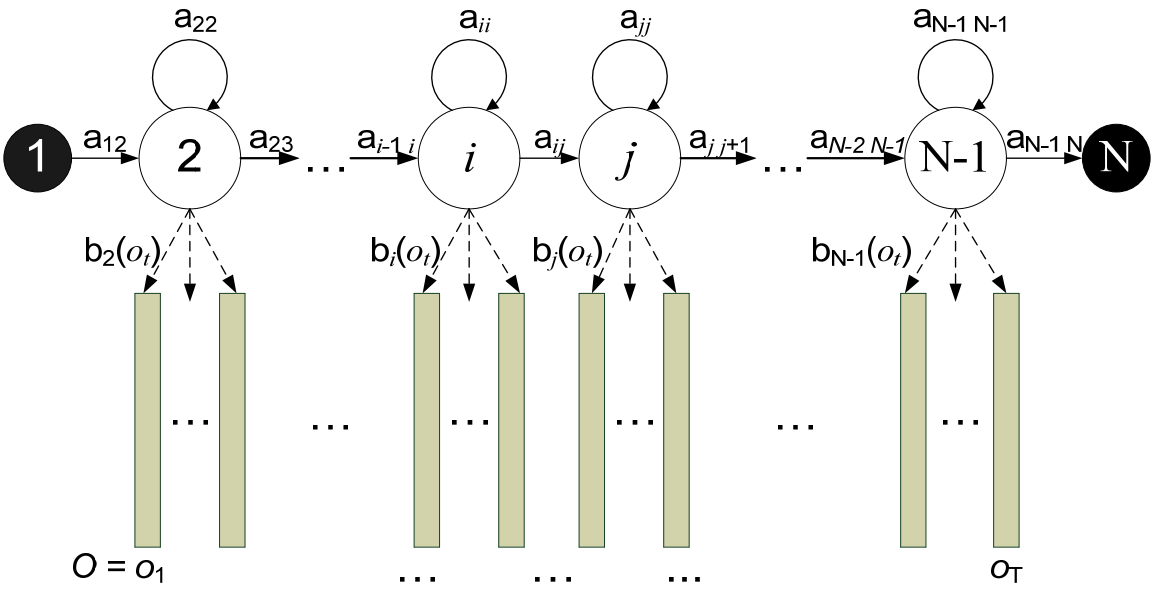

Figure 2.2 A left-to-right HMM with *N*-2 emitting states

The observation distribution $b_j(O_t)$ is represented by either single Gaussian or multiple Gaussian distributions, which is referred to as Gaussian mixture model (GMM). A GMM is defined as the density function

$$b_j(o_t) = \sum_{m=1}^{M_j} c_{jm} N(o_t; \mu_{jm}, \Sigma_{jm}), \tag{2.3}$$

where M_j is the number of mixture components for state j, c_{jm} is the weight of component m of state j subject to the constraint $\sum_{m=1}^{M_j} c_{jm} = 1$. $N(o_t; \mu_{jm}, \Sigma_{jm})$ is the m th normal density function of state j denoted by

$$N(o_t; \mu_{jm}, \Sigma_{jm}) \propto \left|\Sigma_{jm}\right|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(o_t - \mu_{jm})^T \Sigma_{jm}^{-1}(o_t - \mu_{jm})\right]. \tag{2.4}$$

A single Gaussian ($M_j = 1$) may be not suitable especially for a HMM system trained by the speech data from a variety of speakers because there are many pronunciation variations of the same phoneme (Becchetti and Ricotti, 1999). Multiple mixture Gaussians ($M_j > 1$) can approximate continuous pdfs, and more accurately represent the distribution of speech data (Liporace, 1982; Juang, 1985; Juang *et al.*, 1986).

2.1.2 Estimation of HMM Parameters – Expectation-Maximization Algorithm

Training data and machine learning algorithms such as the *expectation maximization* (EM) algorithm (Dempster *et al.*, 1977) are used to find the parameters of the HMM. The EM algorithm is an iterative procedure for approximating maximum likelihood (ML) in the context of incomplete-data cases such as Gaussian mixture density and hidden Markov model estimation problems. For each iteration, the procedure consists of an expectation step (E-step) and a maximization step (M-step). In the E-step, the Baum's auxiliary function $Q(\lambda, \hat{\lambda})$ (Baum *et al.*, 1970) is computed as the expectation of the complete-data (sufficient statistics) given the incomplete-data (observations) and the current model parameters λ. Given this complete-data, the M-step estimates the parameters of the model by maximizing the auxiliary function. The application of re-estimation formulae for HMM implementation of the general EM algorithm is called the Baum-Welch algorithm (Baum *et al.*, 1970; Baum, 1972; Moon, 1996). For an HMM with N number of states, where states 1 and N are the non-emitting initial and final states, the forward probability is defined as the joint probability of the first t observation vectors in state j at time t:

$$\alpha_t(j) = P(o_1, ..., o_t, s_t = j \mid \lambda) = \left[\sum_{i=2}^{N-1} \alpha_{t-1}(i) a_{ij} \right] b_j(o_t) . \tag{2.5}$$

$\alpha_t(j)$ can be computed recursively on the right hand right of equation (2.5). The initial and final conditions for the above recursion are

$$\alpha_t(j) = \begin{cases} 1 & j=1 \quad t=1 \\ a_{1j} b_j(o_t) & 1<j<N \quad t=1 \\ \sum_{i=2}^{N-1} \alpha_T(i) a_{iN} & j=N \quad t=T \end{cases}. \tag{2.6}$$

Likewise, the backward probability is defined as the probability of the observation sequence from $t+1$ to the end, and calculated as

$$\beta_t(j) = P(o_{t+1}, ...o_T \mid s_t = j, \lambda) = \sum_{j=2}^{N-1} a_{ij} b_j(o_{t+1}) \beta_{t+1}(j). \tag{2.7}$$

The initial and final conditions are

$$\beta_t(j) = \begin{cases} a_{jN} & 1<j<N, t=T \\ \sum_{i=2}^{N-1} a_{1i} b_i(o_1) \beta_1(i) & j=1, t=1 \end{cases}. \tag{2.8}$$

The state transition count $\xi_t(i,j)$, which is the probability of being in state i at time $t-1$ and going to state j at time t, is defined and computed as

$$\xi_t(i,j) = P(s_{t-1} = i, s_t = j \mid O, \lambda) = \frac{\alpha_{t-1}(i) a_{ij} b_j(o_t) \beta_t(j)}{P(O \mid \lambda)}, \tag{2.9}$$

where $P(O \mid \lambda)$ is the total likelihood and calculated by either forward probability or backward probability as

$$P(O \mid \lambda) = \alpha_T(N) = \beta_1(1). \tag{2.10}$$

The state occupation count, $\gamma_t(j)$, the probability of being in state j at time t, is defined and calculated as:

$$\gamma_t(j) = P(s_t = j \mid O, \lambda) = \frac{P(O, s_t = j \mid \lambda)}{P(O \mid \lambda)} = \frac{\alpha_t(j) \beta_t(j)}{P(O \mid \lambda)}. \tag{2.11}$$

The HMM parameters λ can be estimated by maximizing the right side of equation (2.2) iteratively. The maximization process is equivalent to maximizing the following auxiliary function (Dempster *et al.*, 1977):

$$Q(\lambda,\hat{\lambda}) = \sum_{t,j} \gamma_t(j) \log b_j(o_t) + \sum_{t,i,j} \xi_t(i,j) \log a_{ij} , \tag{2.12}$$

where $\hat{\lambda}$ is the new HMM parameters derived from λ in the previous iteration. Given the above definitions, the final re-estimation formulae for the parameters of HMM with GMM at state j are as follows:

$$\hat{c}_{jm} = \frac{\sum_t \gamma_t(jm)}{\sum_{m,t} \gamma_t(jm)}, \tag{2.13}$$

$$\hat{\mu}_{jm} = \frac{\sum_t \gamma_t(jm) o_t}{\sum_t \gamma_t(jm)}, \tag{2.14}$$

$$\hat{\Sigma}_{jm} = \frac{\sum_t \gamma_t(jm)(o_t - \hat{\mu}_{jm})(o_t - \hat{\mu}_{jm})^T}{\sum_t \gamma_t(jm)}. \tag{2.15}$$

A similar re-estimation equation can be derived for the transition probability

$$\hat{a}_{ij} = \frac{\sum_{t=2}^{T} \xi_t(i,j)}{\sum_{t=1}^{T} \gamma_t(i)}. \tag{2.16}$$

The Baum-Welch algorithm described above runs iteratively. More specifically in each iteration, the E-step computes the expected state occupancy γ and the expected state transition count ξ using the previous iteration forward and backward probabilities, and the M-step re-estimates the model parameters in equations (2.13), (2.14), (2.15) and (2.16) to maximize the auxiliary Q-function in equation (2.12).

For recognition, the Viterbi algorithm (Forney, 1973) is generally used to identify the

most likely sequence of states that could have produced the input utterance, and the correct acoustic model is selected using ***maximum likelihood*** (ML) (Moon, 1996) method over the set of all HMMs.

Recognition results are commonly reported by *word error rate* (WER) or *word accuracy* (ACC) (Huang *et al.*, 2001). Word error rate is defined as

$$WER = \frac{S + D + I}{N}\%, \tag{2.17}$$

where S is the number of substitutions, D is the number of deletions, I is the number of insertions, and N is the number of words in the reference. Word accuracy is computed as

$$ACC = \frac{N - S - D - I}{N}\% = \frac{H - I}{N}\% = 1 - WER, \tag{2.18}$$

where H is $N - (S + D)$, the number of correctly hit (recognized) words. The Viterbi algorithm (Forney, 1973) generally is used to align a recognized word sequence against the correct word sequence in transcription and compute the number of substitutions, deletions, and insertions. Word accuracy is used to measure the experimental performance in this study.

2.1.3 Acoustic Units and Context Dependency

The fundamental unit of a language (e.g., English) is typically a word. For speech recognition tasks with a small vocabulary (< 1K words), such as English digits, HMMs are often whole-word models with enough states to represent the sequence of all phonemes. An advantage of using word models is that the phonetic coarticulation inherent within these words can be captured.

While whole-word HMMs have been widely used for small vocabulary speech

recognition, they are impractical in terms of training data and number of models needed for speech recognition with medium (1K – 10K words) to large vocabularies (> 10K words). In these situations, the alternative is to use phoneme based models, which can be sufficiently trained with a few hundred sentences. An advantage of using phone models is that there is a standard rule (i.e., dictionary or lexicon) to map words to phonemes allowing words to be easily split into a sequence of phones with capacity for unlimited vocabulary.

There are two major categories for phone models: context-independent and context-dependent models. Monophone models, which are context-independent, are trained using each individual phoneme in the set (e.g., about 50 phonemes in English). Monophone models do not take into account the coarticulatory effect, in which pronunciation of the current phone is strongly affected by its immediately preceding and following phones. Thus, monophone models for many speech recognition tasks lead to less accurate performance.

Context-dependent phone models have been widely used in most state-of-the-art speech recognition systems and give significantly improved performance because they capture the most phonetic coarticulation (Huang *et al.*, 2001). A common context-dependent phone model is the triphone, which takes into consideration both the left and right neighboring phones of the current phone. For example, in the word red, a possible triphone may be [r-e+d], where [r] and [d] are the preceding and following phones of the phone [e], "-" denotes the left context and "+" denotes the right context.

2.2 Speaker Adaptation

As introduced in Chapter 1, speaker adaptation (SA) adapts the pre-trained background model(s) to the ideal user specific (SD) system using much less data than that required to

train the SD model. In HMM-based acoustic models, adaptation techniques change the mean (and possibly covariance) of Gaussian probability density functions (pdfs) (Huang *et al.*, 2001). Figure 2.3 illustrates speaker adaptation in an HMM-based acoustic model in a two-dimensional feature space. Starting from the original Gaussian pdfs of the SI training data for an HMM on the left, the right shows the HMM is trained by SD data in an ideal situation and speaker adaptation is needed to move the distribution towards the SD model in practice.

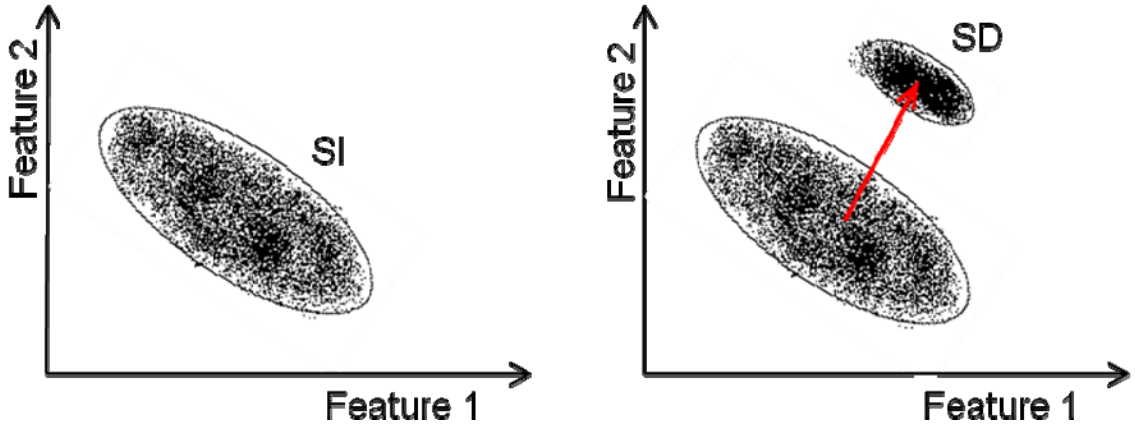

Figure 2.3 Speaker adaptation in acoustic model

The challenge for model adaptation is that we can use only a small amount of observable data to modify model parameters. By obtaining a little information about the current speaker, a speech recognizer should be able to improve its performance by adapting to the characteristics particular to the current utterance. Model-based speaker adaptation algorithms range from the traditional approaches such as Bayesian-based *maximum a posteriori* (MAP) (Gauvain and Lee, 1994) and the transformation-based *maximum likelihood linear regression* (MLLR) (Leggetter and Woodland, 1995) to rapid speaker

adaptation methods such as ***eigenvoice*** (EV) (Kuhn, 1998; Kuhn *et al.*, 2000) and ***reference speaker weighting*** (RSW) (Hazen and Glass, 1997; Hazen, 2000).

2.2.1 Maximum a Posteriori (MAP) – Bayesian-based Approach

The problem of speaker adaptation is one of finding the most likely set of acoustic model parameters given the adaptation data. We assume that an HMM is characterized by a parameter vector λ, and prior knowledge about the vector is available and characterized by *a prior* probability density function, $p(\lambda)$, whose parameters are to be determined experimentally. With a set of *T* observation vectors, $O = o_1, o_2, ..., o_T$, the MAP estimate is defined as

$$\hat{\lambda}_{MAP} = \arg\max_{\lambda} p(\lambda \mid O) = \arg\max_{\lambda} p(O \mid \lambda) p(\lambda). \quad (2.19)$$

If we have no prior information, then $p(\lambda)$ is the uniform distribution, and the MAP estimate becomes identical to the maximum likelihood (ML) estimate. A more accurate prior, $p(\lambda)$, makes MAP estimation more robust. However, MAP does have the desirable property that it will eventually converge to the SD performance. The MAP estimate for the parameters of HMMs can be implemented by the EM algorithm (Gauvain and Lee, 1994). The corresponding auxiliary function for Gaussian components is defined as

$$Q_{MAP}(\lambda, \hat{\lambda}) = \log p(\lambda) - \frac{1}{2} \sum_{t,j,m} \gamma_t(jm) \left[\log |\Sigma_{jm}| + (o_t - \mu_{jm})^T \Sigma_{jm}^{-1} (o_t - \mu_{jm}) \right], \quad (2.20)$$

where $\gamma_t(jm)$ is the occupancy of component m of state j. To yield mathematically simple forms of MAP criterion, the prior distribution $p(\lambda)$ should be a conjugate prior to the likelihood of the observations, assuming the Gaussian mixture weights and component

parameters are independent. Then the mean vector of MAP can be re-estimated as

$$\hat{\mu}_{jm} = \frac{\tau_{jm}\mu_{jm} + \sum_t \gamma_t(jm) o_t}{\tau_{jm} + \sum_t \gamma_t(jm)}, \tag{2.21}$$

where μ_{jm} is the prior mean vector, τ_{jm} is a balancing factor between prior mean and the ML mean (Gauvain and Lee, 1994).

From equation (2.21), as the amount of adaptation data increases infinitely, the estimate converges to the ML estimate. When the adaptation data is limited, an accurate prior mean vector μ_{jm} gives robust estimates. A major limitation is that the MAP adaptation is a local approach to updating the model parameters. Namely, only the model parameters that are observed in the adaptation data can be modified from the prior value. When the system has a large number of free parameters, MAP approach can be very slow and is unsuitable for rapid adaptation.

2.2.2 Maximum Likelihood Linear Regression (MLLR) – Transformation-based Approach

Maximum likelihood linear regression (MLLR) uses a set of linear regression transformation functions to map both the mean vector and covariance matrix of a Gaussian mixture model in order to maximize the likelihood of the adaptation data (Leggetter and Woodland, 1995). The mean vector μ is transformed by:

$$\hat{\mu} = A\mu + b = W\xi, \tag{2.22}$$

where A is a regression transformation matrix, b is an additive bias vector, ξ is the extended mean vector as $[1, \mu^T]^T$, and W is the extended transformation matrix $[b \quad A]$. The same transformation can be used for all Gaussian components across all acoustic

models, or different transformations can be used for different groups of Gaussian components, called regression classes. When the amount of adaptation data is limited or does not include the observations for certain acoustic models, the same transformation can be used for several distributions if they are in the same class that is grouped by similar acoustic characteristics.

The transformation matrix W is obtained using the EM algorithm (Leggetter and Woodland, 1995). A standard auxiliary function for MLLR transform updates can be defined as

$$Q_{MLLR}(\lambda, \hat{\lambda}) = -\frac{1}{2} \sum_{t,j,m} \gamma_t(jm)(o_t - W\xi_{jm})^T \Sigma_{jm}^{-1}(o_t - W\xi_{jm}), \tag{2.23}$$

where $\gamma_t(jm)$ is the occupancy of component m of state j. The detailed solutions for estimating transformation of mean and variance using EM algorithm are derived in (Leggetter and Woodland, 1995) and (Gales and Woodland, 1996; Gales, 1998) respectively.

MLLR has faster adaptation than MAP adaptation when the amount of adaptation data is small because of regression classes, but MLLR becomes less accurate than MAP as adaptation data size increases. Although MLLR requires less adaptation data than MAP, it still needs over a minute of data in most spoken dialogue systems (Glass *et al.*, 1996), which means MLLR is still not considered a rapid adaptation method.

2.2.3 Rapid Speaker Adaptation – Model Combination Based Approach

MAP and MLLR approaches have been studied for many years. Both require at least tens of seconds of adaptation data from the new speaker in order to perform better than a SI system. In contrast, model combination-based adaptation performs effectively when the amount of adaptation data is only a few seconds, and so is called rapid adaptation. A model

combination-based approach, however, requires speaker-dependent models as *reference speakers* to start adapting to a new speaker, instead of using a speaker-independent model directly as in MLLR and MAP adaptation.

The basic premise behind rapid adaptation shown in Figure 2.4 is that the model parameters of a new speaker can be approximated from a weighted combination of model parameters from different reference speakers.

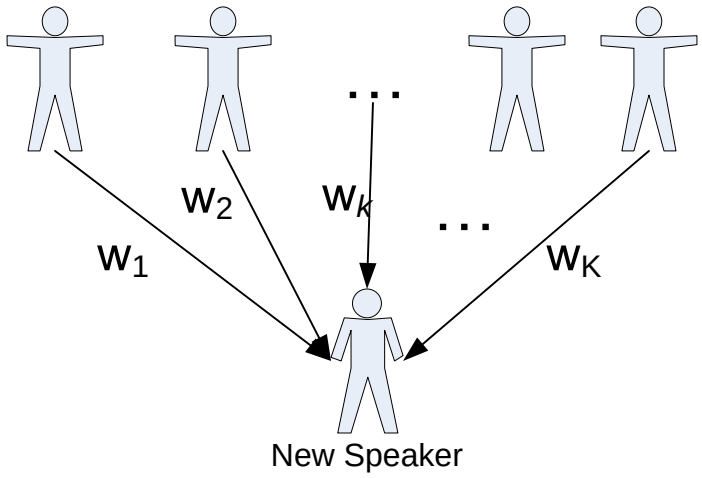

Figure 2.4 Model combination-base speaker adaptation

Each reference speaker is represented by a *supervector* that is composed by concatenating the mean vectors of all acoustic models (Figure 2.5). The ordering of means in the supervector must be the same across all speakers.

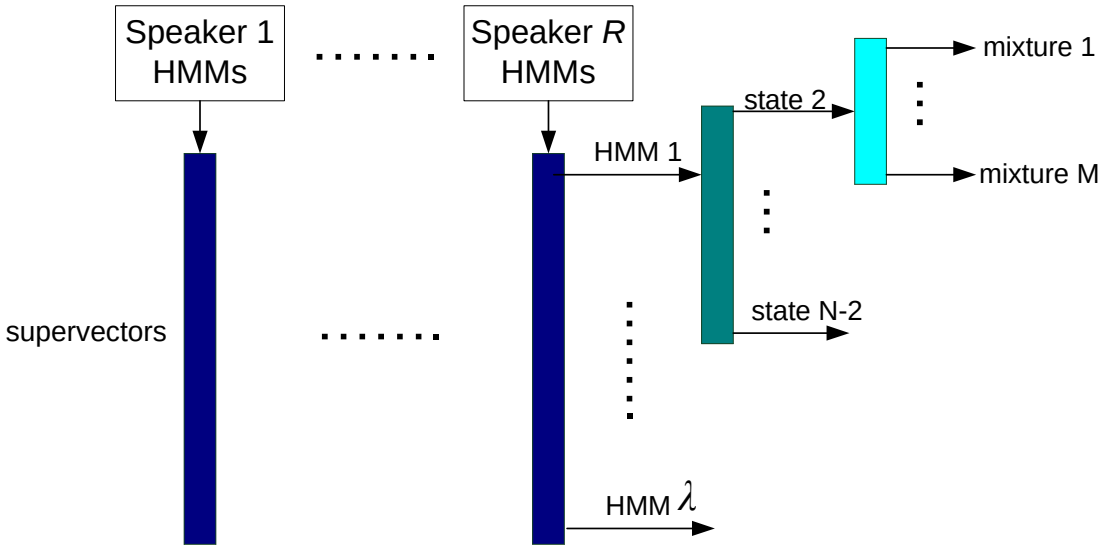

Figure 2.5 Supervector

The mean vector for a new speaker can be estimated by

$$\hat{\mu} = \sum_{k=1}^{K} w_k \mu_k \,, \tag{2.24}$$

where weights w_k are estimated from adaptation data.

Eigenvoice (EV)(Kuhn, 1998; Kuhn *et al.*, 2000) is a model combination-based adaptation technique. The method was motivated by the eigenface approach in face recognition (Turk and Pentland, 1991). The idea is to derive a small set of basis vectors called eigenvoices, considered to represent different voice characteristics (e.g., gender, age, accent, etc.), and each individual speaker is then a point in the eigenspace. Eigenvoice employs principal component analysis (PCA) (Jolliffe, 2002) to find a set of orthogonal basis vectors for this purpose, and these eigenvectors are commonly known as eigenvoices. These eigenvoices may be found by traditional linear PCA, or by nonlinear *kernel* PCA (Scholkopf *et al.*, 1998) using a composite kernel, referred to as kernel eigenvoices (KEV) (Kwok *et al.*, 2003). A new speaker is represented as a linear combination of a few (most important)

eigenvoices.

Eigenvoice adaptation is illustrated in Figure 2.6 and is implemented as follows:

1. Train a set of R speaker-dependent models.
2. For each SD model, concatenate all its mean vectors into a speaker supervector.
3. Perform linear PCA on the supervectors using their correlation matrix.
4. Arrange the eigenvectors in descending order of their eigenvalues and pick the top K eigenvectors; as the required eigenvoices, e_k, $k = 1, 2, ..., K$.
5. Represent a new speaker's supervector, $\hat{\mu}$ by a linear combination of the K chosen eigenvoices:

$$\hat{\mu} = \sum_{k=1}^{K} w_k \cdot e_k \tag{2.25}$$

6. Estimate the eigenvoice weights by maximizing the likelihood of the adaptation data.

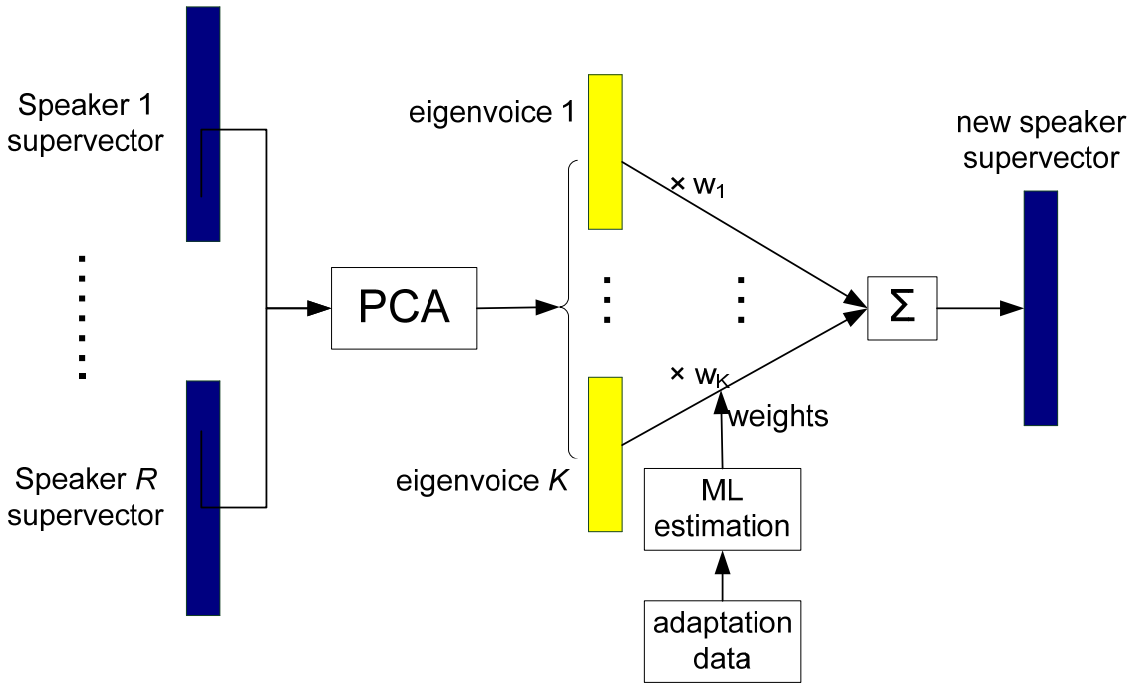

Figure 2.6 Eigenvoice adaptation

In step 6 above, the eigenvoice weights w_k are estimated using the ML algorithm. The auxiliary function is shown as

$$Q_{EV}(\lambda,\hat{\lambda}) = -\frac{1}{2}\sum_{j,m,t}\gamma_t(jm)\left\{\log|\Sigma_{jm}| + \left[o_t - \sum_k w_k e_k(jm)\right]^T \Sigma_{jm}^{-1}\left[o_t - \sum_k w_k e_k(jm)\right]\right\}, \quad (2.26)$$

where $e_k(jm)$ represents the subvector of eigenvoice k corresponding to the mean vector of mixture component m of state j. To maximize Q_{EV} in equation (2.26), set $\frac{\partial Q_{EV}}{\partial w_k} = 0$, $k = 1, 2, ..., K$, then

$$\sum_{j,m,t}\gamma_t(jm)[e_k(jm)]^T \Sigma_{jm}^{-1} o_t = \sum_{s,m,t}\gamma_t(jm)\sum_k w_k[e_k(jm)]^T \Sigma_{jm}^{-1} e_k(jm). \quad (2.27)$$

Equation (2.27) can be rewritten as

$$v = Qw, \quad (2.28)$$

where the left side of the equation is a *K*-dimensional vector,

$$v = \begin{bmatrix} \sum_{j,m,t}\gamma_t(jm)[e_1(jm)]^T \Sigma_{jm}^{-1} o_t \\ \sum_{j,m,t}\gamma_t(jm)[e_2(jm)]^T \Sigma_{jm}^{-1} o_t \\ \vdots \\ \sum_{j,m,t}\gamma_t(jm)[e_K(jm)]^T \Sigma_{jm}^{-1} o_t \end{bmatrix}, \quad (2.29)$$

where Q is a $K \times K$ matrix, a coefficient at *k*th row and *p*th column is denoted as

$$q_{kp} = \sum_{j,m,t}\gamma_t(jm)[e_p(jm)]^T \Sigma_{jm}^{-1} e_k(jm), \quad (2.30)$$

and the weight vector $w = [w_1, w_2, ..., w_K]^T$ is computed by

$$w = Q^{-1}v. \quad (2.31)$$

This estimation process can be iterated until w_k converges.

2.2.4 Speaker Diarization

Speaker adaptation deals with speaker variability when speaker changes occur between training and recognition. Rapid adaptation is required in some applications where speakers turn quite frequently such as conversational telephone speech (CTS) and broadcast news (BN); however, the speaker identities are often difficult to trace in such applications. The task of automatic speaker diarization (ASD) focuses on finding speaker turns. This section highlights the processes of speaker diarization as related to speaker adaptation.

Speaker diarization or speaker segmentation and clustering is a task of marking where speaker changes occur in the detected speech and determining which associated segments of speech coming from the same speaker (Tranter and Douglas, 2006). A typical speaker diarization system is illustrated in Figure 2.7. The framework consists of tasks to perform speech detection, gender and/or bandwidth segmentation, speaker segmentation, and final boundary refinement.

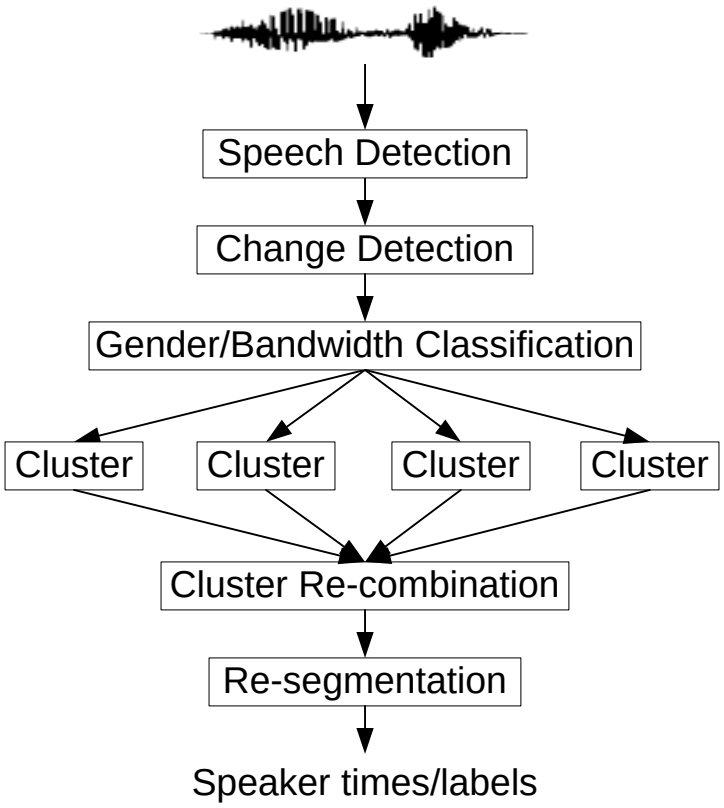

Figure 2.7 Speaker diarization system

Clustering lies at the heart of a speaker diarization system. The goal of the clustering stage is to associate together segments of the same speakers. Ideally, the process results in one cluster for each speaker and segments from a given speaker for one cluster. A typical clustering method utilized in speaker diarization is hierarchical agglomerative clustering with a *Bayesian information criterion* (BIC) (Chen and Gopalakrishnam, 1998), widely used in statistics; or a modification of BIC (Pardo *et al.*, 2007) as a stopping criterion. The clustering stage consists of the following steps:

Create cluster initializations using speech segments

Compute pair-wise distances between clusters

Merge the closest clusters

Update distances of remaining clusters to the new cluster

Repeat steps 2-4 until stopping criterion is met.

The clusters are represented by a single full covariance Gaussian (Nguyen *et al.*, 2002; Moh *et al.*, 2003; Barras *et al.*, 2004; Reynolds and Torres-Carrasquillo, 2004; Sinha *et al.*, 2005) or MAP-adapted GMMs (Moraru *et al.*, 2003; Wooters *et al.*, 2004; Meignier *et al.*, 2005). The distance between the two clusters is measured using ΔBIC analysis. The process is generally stopped when the lowest ΔBIC is higher than a specified threshold.

2.3 Bioacoustics

Bioacoustics, a cross-disciplinary field combining biology and acoustics, usually refers to the research of sound in non-human animals. One major task in bioacoustics is the determination of a species' repertoire of vocalizations (Cleveland and Snowdon, 1982; Berg, 1983; Sjare and Smith, 1986). Normally, this is achieved by analyzing spectrograms of the various vocalizations and then grouping similar sounds into a single call type. Sounds are broken into types based on harmonic structure, pitch contour, whether the vocalization is pulsed, or other criteria. Whenever possible, behavior recorded in conjunction with the vocalization is used to help distinguish between the different types of sounds. Once the basic sound types are identified, a language structure can be hypothesized for those species whose vocalizations consist of a number of different syllables, such as bird or whale song (Clemins, 2005). While there is no substitute for human experts in bioacoustics, automatic animal vocalization classification, adopting modern technologies in engineering areas, especially in speech processing, has made a significant contribution for analyzing animal vocalization (Sczewczyk *et al.*, 2004).

In the last decade, Hidden Markov Models (HMMs) have been successfully applied to animal vocalization classification and detection in a number of species. Kogan and Margoliash (1998) and Anderson *et al.* (1999) have shown that HMM-based classification is more robust to noise and more effective for highly confusable vocalizations than a dynamic time warping (DTW) approach applied to the indigo bunting (*Passerina cyanea*) and zebra finch (*Taeniopygia guttata*). Other species in which HMM-based classification has been investigated include African elephants (*Loxodonta africana*) (Clemins *et al.*, 2005), beluga whale (*Delphinapterus leucas*) (Clemins and Johnson, 2005), ortolan bunting (*Emberiza hortulana* L.) (Trawicki *et al.*, 2005), red deer (*Cervus elaphus*) (Reby *et al.*, 2006), and rhesus macaques (*Macaca mulatta*) (Li *et al.*, 2007). HMM-systems have been widely used to examine vocal repertoire, identify individuals, and classify vocalizations according to social context or behavior.

The above automatic vocalization classification systems are caller-independent (CI), meaning that the vocalization examples used for training the classifier come from a different set of individuals. Previous studies in animal vocalization analysis have found that individual vocal variability is one of the most important cues impacting vocalization related behavior study in bioacoustics (Reby *et al.*, 2006). Individual variability in acoustic structure has been described in many species such as bottlenose dolphins (*Tursiops truncates*) (Parijs *et al.*, 2002; Janik *et al.*, 2006), zebra finches (*Taeniopygia guttata*) (Vignal *et al.*, 2004), and Belding's ground squirrels (*Spermophilus beldingi*) (McCowan and Hooper, 2002). In ortolan buntings, song vocalization has been found to differ significantly between individuals in terms of repertoire content (Osiejuk *et al.*, 2003) and tonality (Osiejuk *et al.*, 2005). Given the similarity to the speaker variability problem between SI and SD recognition systems in human speech, it is

possible to build analogous classification systems for animal vocalizations that are caller-dependent (CD) or caller-adapted (CA). This would imply that a CA system for animal vocalization analysis and classification should yield measurable improvements in overall accuracy and performance. Because both the data collection and analysis/transcription processes are much more difficult and time-consuming for most animal species than for human speech, utilizing a CA system to reduce the overall data requirements for developing automated classification systems may result in significant cost-savings.

C H A P T E R 3 EIGEN-CLUSTERING

In model-combination based speaker adaptation, a set of speaker-dependent models are required as reference speakers to be selected for adapting. Those speaker-dependent models are pre-trained with *a priori* knowledge of the speakers and the utterances associated with each of them in the training data. However, speaker identities are not always available in the real world when there are a large number of different speakers and many switches between them, such as broadcast news. This chapter introduces a new method, eigen-clustering, to adapt the speaker-independent models regardless of speaker identities.

3.1 Motivation

The eigen-clustering method is motivated by principal component analysis (PCA), which is a mathematical tool that transforms a large number of correlated variables into a smaller size of uncorrelated variables called principal components (Jolliffe, 2002). In eigen-clustering, those variables are the utterances in the speaker-independent training set, instead of the reference speakers as in the eigenvoice method, and PCA separates the most uncorrelated utterances. The basic idea here is to map the original model space to the eigenspace, so that each point in eigenspace represents the associated utterance. The mapping shifts and rotates the original axis in the model space to represent as much of the variability among the utterances as possible in the eigenspace. Figure 3.1 illustrates the first principal component for three utterances in a two-dimensional model space to optimally separate the three utterances. The first principal component accounts for the largest variation among the training utterances, and each succeeding component represents as much of the remaining variation as possible. This also implies the meaning of the term "principal"

in PCA, which the lower-order principal components are more important than the higher-order ones. PCA can use these lower-order components in eigenspace to sufficiently describe the characteristics of all utterances in the original model space. This property of PCA makes the calculation more efficient in a dimensionality reduction sense.

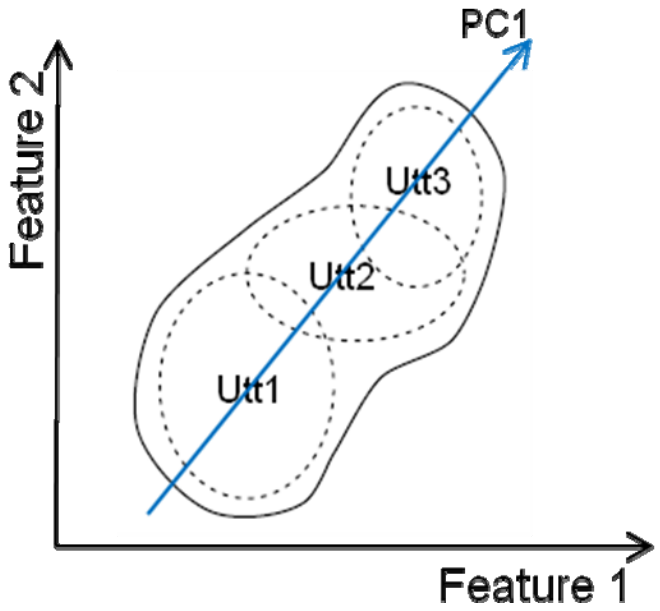

Figure 3.1 The first principal component for three utterances in 2-D model space

3.2 System Design

Assume there are N utterances in a training set, but no knowledge of speaker identities. Suppose each utterance was spoken by one of the P speakers, where $N \geq P$; and P is unknown. The eigen-clustering speaker adaptation is illustrated in Figure 3.2, and the implementation is as follows.

Train an SI model using the N utterances.

Adapt each utterance using MAPLR (MLLR + MAP) to build an acoustic model per utterance.

Construct a mean supervector for each utterance.

Compute the principal components using PCA, and choose the top K principal components ($K < N$) as the eigen-clusters c_k, $k = 1, 2, ..., K$.

Calculate a new speaker's supervector, $\hat{\mu}$, by a linear combination of the K picked clusters:

$$\hat{\mu} = \sum_{k=1}^{K} w_k \cdot c_k , \tag{3.1}$$

where c_k, eigen-cluster weights are estimated by maximizing the likelihood of the adaptation data of the new speaker.

Create the new speaker's acoustic model by substituting the mean vectors in the SI model with the new speaker's supervector.

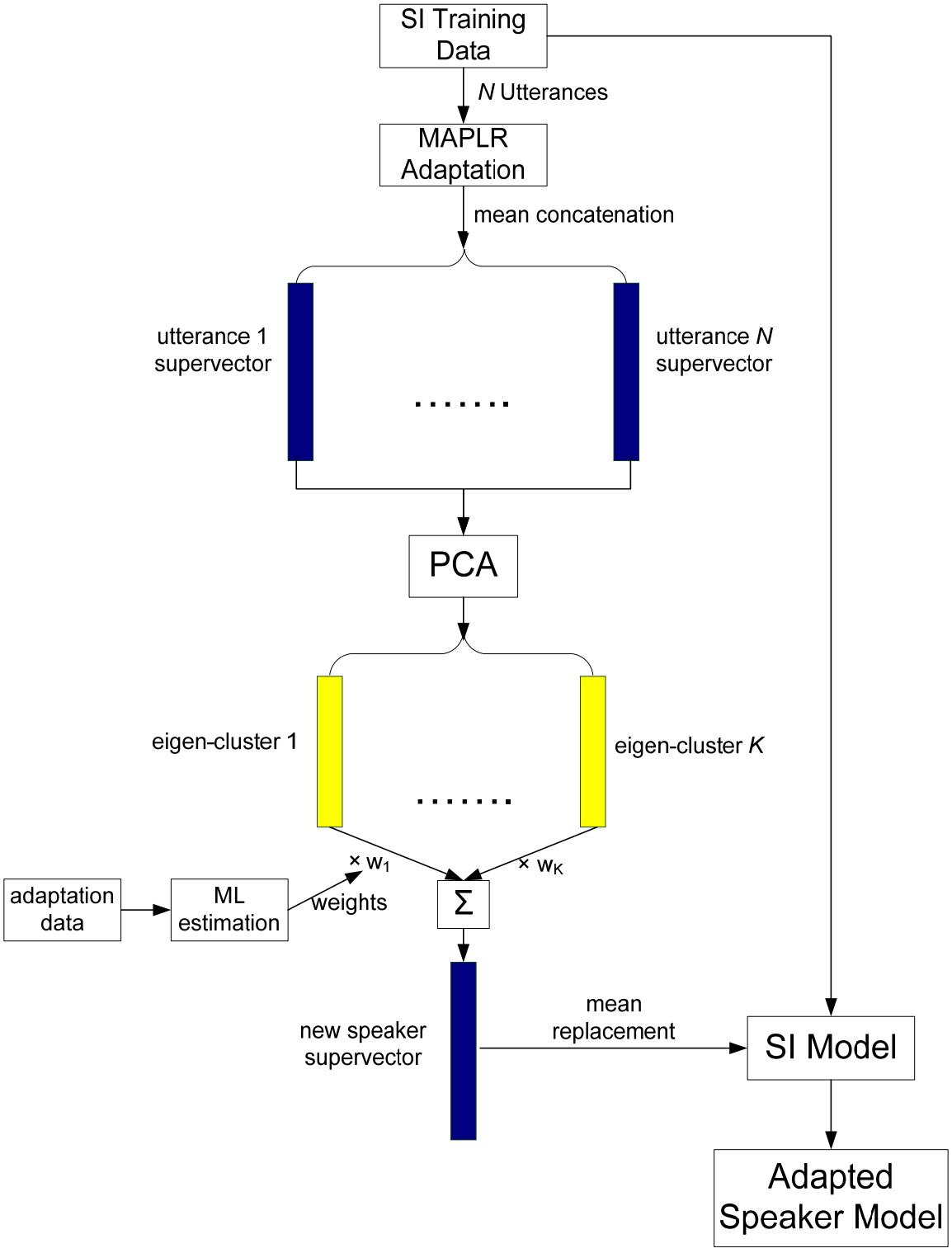

Figure 3.2 Eigen-clustering speaker adaptation

The MAPLR (Chesta *et al.*, 1999; Chou, 1999) is an approach to combine both the MLLR and MAP methods in step 2. Comparing Sections 2.2.1 and 2.2.2, the MLLR has a faster adaptation in terms of the amount of adaptation data than the MAP, whereas the MAP becomes more accurate with the increased amount of adaptation data. The MAPLR integrates the advantages of both the MLLR and MAP utilizing MLLR as "a prior" in MAP adaptation. By substituting the prior mean vector μ_{jm} in equation (2.21) with the equation (2.22), the mean vector of MAPLR is estimated as

$$\hat{\mu}_{jm} = \frac{\tau_{jm}(A\mu_{jm} + b) + \sum_t \gamma_t(jm) o_t}{\tau_{jm} + \sum_t \gamma_t(jm)}. \tag{3.2}$$

An acoustic model is then adapted using the smallest data possible in the speaker-independent training set, only one utterance. Although MAPLR is not for rapid adaptation in terms of a few seconds of adaptation data, it updates all the means (variances are not included because the supervectors are built on the means only) in the HMM towards the model. One advantage of using MAPLR adaptation for each of the utterances is to force the mean vectors of each utterance model into the same order, which guarantees that no mean vectors will be misplaced in concatenating to supervectors.

After all supervectors are constructed in step 2, they are grouped into a $D \times N$ supermatrix X, where the columns are N supervectors with each of dimension D. Then an $N \times N$ covariance matrix C (3.3) is calculated from the supermatrix X as

$$C = E[(X - \bar{X})(X - \bar{X})^T], \tag{3.3}$$

where E is the expectation operator, and each column of $\bar{X}$ is the vector obtained by averaging over all N supervectors. Each covariance coefficient $C(i, j)$, the *i*th row and *j*th column in the covariance matrix C, indicates the degree of statistical dependence between

the i and j pair of the supervectors. Given the covariance matrix C, the PCA transformation of X is

$$Y = V^T X, \tag{3.4}$$

which obtaining the data matrix Y in a new coordinate eigenspace defined by the eigenvectors that are the rows of the matrix V^T.

V^T can be computed by singular value decomposition (SVD), which is one the most common implementation of PCA in computer programs. Given the covariance matrix C, the SVD form is

$$C = \underbrace{\begin{bmatrix} | & & | & & | \\ \vec{u}_1 & \cdots & \vec{u}_k & \cdots & \vec{u}_D \\ | & & | & & | \end{bmatrix}_{D\times D}}_{U} \underbrace{\begin{bmatrix} \lambda_1 & & & & \\ & \ddots & & & \\ & & \lambda_k & & \\ & & & \ddots & \\ & & & & \lambda_D \end{bmatrix}_{diag}}_{S} \underbrace{\begin{bmatrix} | & & | \\ \vec{v}_1 & & \\ \vdots & & \\ \vec{v}_k & & \\ \vdots & & \\ \vec{v}_D & & \\ | & & | \end{bmatrix}_{D\times D}}_{V^T}, \tag{3.5}$$

where the rows of V form a set of orthonormal eigenvectors, and the diagonal matrix S contains the eigenvalues in descending order. The first k principal components are chosen from the first k rows of V.

Although PCA is defined in terms of the covariance matrix, the correlation matrix has also been widely used. The magnitude of the covariance matrix C depends on the standard deviations of the each pair of supervectors. The correlation matrix R has the same dimension as the covariance matrix C. The correlation coefficient, the component in the correlation matrix R, is defined as

$$R(i,j) = \frac{C(i,j)}{\sigma(i,j)}, \tag{3.6}$$

where $\sigma(i,j)$, the *i*th row and *j*th column component of the standard deviation matrix σ, is

calculated by

$$\sigma(i,j) = \sqrt{[(E[X^2] - \bar{X}^2)(E[X^2] - \bar{X}^2)^T](i,j)}\,. \tag{3.7}$$

The correlation coefficient $R(i,j)$ is a normalized covariance coefficient scaling the degree of dependence between the *i* and *j* pair of the supervectors on the interval from -1 to +1.

In step 5, the eigen-cluster weights w_k are estimated using the same algorithm (maximum likelihood) as in the eigenvoice method. The auxiliary function of eigen-cluster weights is

$$Q_{EC}(\lambda, \hat{\lambda}) = -\frac{1}{2}\sum_{j,m,t}\gamma_t(jm)\left\{\log|\Sigma_{jm}| + \left[o_t - \sum_k w_k c_k(jm)\right]^T \Sigma_{jm}^{-1}\left[o_t - \sum_k w_k c_k(jm)\right]\right\}, \tag{3.8}$$

where $c_k(jm)$ represents the subvector of eigen-cluster k corresponding to the mean vector of mixture component m of state j. After maximization of Q_{EC} in equation (3.8) by setting $\frac{\partial Q_{EC}}{\partial w_k} = 0$, the re-arranged equation becomes

$$\sum_{j,m,t}\gamma_t(jm)[c_k(jm)]^T \Sigma_{jm}^{-1} o_t = \sum_{s,m,t}\gamma_t(jm)\sum_k w_k[c_k(jm)]^T \Sigma_{jm}^{-1} c_k(jm)\,. \tag{3.9}$$

Equation (3.9) can be written in the format

$$v = Qw\,, \tag{3.10}$$

where w contains K coefficients of the eigen-cluster model needed to be estimated

$$w = [w_1, w_2, ..., w_K]^T\,. \tag{3.11}$$

The vector v on left hand side of the equation is a $K \times 1$ vector,

$$v = \begin{bmatrix} \sum_{j,m,t}\gamma_t(jm)[c_1(jm)]^T \Sigma_{jm}^{-1} o_t \\ \sum_{j,m,t}\gamma_t(jm)[c_2(jm)]^T \Sigma_{jm}^{-1} o_t \\ \vdots \\ \sum_{j,m,t}\gamma_t(jm)[c_K(jm)]^T \Sigma_{jm}^{-1} o_t \end{bmatrix}; \tag{3.12}$$

and Q in the right hand side is a $K \times K$ matrix with a coefficient at the *k*th row and *p*th column determined by

$$q_{kp} = \sum_{j,m,t} \gamma_t(jm)[c_p(jm)]^T \Sigma_{jm}^{-1} c_k(jm). \tag{3.13}$$

The eigen-cluster weight vector w is calculated by

$$w = Q^{-1}v. \tag{3.14}$$

The weight vector can be estimated iteratively for better convergence. According to the EM algorithm in Section 2.1.2, maximizing the Q_{EC} function in equation (3.8) is equivalent to maximizing the likelihood $P(O|\lambda)$ in equation (2.2) for each iteration. The new HMM parameter vector $\hat{\lambda}$ is derived from the parameter vector λ in the previous iteration, the process guarantees a monotonic likelihood improvement on each iteration, and eventually the likelihood converges to a local maximum (Dempster *et al.*, 1977). The EM algorithm for the eigen-clustering can be described as follows.

Step 1. Initialization: HMM parameter vector λ is obtained using MAPLR for each utterance.

Step 2. E-step: Compute auxiliary function $Q_{EC}(\lambda, \hat{\lambda})$ in equation (3.8) based on λ.

Step 3. M-step: Compute $\hat{\lambda}$ according to the re-estimation equations (3.12) - (3.14) to maximize the auxiliary Q_{EC} function.

Step 4. Iteration: Set $\lambda = \hat{\lambda}$, repeat from step 2 until convergence.

The aim of the eigen-clustering adaptation is to enable model-combination based rapid speaker adaptation without explicit speaker knowledge on speaker-dependent models, opposite of its counterpart, eigenvoice adaptation. The eigen-clustering method has the

same adaptation requirements (e.g., speaker-independent training data only) as any generic speech recognition task, or basic adaptation such as MLLR.

In addition to finding the principal components to represent the most diverse acoustic characteristics of the HMMs, eigen-clustering plays the role of clustering speakers, which is unnecessary in the eigenvoice method. Although both eigen-clustering and eigenvoice use PCA, their meanings in the eigenspace are totally different. In the eigenspace, each point of eigen-clustering represents the projection of the associated utterance in the model (feature) space, whereas each point of eigenvoice is the map of the corresponding speaker from the model space.

The idea of eigen-clustering can be definitely tied to the unsupervised speaker clustering task, which tries to find the identity of speakers from the unlabeled data. Comparing to the strict speaker clustering task, eigen-clustering has a looser requirement, since speaker adaptation does not need explicit identity labeling. The estimate of the number of speakers is implicitly carried out by PCA, which is a dimensional reduction technique. The concept of PCA, which keeps the lower-order eigenvectors and throws away the higher-order ones, is a key in eigen-clustering to reduce computational cost in the following model training step.

Motivated by the MAPLR adaptation method, it is possible to combine eigen-clustering with the MAP method. Although MAPLR combines the advantages of both the MLLR and MAP, it is still not rapid adaptation on the scale of a few seconds. By using eigen-clustering prior to MAP adaptation, it would be possible to both obtain rapid adaptation and still allow continually increasing accuracy as adaptation data increases. To implement this, the mean vector of the eigen-clustering can be combined with MAP by substituting the prior mean vector μ_{jm} in equation (2.21) with the equation (3.1),

$$\hat{\mu}_{jm} = \frac{\tau_{jm}\hat{\mu}_{jm}^{EC} + \sum_t \gamma_t(jm) o_t}{\tau_{jm} + \sum_t \gamma_t(jm)}, \tag{3.15}$$

where EC denotes the mean calculated using the eigen-clustering.

C H A P T E R 4 EXPERIMENTS

In this chapter, experiments concerning rapid speaker adaptation with adaptation data of less than a minute are presented. The experimental evaluation consists of two different task domains. The first is conducted on a medium vocabulary human speech recognition task, while the second is conducted on an animal vocalization classification task. The former is to show the new eigen-clustering method useful in human speech technology, and the latter is to demonstrate the contribution to a more specialized applied task in bioacoustics.

The experimental setups, including the training, evaluation and adaptation datasets, extracting the acoustic features, building of the acoustic models, are fundamentally the same for both the aspects. The adaptation is done offline as all adaptation data is available, and in a supervised mode as all data is transcribed by human experts. The comparisons between the proposed and other state of the art adaptation techniques are illustrated.

4.1 Implementation

Implementation of the new eigen-clustering speaker adaptation technique was done using the HTK library framework (Young *et al.*, 2002). The eigenvoice adaptation method for comparison was implemented by modifying HTK. Existing HTK tools were used to evaluate MLLR, MAP, and MAPLR techniques. The code for SVD, which is used to implement PCA, is modified from the LAPACK routines (Anderson *et al.*, 1999).

The software has three layers with different programming languages. The core layer consists of the modified HTK code, in C. The middle layer used to capsulate HTK tools, options and configuration parameters is done in Perl. And the top layer is a scripting layer in Matlab.

4.2 Human Speech

4.2.1 Data Corpus

The experiments for human speech used the DARPA 1000-word Resource Management continuous speech database part 1 (RM1) (Price *et al.*, 1993) with a variety of American English dialects in reading sentences. RM1 includes three portions, speaker-independent (SI) training data, speaker-dependent (SD) training data, and development and evaluation data. The data was recorded at 16 kHz, with 16-bit resolution, using a Sennheiser HMD-414 headset microphone.

The speaker-independent training set contains 3990 utterances from 109 speakers, a combination of the 72 training set speakers and the 37 development set speakers. Since this research emphasizes speaker adaptation, the speaker-independent evaluation set is used for all experiments here. The speaker-dependent set consists of 12 speakers, each having a set of 600 utterances for speaker-dependent training, 10 utterances (average 3 seconds per utterance) for rapid adaptation task, and 100 utterances for evaluation. Table 4.1 shows the detailed descriptive statistics of the data setup for training, testing and rapid adapting.

	SI Training Set	**SD Training Set**	**Test Set**	**Rapid Adaptation Set**
Number of Speakers	109	12	12	12
Number of Utterances	3990	600×12	100×12	10×12
Mean Utterances/Speaker	36.6	600	100	10

Table 4.1 Distribution of the number of speakers, utterances, and the average utterances per speaker for training, test and rapid adaptation sets

4.2.2 Acoustic Modeling and Feature Extraction

Three different acoustic models were built and tested: single Gaussian monophones, 4-mixture Gaussian context-independent monophones, and single Gaussian context-dependent triphones. Each phoneme model was a left-to-right 3-state hidden Markov model (HMM). In addition, there was a 1-state short pause model and a 3-state silence model.

The acoustic feature vector has 39 components, consisting of 12 mel-frequency cepstral coefficients (MFCCs) (Huang *et al.*, 2001) and the normalized log energy along with their first and second order derivations, extracted from each utterance using 25ms Hamming windows with a 10ms step size.

4.2.3 Experimental Procedure

For each of three acoustic models in Section 4.2.2, a baseline speaker-independent system (Figure 4.1) was constructed by pooling the 3990 utterances for all 109 speakers in the SI training set; and 12 speaker-dependent models (Figure 4.2) were built using the 600 utterances associated each of the 12 speakers in the SD training set.

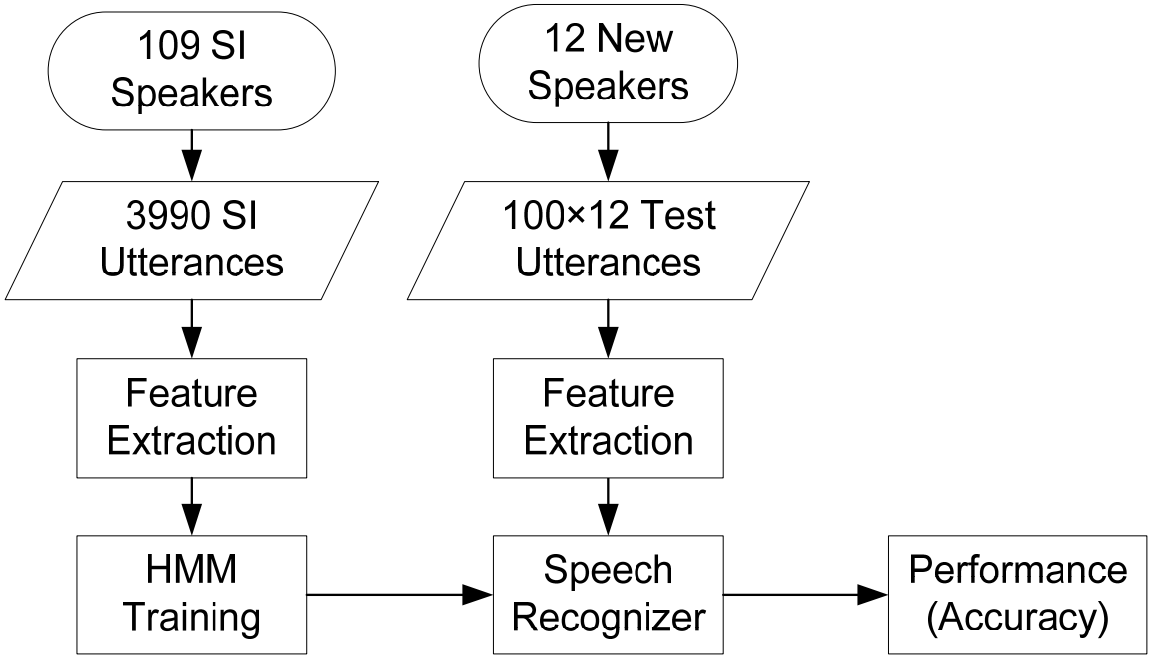

Figure 4.1 Speaker-independent system for RM dataset

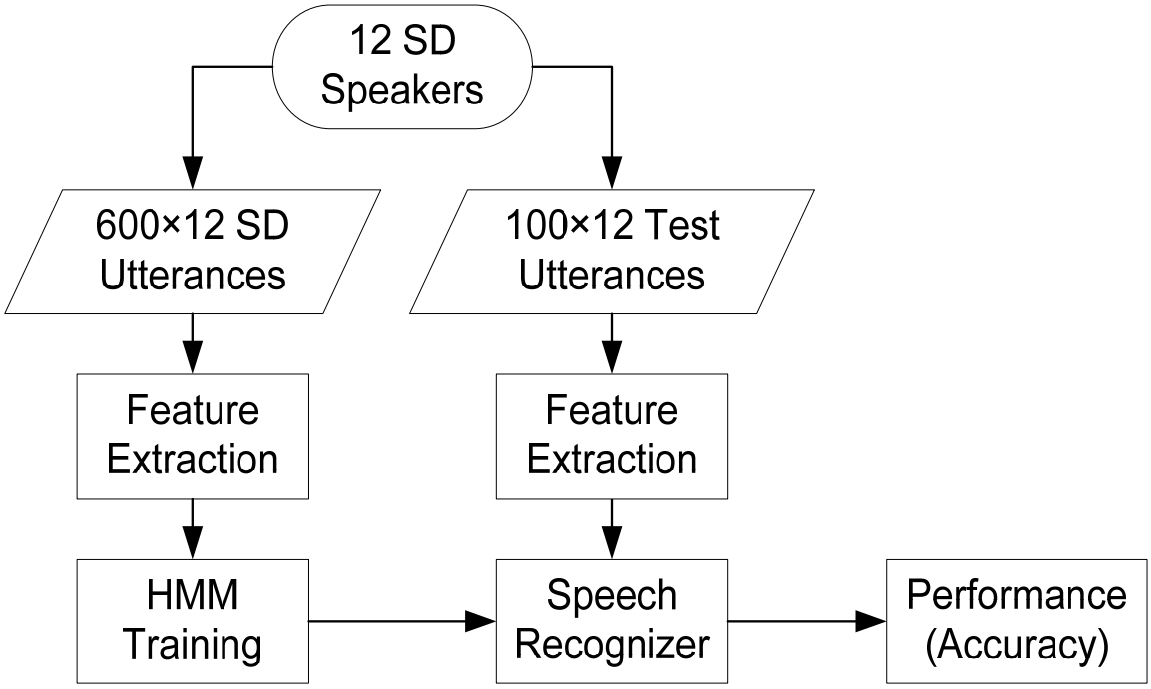

Figure 4.2 Speaker-dependent system for RM dataset

The speaker-adapted systems (Figure 4.3) using MLLR, MAP, and MAPLR respectively

were implemented on the SI model, with the 12 subjects in SD data set treated as new speakers. For better performance, all three methods included both mean and variance adaptation.

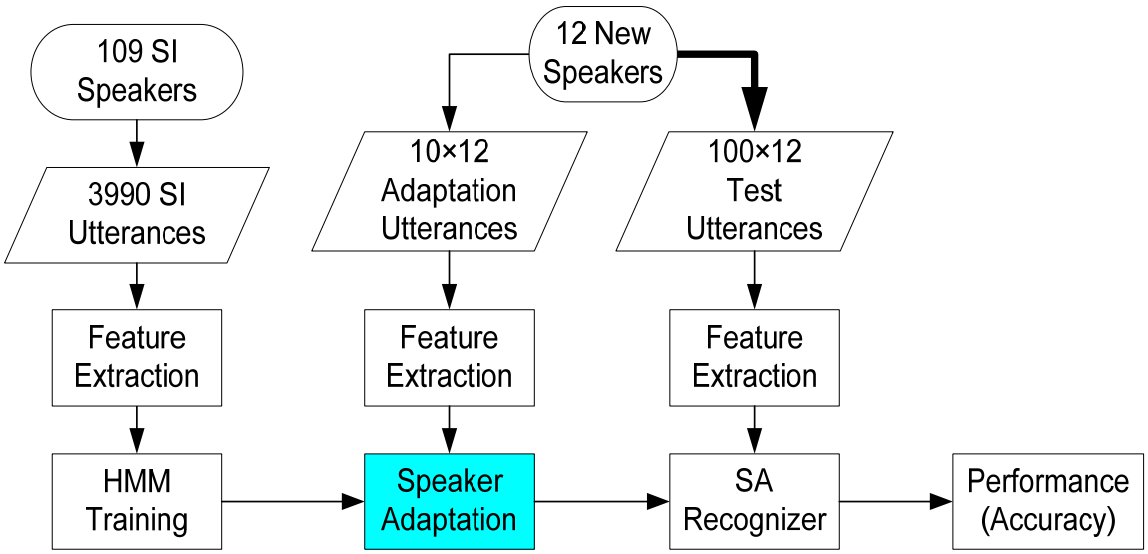

Figure 4.3 Speaker-adapted system for RM data

To investigate the properties of the eigenvoice adaptation, additional speaker-dependent models were required. From the SI model, 109 "SD" models were obtained from the SI training set using MAPLR adaptation. The 109 models were not trained in a traditional way with a large amount of data, e.g., 600 utterances, because 30 – 40 utterances for each speaker were not sufficient. However, that amount of data was enough to build a sufficiently well-trained system using adaptation. The 109 models were treated as "SD" for this case.

For the eigen-clustering adaptation, 3990 models were estimated for all utterances in the SI training set using MAPLR adaptation. Rather than building a model close to the SD one as in the eigenvoice approach, the main objective for using the MAPLR method here is to align the all phoneme mean vectors in the same order for concatenating to supervectors later

on.

For each of 12 speakers in the SD data set, 10 sets of adaptation data ranging from 1 to 10 utterances of the rapid adaptation set (e.g., set 1 has 1 utterance, set 2 has 2 utterances, etc.), were randomly selected. To observe the properties of MAP, MLLR and MAPLR methods with large amount of adaptation data, all 600 utterances from each of 12 speakers were used in 7 sets including 50, and 100 – 600 in increments of 100 utterances. All of the above adapted models were tested on the 100 evaluation utterances in the SD data set using a word-pair grammar (Derr and Schwartz, 1989). The final results are the averages of experiments over all 12 speakers. All adaptation techniques were in supervised mode, which means the correct transcriptions for adaptation utterances were used during adaptation process. PCA implementations on the both correlation and covariance matrix were compared in the experiments for both eigenvoice and eigen-clustering adaptation.

4.2.4 Single Gaussian Monophone Models

The MLLR, MAP and MAPLR adaptation results are shown in Figure 4.4. Given the non-linearity of number of utterances (horizontal axis), the ranges for rapid adaptation (fewer than 10 utterances) and normal adaptation are separated by the vertical green bar. The baseline SI system with single Gaussian monophone HMMs has an accuracy of 76.1%, and the SD system has an accuracy of 89.2%. All three adaptation methods improve the accuracy slightly over the baseline with only one adaptation utterance (average utterance length is 3-second). The MAP adaptation shows consistent improvement with incrementing adaptation data size. MAPLR performs similar to MLLR because it uses the MLLR adapted model as *a prior* in MAP. Both MLLR and MAPLR have degraded performance to the

baseline SI system in the range of 2 – 5 adaptation utterances, because the insufficient adaptation data makes the estimation of the transformation matrix W in equation (2.22) at each regression class inaccurate (Leggetter and Woodland, 1995). All three methods have very similar accuracies near 82% using the full size rapid adaptation data. The MAP becomes more accurate than both the MLLR and MAPLR when the amount of adaptation data increases to 100 utterances per speaker because all the model parameters are updated with the MAP. The MAP approaches the accuracy of the SD system faster than MAPLR with the full 600 utterances of adaptation data, while the MLLR becomes saturated. The properties of all three adaptation methods can be observed in both the rapid adaptation (10 utterances) and normal adaptation (up to 600 utterances) in Figure 4.4. This research emphasizes on the rapid speaker adaptation, so only up to 10 utterances of adaptation data is used for the rest of experiments in this section.

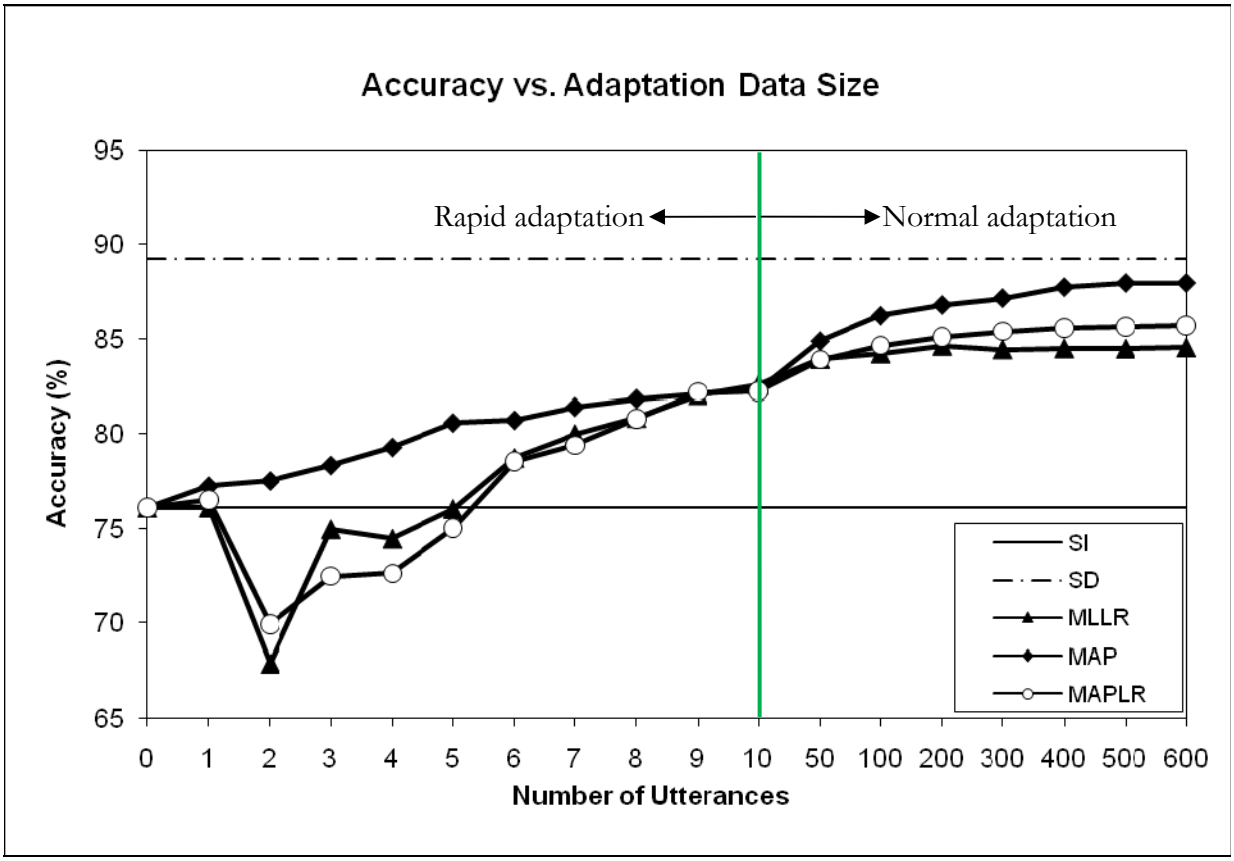

Figure 4.4 Performance of SI, SD, MLLR, MAP and MAPLR adaptation

The recognition accuracies for eigenvoice adaptation using PCA derived, from both correlation and covariance matrices, as a function of adaptation data size and eigenspace dimensions (number of eigenvoices) are compared in Figure 4.5. The number of eigenvoices includes 1 to 15, 20 – 100 in increments of 10; as well as all 109 reference speakers. These same results are also displayed in Table A.1 and Table A.2 respectively. The accuracy range is from 76.3% to 79.2% for the PCA correlation implementation, and from 79% to 82% for the covariance implementation. The covariance implementation shows clearly higher accuracies over the correlation approach at almost every point. The accuracy surfaces of the both implementations are relative flat with increasing number of adaptation utterances, plateauing after 2 – 3 utterances.

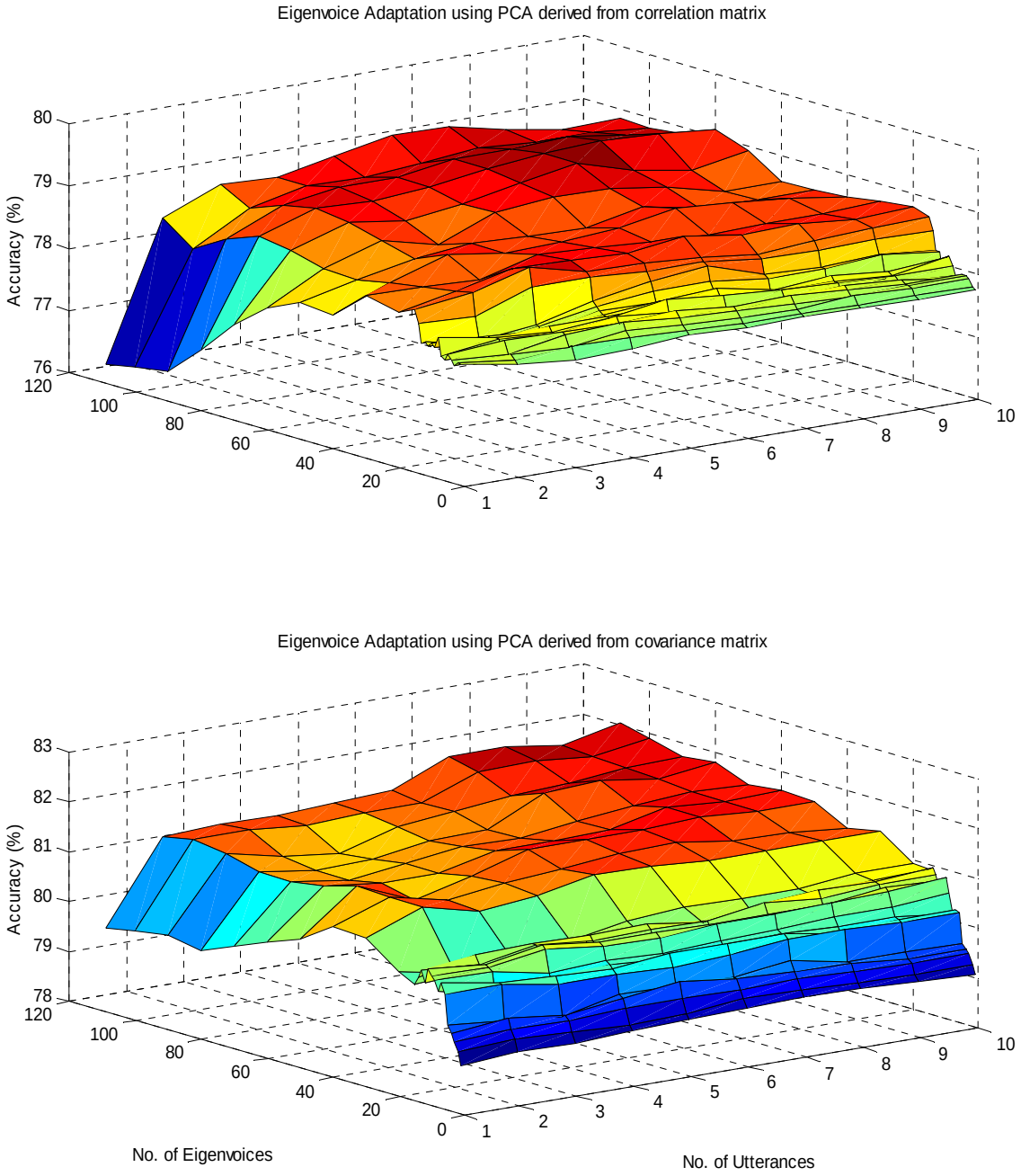

Figure 4.5 Eigenvoice performance comparison on PCA correlation matrix vs. covariance matrix

Similarly, the proposed eigen-clustering adaptation evaluation results are shown in Figure 4.6. The number and the scale of eigen-clusters are the same as in the experiments for the eigenvoice adaptation for comparison purposes. The largest number of eigen-clusters in the experiment is 109 to equal the total number of reference speakers. The number of selected clusters would be unlikely to be so high in practice, because one of properties of PCA is to reduce data dimensions. The PCA correlation approach in Table A.3 gives the recognition accuracies in a range of 72.3% to 79.3%, while the covariance one in Table A.4 shows the results from 70.3% to 80.9%. As before, the covariance implementation has overall better performance than the correlation one.

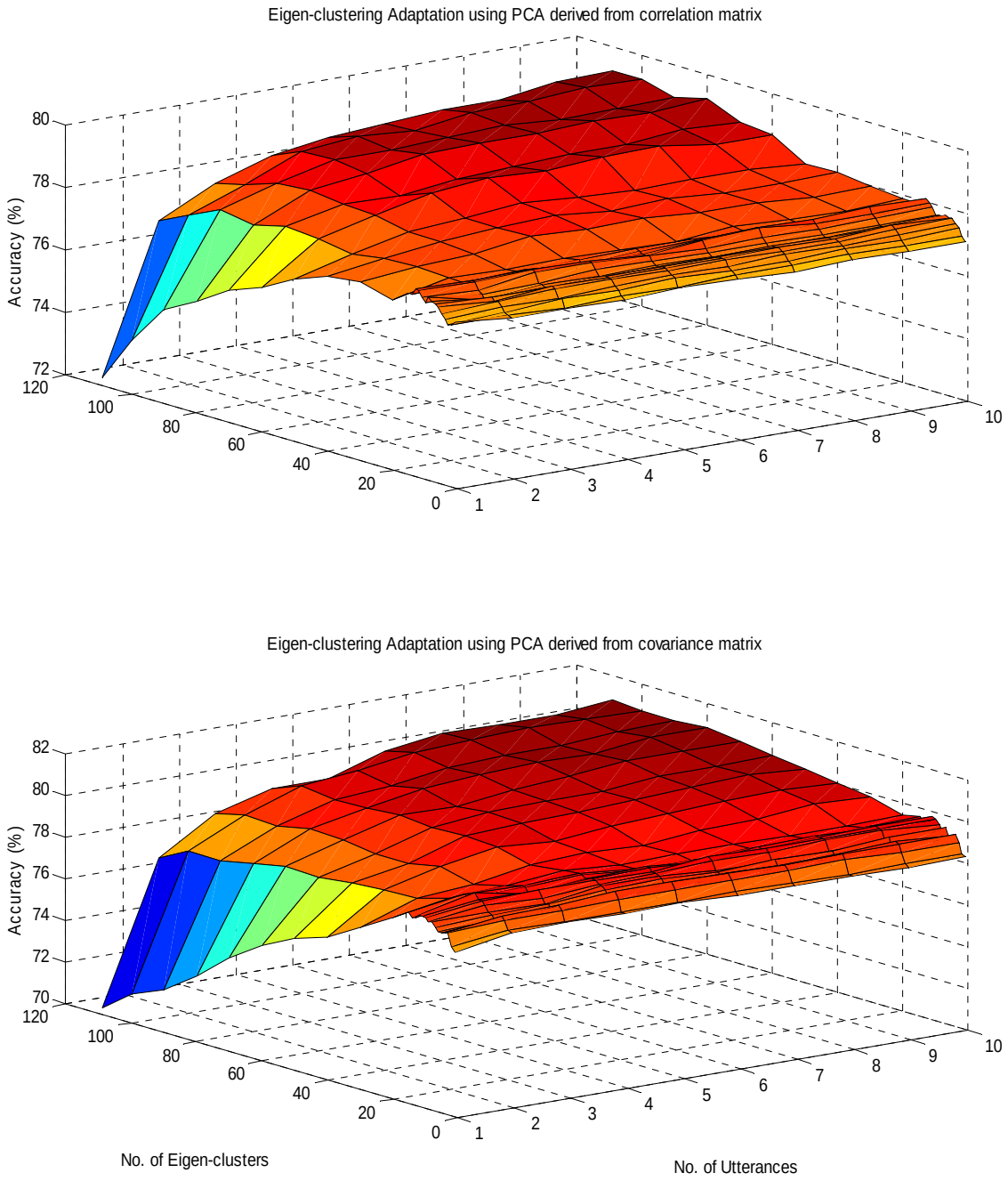

Figure 4.6 Eigen-clustering performance comparison on PCA correlation matrix vs. covariance matrix

Figure 4.7 shows the eigen-clustering (EC) and eigenvoice (EV) adaptations implemented by PCA on the covariance matrix, with the numbers of clusters/voices selected as 5, 10, 15, 20, and 30 and appended after EC/EV. Under the same amount of adaptation data, the overall performance of eigenvoice adaptation (blue lines in Figure 4.7) is better than the eigen-clustering method (green lines in Figure 4.7) by about 2%. This shows that the knowledge of the reference speakers available in the eigenvoice method gives more explicit acoustic characteristics than the reference utterances.

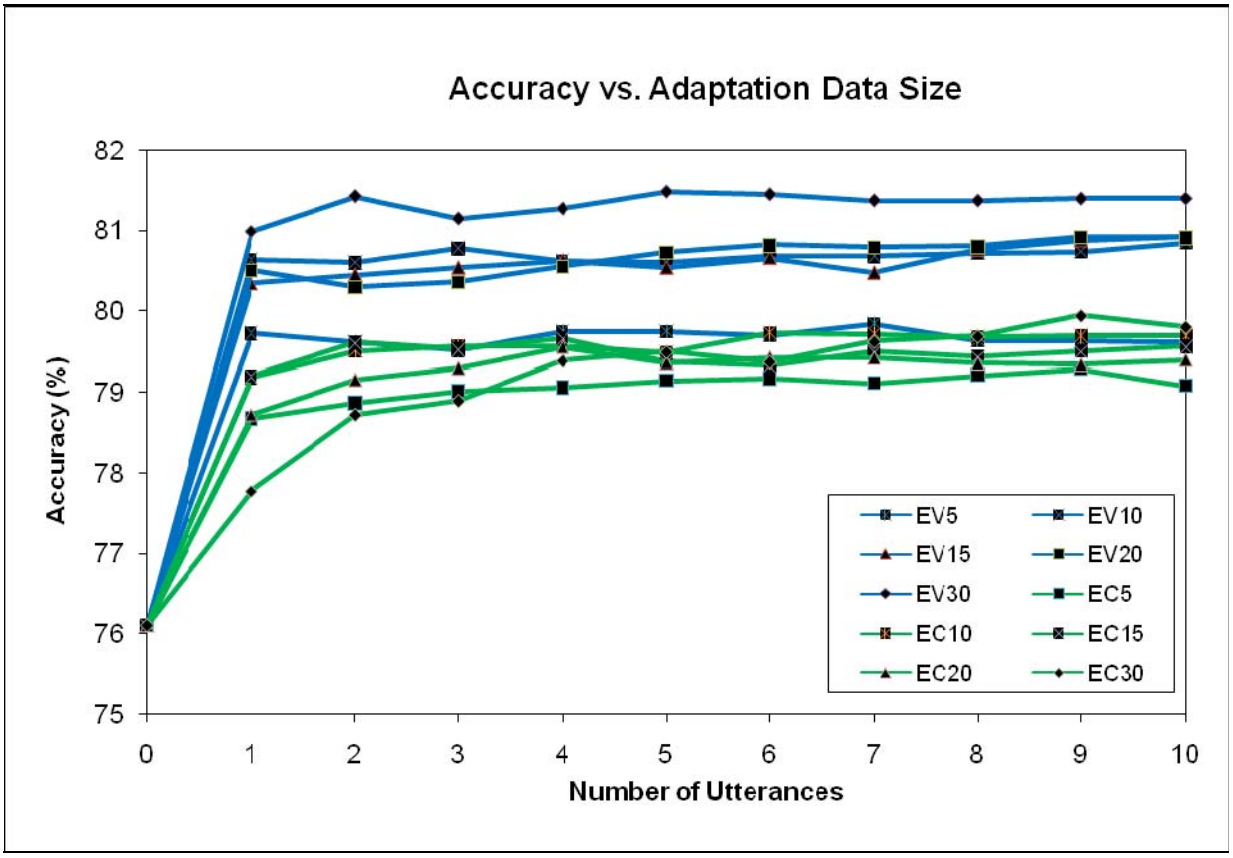

Figure 4.7 Performance of Eigenvoic and Eigen-clustering adaptation

A comparison of eigen-clustering with 30 clusters (EC30) and eigenvoice with 30 voices (EV30) to the other three adaptation methods is shown in Figure 4.8. Both eigen-clustering

and eigenvoice methods outperform the other three when 4 or fewer adaptation utterances (about 12 seconds) are used. The eigen-clustering method gives higher accuracies than MLLR and MAPLR with up to 6 utterances (about 18 seconds), whereas eigenvoice shows better results than MLLR and MAPLR with about 7 utterances (21 seconds). Note that eigenvoice is shown for reference but is not directly comparable since the eigenvoice method requires the speaker identities but none of the other methods do, including eigen-clustering. When eigen-clustering is combined with MAP (red line in Figure 4.8, EC30+MAP, shows the eigen-clustering with 30 clusters as the prior in MAP), the performance is not only better than the other three in the range of 1 – 9 utterances but also shows the potential in consistently improving when the amount of adaptation data increases. Table A.5 shows the detailed accuracies in number in Figure 4.7 and Figure 4.8.

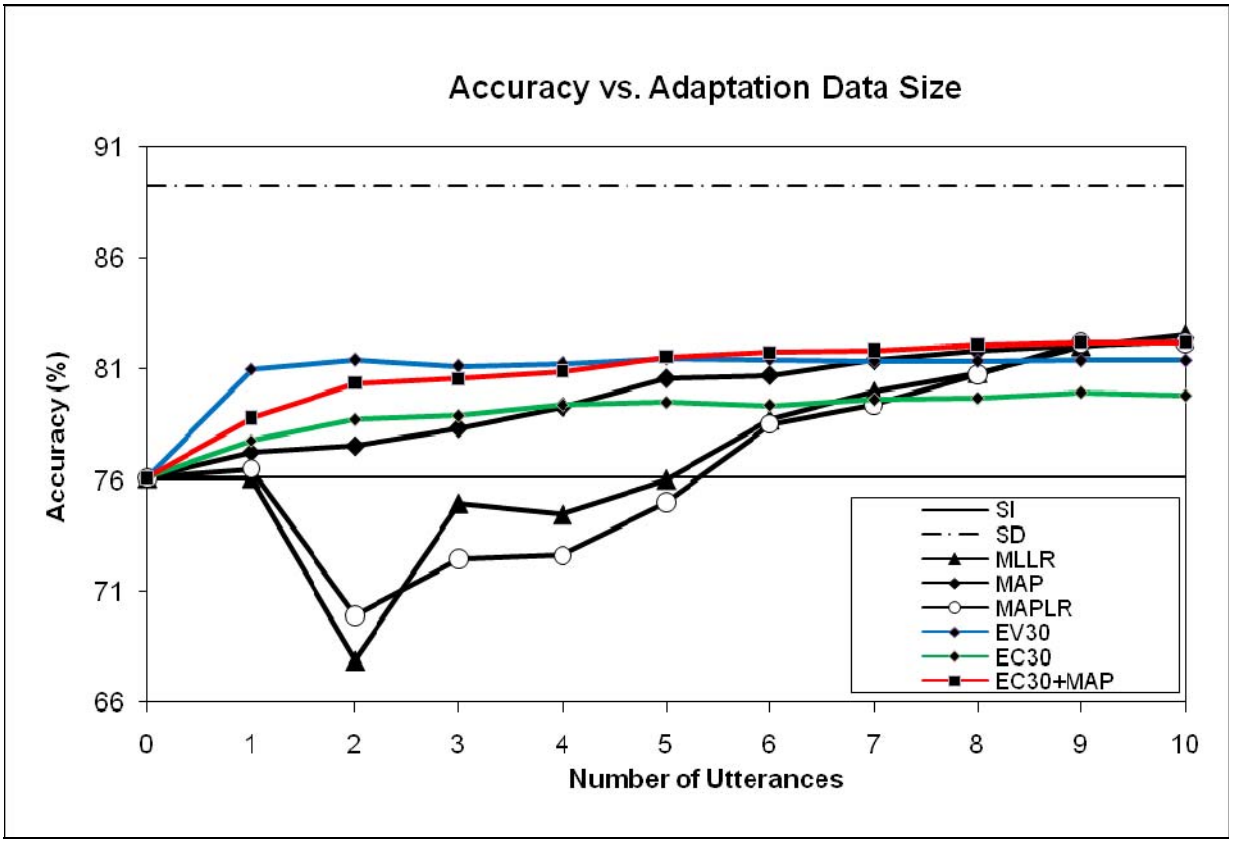

Figure 4.8 Performance of six adaptation methods

4.2.5 Four-Mixture Gaussian Monophone Models

This section demonstrates the performance of the proposed eigen-clustering method for 4-mixture Gaussian monophone HMMs.

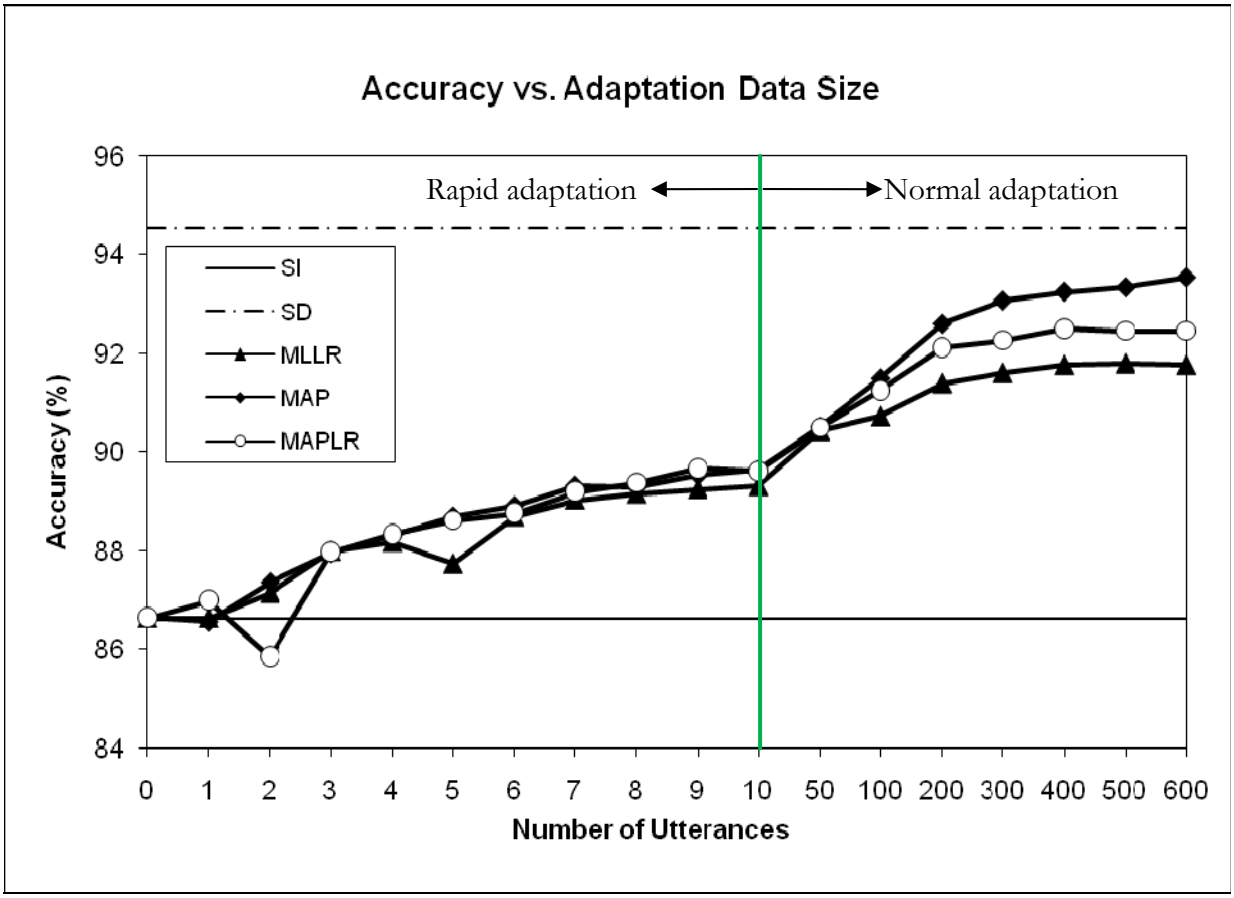

Figure 4.9 Performance of SI, SD, MLLR, MAP and MAPLR adaptation

The performance of the three adaptation methods MLLR, MAP and MAPLR along with SI and SD systems is shown in Figure 4.9. The ranges for rapid adaptation and normal adaptation are separated by the vertical green bar using non-linear scale of number of utterances (horizontal axis). Comparing to the performance of single Gaussian monophone HMMs in Figure 4.4, the 4-mixture Gaussian HMMs show an absolute improvement of 10% over the baseline SI and 5% over the SD system. All three adaptation methods improve accuracy slightly over the baseline when only one adaptation utterance is used. The MAP adaptation shows the improvement more consistently than the others with incrementing adaptation data size. All three methods have very close accuracies when there are more than

6 utterances (about 18 seconds) for adapting. The highest accuracy from the three methods with all 10 utterances still does not approach that of the SD system. In normal adaptation data in a range of 50 to 600 utterances per speaker, all three methods show the same manner of performance as in the single Gaussian monophone, where MAP outperforms over the other two when more than 200 utterances of adaptation data are used, and approaches the accuracy of the SD system faster than MAPLR, while MLLR becomes saturated with the full 600 utterances.

The performance of eigenvoice adaptation using PCA derived from both correlation and covariance matrices as a function of adaptation utterances and eigenspace dimensions (number of eigenvoices) are compared in Figure 4.10. In the same manner as in Section 4.2.4 for single Gaussian monophone HMMs, the number of eigenvoices is chosen from 1 to 15, 20 – 100 in increments of 10, as well as for all 109 reference speakers. The detailed results are also displayed in Table A.6 and Table A.7 respectively. The accuracy range is from 86.7% to 87.6% for the PCA correlation implementation, and from 87.5% to 89.3% for the covariance implementation. The covariance implementation shows clearly higher accuracies over the correlation approach at almost every point. The accuracy surfaces of the both implementations are relatively flat with increasing number of adaptation utterances, plateauing after 2 – 3 utterances.

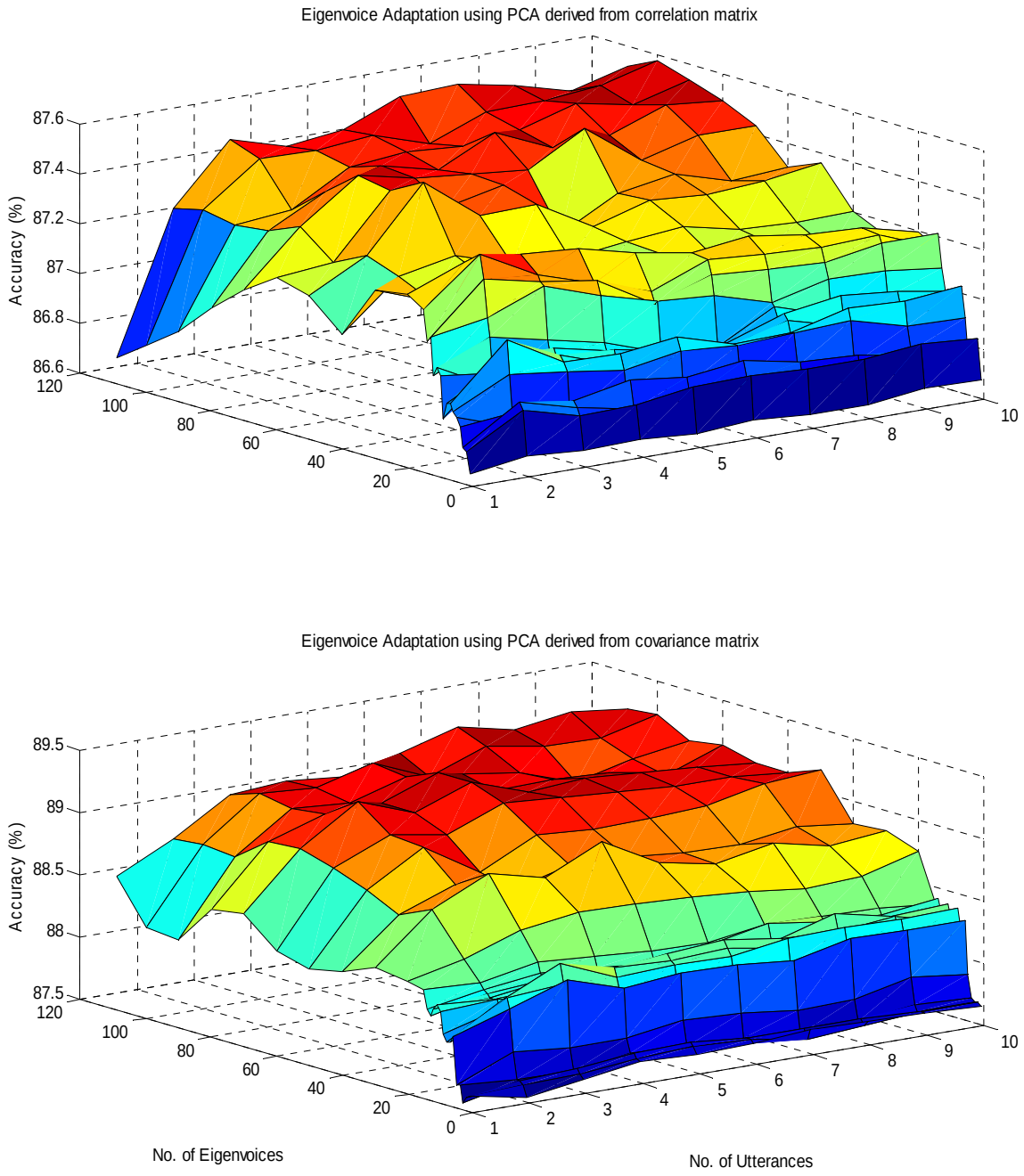

Figure 4.10 Eigenvoice performance comparison on PCA correlation matrix vs. covariance matrix

For the proposed method, eigen-clustering adaptation evaluation results are shown in Figure 4.11. The number and the scale of eigen-clusters are in the same setup as in the experiments for the single Gaussian HMMs. The PCA correlation approach in Table A.8 gives the recognition accuracies in a range of 86.3% to 86.7%, while the covariance one in Table A.9 shows the results from 86.1% to 88%. The covariance implementation has overall better performance than the correlation one as before.

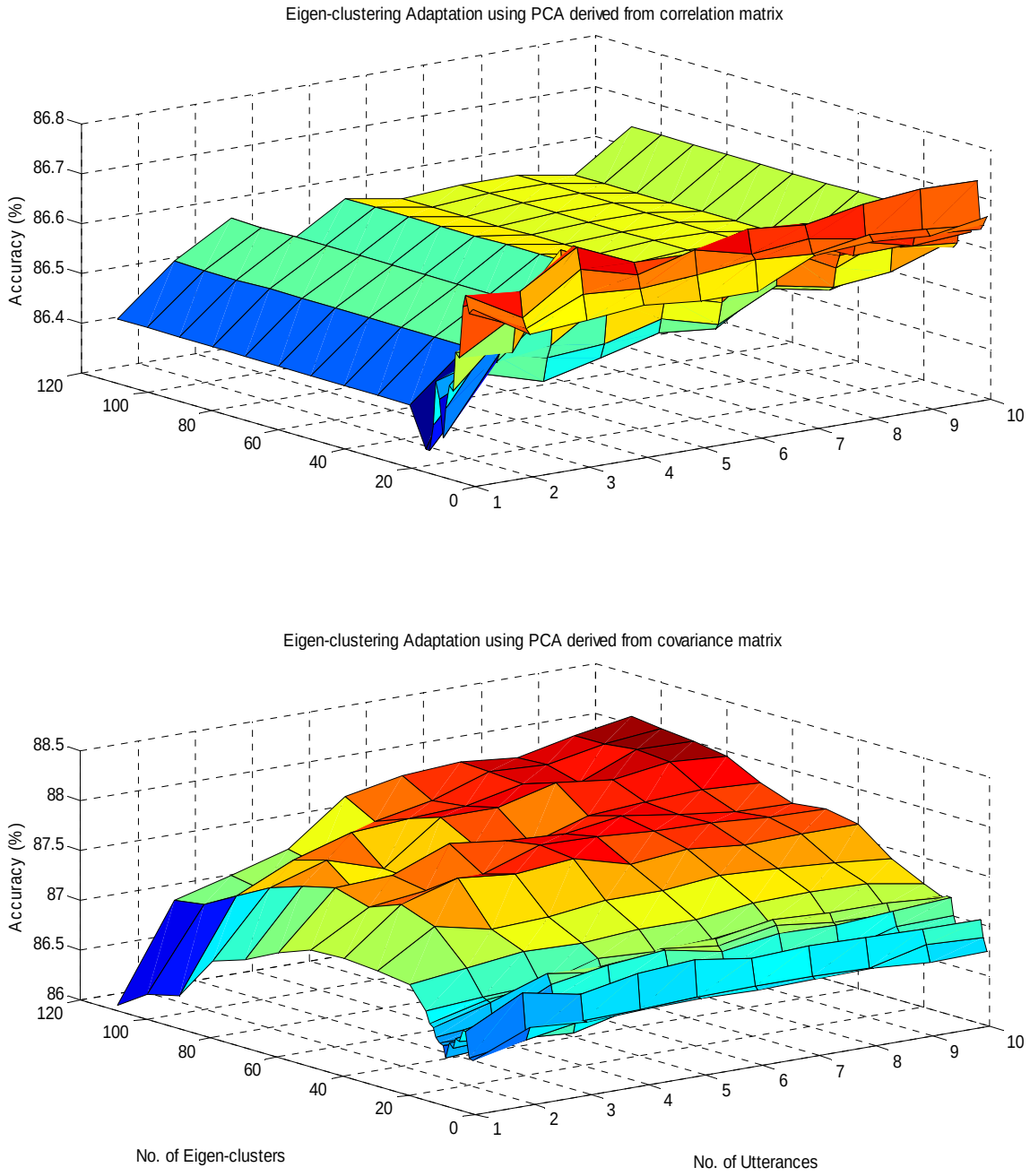

Figure 4.11 Eigen-clustering performance comparison on PCA correlation matrix vs. covariance matrix

The performance of eigen-clustering (EC) and eigenvoice (EV) adaptations is shown in Figure 4.12. The both methods are implemented by PCA on the covariance matrix with the numbers of the clusters/voices selected as 5, 10, 15, 20, and 30. Under the same amount of adaptation data, the overall performance of eigenvoice adaptation (blue lines in Figure 4.12) with knowledge of speaker identities is better than the eigen-clustering method (green lines in Figure 4.12) by about 2%.

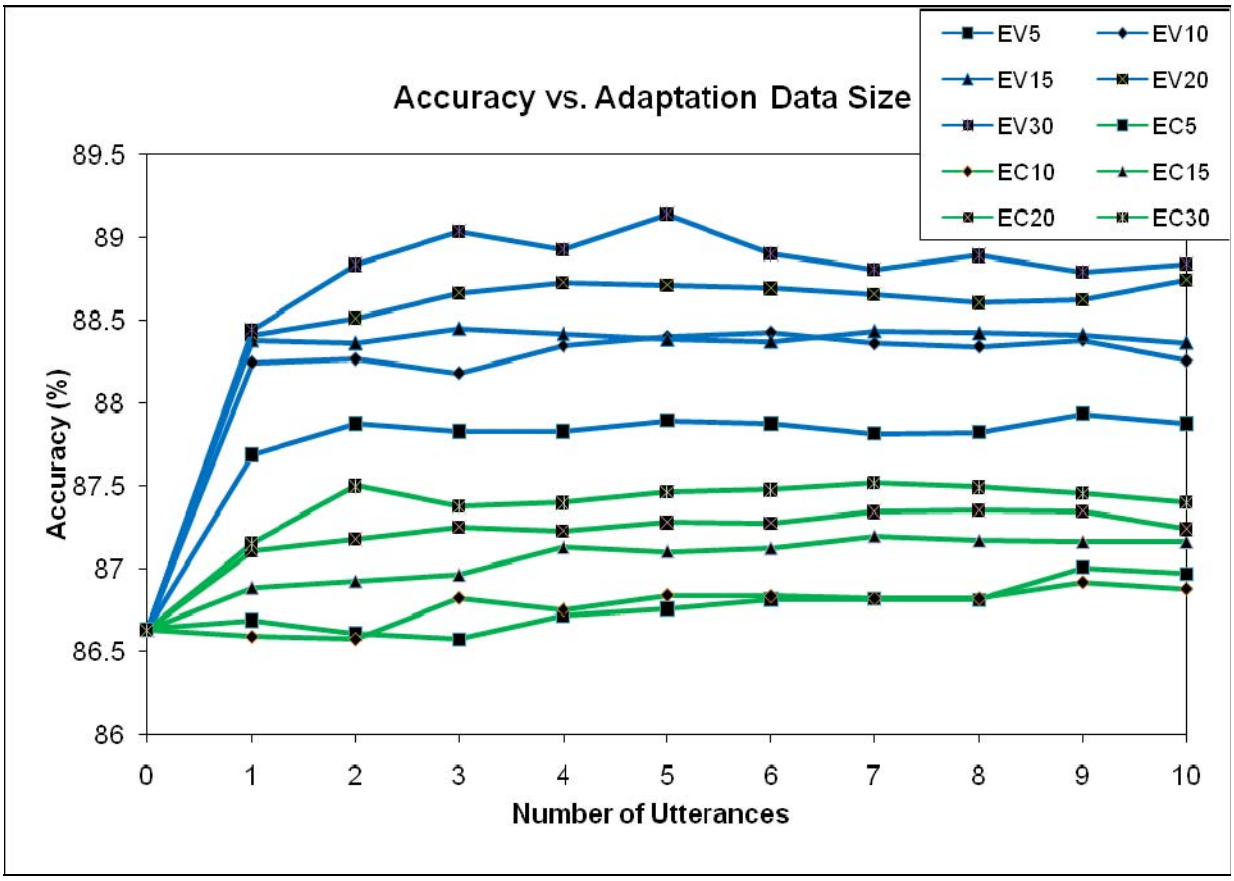

Figure 4.12 Performance of Eigenvoic and Eigen-clustering adaptation

A comparison between eigen-clustering with 30 clusters (EC30) and the other adaptation methods is illustrated in Figure 4.13. The eigenvoice adaptation with 30 voices (EV30) outperforms the other three when 5 or fewer adaptation utterances (about 15 seconds) are

used, whereas the eigen-clustering adaptation is slightly better when one or two adaptation utterances are used. When eigen-clustering with 30 clusters is combined with MAP (red line in Figure 4.13), the performance is not only better than the other three in the range of 1 – 6 utterances but also shows the potential in consistently improving when the amount of adaptation data increases. The detailed accuracy results in Figure 4.12 and Figure 4.13 are displayed in Table A.10.

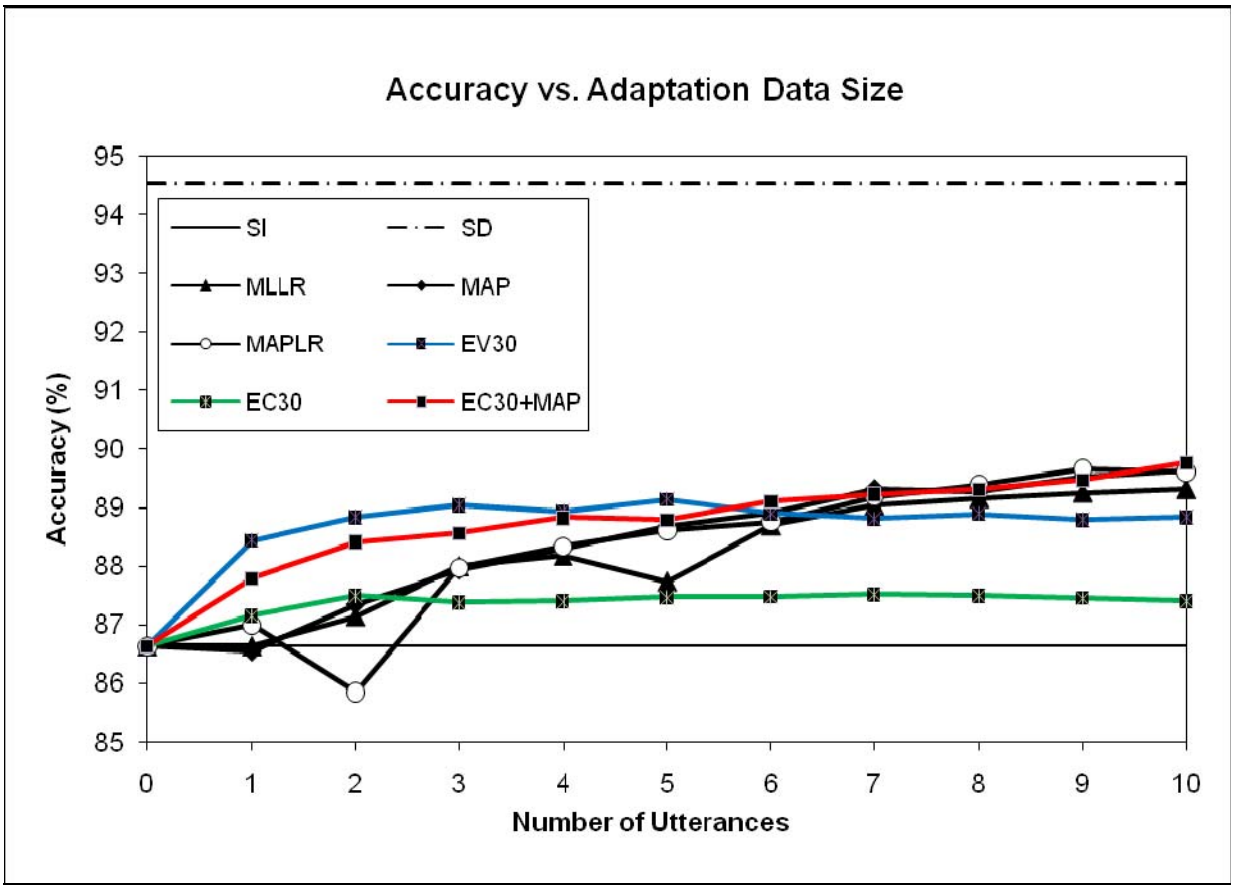

Figure 4.13 Performance of six adaptation methods

4.2.6 Single Gaussian Triphone Models

In sections 4.2.4 and 4.2.5, context-independent HMM systems were evaluated. In this

section, a context-dependent HMM triphone system is implemented to fully investigate the properties of the eigen-clustering method.

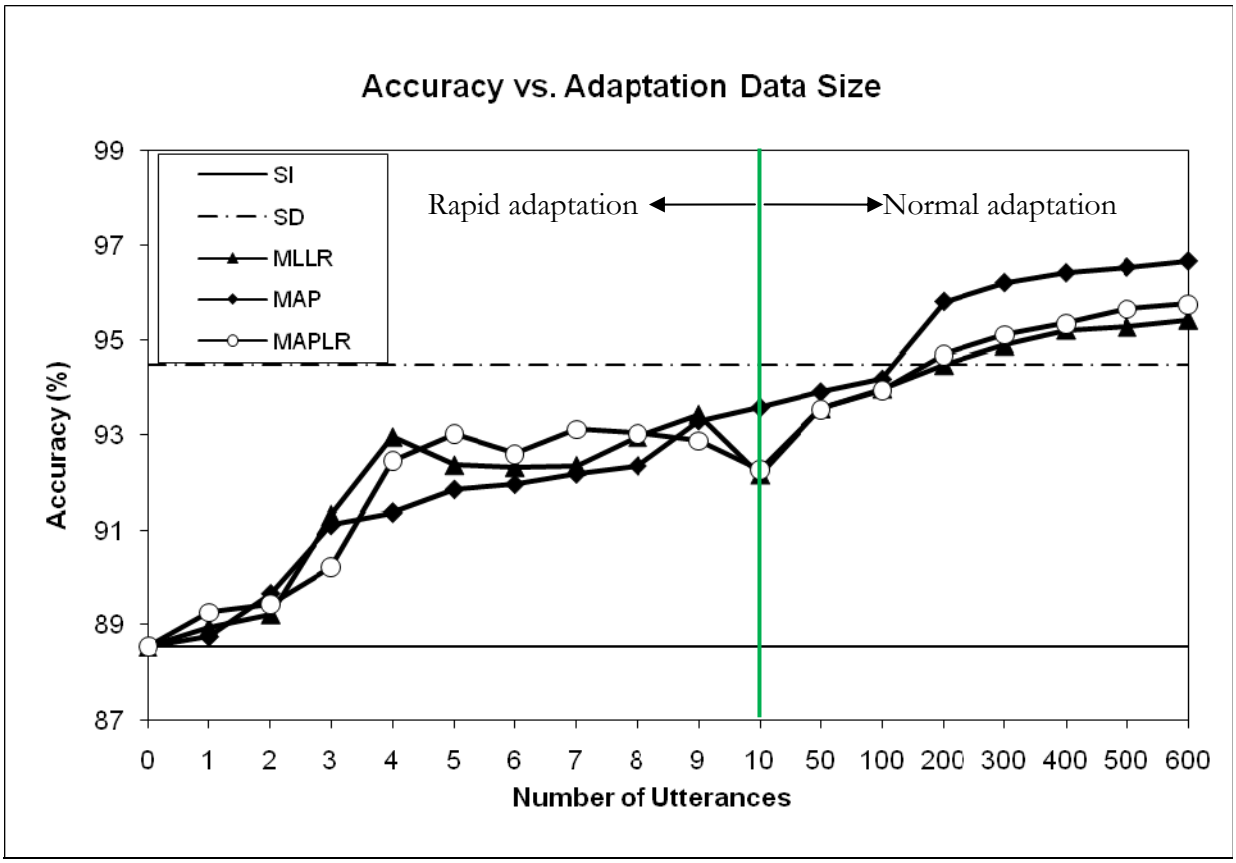

Figure 4.14 Performance of SI, SD, MLLR, MAP and MAPLR adaptation

The performance of the three adaptation methods MLLR, MAP, and MAPLR along with SI and SD systems is shown in Figure 4.14. The ranges for rapid adaptation and normal adaptation are split by the vertical green bar using non-linear scale of number of utterances (horizontal axis). Comparing to the performance of the 4-mixture Gaussian HMMs in Figure 4.9, the single Gaussian triphone HMMs show absolute improvement of 2% over the baseline SI and a similar result in the SD system. All three adaptation methods improve accuracy over the baseline in any conditions. The MAP adaptation shows the

improvement more consistently than the others with incrementing adaptation data size. All three methods have the very close accuracies when there are 3 and less utterances (about 9 seconds) for adapting. The highest accuracy from the three methods with all 10 utterances is close to the SD system. In normal adaptation data ranging from 50 to 600 utterances per speaker, all three methods show the same manner of performance as in the context-independent systems, with the MAP outperforming the other two when more than 200 utterances of adaptation data are used. In the first case, the MAP is superior over the speaker-dependent system and the MLLR and MAPLR nearly tie the SD, while the MLLR is not as obviously saturated as before.

The performance of eigenvoice adaptation using PCA derived from both correlation and covariance matrices as a function of adaptation utterances and eigenspace dimensions (number of eigenvoices) are compared in Figure 4.15. In the same manner as in the experimental sections for context-independent HMMs, the number of eigenvoices includes 1 to 15, 20 – 100 in increments of 10, as well as all 109 reference speakers. The detailed results are also displayed in Table A.11 and Table A.12 respectively. The accuracy range is from 87.6% to 90.5% for the PCA correlation implementation, and from 90.1% to 93.2% for the covariance implementation. The covariance implementation shows clearly higher accuracies over the correlation approach at almost every point. The accuracy surfaces of the both implementations are relatively flat with increasing number of adaptation utterances, plateauing after 2 – 3 utterances.

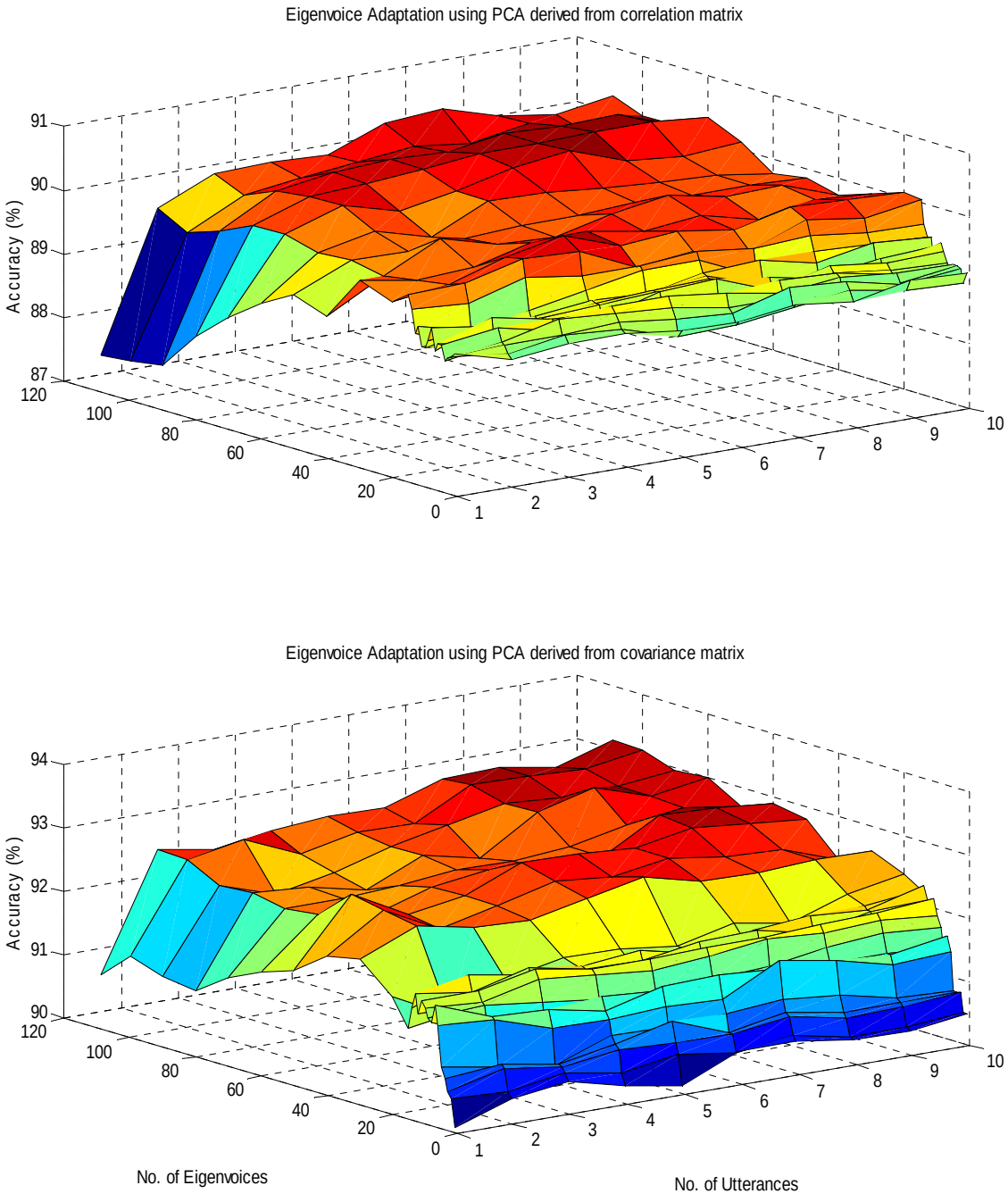

Figure 4.15 Eigenvoice performance comparison on PCA correlation matrix vs. covariance matrix

Figure 4.16 shows the evaluation results for the eigen-clustering adaptation. The number and the scale of eigen-clusters are in the same setup as in the experiments for the context-independent HMMs. The PCA correlation approach in Table A.13 gives the recognition accuracies in a range of 84.1% to 91.3%, while the covariance one in Table A.14 shows the results from 82.2% to 92.9%. The covariance implementation has overall better performance than the correlation one as before.

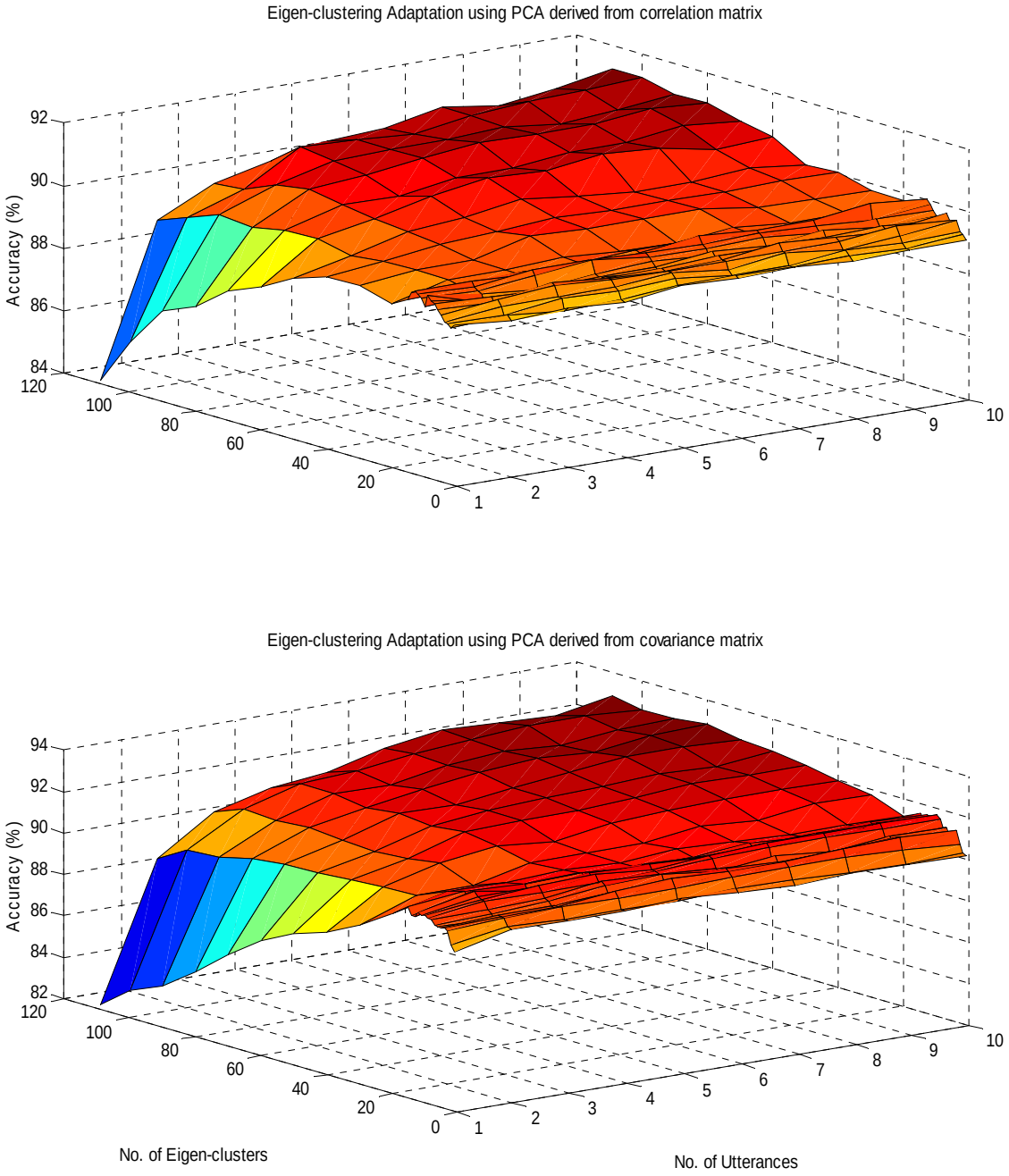

Figure 4.16 Eigen-clustering performance comparison on PCA correlation matrix vs. covariance matrix

Figure 4.17 shows the performance of eigen-clustering and eigenvoice adaptations, both are implemented by PCA on the covariance matrix with the numbers of the clusters/voices selected as 5, 10, 15, 20, and 30. Under the same amount of adaptation data, the overall performance of eigenvoice adaptation (blue lines in Figure 4.17) is better than the eigen-clustering method (green lines in Figure 4.17) by about 2%.

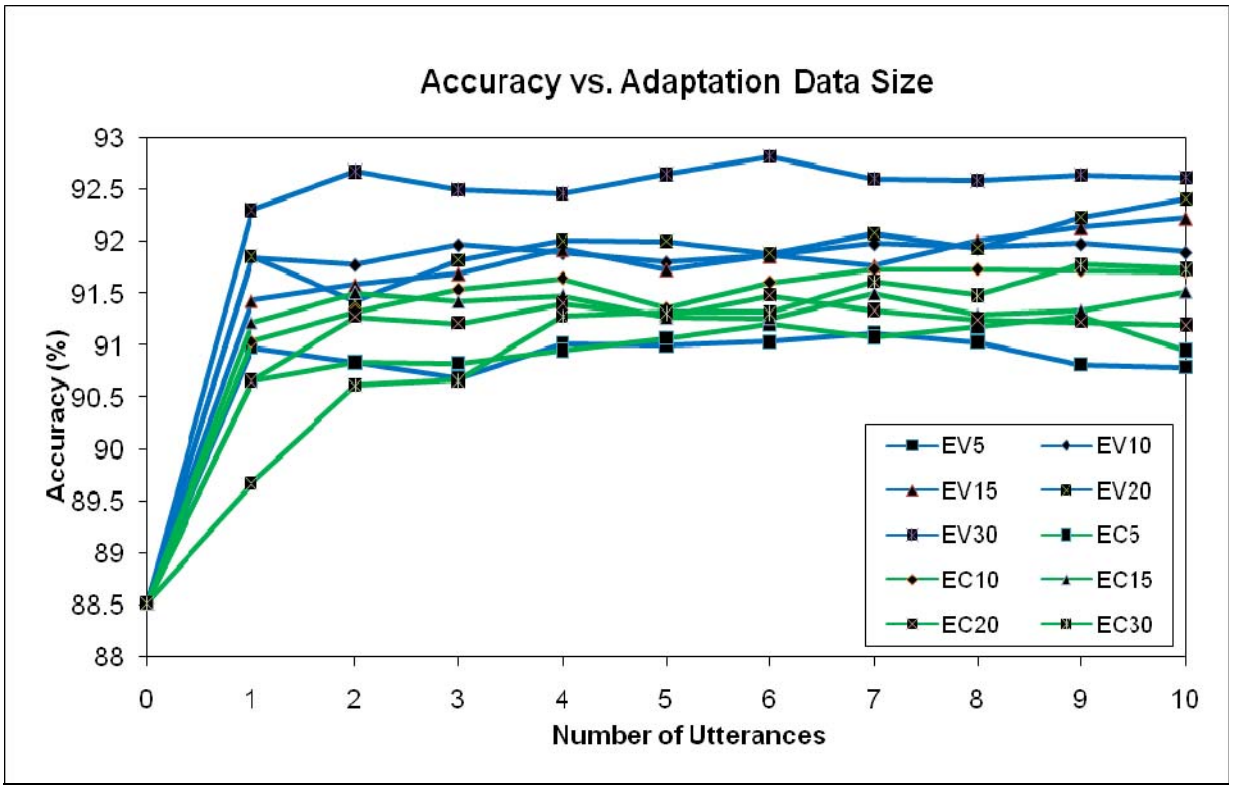

Figure 4.17 Performance of Eigenvoic and Eigen-clustering adaptation

A comparison of eigen-clustering with 30 clusters and eigenvoice with 30 voices to the other three adaptation methods is shown in Figure 4.18. The both eigenvoice and eigen-clustering adaptation methods outperform the other three when 3 or fewer adaptation utterances (about 9 seconds) are used. When eigen-clustering with 30 clusters is combined with MAP (red line in Figure 4.18), the performance is not only better than the other three

in the range of 1 – 3 utterances but also shows the potential in consistently improving when the amount of adaptation data increases. The performance accuracies are illustrated in Table A.15.

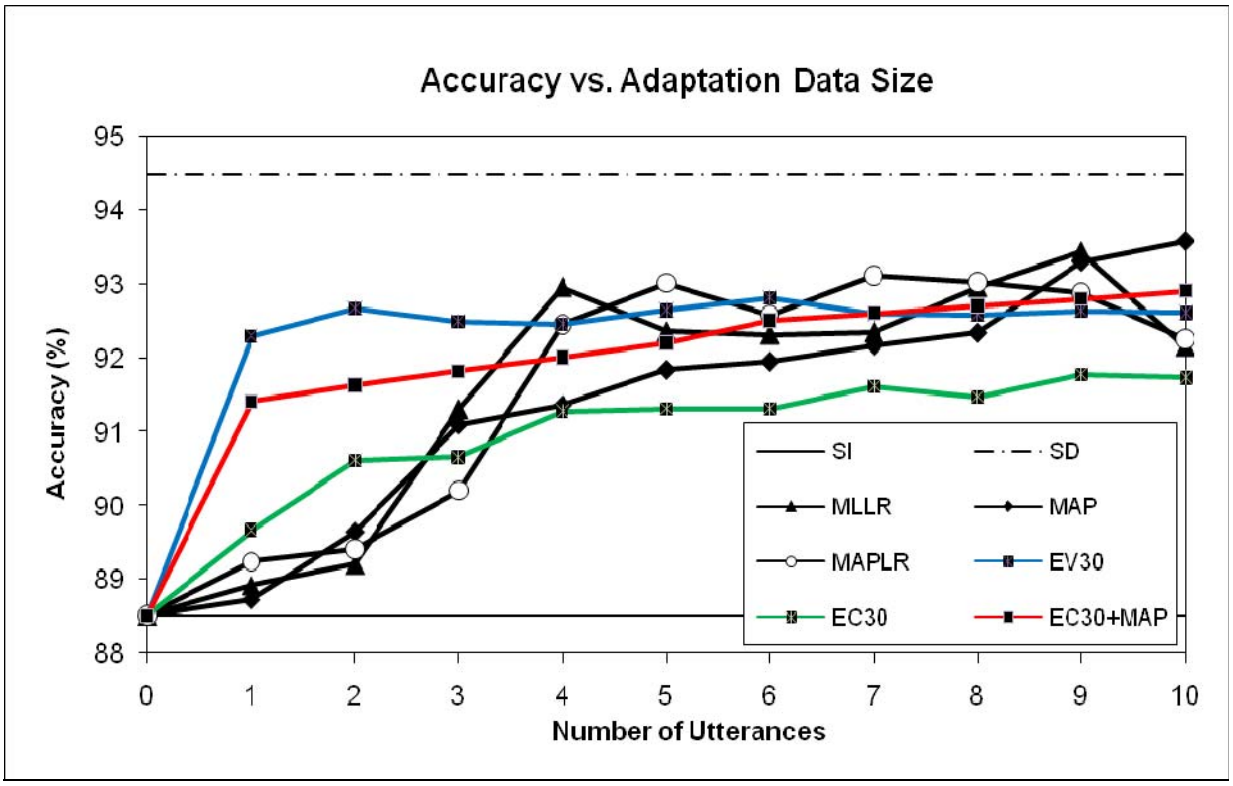

Figure 4.18 Performance of six adaptation methods

4.3 Animal Vocalization

Automatic systems for animal vocalization classification often require fairly large amounts of data to build models. However, animal vocalization data collection and transcription is a difficult and time consuming task, so that it is expensive to create large datasets. One natural solution to this problem is the use of acoustic adaptation methods. Such methods, common in human speech recognition systems, create initial models trained on speaker independent data, then use small amounts of adaptation data to build individual-

specific models. Since, as in human speech, individual vocal variability is a significant source of variation in bioacoustic data, acoustic model adaptation is naturally suited to classification in this domain as well.

4.3.1 Subjects and Data

Ortolan buntings (*Emberiza hortulana* L.) as a species have declined steadily the last fifty years in Western Europe, and have been listed as critically endangered on the Norwegian red-list. The population size is now only about 100 singing males and declines an average of 8% annually (Dale, 2001). The initial decline of the Norwegian population was probably due to the habitat loss related to changes in agriculture practices (Dale, 2001). However, ten years of intensive study revealed that the main reason for the continuous decrease is female-biased dispersal pattern, which in isolated and patchy population seriously affects sex ratio, behavior of males and breeding success measured at the population level (Dale, 2001). It is hoped that increasing our understanding of male ortolan bunting vocalizations will enable us to better understand breeding behavior and reduce the risk of extinction.

Norwegian ortolan bunting vocalizations were collected from County Hedmark, Norway in May of 2001 and 2002 (Osiejuk *et al.*, 2003). The birds covered an area of approximately 500 km^2 on twenty-five sites, and males were recorded on eleven of those sites. A team of one to three research members who recognized and labeled the individual male buntings visited the sites. Overall, the entire sample population in 2001 and 2002 contains 150 males, 115 of which were color-ringed for individual identification. Because there are no known acoustic differences between the ringed and non-ringed males, all data was grouped together for experimental use.

Ortolan buntings communicate through fundamental acoustical units called syllables (Osiejuk *et al.*, 2003, 2005). Figure 4.19 depicts the 19-syllable vocal repertoire used in this dataset. Individual songs are grouped into song type categories, e.g. *ab*, *cb*, that indicate the sequence of syllable types present. Each song type has many specific song variants, e.g. *aaaab*, *aaabb*, which indicate the exact repetition pattern. Figure 4.20 shows spectrograms of three specific type *ab* songs, song variants *aaaab*, *aaabb* and *aaaabb*. The waveforms in Figure 4.19 and Figure 4.20 are low background noise exemplars, taken from different individuals to illustrate the repertoire.

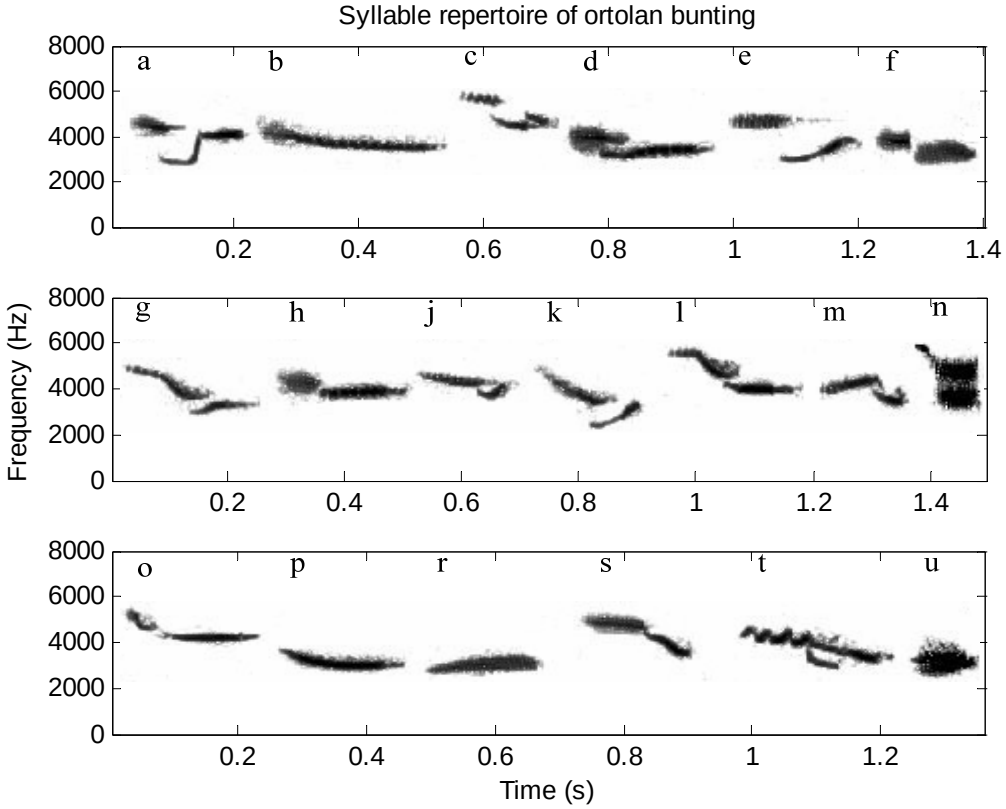

Figure 4.19 Complete set of the 19-syllable repertoire of ortolan bunting.

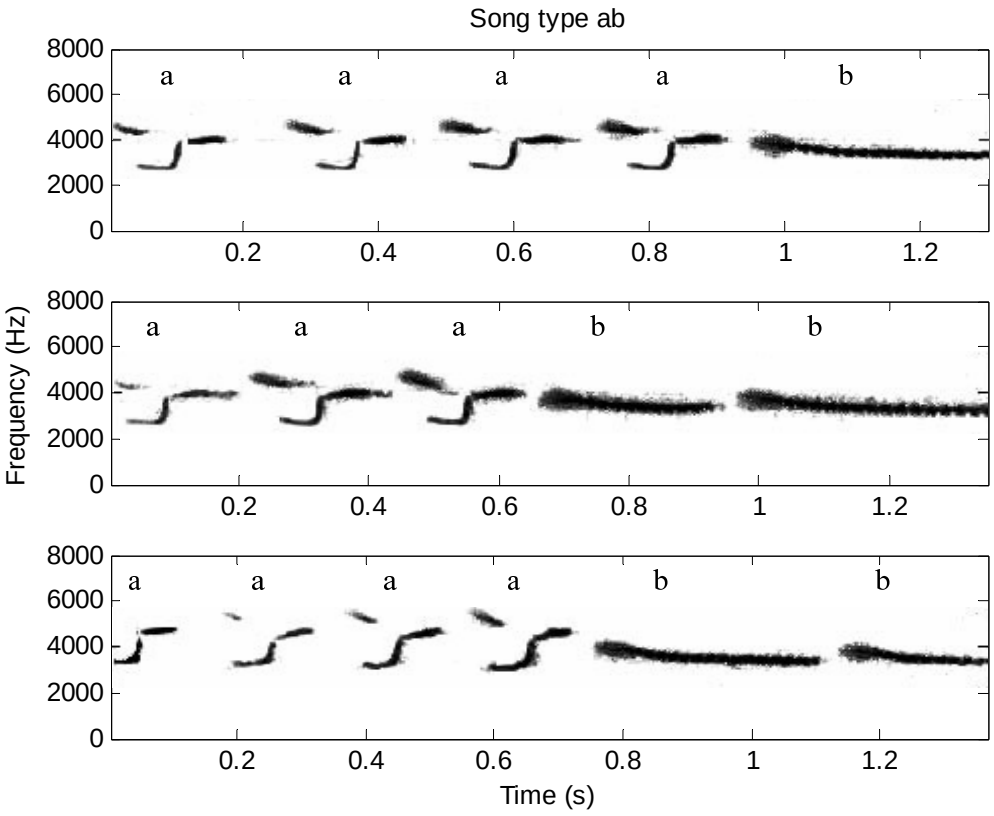

Figure 4.20 ab-type song variation in ortolan bunting

The vocalizations were recorded in the morning hours between 04:00 and 11:00 in each site, using a HHB PDR 1000 Professional DAT recorder with a Telinga V Pro Science parabola, a Sony TCD-D8 DAT recorder with a Sennheiser ME 67 shotgun microphone or an Aiwa HS-200 DAT recorder with a Sennheiser ME 67 shotgun microphone. All recordings were digitally transferred from Technics SV-DA 10 recorder via a SPDIF cable to a PC workstation with SoundBlaster Live! 5.1 at a sampling rate of 48 kHz with 16-bit quantization. For a more detailed description of the methods used to record the vocalizations, see Osiejuk *et al.* (Osiejuk *et al.*, 2003, 2005).

4.3.2 Data Organization

The data set used here is a subset of the data (Osiejuk *et al.*, 2003, 2005) including 60 song types and 19 syllables from 105 individuals. In selecting data for this study, calls containing syllables that were identified in only a single individual or a single song type were not included. Different individuals were selected for the training and testing/adaptation sets, balanced to get full coverage of all syllables in each set.

The protocol used to separate the data into training, test and adaptation sets is as follows (Tao and Johnson, 2008):

1. Choose calls containing syllables identified in more than a single individual or a single song type. This gives a resulting data set of 105 individuals, 60 call types and 19 syllables.
2. Select individuals for testing/adaptation.
    a. Sort song types in ascending order according to number of examples.
    b. Starting with the least common song type, select the individual with the highest number of examples in that song type (minimum 2 examples).
    c. Repeat this process for each song type until the individuals selected for testing cover all 60 types.

    This results in a set of 30 individuals for testing/adaptation.
3. Create explicit test and adaptation data sets by randomly dividing the data into test and adaptation sets for each selected individual, subject to a maximum of 30 vocalizations in each set for any one individual and song type.
4. Group the remaining individuals into a caller-independent training data set, again reducing the number of examples to a maximum of 30 for any one individual and

song type.

Descriptive statistics of the resulting training, test, and adaptation sets are shown in Table 4.2. From the process detailed above it is clear that the 75 individuals in the training set are disjoint from the 30 individuals in the test/adaptation data, while the test and adaptation sets share the same group of individuals. All three sets have a full representation of syllables. Note that the training set does not cover the full range of 60 song types, but is still sufficient for training syllable-level HMMs for classification, as discussed in the next section. The size of the adaptation set is the same as that of the test set to allow the data to be used for training caller-dependent models as well as to allow a large range of variation for examining the impact of adaptation data quantity on performance.

	Training Set	Test Set	Adaptation Set
Number of Individuals	75	30	30
Number of Song Types	53	60	60
Number of Syllables	19	19	19
Number of Vocalizations	2039	864	886
Mean Vocalizations/Caller	27.2	28.8	29.5
Mean Vocalizations/Type	38.5	14.4	14.8
Mean Vocalizations/Syllable	107.3	45.5	46.6

Table 4.2 Distribution of the number of individuals, song-types and vocalizations, and vocalizations with associated frequencies on individual, song-type and syllable for training, test and adaptation sets.

4.3.3 Feature Extraction and Acoustic Modeling

The acoustic features used in this classification system are Greenwood Function Cepstral Coefficients (GFCCs) (Clemins and Johnson, 2006; Clemins *et al.*, 2006). GFCCs are a species-specific generalization of Mel Frequency Cepstral Coefficients (MFCCs) (Huang *et al.*, 2001). The process for computing cepstral coefficients begins with segmenting vocalizations into evenly spaced appropriately sized windows (based on the frequency range

and vocalization patterns of the species). For each window, a log magnitude Fast Fourier Transform (FFT) is computed and grouped into frequency bins. A Discrete Cosine Transform (DCT) is then taken to transform the log magnitude spectrum into cepstral values. For GFCC's, the frequency scale of the FFT is warped according to the Greenwood function (Greenwood, 1961) to provide a perceptually scaled axis. To do this, the parameters of the Greenwood function are estimated from the upper and lower bounds of the species' hearing range along with a warping constant of $k = 0.88$ (LePage, 2003). Details of the warping equations and GFCC feature extraction process can be found in (Clemins and Johnson, 2006; Clemins *et al.*, 2006). Given basic information about a species frequency range, GFCC's provide an accurate and robust set of features to describe spectral characteristics over time.

In addition to the base set of GFCC features, energy is computed on the original time-domain data, and velocity and acceleration coefficients representing the first and second order rates of change are added. For the experiments described here, the vocalizations are segmented using 5ms Hamming windows, with a 2.5ms overlap. 12 GFCCs plus normalized log energy along with velocity and acceleration coefficients are calculated, for a total of 39 features. Frequency warping is done using a given hearing range from 400 Hz to 7200 Hz, with 26 triangular frequency bins spaced across that range. Velocity and acceleration coefficients are computed using a 5-window linear regression.

In this work, each of the 19 ortolan bunting syllables is modeled with a 15-state left-to-right HMM with a diagonal covariance Gaussian model, which is analogous to single Gaussian monphone models in human speech. A song-type grammar model is used simply to treat each song-type as an isolated unit, which is similar to the grammar used in the

isolated word recognition task in human speech (Rabiner and Levinson, 1981). During the training process, the Baum-Welch algorithm for Expectation Maximization (EM) (Baum *et al.*, 1970; Moon, 1996) is used to estimate the HMM parameters that maximize the joint likelihood of all training observation sequences. For classification, the Viterbi algorithm (Forney, 1973) is used to find the model sequence having the highest likelihood match to the sequence of test features.

4.3.4 Song-type Classification

Song-type classification experiments were implemented on the ortolan bunting data set as previously described. The goal of these experiments is to compare how well the proposed eigen-clustering method performs compared to a baseline caller-independent (CI) system, and other available adaptation techniques. For reference, a fully caller-dependent (CD) system was also implemented.

The following song-type classification systems were implemented for comparison:

CI: The baseline caller-independent models. The system diagram for the CI system is shown in Figure 4.21. There was no overlap between the training individuals and test individuals, with 75 and 30 individuals in the two datasets, respectively.

CD: The caller-dependent models. The system diagram for the CD system is shown in Figure 4.22. The training and testing data were separate but came from the same individuals. The training data used for the CD experiments was the same as the adaptation data used for the CA experiments.

CA: The caller-adapted models. The adaptation system is shown in Figure 4.23. The training and testing data were separate but came from the same individuals, and the test data

was further split into adaptation data and final test data. The CA experiments were implemented using supervised mean and variance adaptation.

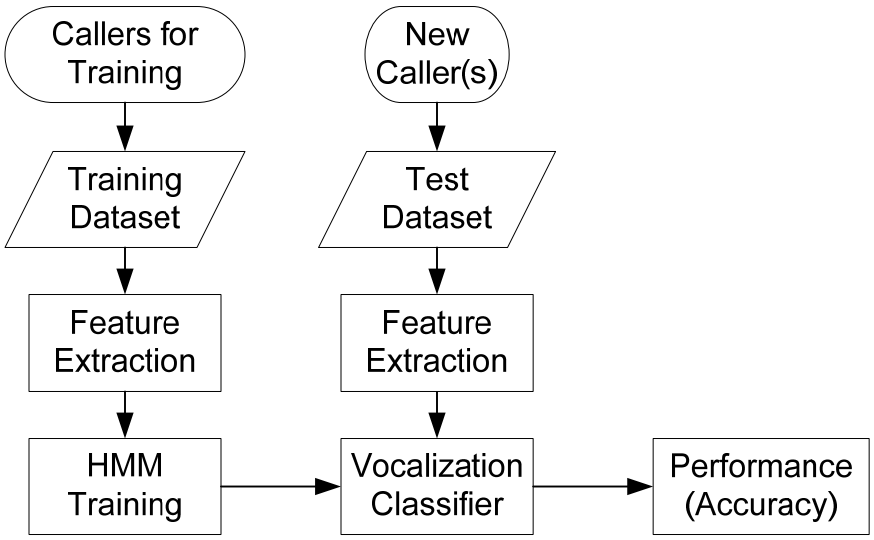

Figure 4.21 Caller-independent (CI) system, with separate individuals for the training and testing data

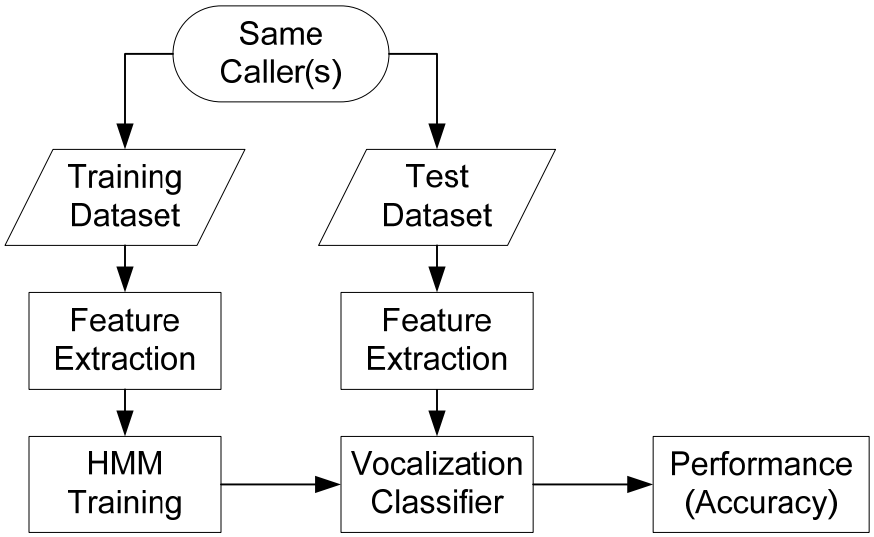

Figure 4.22 Caller-dependent (CD) system, with training and testing data coming from the same group of individuals

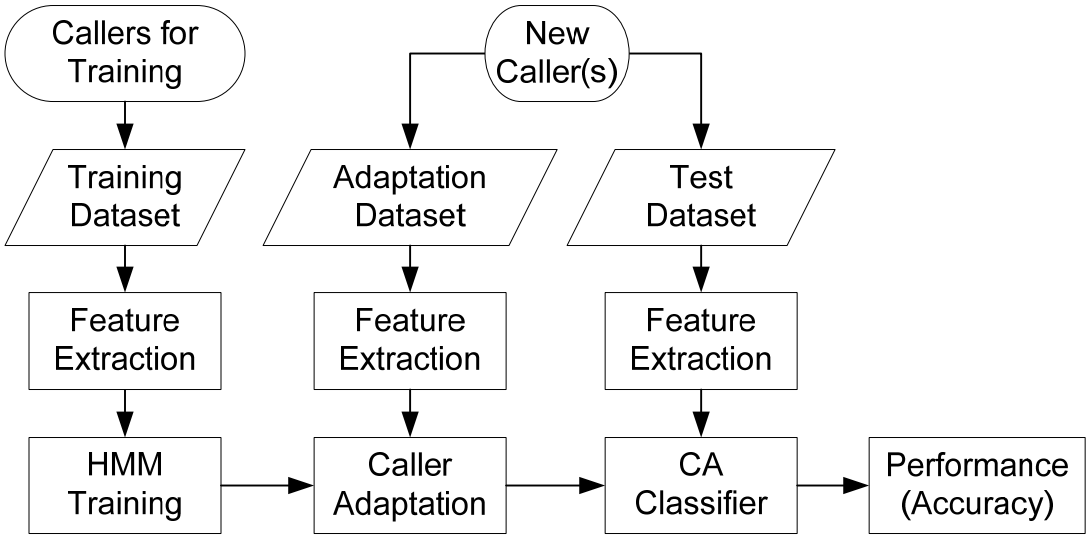

Figure 4.23 Caller-adapted (CA) system, with separate training and testing data, but with a portion of the testing data pulled out and used for adaptation.

Similar to the human speech setup in Section 4.2.3, the 75 callers in the call-independent training set were treated as the reference callers for the eigenvoice adaptation, while each of 2039 vocalizations was used to train a vocalization-level model for the eigen-clustering method.

In order to see how the amount of adaptation data affected the results, each adaptation method was implemented multiple times, using increasing amounts of adaptation data. This was done in 10% increments (about 3 seconds), starting with 0% (no adaptation, equivalent to the initial CI system), then 10%, 20%, and so on up to 100% (full adaptation set in use).

The classification accuracies for eigenvoice adaptation using PCA derived from both correlation and covariance matrices as a function of adaptation data size and eigenspace dimensions (number of eigenvoices) are compared in Figure 4.24. The number of eigenvoices includes 1 to 15, 20 – 70 in increments of 10, as well as all 75 reference callers. The detailed classification results are also displayed in Table A.16 and Table A.17 respectively. The accuracy range is from 77.8% to 89.7% for the PCA correlation

implementation, and from 81.4% to 93.3% for the covariance implementation. The covariance implementation shows clearly higher accuracies over the correlation approach at almost every point. The accuracy surfaces of the both implementations are relatively flat with increasing the number of adaptation utterances, plateauing after 6 – 9 seconds.

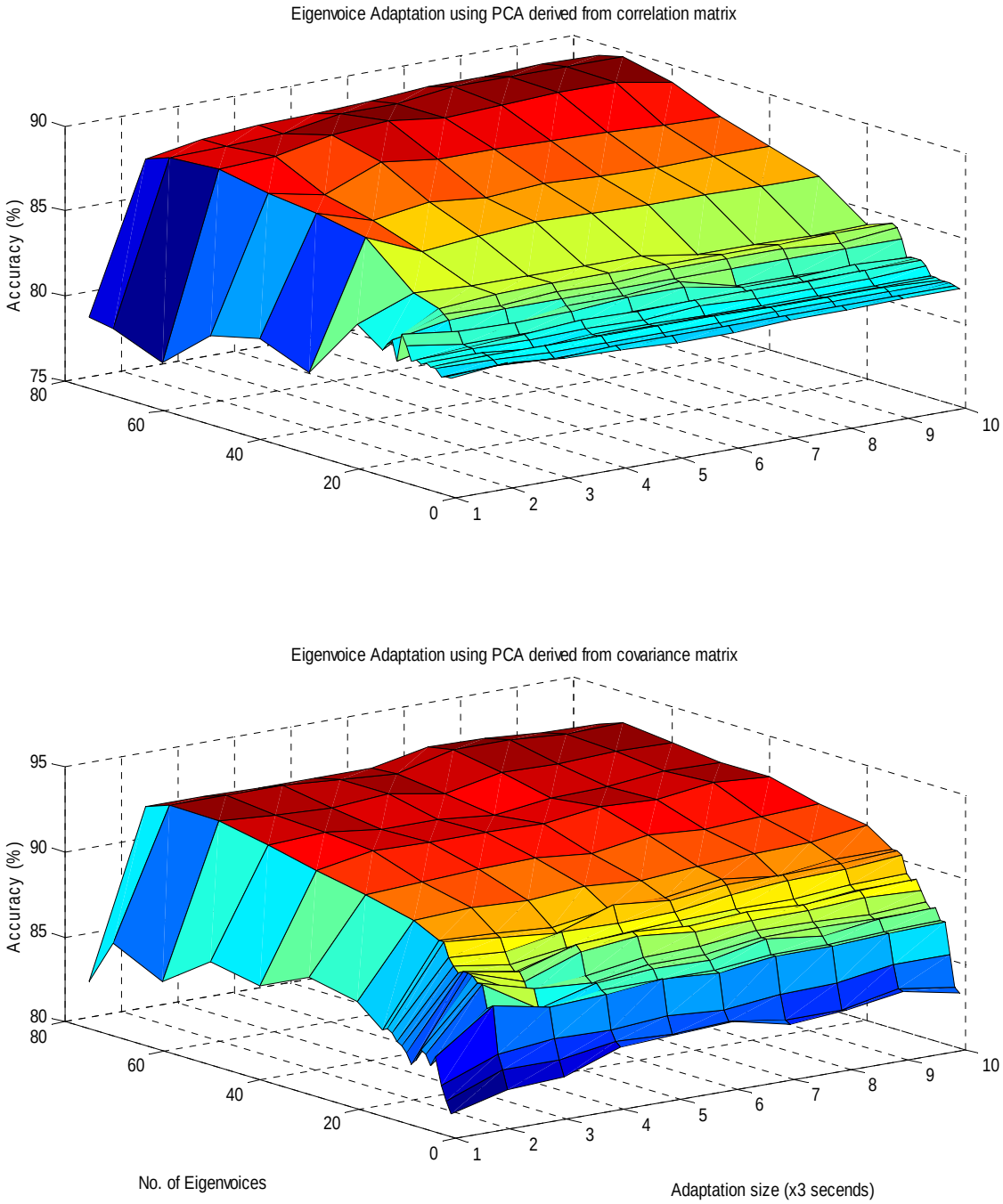

Figure 4.24 Eigenvoice performance comparison on PCA correlation matrix vs. covariance matrix

The proposed eigen-clustering adaptation evaluation results are shown in Figure 4.25. The number and the scale of eigen-clusters are the same as in the experiments for the eigenvoice adaptation for comparison purposes. The largest number of eigen-clusters in the experiment is 75 to equal the total number of reference callers. The number of selected clusters would be unlikely to be so high in practice, because one of properties of PCA is to reduce data dimensions. The PCA correlation approach in Table A.18 gives the recognition accuracies in a range of 84% to 87.9%, while the covariance one in Table A.19 shows the results from 82.9% to 87.7%. The covariance implementation has better performance than the correlation one in general as before.

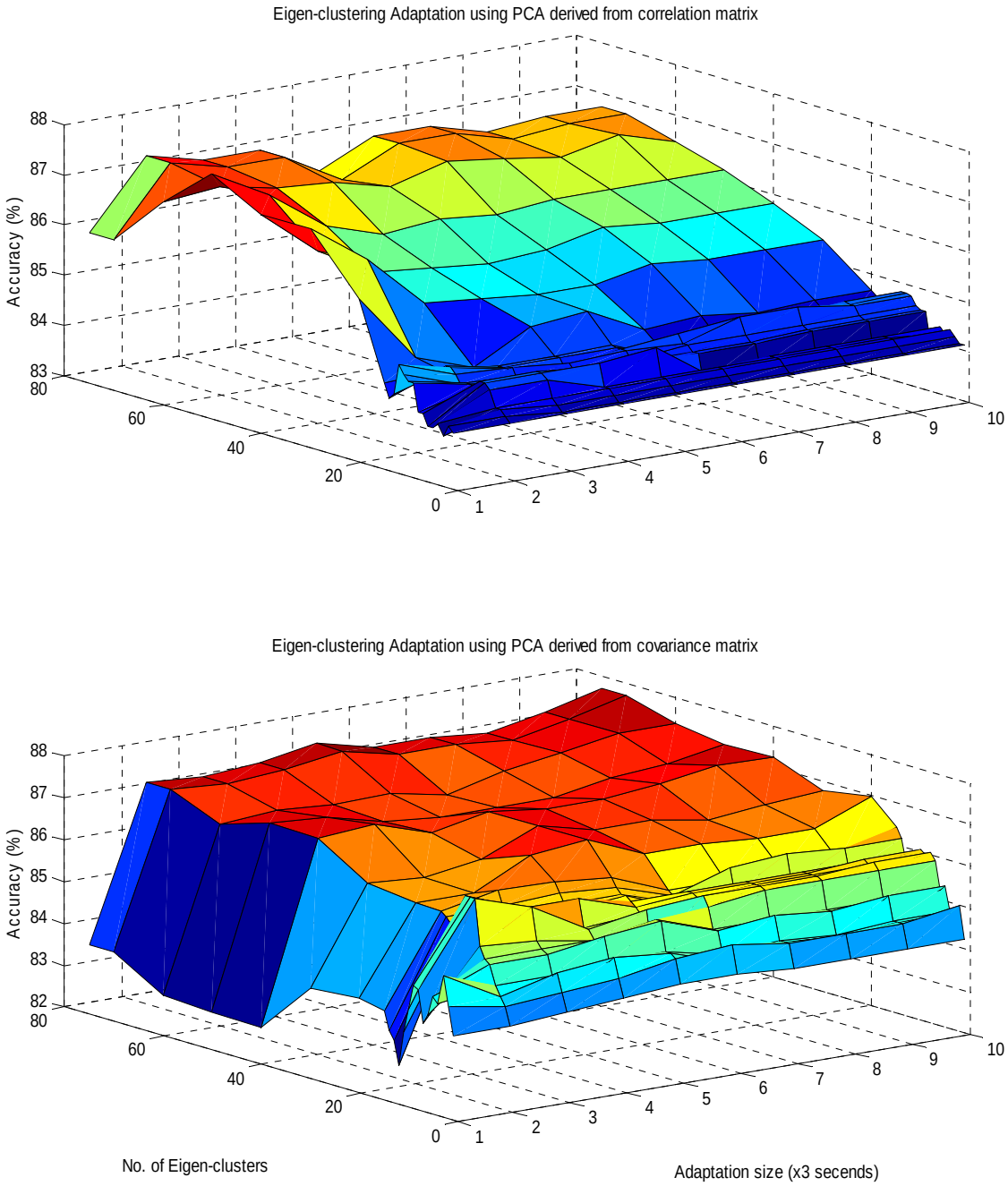

Figure 4.25 Eigen-clustering performance comparison on PCA correlation matrix vs. covariance matrix

The classification results of eigen-clustering and eigenvoice adaptations are shown in Figure 4.26. The both methods are implemented by PCA on the covariance matrix with the numbers of clusters/voices selected as 5, 10, 15, 20 and 30. Under the same amount of adaptation data, the overall performance of eigenvoice adaptation (blue lines in Figure 4.26) is better than the eigen-clustering (green lines in Figure 4.27) method by about 2%.

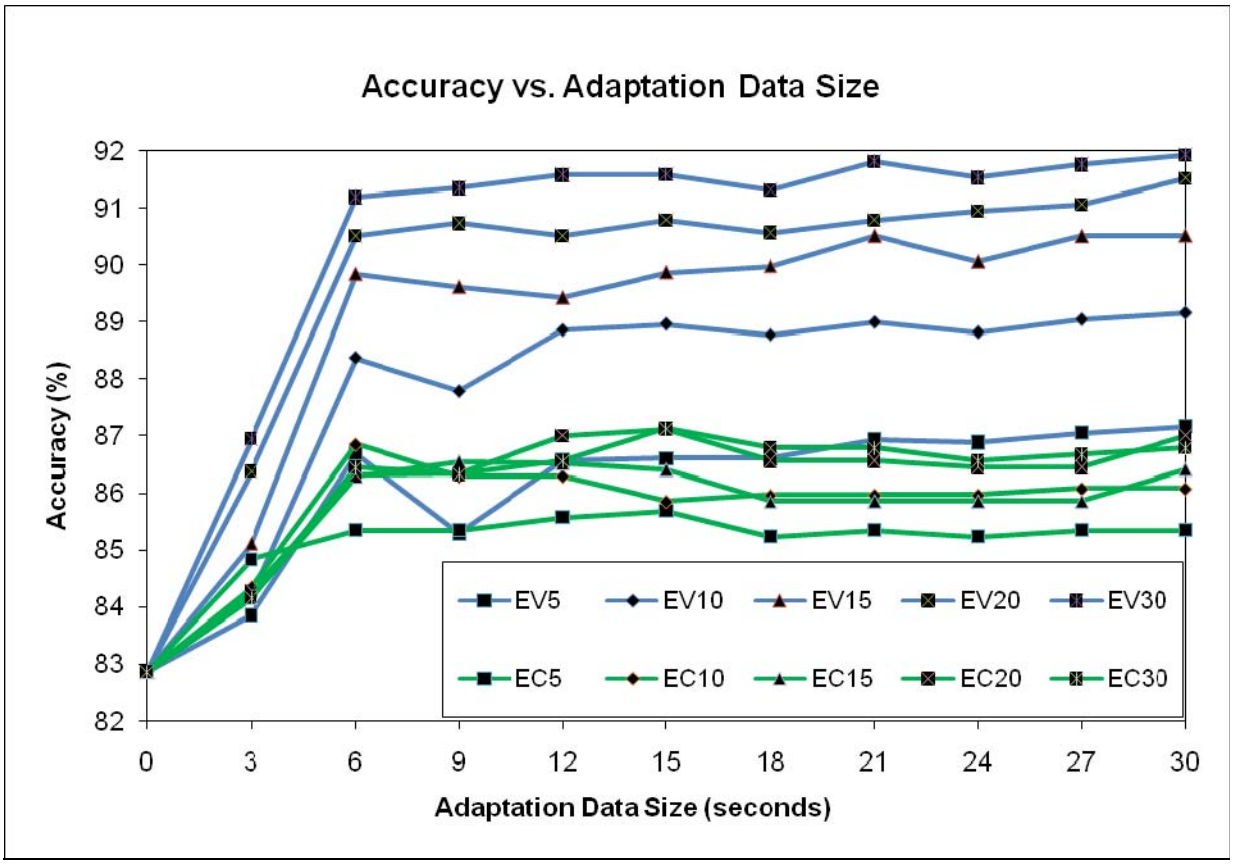

Figure 4.26 Performance of Eigenvoic and Eigen-clustering adaptation

A comparison of eigen-clustering with 30 clusters (EC30) and eigenvoices with 30 voices (EV30) to the other three adaptation methods MLLR, MAP and MAPLR is shown in Figure 4.27. The baseline CI system has an 82.9% accuracy, while the CD system has an 88.1% accuracy. The both eigenvoice and eigen-clustering methods outperform the baseline CI

system. The eigenvoice adaptation shows the highest accuracy using only 3 seconds of adaptation data. When the eigen-clustering with 30 clusters is combined with MAP (EC30 + MAP), the performance is not only better than the other three in the range of 1 – 9 seconds but also shows the potential in consistently improving when the amount of adaptation data increases. All the six adaptation methods improve the accuracy over the baseline with more than 6 seconds of adaptation data. The EC30+MAP, eigenvoice, MAP and MAPLR methods have already outperformed the CD system at 6 seconds of utterances. This clearly demonstrates that the training data for the CD system (30 utterances per caller) is insufficient. The MAP adaptation shows consistent improvement with incrementing adaptation data size. MAP performs comparably to eigenvoice and slightly better than eigen-clustering at the 3-second of adaptation data, because most syllables for each new caller have been covered in such a small amount data. Both MLLR and MAPLR are below the baseline when only the 3 seconds of adaptation data are used, because computational errors happened during estimating the transformation matrix W in equation (2.22) at each regression class using such a tiny amount of adaptation data (Leggetter and Woodland, 1995); and because MAPLR is MAP taking the MLLR trained model as *a prior*. This poor performance of both MLLR and MAPLR at 3-second adaptation data shown here is consistent with the previous reported results (Leggetter and Woodland, 1995; Kuhn *et al.*, 2000). The MAPLR adaptation yields the highest accuracy with the full set of adaptation data, 94.3% overall, representing a net gain of 11.4 percentage points (66% reduction in error) over CI and 6.2 percentage points (52% reduction in error) over CD. This also illustrates that adapted system can effectively use the CI system to adapt towards the CD system. Table A.20 contains the detailed classification accuracies in Figure 4.26 and Figure

4.27.

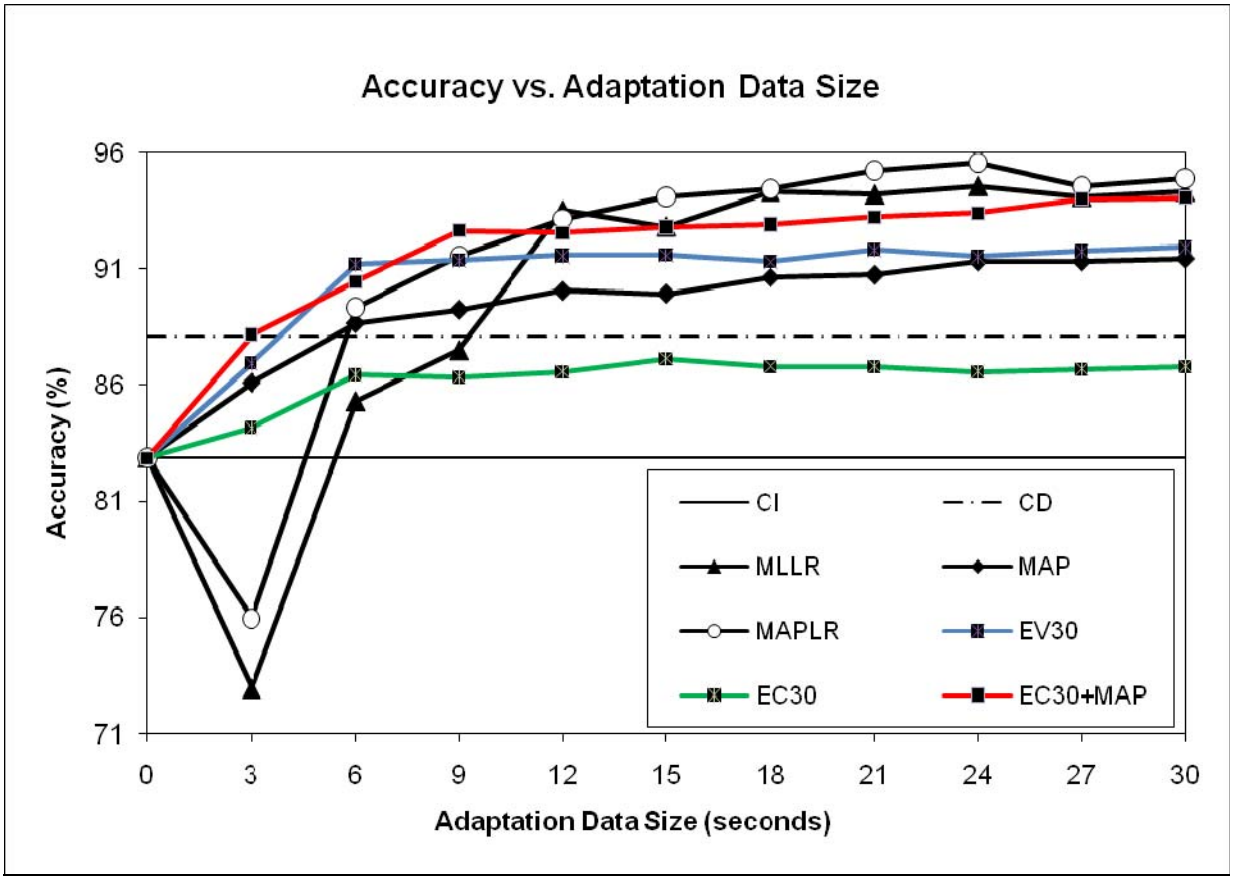

Figure 4.27 Performance of six adaptation methods

C H A P T E R 5 CONCLUSIONS

A new fast (e.g. a few seconds of adaptation data only) speaker adaptation method, eigen-clustering, is presented using a dimensionality reduction technique to map the parameters of utterance-based models into a set of orthogonal basis vectors. The new method was evaluated on the medium-vocabulary DARPA Resource Management database and an ortolan bunting vocalization database. Principal component analysis (PCA) was employed as the dimensional reduction technique to find the eigen clusters, and expectation maximization (EM) algorithm was applied as a maximum likelihood (ML) estimator to compute the coordinates of each new adapted speaker in the eigen space. The eigen-clustering approach showed superior performance on small amounts (up to 30 seconds) of supervised adaptation data comparing to the performance of the speaker-independent/caller-independent state-of-the-art baseline systems as illustrated in Figure 4.8, Figure 4.13, Figure 4.18, and Figure 4.27. With less than 6 seconds of adaptation data in these figures, it outperformed the MAP, MLLR, and MAPLR adaptation techniques across-the-board. It also achieved accuracies about 2% lower than that of eigenvoice, a method which is based on similar concepts but requires explicit knowledge of speaker identities during training. Similar to the eigenvoice method, as the amount of adaptation data increased, the performance of eigen-clustering adaptation seemed to reach a plateau.

The eigen-clustering approach is focused on very rapid adaptation in terms of a few seconds of adaptation data without explicit speaker knowledge in speaker-independent training data, similar conditions to the speaker-independent training, MAP, MLLR, and MAPLR adaptations. This method would be very useful for human speech recognition tasks such as closed-captioning for TV programs and broadcast news, where there is lots of

switching between different speakers without knowing who they are.

For rapid speaker adaptation with a very small amount of adaptation data, the eigen-clustering method has similarities to its counterpart, the eigenvoice method. Both methods use PCA offline to map the supervectors from the original model (feature) space to the eigenspace. However, the meanings of the supervectors are quite different between the two methods. A supervector represents one speaker-independent utterance in the eigen-clustering method, whereas it represents a reference speaker (i.e., all utterances associated with that speaker) in the speaker-independent training set in the eigenvoice approach. A minor difference to the original eigenvoice paper (Kuhn *et al.*, 2000) is that PCA derived from the covariance matrix gave overall better performance than the correlation matrix approach, which agreed with the results shown in the similar work (Hu *et al.*, 1998; Westwood, 1999). The reason to use the correlation matrix in the eigenvoice paper (Kuhn *et al.*, 2000) was to prevent variables (i.e., reference speakers) with large absolute values from dominating the analysis in cases where the variables have different units of measurement or are of different types. This reason did not influence either the eigen-clustering or eigenvoice method in terms of the experimental results in Figure 4.5, Figure 4.6, Figure 4.10, Figure 4.11, Figure 4.15, Figure 4.16, Figure 4.24 and Figure 4.25. The detailed comparison between the properties of eigen-clustering and eigenvoice adaptation is shown in Table 5.1.

	Eigen-clustering	Eigenvoice
SD Models/data required?	No	Yes
Speaker identities required?	No	Yes
Interpretation of supervector	Utterance	Speaker
Recommended PCA approach	Covariance matrix	Correlation Matrix
Offline PCA computation	Yes	Yes
Weight vector estimation	EM	EM
Adaptation data needed (> SI/CI)	3 sec	3 sec
Saturation starting	≥ 6 sec	≥ 6 sec

Table 5.1 Comparison between the properties of eigen-clustering and eigenvoice methods

The performance of eigen-clustering is generally about 2% lower than the eigenvoice adaptation when they are in the same number of *K*-dimension and the same size of adaptation data. This suggests that the *K*-dimensional eigenspace mapped from the "real" reference speaker models presents more accurate speech characteristics than the one created by the "artificial" utterance models.

The eigen-clustering method can be extended by combining it with the other adaptation methods such as MAP with the eigen-clustering adapted model as ***a prior***. The green lines in Figure 4.8, Figure 4.13, Figure 4.18, and Figure 4.27 clearly indicate the advantages and disadvantages of the eigen-clustering approach. The most obvious advantage is an extremely fast adaptability for a very small amount of adaptation data (6 seconds or fewer), but a disadvantage is that the performance quickly reaches a plateau for larger amounts (about 6 – 9 seconds). According to (Kuhn *et al.*, 2000), the reason for the performance saturation at larger amounts is that the *K*-dimensional eigenspace is a constraint in representing a new speaker, meaning some acoustic representations for the new speaker are not seen in the *K*

reference speakers. In other words, the speech characteristics in the *K*-dimensional space do not fully cover the new speakers. For example, SD models trained from all American English speakers do not cover a new speaker with a British English accent. The eigen-clustering and MAP combination retains the advantages of the approach for a very small amount of adaptation data, while improving performance for larger amounts (see the red line in each of these figures). This combination allows the adapted speaker model to leave the *K*-dimensional space, approaching the "true" new speaker model as the data becomes available.

The eigen-clustering and MAP combined method processes the two different adaptations consecutively, where eigen-clustering is used to adapt the model parameters and then MAP uses the adapted model as ***a priori***. This concatenated style makes the final performance purely rely on the adapted model obtained in the first step, so that the accuracy improvement is much less obvious when 15 seconds or more of adaptation data is used. Given this issue, future work will focus on new adaptation methods to combine both rapid and normal adaptation scales by generalizing the estimation of eigen-cluster/voice weights.

Speaker adaptation is a technique to effectively reduce the speaker variability in speech recognition. In a larger view, the concept of adaptation can handle a variety of variation issues. Future work will apply adaptation methods to minimize the intra-speaker variability such as phoneme variations for speaker identification/verification tasks, and to normalize the mismatch across different environment conditions for some tasks where the collected data might be in different background noises, and (or) from different recording equipments.

Overall, the most important contribution of this work is that the eigen-clustering adaptation realizes extremely rapid speaker adaptation without the need for speaker-dependent models. This contribution has the potential to impact applications in human

speech technology such as speech recognition and speaker identification, as well as more customized speech technology applications such as automatic vocalization transcription for bioacoustic data.

BIBLIOGRAPHY

Anderson, E., Bai, Z., Bischof, C., Blackford, S., Demmel, J., Dongarra, J., Croz, J. D., Greenbaum, A., Hammarling, S., McKenney, A., and Sorensen, D. (**1999**). *LAPACK User's Guide* (SIAM, Philadelphia).

Anderson, S. E. (**1999**). "Speech recognition meets bird song: A comparison of statistics-based and template-based techniques," The Journal of the Acoustical Society of America **106**(4), 2130-2130.

Barras, C., Zhu, X., Meignier, S., and Gauvain, J.-L. (**2004**). "Improving speaker diarization," in *Proc. Fall Rich Transcription Workshop (RT-04)* (Palisades, NY).

Baum, L. E. (**1972**). "An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes," Inequalities **3**, 1-8.

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

Becchetti, C., and Ricotti, L. P. (**1999**). *Speech Recognition: Theory and C++ Implementation* (Wiley).

Berg, J. K. (**1983**). "Vocalizations and associated behaviors of the African elephant (*Loxodonta africana*) in captivity," Z. Tierpsychol., 63-79.

Chen, S. S., and Gopalakrishnam, P. S. (**1998**). "Speaker environment and channel change detection and clustering via the Bayesian Information Criterion," in *Proc. DARPA Broadcast News Transcription and Understanding Workshop* (Lansdowne, VA), pp. 127-132.

Chesta, C., Siohan, O., and Lee, C. H. (**1999**). "Maximum a Posteriori Linear Regression for Hidden Markov Model Adaptation," in *Eurospeech* (Budapest), pp. 211-214.

Chou, W. (**1999**). "Maximum a Posteriori Linear Regression with Elliptically Symmetric Matrix Priors," in *Eurospeech* (Budapest), pp. 1-4.

Clemins, P. J. (**2005**). "Automatic Speaker Identification and Classification of Animal Vocalizations. Doctoral Dissertation," (Marquette University, Milwaukee, WI).

Clemins, P. J., and Johnson, M. T. (**2005**). "Unsupervised classification of beluga whale vocalizations," in *150th Meeting of the Acoustical Society of America* (Minneapolis, Minnesota), pp. 2470-2470.

Clemins, P. J., and Johnson, M. T. (**2006**). "Generalized perceptual linear prediction features for animal vocalization analysis," The Journal of the Acoustical Society of America **120**(1), 527-534.

Clemins, P. J., Johnson, M. T., Leong, K. M., and Savage, A. (**2005**). "Automatic classification and speaker identification of African elephant (*Loxodonta africana*) vocalizations," Journal of the Acoustical Society of America **117**(2), 956-963.

Clemins, P. J., Trawicki, M. B., Adi, K., Tao, J., and Johnson, M. T. (**2006**). "Generalized perceptual features for vocalization analysis across multiple species," in *Proceedings of the IEEE ICASSP* (Paris, France), pp. I253 - I256.

Cleveland, J., and Snowdon, C. T. (**1982**). "The complex vocal repertoire of the adult cotton-top tamarin (*Saguinus oedipus oedipus*)," Z. Tierpsychol. **58**, 231-270.

Dale, S. (**2001**). "Causes of population decline in ortolan bunting in Norway," in *Proceedings in 3rd International Ortolan Symposium*, pp. 33-41.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (**1977**). "Maximum likelihood from incomplete data via the EM algorithm," Journal of the Royal Statistical Society **39(1)**, 1–38.

Derr, A., and Schwartz, R. (**1989**). "A simple statistical class grammar for measuring speech recognition performance," in *Proceedings of the workshop on Speech and Natural Language* (Cape Cod, Massachusetts), pp. 147 - 149.

Forney, G. D. (**1973**). "The Viterbi Algorithm," Proceedings of the IEEE **61**, 268-278.

Gales, M. J. F. (**1998**). "Maximum likelihood linear transformations for HMM-based speech recognition," Computer Speech and Language **12**, 75-98.

Gales, M. J. F., and Woodland, P. C. (**1996**). "Mean and variance adaptation within the MLLR framework," Computer Speech and Language **10**, 249-264.

Gauvain, J. L., and Lee, C. H. (**1994**). "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," IEEE Transactions on Acoustics, Speech and Signal Processing **2, No. 2**, 291-298.

Glass, J., Chang, J., and McCandless, M. (**1996**). "A probabilistic framework for feature-based speech recognition," in *Proc. Intl. Conf. on Spoken Language Processing* (Philadelphia), pp. 2277-2280.

Greenwood, D. D. (**1961**). "Critical bandwidth and the frequency coordinates of the basilar membrane," The Journal of the Acoustical Society of America **33**(10), 1344-1356.

Hazen, T. J. (**2000**). "A comparison of novel techniques for rapid speaker adaptation," Speech Communications **31**, 15-33.

Hazen, T. J., and Glass, J. R. (**1997**). "A comparison of novel techniques for instantaneous speaker adaptation," Proceedings of Eurospeech, 2047-2050.

Hu, Z., Barnard, E., and Vermeulen, P. (**1998**). "Speaker normalization using correlations

among classes," in *Proc. Workshop on Speech Recognition, Understanding, Processing* (Hong Kong).

Huang, X., Acero, A., and Hon, H.-W. (**2001**). *Spoken Language Processing* (Prentice Hall, Upper Saddle River, New Jersey).

Janik, V. M., Sayigh, L. S., and Wells, R. S. (**2006**). "Signature whistle shape conveys identity information to bottlenose dolphins," in *Proceedings of the National Academy of Sciences of the USA*, pp. 8293-8297.

Jelinek, F. (**1999**). *Statistical Methods for Speech Recognition* (MIT Press, Cambridge, MA).

Johnson, M. T., Darre, M., Savage, A., Scheifele, P., and vonMuggenthaler, E. (**2003**). "The Dr. Dolittle Project: A Framework for Classification and Understanding of Animal Vocalizations," in *National Science Foundation under Grant No. IIS-0326395*.

Jolliffe, I. T. (**2002**). *Principal Components Analysis* (Springer-Verlag, New York).

Juang, B.-H. (**1984**). "On the Hidden Markov Model and Dynamic Time Warping for Speech Recognition - A Unifies View," AT&T Bell Laboratories Technical Journal **63**(1213-1243).

Juang, B.-H. (**1985**). "Maximum likelihood estimation for mixture multivariate stochastic observation of Markov chains," AT&T Technical Journal **64**(6), 1235-1249.

Juang, B.-H., Levinson, S. E., and Sondhi, M. M. (**1986**). "Maximum likelihood estimation for multivariate mixture observations of markov chains," IEEE Transactions on Information Theory **32**, 307-309.

Kogan, J. A., and Margoliash, D. (**1998**). "Automated recognition of bird song elements from continuous recordings using dynamic time warping and hidden Markov models: A comparative study," The Journal of the Acoustical Society of America **103**(4), 2185-2196.

Kuhn, R. (**1998**). "Eigenvoices for Speaker Adaptation," in *Int. Conf. on Spoken Language Processing* (Sydney, Australia), pp. 1771-1774.

Kuhn, R., Junqua, J.-C., Nguyen, P., and Niedzielski, N. (**2000**). "Rapid speaker adaptation in eigenvoice space," IEEE Transactions on Speech and Audio Processing **8**(6), 695-707.

Kwok, J. T., Mak, B., and Ho, S. (**2003**). "Eigenvoice Speaker Adaptation via Composite Kernel PCA," in *Advances in Neural Information Processing Systems* (Vancouver, Canada).

Leggetter, C. J., and Woodland, P. C. (**1995**). "Maximum likelihood linear regression for speaker adaptation of continuous density hidden markov models," in *Computer Speech and Language*, pp. 171-185.

LePage, E. L. (**2003**). "The mammalian cochlear map is optimally warped," The Journal of

the Acoustical Society of America **114**(2), 896-906.

Li, X., Tao, J., Johnson, M. T., Soltis, J., Savage, A., Leong, K. M., and Newman, J. D. (**2007**). "Stress and emotion classification using jitter and shimmer features," in *Proceedings of the IEEE ICASSP* (Honolulu, Hawaii), pp. IV1081-1084.

Liporace, L. A. (**1982**). "Maximum likelihood estimation for multivariate observations of Markov sources," IEEE Transactions on Information Theory **28**, 729-734.

McCowan, B., and Hooper, S. L. (**2002**). "Individual acoustic variation in Belding's ground squirrel alarm chirps in the High Sierra Nevada," The Journal of the Acoustical Society of America **111**(3), 1157-1160.

Meignier, S., Moraru, D., Fredouille, C., Bonastre, J.-F., and Besacier, L. (**2005**). "Step-by-Step and integrated approaches in broadcast news speaker diarization," Comput. Speech Lang. **20**, 303-330.

Moh, Y., Nguyen, P., and Junqua, J.-C. (**2003**). "Toward domain independent clustering," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* (China), pp. 85-88.

Moon, T. K. (**1996**). "The Expectation-Maximization Algorithm," in *IEEE Signal Processing Magazine*, pp. 47-60.

Moraru, D., Meignier, S., Besacier, L., Bonastre, J.-F., and Magrin-Chagnolleau, I. (**2003**). "The ELISA consortium approaches in speaker segmentation during the NIST 2002 speaker recognition evaluation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*

Nguyen, P., Rigazio, L., Moh, Y., and Junqua, J. C. (**2002**). "Rich transcription 2002 site report," in *Proc. Rich Transcription Workshop (RT-02)* (Panasonic speech technology laboratory (PSTL)).

Osiejuk, T. S., Ratynska, K., Cygan, J. P., and Dale, S. (**2003**). "Song structure and repertoire variation in ortolan bunting (*Emberiza hortulana* L.) from isolated Norweigian population," Annales Zoologici Fennici **40**, 3-16.

Osiejuk, T. S., Ratynska, K., Cygan, J. P., and Dale, S. (**2005**). "Frequency shift in homologue syllables of the Ortolan Bunting *Emberiza hortulana*," Behavioural Processes **68**, 69-83.

Pardo, J. M., Anguera, X., and Wooters, C. (**2007**). "Speaker diarization for multiple-distant-microphone meetings using several sources of information," IEEE Transactions on computers **56**(9).

Parijs, S. M. V., Smith, J., and Corkeron, P. J. (**2002**). "Using calls to estimate the abundance of inshore Dolphins: a case study with Pacific humpback dolphins (Sousa Chinensis)," Journal of Applied Ecology **39**, 853-864.

Price, P., Fisher, W. M., Bernstein, J., and Pallett, D. S. (**1993**). *Resource Management RM1 2.0* (Linguistic Data Consortium, Philadelphia).

Rabiner, L. R. (**1989**). "Tutorial on hidden Markov models and selected applications in speech recognition," Proceedings of the IEEE **77**, 257-286.

Rabiner, L. R., and Juang, B. H. (**1993**). *Fundamentals of Speech Recognition* (Prentice-Hall, Englewood Cliffs, NJ).

Rabiner, L. R., and Levinson, S. E. (**1981**). "Isolated and Connected Word Recognition--Theory and Selected Applications," IEEE Transactions on Communications **29**(5), 621-659.

Reby, D., André-Obrecht, R., Galinier, A., Farinas, J., and Gargnelutti, B. (**2006**). "Cepstral coefficients and hidden Markov models reveal idiosyncratic voice characteristics in red deer (*Cervus elaphus*) stags," Journal of the Acoustical Society of America **120**(6), 4080-4089.

Reynolds, D. A., and Torres-Carrasquillo, P. (**2004**). "The MIT Lincoln Laboratory RT-04F diarization systems: Applications to broadcast audio and telephone conversations," in *Proc. Fall 2004 Rich Transcription Workshop (RT-04)* (Palisades, NY).

Scholkopf, B., Smola, A., and Muller, K. R. (**1998**). "Nonlinear component analysis as a kernel eigenvalue problem," Neural Computation **10**, 1299-1319.

Sczewczyk, R., Osterweil, E., Polastre, J., Hamilton, M., Mainwaring, A., and Estrin, D. (**2004**). "Habitat monitoring with sensor networks," Communications of the ACM **47**, 34-40.

Sinha, R., Tranter, S. E., Gales, M. J. F., and Woodland, P. C. (**2005**). "The Cambridge University March 2005 speaker diarization system," in *Proc. Eur. Conf. Speech Commun. Technol.* (Lisbon, Portugal), pp. 2437-2440.

Sjare, B. L., and Smith, T. G. (**1986**). "The vocal repertoire of white whales, *Delphinapterus leucas*, summering the Cunningham Inlet, Northwest Territories," Canadian Journal of Zoology **64**, 407-415.

Tao, J., and Johnson, M. T. (**2008**). "Maximum likelihood linear regression for acoustic model adaptation in ortolan bunting (Emberiza hortulana L.) vocalization recognition," The Journal of the Acoustical Society of America **123**(3), 1318-1328.

Tranter, S. E., and Douglas, A. R. (**2006**). "An overview of automatic speaker diarization systems," IEEE Transaction on Audio, Speech, and Language Processing **14**(5), 1557-1565.

Trawicki, M. B., Johnson, M. T., and Osiejuk, T. S. (**2005**). "Automatic song-type classification and speaker identification of Norwegian ortolan bunting (*Emberiza hortulana*) vocalizations," in *2005 IEEE Workshop on Machine Learning for Signal Processing* (Mystic, Connecticut, USA), pp. 277- 282.

Turk, M., and Pentland, A. (**1991**). "Face recognition using eigenface," Proceedings of the International Conference on Computer Vision and Pattern Recognition, 586-591.

Vignal, C., Mathevon, N., and Mottin, S. (**2004**). "Audience drives male songbird response to partner's voice," Nature **430**, 448-451.

Westwood, R. (**1999**). "Speaker Adaptation Using Eigenvoices," in *Department of Engineering* (Cambridge University, Lodon, UK), p. 53.

Wooters, C., Fung, J., Peskin, B., and Anguera, X. (**2004**). "Toward Robust speaker segmentation: The ICSI-SRI Fall 2004 Diarization System," in *Proc. Fall 2004 Rich Transcription Workshop (RT-04)* (Palisades, NY).

Young, S., Evermann, G., Hain, T., Kershaw, D., Odell, J., Ollason, D., Povey, D., Valtchev, V., and Woodland, P. (**2002**). *the HTK Book (for HTK Version 3.2.1)* (Cambridge University Engineering Department, UK).

Yu, K. (**2006**). "Adaptive training for large vocabulary continuous speech recognition," in *Department of Engineering* (University of Cambridge, Cambridge, UK).

APPENDIX A EXPERIMENTAL RESULTS

A.1. Human Speech

A.1.1 Single Gaussian Monophone Models

No. of EVs vs. Adaptation Utterances	1	2	3	4	5	6	7	8	9	10
1	78.0	77.8	77.7	77.7	77.8	77.8	77.8	77.8	77.7	77.7
2	78.0	77.9	77.9	77.9	77.9	77.8	77.8	77.8	77.8	77.8
3	77.9	77.9	77.9	78.0	77.9	77.9	77.9	77.9	77.9	77.9
4	78.0	77.9	78.1	78.1	78.0	78.0	78.0	77.9	77.9	77.9
5	78.0	78.1	78.1	78.1	78.1	78.0	78.1	78.0	77.9	77.9
6	78.0	78.0	78.1	78.2	78.1	78.0	78.0	78.1	78.0	77.9
7	78.0	78.1	78.1	78.1	78.0	77.9	78.0	78.0	77.9	77.9
8	78.3	78.1	78.1	78.1	78.3	78.2	78.1	78.1	78.0	78.0
9	78.1	78.0	78.1	78.1	78.1	78.1	78.1	78.1	78.0	78.2
10	78.2	78.1	78.1	78.2	78.2	78.3	78.1	78.1	78.1	78.1
11	78.1	78.0	78.0	78.2	78.2	78.2	78.0	77.9	77.9	77.9
12	78.1	77.9	78.1	78.3	78.3	78.3	78.2	78.2	78.3	78.2
13	78.1	78.0	78.1	78.3	78.5	78.6	78.3	78.2	78.3	78.2
14	78.4	78.4	78.7	78.7	78.6	78.6	78.5	78.5	78.5	78.5
15	78.6	78.7	79.0	78.8	78.7	78.7	78.6	78.6	78.7	78.7
20	78.5	78.7	78.9	78.9	78.8	78.9	78.8	78.8	78.8	78.8
30	78.6	78.6	78.6	78.8	78.8	78.7	78.7	78.7	78.7	78.7
40	78.2	78.5	78.8	78.7	78.7	78.8	78.9	78.9	78.8	78.6
50	78.2	78.4	78.6	78.7	78.6	78.6	78.8	78.6	78.6	78.6
60	78.0	78.4	78.7	78.5	78.9	79.0	78.9	78.8	78.6	78.6
70	77.5	78.6	78.9	78.7	78.9	79.0	79.2	78.9	78.8	78.9
80	77.0	78.7	78.9	78.9	79.0	79.1	79.2	79.2	79.2	79.1
90	76.5	78.4	78.8	79.0	79.0	79.1	79.1	79.2	79.0	78.9
100	76.4	78.1	78.6	78.8	78.8	78.9	79.0	78.9	78.9	78.8
109	76.3	78.5	78.9	78.8	79.0	79.2	79.2	79.0	78.8	78.8

Table A.1 Word accuracies of Eigenvoice adapted single Gaussian monophone system using PCA correlation implementation

No. of EVs vs. Adaptation Utterances	1	2	3	4	5	6	7	8	9	10
1	79.0	79.1	79.0	79.1	79.1	79.2	79.2	79.2	79.1	79.1
2	79.2	79.3	79.3	79.2	79.2	79.3	79.3	79.3	79.2	79.2
3	79.4	79.5	79.4	79.5	79.5	79.5	79.5	79.5	79.5	79.4
4	79.6	79.7	79.5	79.7	79.8	79.7	79.6	79.6	79.5	79.5
5	79.7	79.6	79.5	79.7	79.7	79.7	79.8	79.6	79.6	79.6
6	80.3	80.3	80.1	80.1	80.1	80.1	80.3	80.2	80.2	80.2
7	80.5	80.3	80.3	80.2	80.4	80.3	80.3	80.1	80.2	80.2
8	80.3	80.1	80.1	80.3	80.2	80.3	80.3	80.3	80.3	80.1
9	80.6	80.5	80.3	80.4	80.3	80.3	80.4	80.4	80.4	80.5
10	80.6	80.6	80.8	80.6	80.6	80.7	80.7	80.7	80.7	80.8
11	80.7	80.6	80.8	80.7	80.7	80.7	80.7	80.7	80.8	80.8
12	80.4	80.5	80.7	80.6	80.5	80.6	80.6	80.5	80.6	80.5
13	80.7	80.6	80.8	80.7	80.7	80.7	80.9	80.7	80.8	80.8
14	80.5	80.8	80.8	80.8	80.8	80.6	80.8	80.9	80.9	81.0
15	80.3	80.4	80.5	80.6	80.5	80.7	80.5	80.8	80.9	80.9
20	80.5	80.3	80.4	80.6	80.7	80.8	80.8	80.8	80.9	80.9
30	81.0	81.4	81.1	81.3	81.5	81.4	81.4	81.4	81.4	81.4
40	81.0	81.4	81.2	81.3	81.4	81.5	81.6	81.4	81.4	81.2
50	80.6	81.5	81.0	81.3	81.6	81.6	81.6	81.7	81.6	81.6
60	80.3	81.3	81.1	81.1	81.4	81.4	81.4	81.7	81.6	81.6
70	80.1	81.1	81.0	81.1	81.3	81.3	81.4	81.5	81.6	81.6
80	79.8	81.2	81.0	81.0	81.2	81.2	81.8	81.8	81.7	81.8
90	79.9	81.3	81.1	80.9	81.3	81.4	81.5	81.6	81.7	81.7
100	79.8	81.4	81.4	81.3	81.4	81.4	81.8	81.8	81.7	81.9
109	79.7	81.3	81.4	81.4	81.4	81.5	82.0	82.0	81.8	82.0

Table A.2 Word accuracies of Eigenvoice adapted single Gaussian monophone system using PCA covariance implementation

No. of ECs vs. Adaptation Utterances	1	2	3	4	5	6	7	8	9	10
1	77.3	77.1	77.1	77.1	77.2	77.2	77.1	77.1	77.1	77.1
2	77.2	77.2	77.1	77.2	77.1	77.3	77.2	77.3	77.2	77.2
3	77.2	77.3	77.4	77.3	77.4	77.4	77.3	77.4	77.4	77.5
4	77.3	77.4	77.4	77.4	77.5	77.5	77.5	77.5	77.5	77.6
5	77.5	77.6	77.6	77.6	77.6	77.6	77.6	77.6	77.7	77.7
6	77.7	77.5	77.7	77.7	77.6	77.6	77.7	77.7	77.8	77.7
7	77.7	77.6	77.6	77.7	77.7	77.7	77.7	77.6	77.6	77.7
8	77.8	77.6	77.6	77.6	77.6	77.6	77.7	77.6	77.6	77.7
9	77.9	77.5	77.7	77.6	77.6	77.6	77.6	77.6	77.6	77.7
10	77.7	77.6	77.6	77.8	77.8	77.7	77.7	77.6	77.6	77.7
11	77.8	77.7	78.0	78.0	77.9	77.8	77.9	77.8	77.8	77.9
12	78.0	78.0	78.0	78.0	77.9	78.0	78.1	78.0	78.0	78.0
13	78.0	78.0	78.0	77.9	77.9	78.0	78.1	78.0	78.1	78.1
14	77.8	77.9	78.0	77.8	77.9	78.0	78.0	77.9	77.9	77.9
15	77.8	77.8	77.8	78.0	78.0	78.0	77.9	77.8	77.9	77.9
20	77.5	77.8	77.8	77.8	77.8	77.8	77.9	77.8	77.8	77.9
30	77.7	77.9	78.0	78.2	78.2	78.2	78.1	78.1	77.9	78.0
40	77.6	77.9	78.1	78.2	78.3	78.1	78.2	78.1	78.1	78.1
50	77.2	77.7	78.1	78.2	78.4	78.3	78.2	78.2	78.2	78.1
60	76.6	77.9	78.3	78.7	78.9	78.8	78.9	79.0	78.7	78.7
70	76.2	78.0	78.4	78.3	78.5	78.6	78.7	78.7	78.9	78.8
80	75.6	77.7	78.5	78.7	79.0	79.1	79.1	79.2	79.1	79.2
90	75.0	77.9	78.4	78.9	78.9	78.9	79.0	78.9	78.9	79.0
100	73.7	77.4	78.1	78.8	79.0	78.9	79.1	79.2	79.3	79.3
109	72.3	76.9	77.8	78.4	78.7	78.8	78.9	79.0	79.2	79.2

Table A.3 Word accuracies of Eigen-clustering adapted single Gaussian monophone system using PCA correlation implementation

No. of ECs vs. Adaptation Utterances	1	2	3	4	5	6	7	8	9	10
1	77.9	78.4	78.4	78.3	78.5	78.3	78.3	78.3	78.2	78.3
2	78.1	78.4	78.4	78.4	78.4	78.5	78.4	78.3	78.4	78.4
3	78.8	78.9	78.9	78.9	78.9	78.9	79.0	79.0	78.9	78.8
4	78.7	79.0	79.1	79.3	79.3	79.2	79.2	79.2	79.2	79.2
5	78.7	78.9	79.0	79.1	79.1	79.2	79.1	79.2	79.3	79.1
6	78.7	78.8	78.7	78.9	78.9	78.9	78.9	79.0	78.9	78.9
7	78.8	78.9	79.0	79.3	79.3	79.2	79.3	79.2	79.4	79.4
8	79.0	79.1	79.0	79.4	79.4	79.2	79.1	79.2	79.1	79.2
9	79.0	79.4	79.2	79.4	79.5	79.6	79.5	79.7	79.7	79.5
10	79.2	79.5	79.6	79.6	79.5	79.7	79.7	79.7	79.7	79.7
11	79.2	79.5	79.5	79.6	79.6	79.5	79.5	79.7	79.8	79.7
12	79.2	79.8	79.3	79.5	79.6	79.4	79.6	79.7	79.6	79.5
13	79.0	79.6	79.3	79.5	79.5	79.4	79.4	79.6	79.7	79.5
14	79.0	79.5	79.5	79.6	79.4	79.4	79.3	79.4	79.6	79.5
15	79.2	79.6	79.5	79.7	79.4	79.3	79.5	79.4	79.5	79.6
20	78.7	79.2	79.3	79.5	79.4	79.4	79.4	79.4	79.3	79.4
30	77.8	78.7	78.9	79.4	79.5	79.4	79.6	79.7	79.9	79.8
40	76.9	78.7	79.3	79.8	79.6	79.8	80.0	80.1	80.1	80.1
50	76.3	78.6	79.0	79.8	80.0	80.0	80.2	80.3	80.4	80.2
60	75.6	78.3	79.0	79.8	80.0	80.1	80.1	80.4	80.4	80.4
70	74.5	78.4	79.2	80.0	79.9	80.2	80.4	80.7	80.7	80.6
80	73.2	78.1	79.3	79.8	80.3	80.5	80.6	80.5	80.9	80.9
90	72.0	77.8	79.0	79.8	80.0	80.4	80.5	80.5	80.7	80.7
100	71.4	77.7	79.0	80.0	80.2	80.6	80.7	80.9	80.8	80.8
109	70.3	77.1	78.7	79.4	79.4	80.3	80.7	80.6	80.6	80.9

Table A.4 Word accuracies of Eigen-clustering adapted single Gaussian monophone system using PCA covariance implementation

System vs. Adaptation data size	1	2	3	4	5	6	7	8	9	10
MLLR	77.3	77.5	78.4	79.3	80.6	80.7	81.4	81.9	82.1	82.3
MAP	76.5	69.9	72.5	72.6	75.0	78.5	79.4	80.8	82.2	82.2
MAPLR	79.7	79.6	79.5	79.7	79.7	79.7	79.8	79.6	79.6	79.6
EV5	80.6	80.6	80.8	80.6	80.6	80.7	80.7	80.7	80.7	80.8
EV10	80.3	80.4	80.5	80.6	80.5	80.7	80.5	80.8	80.9	80.9
EV15	80.5	80.3	80.4	80.6	80.7	80.8	80.8	80.8	80.9	80.9
EV20	81.0	81.4	81.1	81.3	81.5	81.4	81.4	81.4	81.4	81.4
EV30	78.7	78.9	79.0	79.1	79.1	79.2	79.1	79.2	79.3	79.1
EC5	79.2	79.5	79.6	79.6	79.5	79.7	79.7	79.7	79.7	79.7
EC10	79.2	79.6	79.5	79.7	79.4	79.3	79.5	79.4	79.5	79.6
EC15	78.7	79.2	79.3	79.5	79.4	79.4	79.4	79.4	79.3	79.4
EC20	77.8	78.7	78.9	79.4	79.5	79.4	79.6	79.7	79.9	79.8
EC30	78.8	80.4	80.6	80.9	81.6	81.7	81.8	82.1	82.2	82.2
EC30+MAP	77.3	77.5	78.4	79.3	80.6	80.7	81.4	81.9	82.1	82.3

Table A.5 Word accuracies of the six adaptation methods on single Gaussian monophone system

A.1.2 Four-Mixture Gaussian Monophone Models

No. of EVs vs. Adaptation Utterances	1	2	3	4	5	6	7	8	9	10
1	86.6	86.7	86.7	86.7	86.7	86.7	86.7	86.7	86.7	86.7
2	86.7	86.9	86.8	86.8	86.8	86.8	86.8	86.8	86.9	86.8
3	86.7	86.9	86.8	86.8	86.8	86.8	86.7	86.8	86.7	86.8
4	86.8	86.8	86.9	86.9	86.9	86.9	86.9	86.8	86.8	86.8
5	86.9	86.8	86.8	86.8	86.8	86.8	86.8	86.8	86.8	86.8
6	86.9	87.0	87.0	86.9	87.0	87.0	87.0	87.0	86.9	86.9
7	86.9	87.1	87.0	87.0	87.0	87.0	87.0	87.0	87.0	87.0
8	86.9	87.0	87.0	86.9	86.9	86.9	87.0	86.9	87.0	86.9
9	86.8	87.1	87.0	87.0	87.0	87.0	87.0	87.1	87.0	87.0
10	87.0	87.0	87.0	87.0	86.9	86.9	87.0	87.0	87.0	87.0
11	87.0	86.9	87.0	87.0	86.9	86.9	87.0	87.0	87.0	87.0
12	87.0	87.1	87.1	87.0	86.9	86.9	87.0	87.0	87.0	87.0
13	87.2	87.2	87.3	87.2	87.1	87.1	87.1	87.1	87.1	87.1
14	87.1	87.3	87.3	87.3	87.1	87.2	87.2	87.2	87.2	87.2
15	87.3	87.4	87.3	87.2	87.2	87.3	87.2	87.2	87.2	87.2
20	87.3	87.4	87.4	87.4	87.3	87.3	87.3	87.3	87.3	87.2
30	87.3	87.2	87.2	87.2	87.1	87.2	87.2	87.2	87.1	87.1
40	87.1	87.3	87.3	87.2	87.2	87.2	87.3	87.2	87.2	87.2
50	87.2	87.3	87.6	87.4	87.4	87.3	87.3	87.3	87.3	87.4
60	87.2	87.2	87.5	87.4	87.4	87.2	87.3	87.4	87.3	87.3
70	87.1	87.3	87.5	87.5	87.4	87.4	87.5	87.4	87.4	87.4
80	87.0	87.3	87.4	87.5	87.4	87.5	87.4	87.4	87.5	87.5
90	86.9	87.3	87.3	87.4	87.4	87.4	87.5	87.5	87.5	87.5
100	86.8	87.3	87.5	87.5	87.5	87.4	87.5	87.5	87.5	87.6
109	86.7	87.3	87.5	87.4	87.5	87.6	87.6	87.5	87.5	87.5

Table A.6 Word accuracies of Eigenvoice adapted four-mixture Gaussian monophone system using PCA correlation implementation

No. of EVs vs. Adaptation Utterances	1	2	3	4	5	6	7	8	9	10
1	87.6	87.5	87.6	87.7	87.7	87.7	87.7	87.7	87.7	87.7
2	87.6	87.6	87.6	87.7	87.7	87.7	87.6	87.6	87.7	87.7
3	87.6	87.7	87.7	87.7	87.7	87.7	87.7	87.7	87.7	87.7
4	87.7	87.7	87.6	87.7	87.7	87.7	87.6	87.7	87.7	87.7
5	87.7	87.9	87.8	87.8	87.9	87.9	87.8	87.8	87.9	87.9
6	88.1	88.2	88.4	88.2	88.3	88.3	88.2	88.3	88.3	88.3
7	88.0	88.3	88.3	88.2	88.2	88.2	88.2	88.2	88.2	88.2
8	88.0	88.3	88.5	88.3	88.3	88.3	88.4	88.4	88.4	88.4
9	88.1	88.2	88.4	88.4	88.3	88.4	88.3	88.3	88.3	88.3
10	88.2	88.3	88.2	88.3	88.4	88.4	88.4	88.3	88.4	88.3
11	88.3	88.2	88.3	88.3	88.3	88.3	88.3	88.5	88.4	88.4
12	88.3	88.3	88.3	88.2	88.2	88.2	88.2	88.3	88.2	88.2
13	88.2	88.3	88.3	88.2	88.2	88.3	88.3	88.4	88.3	88.3
14	88.2	88.3	88.4	88.3	88.3	88.4	88.5	88.4	88.4	88.4
15	88.4	88.4	88.4	88.4	88.4	88.4	88.4	88.4	88.4	88.4
20	88.4	88.5	88.7	88.7	88.7	88.7	88.7	88.6	88.6	88.7
30	88.4	88.8	89.0	88.9	89.1	88.9	88.8	88.9	88.8	88.8
40	88.3	88.7	88.9	88.7	88.9	88.9	88.8	88.8	88.9	88.8
50	88.3	88.8	89.1	88.8	89.0	89.0	89.0	89.0	89.1	89.2
60	88.3	88.9	89.1	89.0	89.2	89.2	89.2	89.1	89.1	89.1
70	88.6	89.0	89.3	89.2	89.3	89.2	89.2	89.2	89.1	89.1
80	88.5	89.0	89.1	89.1	89.3	89.1	89.1	89.1	89.0	89.1
90	88.2	88.8	89.1	89.1	89.2	89.1	89.1	88.9	89.0	89.1
100	88.2	88.8	89.2	89.1	89.2	89.1	89.2	89.1	89.1	89.2
109	88.6	88.8	89.1	89.1	89.1	89.2	89.3	89.2	89.3	89.2

Table A.7 Word accuracies of Eigenvoice adapted four-mixture Gaussian monophone system using PCA covariance implementation

No. of ECs vs. Adaptation Utterances	1	2	3	4	5	6	7	8	9	10
1	86.7	86.6	86.6	86.6	86.6	86.6	86.6	86.7	86.6	86.7
2	86.6	86.6	86.6	86.6	86.6	86.7	86.6	86.7	86.7	86.6
3	86.7	86.6	86.7	86.7	86.7	86.7	86.7	86.7	86.6	86.6
4	86.7	86.7	86.7	86.7	86.7	86.7	86.7	86.7	86.7	86.7
5	86.6	86.6	86.7	86.6	86.6	86.7	86.7	86.7	86.7	86.7
6	86.6	86.5	86.7	86.7	86.6	86.7	86.7	86.6	86.6	86.6
7	86.5	86.6	86.7	86.6	86.6	86.7	86.7	86.6	86.6	86.6
8	86.5	86.6	86.6	86.7	86.6	86.6	86.7	86.7	86.7	86.7
9	86.4	86.6	86.7	86.7	86.7	86.7	86.7	86.7	86.7	86.7
10	86.4	86.5	86.6	86.6	86.6	86.6	86.6	86.7	86.6	86.6
11	86.5	86.5	86.6	86.6	86.6	86.6	86.6	86.6	86.6	86.6
12	86.5	86.5	86.5	86.6	86.6	86.6	86.6	86.6	86.6	86.6
13	86.4	86.5	86.5	86.5	86.5	86.5	86.6	86.5	86.5	86.6
14	86.3	86.5	86.4	86.5	86.5	86.5	86.6	86.5	86.6	86.6
15	86.3	86.6	86.6	86.6	86.5	86.5	86.6	86.5	86.6	86.6
20	86.4	86.5	86.6	86.5	86.6	86.6	86.6	86.6	86.6	86.6
30	86.4	86.5	86.6	86.5	86.6	86.6	86.6	86.6	86.6	86.6
40	86.4	86.5	86.6	86.5	86.6	86.6	86.6	86.6	86.6	86.6
50	86.4	86.5	86.6	86.5	86.6	86.6	86.6	86.6	86.6	86.6
60	86.4	86.5	86.6	86.5	86.6	86.6	86.6	86.6	86.6	86.6
70	86.4	86.5	86.6	86.5	86.6	86.6	86.6	86.6	86.6	86.6
80	86.4	86.5	86.6	86.5	86.6	86.6	86.6	86.6	86.6	86.6
90	86.4	86.5	86.6	86.5	86.6	86.6	86.6	86.6	86.6	86.6
100	86.4	86.5	86.6	86.5	86.6	86.6	86.6	86.6	86.6	86.6
109	86.4	86.5	86.6	86.5	86.6	86.6	86.6	86.6	86.6	86.6

Table A.8 Word accuracies of Eigen-clustering adapted four-mixture Gaussian monophone system using PCA correlation implementation

No. of ECs vs. Adaptation Utterances	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	86.5	86.8	86.7	86.7	86.8	86.7	86.8	86.8	86.7	86.7
2	86.5	86.6	86.7	86.7	86.8	86.8	86.8	86.8	86.9	86.9
3	86.8	87.1	87.0	87.1	87.1	87.0	87.1	87.1	87.0	87.0
4	86.8	86.9	86.8	86.8	86.9	86.8	86.8	86.9	86.8	86.9
5	86.7	86.6	86.6	86.7	86.8	86.8	86.8	86.8	87.0	87.0
6	86.7	86.8	86.9	86.7	86.8	86.9	86.9	87.0	87.0	87.0
7	86.6	86.6	86.7	86.7	86.7	86.7	86.8	86.8	86.8	86.8
8	86.6	86.6	86.8	86.7	86.8	86.8	86.8	86.8	86.9	86.9
9	86.5	86.5	86.8	86.7	86.9	86.8	86.8	86.8	86.9	86.9
10	86.6	86.6	86.8	86.8	86.8	86.8	86.8	86.8	86.9	86.9
11	86.6	86.6	86.9	86.9	86.9	86.8	86.9	86.8	87.0	87.0
12	86.7	86.7	87.0	87.1	87.1	87.0	87.1	87.1	87.1	87.2
13	86.7	86.9	87.1	87.0	87.1	87.1	87.2	87.1	87.1	87.2
14	86.8	86.9	87.0	87.0	87.1	87.1	87.2	87.2	87.2	87.2
15	86.9	86.9	87.0	87.1	87.1	87.1	87.2	87.2	87.2	87.2
20	87.1	87.2	87.2	87.2	87.3	87.3	87.3	87.4	87.3	87.2
30	87.2	87.5	87.4	87.4	87.5	87.5	87.5	87.5	87.5	87.4
40	87.2	87.6	87.8	87.7	87.8	87.6	87.7	87.6	87.7	87.6
50	87.2	87.5	87.6	87.6	87.8	87.8	87.8	87.8	87.7	87.7
60	87.0	87.6	87.6	87.8	87.8	87.7	87.8	87.7	87.7	87.7
70	86.9	87.5	87.3	87.4	87.6	87.6	87.7	87.8	87.8	87.8
80	86.8	87.4	87.6	87.4	87.8	87.8	88.0	87.9	88.0	87.9
90	86.3	87.2	87.5	87.7	87.7	87.7	87.9	87.8	88.0	88.0
100	86.2	87.0	87.1	87.3	87.6	87.7	87.9	87.9	88.0	88.0
109	86.1	87.0	87.2	87.3	87.7	87.9	87.9	87.9	88.0	88.1

Table A.9 Word accuracies of Eigen-clustering adapted four-mixture Gaussian monophone system using PCA covariance implementation

System vs. Adaptation data size	1	2	3	4	5	6	7	8	9	10
MLLR	86.6	87.1	88.0	88.2	87.7	88.7	89.0	89.2	89.3	89.3
MAP	86.6	87.4	87.9	88.3	88.7	88.9	89.3	89.3	89.5	89.6
MAPLR	87.0	85.8	88.0	88.3	88.6	88.8	89.2	89.4	89.7	89.6
EV5	87.7	87.9	87.8	87.8	87.9	87.9	87.8	87.8	87.9	87.9
EV10	88.2	88.3	88.2	88.3	88.4	88.4	88.4	88.3	88.4	88.3
EV15	88.4	88.4	88.4	88.4	88.4	88.4	88.4	88.4	88.4	88.4
EV20	88.4	88.5	88.7	88.7	88.7	88.7	88.7	88.6	88.6	88.7
EV30	88.4	88.8	89.0	88.9	89.1	88.9	88.8	88.9	88.8	88.8
EC5	86.7	86.6	86.6	86.7	86.8	86.8	86.8	86.8	87.0	87.0
EC10	86.6	86.6	86.8	86.8	86.8	86.8	86.8	86.8	86.9	86.9
EC15	86.9	86.9	87.0	87.1	87.1	87.1	87.2	87.2	87.2	87.2
EC20	87.1	87.2	87.2	87.2	87.3	87.3	87.3	87.4	87.3	87.2
EC30	87.2	87.5	87.4	87.4	87.5	87.5	87.5	87.5	87.5	87.4
EC30+MAP	87.8	88.4	88.6	88.8	88.8	89.1	89.2	89.3	89.5	89.8

Table A.10 Word accuracies of the six adaptation methods on four-mixture Gaussian monophone system

A.1.3 Single Gaussian Triphone Models

No. of EVs vs. Adaptation Utterances	1	2	3	4	5	6	7	8	9	10
1	89.2	89.0	89.1	89.1	89.0	89.0	89.1	89.0	89.2	89.1
2	89.2	89.1	89.2	89.1	88.9	88.9	89.0	89.1	89.0	88.9
3	89.1	89.2	89.2	89.3	89.1	89.1	89.3	89.3	89.0	89.1
4	89.1	89.2	89.4	89.4	89.3	89.1	89.1	89.0	89.1	89.2
5	89.3	89.4	89.2	89.5	89.5	89.3	89.2	89.3	89.2	89.2
6	89.3	89.2	89.4	89.4	89.3	89.1	89.4	89.3	89.2	89.0
7	89.2	89.6	89.4	89.2	89.3	89.1	89.2	89.3	89.2	89.2
8	89.6	89.4	89.3	89.2	89.5	89.5	89.2	89.4	89.3	89.3
9	89.3	89.4	89.4	89.4	89.5	89.3	89.4	89.2	89.2	89.5
10	89.2	89.4	89.3	89.4	89.5	89.5	89.3	89.2	89.4	89.4
11	89.2	89.1	89.4	89.5	89.4	89.5	89.2	89.1	89.2	89.2
12	89.5	89.3	89.2	89.4	89.4	89.6	89.6	89.4	89.6	89.5
13	89.4	89.1	89.4	89.6	89.8	89.7	89.4	89.6	89.5	89.5
14	89.7	89.7	89.9	89.8	89.8	89.9	89.8	89.8	89.7	89.7
15	90.0	90.0	90.2	90.2	89.8	89.9	89.7	90.1	89.9	90.1
20	89.8	89.9	90.2	90.3	90.0	90.1	90.0	90.2	90.1	90.1
30	89.9	89.8	89.8	90.1	90.1	89.9	90.0	89.8	90.0	89.9
40	89.2	89.7	90.2	89.9	89.9	90.1	90.2	90.0	90.1	89.7
50	89.4	89.8	89.8	89.8	89.9	89.9	89.9	89.9	89.9	89.9
60	89.1	89.8	89.9	89.7	90.1	90.1	90.0	90.0	89.9	89.8
70	88.7	89.9	90.2	90.0	90.1	90.3	90.5	90.1	90.0	90.1
80	88.3	89.9	90.2	90.2	90.3	90.4	90.4	90.4	90.4	90.3
90	87.7	89.7	90.1	90.3	90.3	90.3	90.4	90.4	90.4	90.1
100	87.6	89.5	89.9	90.0	90.1	90.2	90.1	90.3	90.0	90.1
109	87.6	89.7	90.1	90.1	90.1	90.5	90.5	90.2	90.1	90.2

Table A.11 Word accuracies of Eigenvoice adapted single Gaussian triphone system using PCA correlation implementation

No. of EVs vs. Adaptation Utterances	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	90.1	90.4	90.5	90.3	90.1	90.5	90.5	90.4	90.4	90.5
2	90.5	90.5	90.6	90.5	90.6	90.5	90.6	90.4	90.4	90.5
3	90.6	90.8	90.7	90.8	90.8	90.8	90.9	90.7	90.7	90.8
4	90.8	90.9	90.7	90.9	91.1	90.8	90.9	90.8	90.8	90.8
5	91.0	90.8	90.7	91.0	91.0	91.0	91.1	91.0	90.8	90.8
6	91.6	91.6	91.3	91.2	91.3	91.3	91.7	91.6	91.3	91.4
7	91.9	91.6	91.6	91.4	91.6	91.6	91.6	91.3	91.5	91.6
8	91.7	91.3	91.3	91.6	91.5	91.5	91.6	91.5	91.5	91.4
9	91.7	91.8	91.6	91.5	91.5	91.5	91.6	91.7	91.8	91.7
10	91.8	91.8	92.0	91.9	91.8	91.9	92.0	91.9	92.0	91.9
11	91.9	91.9	92.1	92.0	92.0	92.0	92.0	92.1	92.1	92.2
12	91.7	91.7	91.8	91.8	91.9	91.8	91.9	91.8	91.9	91.7
13	92.0	91.9	92.1	91.9	91.9	92.1	92.1	91.9	92.1	92.1
14	91.8	92.1	92.0	92.0	92.1	91.9	92.1	92.1	92.1	92.3
15	91.4	91.6	91.7	91.9	91.7	91.9	91.8	92.0	92.1	92.2
20	91.9	91.4	91.8	92.0	92.0	91.9	92.1	91.9	92.2	92.4
30	92.3	92.7	92.5	92.5	92.6	92.8	92.6	92.6	92.6	92.6
40	92.2	92.8	92.3	92.5	92.8	92.7	92.8	92.7	92.6	92.3
50	91.8	92.9	92.4	92.5	92.9	92.9	92.8	93.0	92.9	92.8
60	91.7	92.4	92.2	92.5	92.6	92.5	92.5	93.0	93.0	92.9
70	91.4	92.3	92.3	92.4	92.5	92.5	92.5	92.7	92.8	92.7
80	91.0	92.4	92.4	92.2	92.6	92.4	92.9	93.2	92.9	93.0
90	91.1	92.4	92.1	92.3	92.6	92.7	92.9	92.8	92.9	93.0
100	91.3	92.7	92.8	92.4	92.6	92.6	93.1	93.0	93.0	93.1
109	90.8	92.7	92.5	92.7	92.7	92.7	93.0	93.0	92.9	93.2

Table A.12 Word accuracies of Eigenvoice adapted single Gaussian triphone system using PCA covariance implementation

No. of ECs vs. Adaptation Utterances	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	89.2	89.0	89.1	89.0	89.2	89.1	89.2	89.0	89.1	89.1
2	89.1	89.2	89.0	89.0	89.1	89.2	89.1	89.2	89.1	89.2
3	89.1	89.1	89.4	89.2	89.3	89.1	89.1	89.3	89.3	89.3
4	89.2	89.4	89.4	89.4	89.5	89.3	89.5	89.3	89.4	89.6
5	89.4	89.5	89.7	89.6	89.5	89.5	89.5	89.5	89.7	89.6
6	89.5	89.4	89.8	89.6	89.5	89.5	89.8	89.6	89.5	89.5
7	89.6	89.6	89.5	89.5	89.5	89.6	89.6	89.4	89.5	89.7
8	89.7	89.5	89.5	89.6	89.4	89.5	89.6	89.4	89.5	89.5
9	89.9	89.3	89.5	89.5	89.5	89.7	89.6	89.5	89.6	89.5
10	89.5	89.6	89.4	89.6	89.7	89.5	89.5	89.7	89.5	89.6
11	89.7	89.6	89.8	90.0	89.8	89.8	89.8	89.6	89.7	89.7
12	89.8	89.8	90.0	90.0	89.9	89.8	89.9	89.8	89.7	89.9
13	89.9	89.9	89.8	89.8	89.7	90.1	90.0	90.0	90.2	90.0
14	89.8	89.8	89.8	89.5	89.8	90.0	89.8	89.6	89.8	89.8
15	89.6	89.6	89.7	89.8	89.9	89.9	89.8	89.7	89.8	89.9
20	89.3	89.7	89.7	89.8	89.7	89.7	89.8	89.7	89.7	89.8
30	89.6	89.8	89.7	90.2	90.1	90.1	90.1	89.9	89.8	89.8
40	89.5	89.7	89.9	90.0	90.2	90.1	90.1	90.1	90.1	90.0
50	89.2	89.6	90.1	90.2	90.5	90.2	90.1	90.1	90.2	90.0
60	88.6	89.8	90.1	90.5	90.8	90.7	91.0	90.9	90.5	90.6
70	88.2	89.8	90.3	90.3	90.5	90.5	90.7	90.7	91.0	90.8
80	87.3	89.6	90.5	90.6	90.9	91.0	91.0	91.0	91.2	91.1
90	86.9	89.7	90.3	90.8	90.8	90.8	90.9	90.7	90.9	91.0
100	85.6	89.3	89.9	90.9	90.9	90.7	91.0	91.1	91.2	91.2
109	84.1	88.9	89.8	90.2	90.5	90.6	91.0	90.7	90.9	91.3

Table A.13 Word accuracies of Eigen-clustering adapted single Gaussian triphone system using PCA correlation implementation

No. of ECs vs. Adaptation Utterances	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	89.7	90.3	90.2	90.3	90.5	90.2	90.1	90.2	90.2	90.1
2	90.0	90.4	90.3	90.3	90.3	90.4	90.3	90.2	90.2	90.2
3	90.6	90.8	90.9	90.8	90.8	90.9	90.9	90.9	90.8	90.6
4	90.7	90.8	90.8	91.1	91.2	91.1	91.1	91.3	91.3	91.2
5	90.7	90.8	90.8	90.9	91.1	91.2	91.1	91.2	91.3	90.9
6	90.7	90.6	90.7	90.8	90.8	90.8	90.8	90.8	90.8	90.7
7	90.7	90.8	90.9	91.3	91.3	91.0	91.3	91.0	91.3	91.2
8	90.9	91.0	91.0	91.4	91.2	91.1	90.9	90.9	91.0	91.2
9	91.0	91.4	91.3	91.3	91.3	91.5	91.5	91.6	91.5	91.5
10	91.0	91.3	91.5	91.6	91.4	91.6	91.7	91.7	91.7	91.7
11	91.0	91.4	91.4	91.5	91.6	91.5	91.5	91.6	91.7	91.6
12	91.1	91.7	91.2	91.6	91.5	91.3	91.6	91.7	91.6	91.5
13	90.9	91.6	91.0	91.3	91.5	91.4	91.3	91.6	91.7	91.5
14	90.9	91.2	91.5	91.5	91.4	91.4	91.3	91.5	91.5	91.3
15	91.2	91.5	91.4	91.5	91.3	91.3	91.5	91.3	91.3	91.5
20	90.7	91.3	91.2	91.4	91.3	91.5	91.3	91.2	91.2	91.2
30	89.7	90.6	90.7	91.3	91.3	91.3	91.6	91.5	91.8	91.7
40	88.9	90.6	91.3	91.6	91.4	91.8	91.9	92.0	92.0	92.0
50	88.3	90.5	91.0	91.8	92.0	92.0	92.1	92.3	92.3	92.2
60	87.6	90.3	90.9	91.7	92.0	92.1	92.2	92.3	92.2	92.5
70	86.4	90.3	91.0	91.8	91.8	92.1	92.4	92.7	92.7	92.6
80	85.1	90.1	91.1	91.6	92.1	92.4	92.6	92.4	92.8	92.8
90	84.0	89.7	91.0	91.9	91.8	92.3	92.5	92.3	92.6	92.7
100	83.3	89.6	91.1	91.9	92.0	92.4	92.6	92.8	92.8	92.6
109	82.2	88.8	90.6	91.3	91.5	92.3	92.7	92.5	92.4	92.9

Table A.14 Word accuracies of Eigen-clustering adapted single Gaussian triphone system using PCA covariance implementation

System vs. Adaptation data size	1	2	3	4	5	6	7	8	9	10
MLLR	88.9	89.2	91.3	93.0	92.4	92.3	92.4	93.0	93.4	92.2
MAP	88.7	89.7	91.1	91.4	91.8	92.0	92.2	92.3	93.3	93.6
MAPLR	89.3	89.4	90.2	92.5	93.0	92.6	93.1	93.0	92.9	92.3
EV5	91.0	90.8	90.7	91.0	91.0	91.0	91.1	91.0	90.8	90.8
EV10	91.8	91.8	92.0	91.9	91.8	91.9	92.0	91.9	92.0	91.9
EV15	91.4	91.6	91.7	91.9	91.7	91.9	91.8	92.0	92.1	92.2
EV20	91.9	91.4	91.8	92.0	92.0	91.9	92.1	91.9	92.2	92.4
EV30	92.3	92.7	92.5	92.5	92.6	92.8	92.6	92.6	92.6	92.6
EC5	90.7	90.8	90.8	90.9	91.1	91.2	91.1	91.2	91.3	90.9
EC10	91.0	91.3	91.5	91.6	91.4	91.6	91.7	91.7	91.7	91.7
EC15	91.2	91.5	91.4	91.5	91.3	91.3	91.5	91.3	91.3	91.5
EC20	90.7	91.3	91.2	91.4	91.3	91.5	91.3	91.2	91.2	91.2
EC30	89.7	90.6	90.7	91.3	91.3	91.3	91.6	91.5	91.8	91.7
EC30+MAP	91.4	91.6	91.8	92.0	92.2	92.5	92.6	92.7	92.8	92.9

Table A.15 Word accuracies of the six adaptation methods on single Gaussian triphone system

A.2. Animal Vocalization

No. of EVs vs. Adaptation Data (s)	3	6	9	12	15	18	21	24	27	30
1	82.0	82.3	82.0	81.8	81.7	81.7	81.8	81.8	82.0	82.0
2	82.0	82.1	81.9	82.1	82.1	82.1	82.1	82.1	82.1	82.1
3	82.0	82.3	82.0	81.8	81.7	81.7	81.8	81.8	82.0	82.0
4	82.0	82.1	81.9	82.1	82.1	82.1	82.1	82.1	82.1	82.1
5	81.9	82.0	81.8	81.8	81.8	81.8	81.8	81.8	81.8	82.1
6	82.3	82.2	82.1	82.1	82.1	82.1	82.1	82.1	82.1	82.1
7	82.3	82.4	82.4	82.4	82.4	82.4	82.3	82.3	82.3	82.1
8	82.4	82.4	82.1	82.3	82.3	82.3	82.2	82.2	82.2	82.2
9	82.3	82.1	82.0	82.3	82.3	82.2	82.2	82.2	82.2	82.2
10	82.5	82.6	82.4	82.6	82.4	82.7	82.6	82.6	82.6	82.7
11	82.3	82.5	82.7	83.0	82.9	83.0	83.0	82.9	83.0	83.0
12	83.1	82.6	82.6	82.5	82.8	82.8	82.8	82.8	82.8	83.0
13	83.7	83.3	83.2	83.2	83.5	83.5	83.0	83.0	83.1	83.0
14	82.0	82.8	82.8	83.5	83.5	83.8	83.0	83.0	82.8	82.8
15	83.2	83.8	83.8	83.8	83.8	83.8	83.8	83.8	83.8	83.8
20	82.9	84.0	84.0	84.2	84.1	84.4	84.4	84.4	84.1	84.4
30	82.4	84.3	84.5	84.6	84.6	84.5	84.5	84.5	84.6	84.6
40	83.5	84.8	84.6	84.6	84.6	84.6	84.1	84.1	84.0	84.1
50	79.8	87.2	85.8	86.1	86.3	86.2	86.2	86.1	86.0	86.0
60	81.0	87.7	86.7	87.2	86.9	87.1	87.0	86.9	87.0	87.0
70	80.3	88.0	87.4	88.7	88.3	88.3	88.2	88.1	88.0	87.9
75	77.8	88.6	88.8	89.4	89.6	89.2	89.5	89.6	89.6	89.0

Table A.16 Song-type classification accuracies of Eigenvoice adapted single Gaussian syllable model system using PCA correlation implementation

No. of EVs vs. Adaptation Data (s)	3	6	9	12	15	18	21	24	27	30
1	81.4	82.3	82.4	83.6	83.7	83.9	83.1	83.4	83.9	83.3
2	82.1	83.3	83.3	83.8	83.7	83.8	83.3	83.3	84.0	83.5
3	82.9	84.1	84.5	84.4	84.6	84.4	84.7	84.8	85.3	85.3
4	84.5	86.9	86.2	86.4	86.7	86.5	86.9	86.9	87.1	87.2
5	83.9	86.7	85.3	86.6	86.6	86.6	86.9	86.9	87.1	87.2
6	84.5	88.0	85.9	86.9	87.2	87.1	87.7	87.3	87.7	87.8
7	84.6	88.0	87.1	86.8	87.3	87.3	87.7	87.7	87.6	87.6
8	83.9	88.3	86.5	87.1	87.6	87.7	88.1	88.0	87.8	88.0
9	83.6	88.4	87.3	88.5	88.8	88.5	88.6	88.5	88.7	88.6
10	84.2	88.4	87.8	88.9	89.0	88.8	89.0	88.8	89.0	89.2
11	84.5	88.2	87.7	88.6	88.6	88.4	88.8	88.7	89.0	89.0
12	84.7	88.5	88.0	88.5	88.8	88.9	89.0	89.0	89.2	89.4
13	85.1	88.9	88.7	89.3	89.3	89.3	89.7	90.0	90.1	90.3
14	85.3	89.8	89.4	89.2	90.2	89.9	90.2	90.2	90.4	90.5
15	85.1	89.8	89.6	89.4	89.9	90.0	90.5	90.1	90.5	90.5
20	86.4	90.5	90.7	90.5	90.8	90.6	90.8	91.0	91.1	91.5
30	86.9	91.2	91.3	91.6	91.6	91.3	91.8	91.5	91.8	91.9
40	85.6	91.8	92.4	92.3	92.3	92.2	92.6	92.4	92.8	92.6
50	86.1	92.4	92.6	92.2	92.2	91.8	92.3	92.3	92.7	92.6
60	84.1	93.0	92.8	92.6	92.8	92.2	92.9	92.8	93.0	92.9
70	85.5	93.1	93.1	93.0	92.8	92.6	93.3	93.3	93.1	93.2
75	82.9	92.6	92.6	92.5	92.5	92.5	93.1	93.1	92.8	92.7

Table A.17 Song-type classification accuracies of Eigenvoice adapted single Gaussian syllable model system using PCA covariance implementation

No. of ECs vs. Adaptation Data (s)	3	6	9	12	15	18	21	24	27	30
1	84.1	84.1	84.1	84.1	84.1	84.1	84.1	84.1	84.1	84.1
2	84.2	84.2	84.1	84.1	84.1	84.1	84.1	84.1	84.1	84.1
3	84.0	84.4	84.2	84.2	84.2	84.2	84.2	84.2	84.2	84.2
4	84.2	84.4	84.4	84.4	84.4	84.4	84.4	84.4	84.4	84.4
5	84.1	84.7	84.7	84.4	84.4	84.4	84.4	84.4	84.4	84.2
6	84.1	84.8	84.8	84.8	84.9	84.1	84.4	84.4	84.4	84.4
7	84.4	84.8	84.8	84.7	84.8	84.1	84.1	84.2	84.2	84.2
8	84.4	84.6	84.5	84.5	84.6	84.0	84.1	84.0	84.0	84.0
9	84.9	84.7	84.6	84.5	84.5	84.5	84.5	84.6	84.6	84.6
10	84.9	84.7	84.6	84.4	84.5	84.5	84.5	84.6	84.6	84.6
11	85.0	84.6	84.5	84.5	84.5	84.6	84.6	84.6	84.6	84.6
12	85.2	84.8	84.7	84.6	84.7	84.6	84.9	84.9	84.8	84.8
13	84.6	84.9	84.7	84.5	84.7	84.8	84.9	84.8	84.8	84.8
14	84.5	84.9	84.7	84.7	84.7	84.4	84.4	84.4	84.8	84.8
15	84.7	84.8	84.7	84.7	84.7	84.2	84.2	84.2	84.2	84.2
20	86.3	84.9	84.5	85.1	85.2	84.7	84.8	84.6	84.7	84.7
30	87.4	86.5	85.5	85.4	85.3	85.3	85.5	85.4	85.4	85.4
40	87.4	86.4	85.8	85.7	85.6	85.7	85.9	85.8	85.8	85.8
50	87.9	87.3	86.5	86.1	86.2	86.1	86.2	86.2	86.2	86.2
60	87.0	87.2	86.9	86.8	86.6	86.9	86.7	86.6	86.8	86.5
70	86.0	87.3	87.0	87.0	86.4	86.9	86.9	86.6	86.7	86.7
75	86.0	87.3	87.0	87.0	86.4	86.9	86.9	86.6	86.7	86.7

Table A.18 Song-type classification accuracies of Eigen-clustering adapted single Gaussian syllable model system using PCA correlation implementation

No. of ECs vs. Adaptation Data (s)	3	6	9	12	15	18	21	24	27	30
1	84.0	84.0	84.1	84.2	84.3	84.3	84.2	84.2	84.2	84.2
2	84.7	84.5	84.6	84.7	84.9	85.0	84.8	84.8	84.8	85.0
3	85.4	84.7	85.0	85.1	85.0	84.6	84.8	84.7	84.8	84.8
4	84.7	84.9	84.9	85.0	85.0	84.7	84.8	84.7	84.8	84.8
5	84.8	85.4	85.4	85.6	85.7	85.2	85.4	85.2	85.4	85.4
6	84.4	85.5	85.5	85.5	85.7	85.2	85.2	85.2	85.2	85.2
7	84.1	85.7	85.6	85.4	85.0	85.7	85.9	85.9	85.9	85.9
8	84.9	86.8	86.4	85.6	85.9	86.0	86.0	86.0	86.1	86.1
9	85.0	86.8	86.3	85.6	85.7	85.9	86.0	86.0	86.0	86.0
10	84.4	86.8	86.3	86.3	85.9	86.0	86.0	86.0	86.1	86.1
11	83.8	86.2	85.7	85.8	86.0	86.0	86.0	86.0	86.0	86.0
12	82.9	86.1	86.1	86.0	86.1	85.5	85.5	85.5	85.5	85.4
13	83.5	86.1	86.0	86.1	86.1	85.6	85.6	85.6	85.6	85.6
14	83.7	86.2	86.2	86.3	86.2	85.7	86.2	86.3	86.2	86.2
15	84.2	86.3	86.6	86.5	86.4	85.9	85.9	85.9	85.9	86.4
20	84.3	86.3	86.3	87.0	87.1	86.6	86.6	86.5	86.5	87.0
30	84.2	86.4	86.3	86.6	87.1	86.8	86.8	86.6	86.7	86.8
40	82.9	87.2	87.3	86.8	87.1	87.0	87.0	87.1	87.3	87.2
50	82.9	87.2	87.1	86.8	86.6	86.6	86.6	86.9	87.0	87.2
60	83.0	86.8	86.9	86.8	86.9	86.6	86.9	87.1	87.3	87.3
70	83.7	87.3	87.2	87.3	87.5	87.2	87.2	87.1	87.4	87.7
75	83.7	87.3	87.2	87.3	87.5	87.2	87.2	87.1	87.4	87.7

Table A.19 Song-type classification accuracies of Eigen-clustering adapted single Gaussian syllable model system using PCA covariance implementation

System vs. Adaptation data size	1	2	3	4	5	6	7	8	9	10
MLLR	88.9	89.2	91.3	93.0	92.4	92.3	92.4	93.0	93.4	92.2
MAP	88.7	89.7	91.1	91.4	91.8	92.0	92.2	92.3	93.3	93.6
MAPLR	89.3	89.4	90.2	92.5	93.0	92.6	93.1	93.0	92.9	92.3
EV5	91.0	90.8	90.7	91.0	91.0	91.0	91.1	91.0	90.8	90.8
EV10	91.8	91.8	92.0	91.9	91.8	91.9	92.0	91.9	92.0	91.9
EV15	91.4	91.6	91.7	91.9	91.7	91.9	91.8	92.0	92.1	92.2
EV20	91.9	91.4	91.8	92.0	92.0	91.9	92.1	91.9	92.2	92.4
EV30	92.3	92.7	92.5	92.5	92.6	92.8	92.6	92.6	92.6	92.6
EC5	90.7	90.8	90.8	90.9	91.1	91.2	91.1	91.2	91.3	90.9
EC10	91.0	91.3	91.5	91.6	91.4	91.6	91.7	91.7	91.7	91.7
EC15	91.2	91.5	91.4	91.5	91.3	91.3	91.5	91.3	91.3	91.5
EC20	90.7	91.3	91.2	91.4	91.3	91.5	91.3	91.2	91.2	91.2
EC30	89.7	90.6	90.7	91.3	91.3	91.3	91.6	91.5	91.8	91.7
EC30+MAP	91.4	91.6	91.8	92.0	92.2	92.5	92.6	92.7	92.8	92.9

Table A.20 Song-type classification accuracies of the six adaptation methods on single Gaussian syllable model system

Marquette University

This is to certify that we have examined this copy of the dissertation by

Jidong Tao, B.Eng., M.S.

and have found that it is complete and satisfactory in all respects.

This dissertation has been approved by:

_____
Michael T. Johnson, Ph.D., P.E.
Dissertation Director, Department of Electrical and Computer Engineering

_____
Edwin E. Yaz, Ph.D., P.E., Committee Member

_____
James A. Heinen, Ph.D., Committee Member

_____
Richard J. Povinelli, Ph.D., P.E., Committee Member

_____
Craig A. Struble, Ph.D., Committee Member

Approved on

_____

Marquette University

This is to certify that we have examined this copy of the dissertation by

Jidong Tao, B.Eng., M.S.

and have found that it is complete and satisfactory in all respects.

This dissertation has been approved by:

_____
Michael T. Johnson, Ph.D., P.E.
Dissertation Director, Department of Electrical and Computer Engineering

_____
Edwin E. Yaz, Ph.D., P.E., Committee Member

_____
James A. Heinen, Ph.D., Committee Member

_____
Richard J. Povinelli, Ph.D., P.E., Committee Member

_____
Craig A. Struble, Ph.D., Committee Member

Approved on

_____