Acoustic Feature Transformation Based on Generalized Criteria for Speech Recognition

Makoto Sakai

Abstract

This thesis deals with acoustic feature transformations in automatic speech recognition to improve basic performance of a speech recognizer. The aim of acoustic feature transformations is to reduce dimensionality of long-term speech features without losing discriminative information among the different phonetic classes.

First, we focus on optimizing acoustic feature transformations using criteria with which to maximize the ratio of between-class scatter to within-class scatter. This approach is based on a family of functions of scatter or covariance matrices, which is frequently used in practice. Typical methods in this approach include linear discriminant analysis (LDA), heteroscedastic linear discriminant analysis (HLDA), and heteroscedastic discriminant analysis (HDA). Although LDA, HLDA and HDA are the most widely used in speech recognition, the connections between them have been disregarded so far. By developing a unified mathematical framework, close relationships between them are identified and analyzed in detail. The framework termed power LDA (PLDA) can describe various criteria by varying its control parameter. PLDA includes LDA, HLDA and HDA as special cases. In order to determine a sub-optimal control parameter automatically, a control parameter selection method is also provided.

The effectiveness of the combinations of acoustic feature transformations and discriminative training techniques of acoustic models is investigated and additional performance improvement is obtained. Unfortunately, the transformation methods mentioned above may result in an unexpected dimensionality reduction if the data in a certain class consist of several clusters, because they implicitly assume that data are generated from a single Gaussian distribution. This study provides extensions of HDA and PLDA to deal with class distributions with several clusters.

Second, we focus attention on acoustic feature transformations which minimize a kind of classification error between different phonetic classes. As the performance of speech recognition systems generally correlates strongly with the classification accuracy of features, the features should have the power to discriminate between different classes. The existing methods for this approach attempt to minimize the average classification error between different classes. Although minimizing the average classification error suppresses total classification error, it cannot prevent the occurrence of considerable overlaps between distributions of some different classes with low frequencies, which is critical for speech recognition because there may be class pairs that have little or no discriminative information on each other. Instead of the average classification error, minimization methods of maximum classification error are proposed herewith so as to avoid considerable error between different classes. In addition, interpolation methods that minimize the maximization error while minimizing the average classification error are also proposed and achieved the best results.

Acknowledgments

First of all, I am deeply grateful to my supervisor Prof. Norihide Kitaoka for giving me the opportunity to do my PhD at Nagoya University and for being always willing to answer questions or discuss problems. He also had been a former senior colleague at DENSO CORPORATION. Things I learned from him during his years at DENSO CORPORATION have helped me in work even after he left. I am also grateful to my vice supervisors, Profs. Kazuya Takeda and Hiroshi Murase, for their in-depth reading of this thesis and the valuable comments. They provided useful and valuable feedback on my thesis and helped guide it to completion. My special thanks to Prof. Kazuya Takeda for the numerous interesting and enlightening discussions.

My gratitude goes to Profs. Yukio Sato and Jun Sato of Nagoya Institute of Technology where I started my research career. Prof. Yukio Sato (currently a Professor at Keio University and a Professor emeritus at Nagoya Institute of Technology) introduced me to the field of pattern recognition, and taught me so much about it. I must also thank the late Research Assistant Kazuyuki Hattori for infusing me with his intellectual honesty. I have been greatly influenced by his attitude to research, and I was immensely saddened by his sudden passing.

My research on speech recognition has been supported and encouraged by Dr. Yoshiki Ueno, Dr. Nobuaki Kawahara, Ichiro Akahori, and Dr. Manabu Otsuka of DENSO CORPORATION. I would like to thank Yuya Hattori, who generously helped to run a part of the experiment in Chapter 3. I would also like to express my deep appreciation to all current and former colleagues of the Speech Group. Without them I could never have realized my study while working at DENSO CORPORATION.

I am extremely grateful to Prof. Seiichi Nakagawa of Toyohashi University of Technology. His expertise, vast knowledge, and suggestions have been very useful on improving my work. My thanks also to the doctoral meeting members of Nakagawa Lab. for their valuable suggestions. My research has also benefited from constructive advice and suggestions from Assistant Profs. Takanori Nishino (currently a Professor at Mie University) and Chiyomi Miyajima, and the doctoral meeting members of Takeda Lab. of Nagoya University.

Finally, special thanks to my friends, my parents, my brother and my wife who have always been there for me with their unfailing love and assistance.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Human speech is the most natural communication tool between human beings. If a computer recognizes human speech, then we can expect realization of a natural interface that links a person and a computer. Therefore, automatic speech recognition (ASR) technology has long been studied, and has been available for various applications, for example, an automated collect call, a word processor, a mobile phone, remote control, voice portal, disabled accessibility, a home-use game, and a robot. Moreover, speech communication has such a distinct advantage that we can make use of it without line-of-sight movement. Hence, ASR can be used in a restricted condition such as an in-car situation. Therefore, ASR has been commercially produced to operate a car navigation system.

Despite the significant progress of ASR in the past several decades, ASR technology use is hardly widespread. The key reason for this is that recent speech recognizers have not yet delivered adequate speech recognition performance. In other word, basic recognition performance of a speech recognizer does not yet reached the level of user satisfaction. Only one false recognition may give a speech recognizer bad reputation. There is still a considerable gap of speech recognition performance between human and machine even if we attempt to recognize an isolated word from among a small vocabulary. Therefore, we should improve the basic recognition performance of a speech recognizer.

1.2 Approaches

Hidden Markov model (HMM) [1, 2] is one of the fundamental theories concerning general automatic speech recognition. It is a stochastic method, into which some temporal information can be incorporated. With recent advances, speech recognizers based on HMMs have achieved a high level of recognition performance. Since HMM has several advantages in modeling temporal

sequence data, it has been applied with considerable success to other pattern recognition fields such as handwriting character recognition, gesture recognition, part-of-speech tagging, machine translation and bioinformatics [3].

However, several assumptions are imposed in the theory of HMMs for the sake of mathematical and computational tractability. HMMs usually assume successive observations are independent given a hidden state variable. Since this assumption may not sufficient to model speech, a hidden Markov model-based speech recognition system cannot precisely model the temporal change of speech. Therefore, to improve basic recognition performance of a speech recognizer, it is important to capture speech dynamics. To overcome this limitation of HMMs, a number of extensions have been proposed [4–7]. For example, Gupta et al. introduced regression coefficients along a time axis [8]. Deng et al. used HMMs with a polynomial regression function as a non-stationary state [9]. Wellekens provided explicit correlation in HMM [10]. Ming et al. used a conditional Gaussian mixture to model the inter-frame dependence in an HMM [11]. Ostendorf et al. proposed a stochastic segment model for phoneme-based speech recognition [12].

The simplest and most effective approach to represent long-term dynamic information of speech signals is to concatenate several successive frames as an input vector [13]. The concatenated vector may give higher speech recognition performance because it contains useful information for discrimination. The present study concentrates on this approach. Unfortunately, the concatenated high-dimensional vectors often include nonessential information and incur a heavy computational load. Added to this, it is generally known that needless high-dimensional vectors may cause degradation of speech recognition performance because an increase in feature dimension increases the number of model parameters to be estimated. This phenomenon is generally known as "curse of dimensionality" [14, 15]. The drawback of high-dimensional concatenated vectors would especially become a serious problem in the case of an application to an embedded device such as a car navigation system. In order to avoid this, an acoustic feature transformation method is usually applied to concatenated vectors to reduce dimensionality. In an acoustic feature transformation, it is important to preserve discrimination power between different phonetic classes, while reducing dimensionality.

In this study, acoustic feature transformations to reduce dimensionality of feature vectors are investigated in detail. Acoustic feature transformations with dimensionality reduction can be divided into two groups as follows:

1. Maximization of the ratio of between-class scatter to within-class scatter

2. Minimization criteria of classification error

The former approach is based on a family of functions of scatter (or covariance) matrices, which is frequently used in practice. The latter approach is based on a family of criteria which give

some sort of classification error such as upper bounds of the Bayes error [16]. More detailed explanations of the two approaches are given in the following.

1.2.1 Maximization of Ratio of Between-class Scatter to Within-class Scatter

We first describe a criterion which maximizes between-class scatter and minimizes within-class scatter. The basic idea of the criterion is that data in the same class are close to each other while data in different classes are separate from each other. The criterion is usually defined as the ratio of the between-class scatter to the within-class scatter. The most popular method using the criterion is linear discriminant analysis (LDA) [16,17]. LDA is widely used to reduce dimensionality and serves as a powerful tool to preserve discriminative information. Its objective function to be maximized is defined as the between-class scatter normalized by the within-class scatter. LDA assumes that the class distributions are Gaussians with different means and common covariance [18]. Due to this constraint of common covariance, LDA may give unsatisfactory performance when the class distributions are heteroscedastic. In order to overcome this limitation, several extensions have been proposed. Heteroscedastic linear discriminant analysis (HLDA) can deal with unequal covariances because the maximum likelihood estimation was used to estimate parameters for different Gaussians with unequal covariances [19]. Heteroscedastic discriminant analysis (HDA) was proposed as another objective function which employed individual weighted contributions of the classes [20].

The work focuses on the conventional three methods in the speech recognition field: LDA, HDA and HLDA. The effectiveness of these methods for some data sets has been experimentally demonstrated. We first point out that there exists a close relationship among them. Then, we provide a unified view of them and a generalization framework to integrate them. However, these methods may result in an unexpected dimensionality reduction if the data in a certain class consist of several clusters, i.e., multimodal, because they implicitly assume that data are generated from a single Gaussian distribution. We provide two extensions of these conventional methods in order to overcome the drawbacks.

(1) Generalization of LDA, HDA and HLDA (Chapter 3)

First, we demonstrate that these three methods have a strong mutual relationship although they were proposed independently. Then, a unified view of the three methods is presented. All three methods can be formally described in a common framework. The novel framework, called *power linear discriminant analysis* (PLDA), can describe various criteria by varying a control parameter of PLDA. PLDA includes LDA, HLDA, and HDA as special cases. Since PLDA can describe various criteria for dimensionality reduction, it can flexibly adapt to various environments. Thus, PLDA can provide robustness to a speech recognizer. Unfortunately, we cannot know which control parameter is the most effective before training HMMs and testing

the performances of each control parameter. In general, this training and testing process incur high computational costs. Moreover, the computational time is proportional to the number of variations of the control parameters tested PLDA requires considerable time to find an optimal control parameter because its control parameter can be chosen within a real number. In order to slash time, this work provides an efficient selection method of a sub-optimal control parameter without training of HMMs or testing recognition performance.

(2) Locality Preserving Extensions (Chapter 4)

Speech signals for acoustic model training tend to be multimodal because they are generally collected under various conditions, such as gender, age and noise environment. Therefore, each class such as a phone is generally represented as a Gaussian mixture model (GMM) or HMM, whose states are represented by GMMs, in a speech recognizer. Hence, dimensionality reduction methods without handling multimodality may give unsatisfactory performance, so a dimensionality reduction method for multimodal data is desired to improve speech recognition performance.

Recently, several methods have been proposed to reduce the dimensionality of multimodal data in the machine learning community [21–24]. It is important to preserve the local structure of data in reducing the dimensionality of multimodal data appropriately. Locality preserving projection (LPP) [22] finds a projection so that the data pairs close to each other in the original space remain close in the projected space. Thus, LPP reduces dimensionality without losing information on local structure. Local Fisher discriminant analysis (LFDA) [23] is also proposed as a supervised method for multimodal data, while LPP is an unsupervised method. To deal with multimodal data, LFDA combines the ideas of LPP and LDA. LFDA maximizes between-class separability and preserves within-class local structure. Thus, LFDA is an extension of LDA to reduce the dimensionality of multimodal data.

Since LFDA is based on LDA which assumes homoscedasticity, the effectiveness of LFDA may be limited. We extend HDA which assumes heteroscedasticity to reduce the dimensionality of multimodal data appropriately. To deal with multimodal data using HDA, we combine the ideas of LPP and HDA, and propose a locality-preserving HDA. In addition, we also propose a locality-preserving PLDA. These extensions can be expected to yield better performance because they reduce the dimensionality of multimodal data appropriately.

Locality-preserving methods such as LFDA and the proposed methods require considerable computational time to obtain optimal projections when there are many features. In order to reduce computational time, we propose an approximate calculation scheme.

1.2.2 Minimization of Classification Error

The other acoustic feature transformation approach is to minimize some sort of classification error among classes. The criterion has been extensively applied to several pattern recognition

problems [25–29]. However, their performance in speech recognition has not been carefully investigated. If classification error becomes small after acoustic feature transformation, promising speech recognition performance can be obtained. The most natural criterion regarding a classification error is the Bayes error [16]. The Bayes error is the best criterion to classify features. If this error becomes small after an acoustic feature transformation, discriminative information would be preserved to classify features. Unfortunately, to directly measure Bayes error is difficult, although it is a superior criterion to reduce dimensionality.

Instead of calculating the Bayes error directly, several methods are taken to represent the Bayes error as an indirect approach. Several researchers proposed an objective function which uses symmetric KL divergence as a measure of the distance between two class distributions [25–27]. KL divergence can be considered as the dissimilarity between two distributions. Torre et al. [27] introduced an effective computation scheme, called oriented discriminant analysis (ODA), for calculating the objective function using symmetric KL divergence. Nenadic introduced another measure of distance, called μ-measure, which is based on mutual information [29]. The resulting objective function with this μ-measure is similar to HDA. Loog et al. proposed a heteroscedastic extension of LDA using the Chernoff distance [28]. This measure of affinity of two probability densities considers the mean difference as well as the covariance difference. Torkkola [30] used measures based on Renyi entropy [31,32] instead of Shannon's entropy. As another technique, Decell et al. and Saon et al. employed the Bhattacharyya coefficient to measure the Bayes error indirectly [25,26]. The Bhattacharyya coefficient is an upper bound of the Bayes error. This coefficient can be regarded as a dissimilarity between two classes. If the Bhattacharyya coefficient in the reduced space becomes small, the reduced data will preserve discriminative information. This study focuses on the Bhattacharyya coefficient instead of the Bayes error. To deal with multi-class problem, extended objective functions that minimize the average Bhattacharyya coefficient were proposed [25,26]. That is, the conventional methods are used to search for a projection so that the average classification error is minimized. Although minimizing the average classification error can suppress total classification error among classes, it cannot prevent the occurrence of considerable overlaps between distributions of some classes, which is critical for speech recognition because there may be class pairs that have little or no discriminative information on each other.

Another issue regarding the Bhattacharyya coefficient must be addressed. While it has a closed-form expression for two Gaussians, no closed-form expression exists for two GMMs. In other words, the Bhattacharyya coefficient for GMMs is not feasible. Therefore, the existing methods impose the single Gaussian assumption for classes, although a single Gaussian is too simple to represent a class distribution. Recently, efficient approximations of the Bhattacharyya coefficient under a GMM assumption have been derived [33]. Nevertheless, this study also assumes a single Gaussian for a class distribution because of tractability. We do not further discuss the Bhattacharyya coefficient under a GMM assumption because the proposed method

in the present study, which assumes a single Gaussian for classes, could be extended to the approximations dealing with GMMs.

(1) Minimization of Maximum Classification Error (Chapter 5)

As described above, although minimizing the average classification error can suppress total classification error among classes, it cannot prevent the occurrence of considerable overlaps between distributions of some classes. Therefore, the conventional method is critical for speech recognition because there may be class pairs that have little or no discriminative information on each other. For example, when the discrimination power between a certain class pair becomes small after an acoustic feature transformation, all words which include either of the two classes may degrade recognition performance. To overcome the drawback of the conventional method, we introduce an alternative objective function that minimizes the maximum classification error among distributions of all class pairs. The method is able to avoid considerable error between classes, unlike the method which minimizes the average classification error.

However, there is still a problem with this approach. If a large number of class pairs have an overlap comparable to the maximum one, the total error increases significantly. In such a situation, speech recognition performance will deteriorate because most class pairs have only small discrimination power. Therefore, an interpolated criterion that minimizes maximum classification error while minimizing average classification error would be effective. We here introduce two types of interpolated criteria between the average and the maximum classification errors.

1.3 Overview of Thesis

The remainder of the thesis is organized as follows.

Chapter 2 describes a standard framework of an ASR system. Especially, acoustic feature transformation techniques are described in detail.

Chapters 3 and 4 investigate the criterion that maximizes the ratio of between-class scatter to within-class scatter. Chapter 3 proposes a new framework which can describe various criteria including LDA, HLDA, and HDA with one control parameter. In addition, the chapter provides an efficient control parameter selection method, which can find a sub-optimal control parameter without training HMMs nor testing recognition performance. The effectiveness of acoustic feature transformations, discriminative training techniques, and their combinations is also investigated. In Chapter 4, two extensions of HDA and PLDA are provided to reduce the dimensionality of multimodal data appropriately. The chapter also proposes an approximate calculation scheme to calculate sub-optimal projections rapidly.

Chapter 5 investigates the criterion that minimizes some sort of classification error. We propose a novel criterion that minimizes the maximum classification error among distributions

of all class pairs. In addition, we propose two types of interpolated criteria between the average and the maximum classification errors.

Chapter 6 gives the overall conclusions and suggests research directions for the future.

Chapter 2

Theoretical Framework for Automatic Speech Recognition

2.1 Overview of Automatic Speech Recognition Systems

The purpose of an automatic speech recognition system is to convert input speech waveforms to correct transcriptions in text. The speech waveform is typically recorded by a microphone and samples a certain fixed frequency (typically, 8 kHz or 16 kHz). Then, the recorded speech waveform is transformed into a sequence of observation vectors, $\mathbf{O} = \{\mathbf{o}_1, \ldots, \mathbf{o}_T\}$, where $\mathbf{o}_t$ is the short-time speech vector observed at time t. We would like to obtain a correct hypothesis sequence $\mathbf{w} = \{w_1, \ldots, w_N\}$ given the observation $\mathbf{O}$. Finally, the system outputs the hypothesis transcription with the highest probability. We can obtain the desired output given the observation through the maximum a posteriori decision:

$$\hat{\mathbf{w}} = \arg\max_{\mathbf{w}} P(\mathbf{w}|\mathbf{O}), \tag{2.1}$$

where $P(\mathbf{w}|\mathbf{O})$ denotes the posterior probability given the observation. The desired hypothesis, $\hat{\mathbf{w}}$, is the word sequence with the highest probability given the observations. In general, this probability is not feasible directly. Instead, the following expression can be used:

$$\hat{\mathbf{w}} = \arg\max_{\mathbf{w}} \frac{p(\mathbf{O}|\mathbf{w})P(\mathbf{w})}{p(\mathbf{O})}, \tag{2.2}$$

where Bayes' theorem has been applied. It plays a central role in pattern recognition and machine learning. Of the two terms on the right-hand side, the first, $p(\mathbf{O}|\mathbf{w})$, is the probability of the observation given the word sequence, and the second term, $P(\mathbf{w})$, is the prior distribution of the word sequence. These probabilities are represented as an acoustic model and a language model, respectively.

Since the likelihood of the observation sequence $p(\mathbf{O})$ is independent of the hypothesis sequence $\mathbf{w}$, it can be omitted as follows:

$$\hat{\mathbf{w}} = \arg\max_{\mathbf{w}} p(\mathbf{O}|\mathbf{w})P(\mathbf{w}). \tag{2.3}$$

Thus, the right-hand side of Equation (2.1) is replaced with the combined score from the acoustic and the language model.

Since the present study attempts only to carry out an isolated word recognition with a uniform prior, the language model $P(\mathbf{w})$ is not needed. Hence, language modeling is not discussed in this thesis. Therefore, the following expression which omitted $P(\mathbf{w})$ from Equation (2.3) and replaced $\mathbf{w}$ with w_i is used to obtain the output in an isolated word recognition task as follows:

$$\hat{w} = \arg\max_{i} p(\mathbf{O}|w_i), \tag{2.4}$$

where $\hat{w}$ denotes the isolated word with the highest probability.

Recently, direct modelings of posteriori probability $P(\mathbf{w}|\mathbf{O})$ in Equation (2.1) have been proposed, including, for example, speech recognizers based on Support Vector Machine (SVM) [34, 35], Maximum Entropy Markov Model (MEMM) [36], and Hidden Conditional Random Field (HCRF) [37, 38] which extends the Conditional Random Field (CRF) [39] to a sequential problem. However, these SVM-, MEMM-, and HCRF-based speech recognizers are applied only to simple recognition tasks such as the phone classification task and the phone recognition task. Application of these models to isolated word recognition and continuous speech recognition tasks remains an unsolved issue.

2.1.1 Basic Structure of ASR System

The basic structure of the general speech recognition system is illustrated in Figure 2.1. A typical ASR system consists of four components for an isolated word recognition task: front-end processing, acoustic models, dictionary, and decoding. In the front-end processing, an input speech waveform is converted into feature vectors of typically 20-40 dimensions. The vector represents the short-time (approximately 10-20 ms) spectral envelope of the speech signal, which describes the characteristics of the speech. The acoustic models are used to evaluate the likelihood $p(\mathbf{O}|w_i)$ in Equation (2.4). The dictionary defines recognizable vocabulary by a speech recognizer. In decoding, the maximization in Equation (2.4) is carried out using the acoustic models while system vocabulary is restricted by using the dictionary. After searching the system outputs the hypothesis with the highest score. The performance of speech recognition systems is evaluated by comparing the recognized word to a reference transcription.

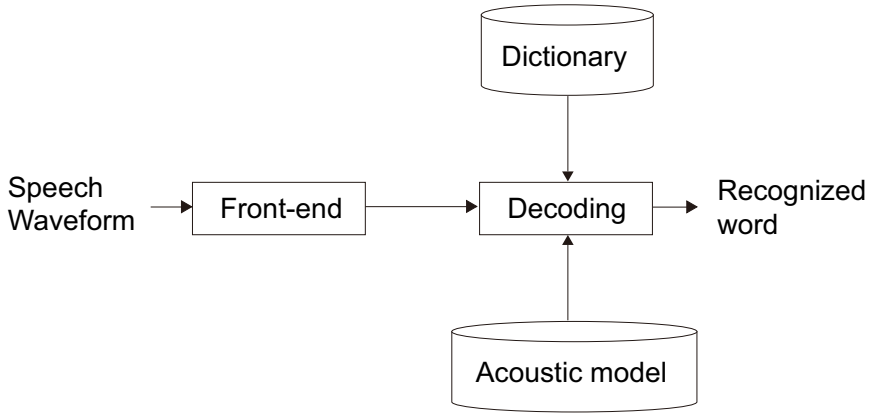


Figure 2.1: Basic structure of ASR system.

2.2 Front-end Processing

In front-end processing, acoustic information for speech recognition is extracted from the speech waveform. Generally, sampled speech waveforms are converted to 20-40 dimensional vectors which represent the short-time spectral envelope of the waveforms because the spectrum envelope conveys most of the significant information for speech recognition [40].

2.2.1 Pre-processing

There are some pre-processing operations that can be applied prior to performing the actual signal processing. In order to flatten the spectrum slope, it is common practice to pre-emphasize the signal by applying the first-order difference equation:

$$x'(n) = x(n) - \alpha x(n-1), \tag{2.5}$$

where $x(n)$ denotes short segment of input speech waveform at time n, and α denotes a pre-emphasis coefficient which lies between zero and one. Typically, the value of α is taken in the range $0.9 \leq \alpha \leq 1.0$. This study used $\alpha = 0.97$. This process can reduce a quantization error caused by a limitation of significant digits. Then, a window function is usually applied to the

emphasized speech signal to attenuate boundary effects in the following analysis. As a time domain window function, hamming or Blackman-Harris windows are often applied to speech frames. This study used a hamming window function:

$$w(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N_s - 1}\right), \tag{2.6}$$

where N_s denotes the number of samples to be processed.

2.2.2 Cepstral Analysis

After some pre-processing, the Fourier spectrum of the speech waveform is computed for each time frame with a typical frame shift of 10 ms and a typical window length of 25 ms. Afterwards, several procedures for further processing are adopted. These include Linear Predictive Coding (LPC), Mel Frequency Cepstral Coefficient (MFCC) [41] and Perceptual Linear Prediction (PLP) [42]. The most commonly used parameterization is MFCCs. This study used MFCCs as acoustic features.

In order to adjust the spectral resolution to that of the human ear, the frequency axis is warped nonlinearly according to the Mel frequency scale:

$$Mel(f) = 2595 \log_{10}\left(1 + \frac{f}{700}\right). \tag{2.7}$$

Then, a triangular filter bank is applied to the warped spectrum. Normally, the triangular filters are spread over the whole frequency range from zero up to the Nyquist frequency in the Mel scale space. Band-limiting such as a high pass cut-off is often useful to reject unwanted frequencies. because a low-cut filter of typically 200 Hz is effective with an in-car environment.

Then, the logarithm is taken and a discrete cosine transformation (DCT) is applied to the log filter bank coefficients $\{m_j\}$ to remove the correlation between the different outputs:

$$c_i = \sqrt{\frac{2}{N_f}} \sum_{j=1}^{N_f} m_j \cos\left(\frac{\pi i}{N_f}(j - 0.5)\right), \tag{2.8}$$

where N_f is the number of filterbank channels. Afterwards, liftering is applied to the cepstral coefficients to reduce dimensionality by omitting the highest cepstral coefficients. The resulting feature vector at time t is denoted by $\mathbf{o}_t$ in this thesis.

The flowchart of the front-end processing is depicted in Figure 2.2. Figure 2.3 shows an example of speech waveform, spectrogram, and cepstrum feature vectors.

2.3 Hidden Markov Models for Acoustic Modeling

Nowadays, almost every speech recognizer employs hidden Markov models (HMMs) [1, 2] to model speech signals. The HMMs are used to provide the estimates of $p(\mathbf{O}|w)$. In this section,

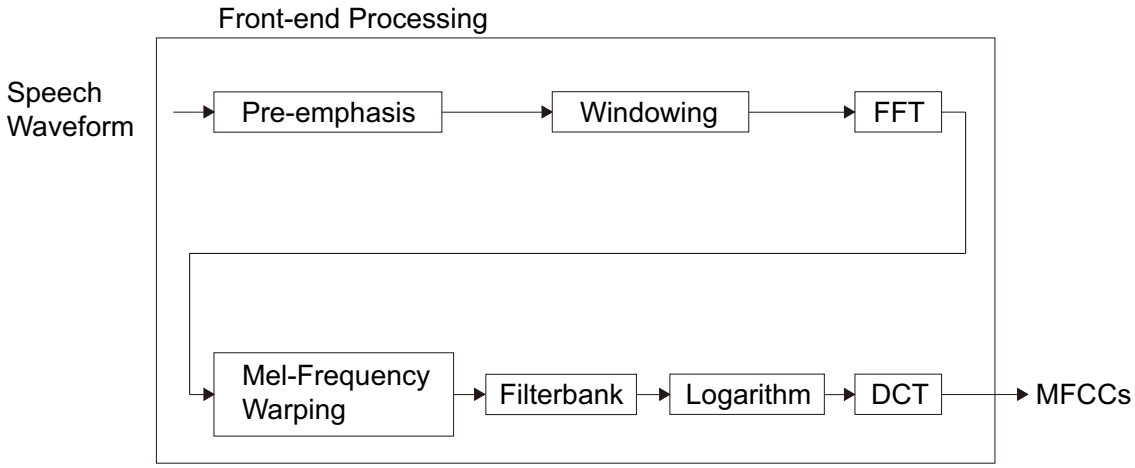
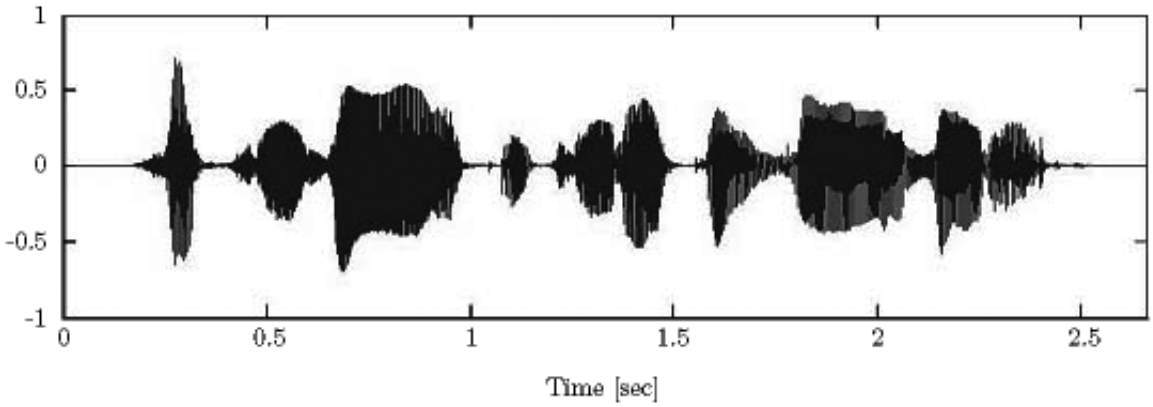


Figure 2.2: Front-end processing.

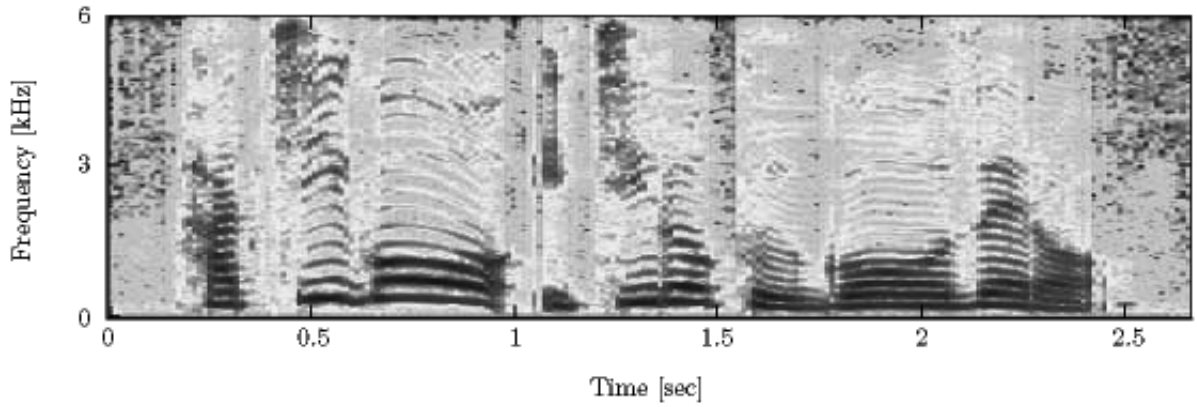
we describe how speech is represented by an HMM. First, acoustic modeling by HMMs is presented. Then, the maximum likelihood (ML) parameter estimation is briefly reviewed. Several discriminative training techniques are introduced. It is well known that discriminative training can yield better performance than ML. We investigate the effectiveness of combinational use of an acoustic feature transformation and discriminative training in Section 3.5.

2.3.1 Acoustic Model

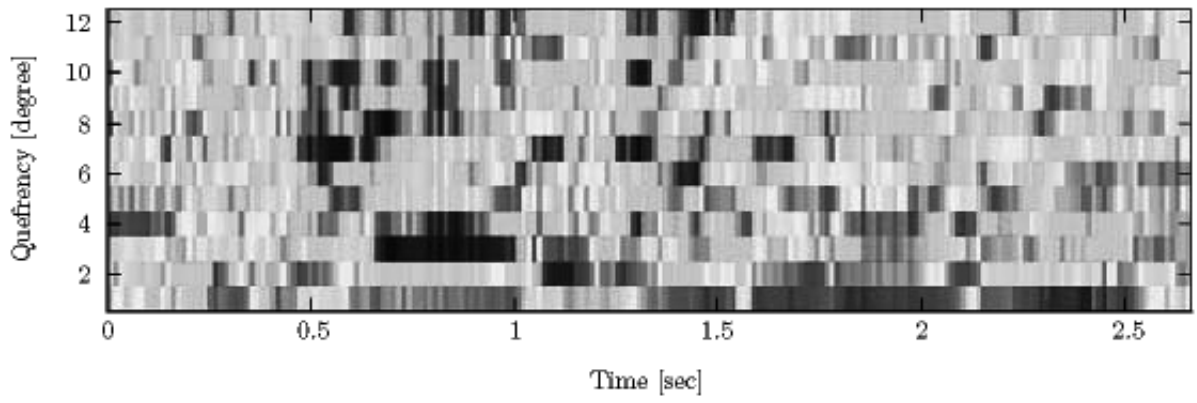
An HMM is a stochastic finite state automaton consisting of a number of states and transitions among states. The acoustic model is a set of HMMs for the basic sub-word units. The most commonly used sub-word units are phones, which are the basic sound of speech. An HMM for each word in vocabulary is made by concatenating the individual sub-word HMMs. Figure 2.4 shows an example of an HMM for a certain sub-word unit. The model in the figure has three emitting states with the probabilistic density function (pdf). The states are labeled by integers. The model is a first-order left-to-right HMM so there are only forward links and self loops, and each transition depends only on its source and destination states. This left-to-right HMM is widely used to represent a phone in speech recognition. Each arrow between states a_{ij} denotes



(a) Waveform.



(b) Spectrogram.



(c) Cepstrum.

Figure 2.3: Example of speech waveform, spectrogram, and cepstrum.

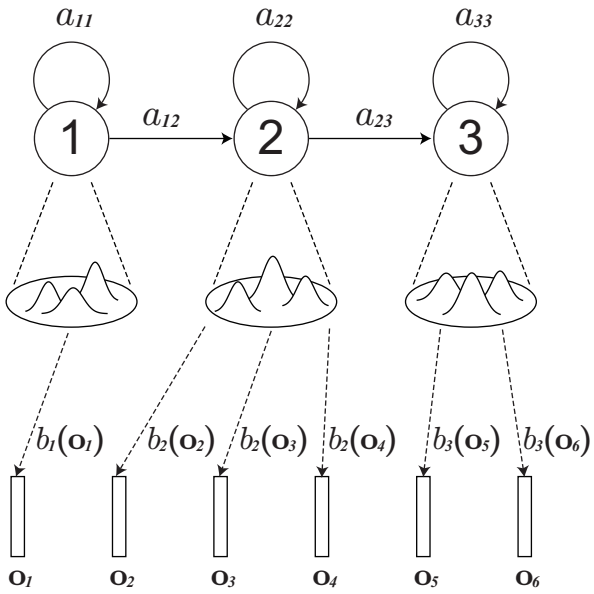


Figure 2.4: Example of an HMM.

a state transition probability from the state i to the state j:

$$a_{ij} = P(q_t = j | q_{t-1} = i), \tag{2.9}$$

for arbitrary t. The transition probabilities must satisfy $0 \leq a_{ij} \leq 1$ and $\sum_j a_{ij} = 1$, and are assumed to be stationary in time. Therefore, the transition does not depend on the time t at which it occurs.

The state conditional output probability $b_j(\mathbf{o}_t)$ with which an observation $\mathbf{o}_t$ is generated by the state $q_t = j$ is modeled as a probability density function:

$$b_j(\mathbf{o}_t) = p(\mathbf{o}_t | q_t = j). \tag{2.10}$$

The most common representation of the pdf is a finite mixture of multivariate Gaussian distribution. Let $\mathcal{N}(\cdot; \boldsymbol{\mu}, \mathbf{C})$ be a multivariate Gaussian with a mean vector $\boldsymbol{\mu}$ and a covariance matrix $\mathbf{C}$:

$$\mathcal{N}(\mathbf{o}; \boldsymbol{\mu}, \mathbf{C}) = \frac{1}{|2\pi\mathbf{C}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{o} - \boldsymbol{\mu})^\top \mathbf{C}^{-1}(\mathbf{o} - \boldsymbol{\mu})\right). \tag{2.11}$$

Then, the pdf given the state j is typically defined as:

$$b_j(\mathbf{o}) = \sum_{m=1}^{M} w_{jm} \mathcal{N}(\mathbf{o}; \boldsymbol{\mu}_{jm}, \mathbf{C}_{jm}), \tag{2.12}$$

where $\boldsymbol{\mu}_{jm}$ and $\mathbf{C}_{jm}$ denote a mean vector and a full covariance matrix of m-th mixture of state j, respectively, and the mixing proportions w_{jm} satisfy $0 < w_{jm} < 1$ for $m = 1, \ldots, M$ and $\sum_{m=1}^{M} w_{jm} = 1$ since they obey standard stochastic constraints. Full covariance modeling often increases computational costs and causes a data sparseness problem for parameter estimation. In order to simplify the probability computation of Gaussians and to reduce the number of parameters, each dimension of the speech feature vector is often assumed to be independent. This assumption leads to diagonal covariance matrices for robustness of parameter estimation and speed-up in the computation.

$$\mathcal{N}(\mathbf{o}; \boldsymbol{\mu}, \boldsymbol{\Lambda}) = \frac{1}{|2\pi\boldsymbol{\Lambda}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{o}-\boldsymbol{\mu})^{\top} \boldsymbol{\Lambda}^{-1} (\mathbf{o}-\boldsymbol{\mu})\right) \tag{2.13}$$

$$= \prod_{d=1}^{D} \frac{1}{\sqrt{2\pi\sigma_d^2}} \exp\left(-\frac{(o_d-\mu_d)^2}{2\sigma_d^2}\right), \tag{2.14}$$

where $\boldsymbol{\Lambda}$ denotes a $D \times D$ diagonal covariance matrix with the i-th diagonal element being σ_i^2, and o_i, μ_i and σ_i^2 are the i-th elements of observation, mean and variance, respectively.

2.3.2 Parameter Estimation for HMMs

HMMs are composed of model parameters such as the state transition probabilities, mixture weights, mean vectors, and covariance matrices. To find optimal model parameters, the following objective function is maximized with respect to the model parameters:

$$\mathcal{F}_{ML}(\lambda) = \sum_{r=1}^{R} \log p_{\lambda}(\mathbf{O}_r | s_r), \tag{2.15}$$

where λ is the set of HMM parameters, $\mathbf{O}_r$ is the r-th training utterance (a word or a sentence), s_r is the r-th correct transcription, R denotes the number of training utterances, and $p_{\lambda}(\mathbf{O}_r|s)$ is the likelihood given transcription s. This optimization is called the maximum likelihood (ML) parameter estimation. Since the discrete state sequence generating the observation sequence is unknown, the Baum-Welch algorithm [43,44] is usually carried out to find the model parameters, which is an instance of the expectation maximization (EM) algorithm [45]. The algorithm iteratively finds discrete state posterior probabilities given the observation sequence and the current model parameters, and finds the expected values for the state conditional densities. In practice, an efficient algorithm known as the forward-backward algorithm [43] is often used to find the discrete state posteriors, which involves the calculating probabilities of partial state sequences forwards and backwards through the observation sequence.

2.3.3 Discriminative Training of HMMs

If the modeling assumptions of acoustic features were completely correct and infinite data were given, the maximum likelihood criterion would be optimal. However, these assumptions are not necessarily accepted in nature. Recently, the field of discriminative training has witnessed considerable activity. The HMM parameters via discriminative training adjust to improve the classification performance of the HMMs on the training data. Discriminative training includes maximum mutual information (MMI) [46], minimum classification error rate (MCE) [47], frame discrimination [48, 49], minimum Bayes risk [50], and minimum phone error (MPE) [51–53]. Many experimental results have shown that discriminative training techniques yields better performance than traditional maximum likelihood (ML) training. In Section 3.5, we investigate the effectiveness of combinations of two discriminative training techniques described in detail below and acoustic feature transformations.

Maximum Mutual Information (MMI)

In MMI training, the mutual information [54, 55] between an acoustic observation and correctly transcribed string is maximized. The MMI criterion is defined as follows [46, 56]:

$$\mathcal{F}_{MMI}(\lambda) = \sum_{r=1}^{R} \log \frac{p_\lambda(\mathbf{O}_r|s_r)^\kappa P(s_r)}{\sum_s p_\lambda(\mathbf{O}_r|s)^\kappa P(s)}, \tag{2.16}$$

where κ is an acoustic de-weighting factor which can be adjusted to improve the test set performance, and $P(s)$ is the language model probability for sentence s. The MMI criterion equals the multiplication of the posterior probabilities of the correct sentences s_r.

As an extension to MMI, a discriminative training method which emphasizes posteriori probability, called Boosted MMI, is proposed by Povey [57].

Minimum Phone Error (MPE)

MPE training aims to minimize the phone classification error (or maximize the phone accuracy) [53]. The objective function to be maximized by the MPE training is expressed as

$$\mathcal{F}_{MPE}(\lambda) = \sum_{r=1}^{R} \frac{\sum_s p_\lambda(\mathbf{O}_r|s)^\kappa P(s) \mathrm{A}(s, s_r)}{\sum_s p_\lambda(\mathbf{O}_r|s)^\kappa P(s)}, \tag{2.17}$$

where $\mathrm{A}(s, s_r)$ represents the raw phone transcription accuracy of the sentence s given the correct sentence s_r, which equals the number of correct phones minus the number of errors.

A Unified View of Discriminative Training Techniques

Recently, a generalization framework on several discriminative training criteria has been proposed [58,59]. In [58], Ψ-probability is introduced, which is the sum of modified joint probability

of $\mathbf{O}_r$ and s_r weighted by the negative exponential of the difference measure between correct and recognized strings:

$$\Psi_\sigma(\mathbf{O}_r, s_r) = \sum_s p_\lambda(\mathbf{O}_r|s)^\kappa P(s) \exp(-\sigma\Delta(s, s_r)). \tag{2.18}$$

where σ and $\Delta(s, s_r)$ denote an exponential decay factor and the difference measure between s and s_r such as Levenshtein distance [60]. Using Ψ-probability, MMI and MPE objective functions are given by:

$$\mathcal{F}_{MMI}(\lambda) = \frac{\Psi_\infty}{\Psi_0}, \tag{2.19}$$

$$\mathcal{F}_{MPE}(\lambda) = -\frac{\Psi_0^{'}}{\Psi_0}, \tag{2.20}$$

where $\Psi_\sigma^{'}$ denotes the partial derivative of Ψ-probability with respect to σ defined by

$$\Psi_\sigma^{'} = \left.\frac{\partial\Psi_\nu}{\partial\nu}\right|_{\nu=\sigma} = -\sum_s p_\lambda(\mathbf{O}_r|s)^\kappa P(s)\Delta(s, s_r) \exp(-\sigma\Delta(s, s_r)). \tag{2.21}$$

Similarly, minimum classification error (MCE) [47] and boosted MMI [57] can be also derived by using the Ψ-probability.

2.4 Decoding Using HMMs

The decoding step in Figure 2.1 must be able to search through all the hypotheses to find the one yielding the maximum likelihood from the acoustic models. Therefore, a decoding algorithm is required to solve Equation (2.4). The likelihood given a word is expanded as follows:

$$\begin{aligned} p(\mathbf{O}|w) &= \sum_{\mathbf{q}} p(\mathbf{O}, \mathbf{q}|w) \\ &= \sum_{\mathbf{q}} \prod_{t=1}^{T} p(\mathbf{o}_t|q_t; w) P(q_t|q_{t-1}; w) \end{aligned} \tag{2.22}$$

where $\mathbf{q}$. Since direct implementation of Equation (2.22) is not practical, the sum in the Equation is approximated by the maximum;

$$p(\mathbf{O}|w) \approx \max_{\mathbf{q}} \prod_{t=1}^{T} p(\mathbf{o}_t|q_t; w) P(q_t|q_{t-1}; w) \tag{2.23}$$

This approximation is known as the Viterbi algorithm [61]. In Viterbi algorithm, the probability of the local best path plays the principal role, which represents the maximum likelihood

of observing vectors $\mathbf{o}_{1:t} \equiv (\mathbf{o}_1, \ldots, \mathbf{o}_t)$ and being in state s at time t. The probability is given by:

$$\delta_t(j) = \max_{q_1, q_2, \ldots, q_{t-1}} P(q_1, q_2, \ldots, q_{t-1}, q_t = i, \mathbf{o}_{1:t}|\lambda). \tag{2.24}$$

where λ denotes the model parameter set.

In practice, $\delta_t(j)$ can be recursively implemented by:

$$\delta_{t+1}(j) = \left[\max_i \delta_t(i) a_{ij}\right] b_j(\mathbf{o}_{t+1}), \tag{2.25}$$

where the initial conditions are given by

$$\begin{aligned} \delta_1(1) &= 1, \\ \delta_1(j) &= a_{1j} b_j(\mathbf{o}_t) \end{aligned}$$

for $1 < j < N$. The Viterbi algorithm results in the joint likelihood of the observation sequence $\mathbf{O}$ and the most likely state sequence $\hat{\mathbf{q}} = \hat{q}_1, \ldots, \hat{q}_T$ given the model parameters:

$$\delta_T(N) = p(\mathbf{O}, \hat{\mathbf{q}}|\lambda) = \max_i \delta_T(i) \tag{2.26}$$

In practice, the Viterbi algorithm can be implemented by taking logarithms of the model parameters because of rapid calculation and underflow prevention.

2.5 Acoustic Feature Transformation

To obtain additional improvement of speech recognition performance, an acoustic feature transformation is often applied to the extracted feature vectors after a front-end processing step. An ASR system added an acoustic feature transformation step is illustrated in Figure 2.5. This study deals with an acoustic feature transformation to improve speech recognition performance. While there are several kinds of acoustic feature transformations, this study concentrates on the feature transformations for addition of dynamic information, which aims at capturing speech dynamics in speech signals. This approach is generally known to improve the basic performance of a speech recognizer.

2.5.1 Feature Transformation for Addition of Dynamic Information

Cepstrum-based feature vectors which are calculated over 20 to 30 milliseconds (ms) accurately extract short-term static information from speech signals. In addition to static feature vectors, dynamic information that describes temporal change among several successive features (typically 50-100 ms) is usually appended to the feature vectors [62]. Using acoustic dynamic information

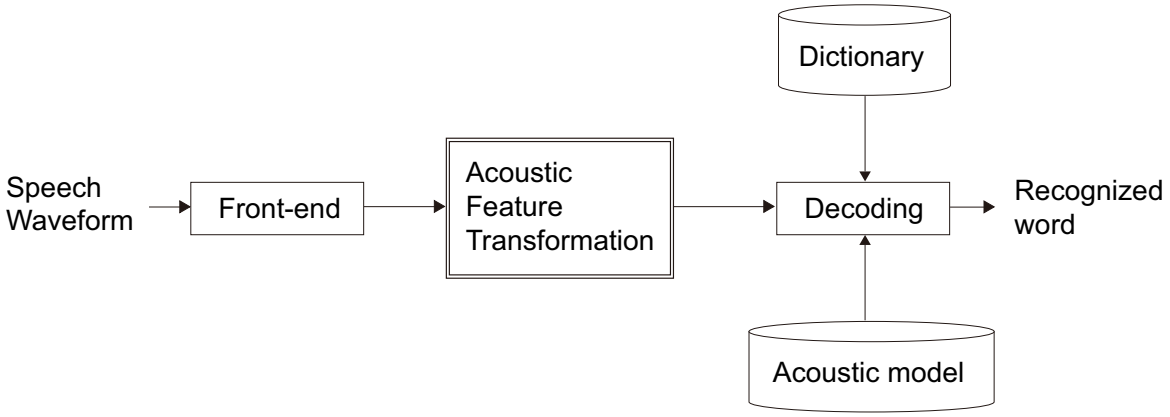


Figure 2.5: Block diagram of ASR system added an acoustic feature transformation step.

that expresses temporal change in speech signals can serve to improve speech recognition performance. Several methods for integrating dynamic information have been proposed. Delta and acceleration coefficients [63] are the most widely used approach in speech recognition. These are first-order and second-order regression coefficients, respectively. Other approach includes a concatenation of several successive frames with dimensionality reduction. Both two approaches can be interpreted as a linear transformation of speech features. On the other hand, nonlinear transformations of speech features such as kernel-based techniques have also been applied in speech recognition. In [64], kernel principal component analysis (KPCA) [65, 66] was applied to transform speech features nonlinearly. Kernel discriminant analysis (KDA) [67, 68] was applied to a phoneme classification task [69]. Extensions of KPCA and KDA such as sparse KPCA [70] and subspace KDA [71] are also applied in speech recognition.

One problem for kernel-based nonlinear dimensionality reduction for speech recognition is that the original derivation requires computation of features in an N-dimensional space where N is the number of training data. Since acoustic models in a speech recognition system are generally trained using a large amount of training data, the value of N tends to become large, e.g., 10^7 to 10^9. Moreover, it will be difficult to generate classifiers such as HMMs in the high-dimensional space. Thus, it becomes impractical to apply kernel-based dimensionality reduction for speech recognition. Hence, we focus on linear transformation approaches. The following

describes two linear transformations in detail: delta and acceleration coefficients, and linear transformation with dimensionality reduction.

Delta and Acceleration Coefficients

In order to capture dynamic information along with successive frames, one popular approach is to compute first-order (delta) and second-order (acceleration) regression coefficients of static frames. The coefficients are appended to static features as inputs. Delta and acceleration coefficients can enhance speech recognition performance, although they are heuristic information from a priori knowledge.

The delta coefficient at time t, $\mathbf{\Delta}_t$, is given by the following regression formula:

$$\mathbf{\Delta}_t = \frac{\sum_{k=1}^{L_{delta}} k \left(\mathbf{o}_{t+k} - \mathbf{o}_{t-k}\right)}{2\sum_{k=1}^{L_{delta}} k^2}, \tag{2.27}$$

where the first-order regression window size is $2L_{delta} + 1$. $\mathbf{\Delta}_t$ is computed in terms of the corresponding static coefficients $\mathbf{o}_{t-L_{delta}}$ to $\mathbf{o}_{t+L_{delta}}$. Thus, a delta coefficient is the linear combination of successive static frames. The same formula is applied to the delta coefficients to obtain acceleration coefficients except that feature vectors are replaced with delta coefficients and the window size is $2L_{acc} + 1$:

$$\mathbf{\Delta\Delta}_t = \frac{\sum_{k=1}^{L_{acc}} k \left(\mathbf{\Delta}_{t+k} - \mathbf{\Delta}_{t-k}\right)}{2\sum_{k=1}^{L_{acc}} k^2}, \tag{2.28}$$

An acceleration coefficient is a combination of successive delta coefficients, and thus is also a linear combination of static ones.

Linear Transformation with Dimensionality Reduction

The simplest, most effective approach to represent long-term dynamic information of speech signal is to concatenate several successive frames as an input vector [13]. The concatenated vector may give higher speech recognition performance because it contains useful information for discrimination. On the other hand, it is generally known that needless high-dimensional vectors may cause the degradation of speech recognition performance because an increase in feature dimension increases the number of model parameters to be estimated. This phenomenon is generally known as the curse of dimensionality [14, 15]. Such high-dimensional vectors require huge computational cost and a large amount of memory. These drawbacks of high-dimensional concatenated vectors would become an especially serious problem in the case of an application to an embedded device such as a car navigation system.

Ordinarily, an acoustic feature transformation with reducing dimensionality is applied to the concatenated features. In order to reduce dimensionality, linear feature transformations are

often used. Let $\mathbf{x}$ and $\mathbf{z}$ denote a concatenated vector and a transformed vector, respectively. A linear feature transformation by a transformation matrix $\mathbf{B}$ is given by:

$$\mathbf{z} = \mathbf{B}^\top \mathbf{x}. \tag{2.29}$$

This linear transformation can describe delta and acceleration coefficients as a special case. That is, linear transformations include delta and acceleration coefficients.

As mentioned in Chapter 1, acoustic feature transformations with dimensionality reduction can be divided into two groups as follows:

1. Maximization of the ratio of between-class scatter to within-class scatter

2. Minimization criteria of classification error

Both LDA and HDA, which are typically used in the speech recognition field, belong to the former approach, which is studied in Chapters 3 and 4. The latter approach is studied in Chapter 5.

2.5.2 Other Transformations

There are several acoustic feature transformations besides integrating dynamic information. One of these aims at transforming the feature space so that the resulting covariance matrices are as diagonal dominant as possible (that is, decorrelation) without reducing dimensionality. In the acoustic feature space after the transformation, diagonal-constraint acoustic modeling would achieve comparable performance to full covariance modeling, while the constraint modeling uses fewer parameters than the full covariance one. This transformation was independently proposed by Gales [72,73] and Gopinath [74], and was called semi-tied covariances in [72,73] and maximum likelihood linear transformation (MLLT) in [74], respectively. Moreover, the transformation has been extended by increasing the degrees of freedom of the transformations such as extended MLLT (EMLLT) [75], mixture of inverse covariances (MIC) [76] and subspace precision and mean (SPAM) [77]. In EMLLT, MIC and SPAM, the inverse covariances are modeled as a weighted sum of rank-one matrices, symmetric matrices, and globally shared full-rank matrices, respectively.

As a speaker normalization technique, frequency or quefrency warping of speech signals has been proposed to deal with a difference in vocal tract length [78–82]. This acoustic feature transformation is called vocal tract length normalization (VTLN). VTLN can be implemented by warping the frequency axis in the filterbank analysis or the quefrency axis in the cepstrum analysis.

There are several ways to apply linear transformations for speaker adaptation. In [83, 84], the test data from the target speaker are transformed by meas of spectral mapping. Leggetter et al. [85] proposed a maximum likelihood linear regression (MLLR) in which an affine

transformation was applied to the mean vectors of the probability density function of acoustic models. In [86–88], extended transformations of MLLR were proposed. The speaker-dependent transformations did not only affect the mean vectors of the pdf but also its covariance matrix.

Since these transformations may improve condition-specific performance rather than basic performance, we do not discuss them here.

2.6 Summary

This chapter introduces a theoretical framework for automatic speech recognition. First, the front-end processing was presented. Then, hidden Markov models for acoustic modeling and the conventional decoding algorithm were presented. Finally, acoustic feature transformation, which is mainly discussed in the thesis, was described in detail.

Chapter 3

Generalization of LDA, HDA and HLDA

This and the following chapter investigate linear feature transformation methods that maximize the ratio of the between-class covariance to the within-class one. This chapter starts with an introduction and a review of conventional linear feature transformations in speech recognition. This chapter shows that the conventional methods have a close relationship. Then, a unified view of the conventional methods is described. This chapter proposes a common framework of the conventional methods, which can describe various criteria by varying its control parameter. This chapter also provides an efficient selection method of a sub-optimal control parameter without training of HMMs or testing recognition performance. Finally, this chapter investigates combinations of discriminant analysis-based feature transformation and discriminative training of acoustic models.

3.1 Introduction

To enhance basic performance of a speech recognizer, acoustic feature transformations according to a criterion which maximizes between-class scatter and minimizes within-class scatter are successfully applied to a speech recognizer. The basic idea of the criterion is that data in the same class are close to each other while data in different classes are separate from each other. The criterion is usually defined as the ratio of the between-class scatter to the within-class scatter. The most popular method using the criterion is linear discriminant analysis (LDA) [16, 17]. LDA is widely used to reduce dimensionality in speech recognition and a powerful tool to preserve discriminative information. Campbell [18] pointed out that LDA can be derived from the maximum likelihood parameter estimation method for class distributions assumed Gaussians with different means and common covariance. Therefore, LDA assumes each class share the same class covariance. However, this assumption does not necessarily hold for a

real data set. In order to overcome this limitation, several extensions have been proposed. Heteroscedastic linear discriminant analysis (HLDA) can deal with unequal covariances because the maximum likelihood estimation is used to estimate parameters for different Gaussians with unequal covariances [19]. Heteroscedastic discriminant analysis (HDA) was proposed as another objective function which employed individual weighted contributions of the classes [20]. The effectiveness of these methods for some data sets has been experimentally demonstrated.

This chapter shows that these three conventional methods have a close relationship. Then, a unified view of the three methods is described. All these three methods can be formally described in a common framework. The novel framework, called *power linear discriminant analysis* (PLDA) [89, 90], can describe various criteria by varying a control parameter of PLDA. PLDA includes LDA, HLDA, and HDA as special cases. Since PLDA can describe various criteria for dimensionality reduction, it can flexibly adapt to various environments such as a noisy environment. Thus, PLDA can provide robustness to a speech recognizer in realistic environments. Unfortunately, we cannot know which control parameter is the most effective before training HMMs and testing the performance of each control parameter. In general, this training and testing process incurs more than several dozen hours. Moreover, the computational time is proportional to the number of variations of the control parameters under test. Therefore, PLDA incurs considerable time to find an optimal control parameter because its control parameter can be set to a real number. This chapter provides an efficient selection method of a sub-optimal control parameter without training of HMMs or testing recognition performance [91].

Besides acoustic feature transformations, discriminative training techniques for acoustic models have also led to significant improvements in speech recognition performance on many tasks. Various criteria for discriminative training of acoustic models have been studied. Maximum mutual information (MMI) and minimum phone error (MPE) criteria have been successfully applied to many speech recognition systems [46, 53, 56]. MPE training has been shown to produce a more accurate model than MMI training, and therefore has been adopted widely. Both acoustic feature transformation techniques and discriminative training techniques aim to improve speech recognition performance at different levels. The combination of these two techniques can further improve speech recognition performance [90, 92–96]. This chapter investigates combinations of discriminant analysis-based feature transformation and discriminative training through experiments using in-car speech [96]. For feature transformation techniques, we evaluate not only traditional techniques, such as LDA and HDA, but also state-of-the-art techniques such as PLDA [89], oriented discriminant analysis [27] and heteroscedastic extension of LDA using Chernoff criterion [28]. The robustness against mismatched noise conditions between training and evaluation environments is also investigated.

3.2 Maximization Criteria of Ratio of Between-class Scatter to Within-class Scatter

This section defines the problem of dimensionality reduction, briefly reviews LDA, HLDA and HDA, and then investigates the effectiveness of these methods for some artificial data sets.

3.2.1 Definition of Problem of Dimensionality Reduction

Dimensionality reducing transformations project an n-dimensional feature space into a lower dimensional one with dimension $p < n$. The goal of dimensionality reduction is to find a representation of a manifold, i.e., a coordinate system, that will allow to project the data vectors on it and obtain a low-dimensional compact representation of the data, which preserves discriminative information. Suppose we have N vectors $\mathbf{x}_j \in \mathbb{R}^n (j = 1, 2, \ldots, N)$, where $\mathbf{x}_j$ consists of several successive features $\mathbf{x}_j = \left[\mathbf{o}_{j-(d-1)}^\top, \cdots, \mathbf{o}_j^\top\right]^\top$, and associated class labels $y_j \in \{1, 2, \ldots, K\}$ such as phones. The following notation are used in this thesis: capital bold letters refer to matrices, e.g., $\mathbf{A}$, bold letters refer to vectors, e.g., $\mathbf{b}$, and scalars are not bold, e.g., c. $\mathbf{A}^\top$ is the transpose of the matrix.

The dimensionality reduction mapping function is given by:

$$\begin{array}{rccc} \mathbf{F}: & \mathbb{R}^n & \rightarrow & \mathbb{R}^p \\ & \mathbf{x} & \mapsto & \mathbf{z} = \mathbf{F}(\mathbf{x}), \end{array} \tag{3.1}$$

where $\mathbf{z}$ is a reduced vector lying in $\mathbb{R}^p$. The function $\mathbf{F}$ maps an n-dimensional feature vector into a p-dimensional one.

In the case of linear transformations discussed in this thesis, a mapping function is given by:

$$\mathbf{F}(\mathbf{x}) = \mathbf{B}_{n\times p}^\top \mathbf{x}, \tag{3.2}$$

where $\mathbf{B}_{n\times p}$ denotes a transformation matrix in $\mathbb{R}^{n\times p}$, whose column vectors span a p-dimensional subspace. The purpose of linear dimensionality reduction is to find an optimal transformation matrix $\mathbf{B}_{n\times p}$ preserving discriminative information.

3.2.2 Linear Discriminant Analysis

The most popular method to reduce dimensionality is linear discriminant analysis (LDA) [16,17]. In LDA, within-class, between-class and mixture covariance matrices are used to formulate its

objective function. These covariance matrices are defined as follows [16, 17]:

$$\begin{aligned}
\mathbf{C}^{(W)} &= \frac{1}{N}\sum_{k=1}^{K}\sum_{j:y_j=k}(\mathbf{x}_j-\boldsymbol{\mu}_k)(\mathbf{x}_j-\boldsymbol{\mu}_k)^\top \\
&= \sum_{k=1}^{K} P_k\mathbf{C}_k, \\
\mathbf{C}^{(B)} &= \sum_{k=1}^{K} P_k(\boldsymbol{\mu}_k-\boldsymbol{\mu})(\boldsymbol{\mu}_k-\boldsymbol{\mu})^\top, \\
\mathbf{C}^{(M)} &= \frac{1}{N}\sum_{j=1}^{N}(\mathbf{x}_j-\boldsymbol{\mu})(\mathbf{x}_j-\boldsymbol{\mu})^\top \\
&= \mathbf{C}^{(W)}+\mathbf{C}^{(B)},
\end{aligned} \tag{3.3}$$

where $\boldsymbol{\mu}_k$ is the mean of features in class k, $\boldsymbol{\mu}$ is the mean of all features regardless of their class assignments, $\mathbf{C}_k$ is the class covariance of class k, and P_k is the weight for class k. In general, P_k is empirically given by $P_k = N_k/N$, where N_k is the number of features in class k. Each class covariance $\mathbf{C}_k$ is defined as:

$$\mathbf{C}_k = \frac{1}{N_k}\sum_{j:y_j=k}(\mathbf{x}_j-\boldsymbol{\mu}_k)(\mathbf{x}_j-\boldsymbol{\mu}_k)^\top.$$

There are several definitions of LDA objective functions. Typical objective functions are the following [16, 17]:

$$J_{LDA}\left(\mathbf{B}_{n\times p}\right) = \frac{\left|\mathbf{B}_{n\times p}^\top\mathbf{C}^{(B)}\mathbf{B}_{n\times p}\right|}{\left|\mathbf{B}_{n\times p}^\top\mathbf{C}^{(W)}\mathbf{B}_{n\times p}\right|}, \tag{3.4}$$

$$J_{LDA}\left(\mathbf{B}_{n\times p}\right) = \frac{\left|\mathbf{B}_{n\times p}^\top\mathbf{C}^{(M)}\mathbf{B}_{n\times p}\right|}{\left|\mathbf{B}_{n\times p}^\top\mathbf{C}^{(W)}\mathbf{B}_{n\times p}\right|}, \tag{3.5}$$

where $|\mathbf{X}|$ is the determinant of the matrix $\mathbf{X}$. A projection matrix is obtained by maximizing the objective function with respect to $\mathbf{B}_{n\times p}$. The optimizations of Equations (3.4) and (3.5) result in the same projection [16].

3.2.3 Heteroscedastic Extensions

LDA is not the optimal projection when the class distributions are heteroscedastic. Campbell [18] has shown that LDA is related to the maximum likelihood estimation of parameters for

a Gaussian model with an identical class covariance. However, this assumption is not necessarily satisfied for a real data set.

In order to overcome this limitation, several extensions have been proposed [19, 20, 27, 28]. This section focuses on two heteroscedastic extensions called heteroscedastic linear discriminant analysis (HLDA) and heteroscedastic discriminant analysis (HDA) [19, 20].

Heteroscedastic Linear Discriminant Analysis

In HLDA, the full-rank linear projection matrix $\mathbf{B}_{n\times n} \in \mathbb{R}^{n\times n}$ is constrained as follows: the first p columns of $\mathbf{B}_{n\times n}$ span the p-dimensional subspace in which the class means and variances are different and the remaining $n-p$ columns of $\mathbf{B}_{n\times n}$ span the $(n-p)$-dimensional subspace in which the class means and variances are identical. Let the parameters that describe the class means and covariances of $\mathbf{B}_{n\times n}^{\top}\mathbf{x}$ be $\hat{\boldsymbol{\mu}}_k$ and $\hat{\mathbf{C}}_k$, respectively:

$$\hat{\boldsymbol{\mu}}_k = \begin{bmatrix} \mathbf{B}_{n\times p}^{\top}\boldsymbol{\mu}_k \\ \mathbf{B}_{n\times(n-p)}^{\top}\boldsymbol{\mu} \end{bmatrix}, \tag{3.6}$$

$$\hat{\mathbf{C}}_k = \begin{bmatrix} \mathbf{B}_{n\times p}^{\top}\mathbf{C}_k\mathbf{B}_{n\times p} & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_{n\times(n-p)}^{\top}\mathbf{C}^{(M)}\mathbf{B}_{n\times(n-p)} \end{bmatrix}, \tag{3.7}$$

where full rank matrix $\mathbf{B}_{n\times n} = \left[\mathbf{B}_{n\times p}|\mathbf{B}_{n\times(n-p)}\right]$ and $\mathbf{B}_{n\times(n-p)} \in \mathbb{R}^{n\times(n-p)}$, and $\mathbf{0}$ denotes a zero matrix whose entries are zero.

Kumar et al. [19] incorporated the maximum likelihood estimation of parameters for differently distributed Gaussians. The probability density of $\mathbf{x}_i$ under the preceding model is given as:

$$P(\mathbf{x}_i) = \frac{|\mathbf{B}_{n\times n}|}{\left|2\pi\hat{\mathbf{C}}_{y_i}\right|}\exp\left(-\frac{(\mathbf{z}_i-\hat{\boldsymbol{\mu}}_{y_i})^{\top}\hat{\mathbf{C}}_{y_i}(\mathbf{z}_i-\hat{\boldsymbol{\mu}}_{y_i})}{2}\right),$$

where $\mathbf{z}_i = \mathbf{B}_{n\times n}\mathbf{x}_i$ and $\mathbf{x}_i$ belongs to the group y_i. The log-likelihood of the data $L = \sum_{i=1}^{N}\log P(\mathbf{x}_i)$ under the linear transformation $\mathbf{B}_{n\times n}$ and under the constrained Gaussian model assumption for each class is:

$$\log L = -\frac{1}{2}\sum_{i=1}^{N}\left\{(\mathbf{z}_i-\hat{\boldsymbol{\mu}}_{y_i})^{\top}\hat{\mathbf{C}}_{y_i}(\mathbf{z}_i-\hat{\boldsymbol{\mu}}_{y_i}) + \log\left|2\pi\hat{\mathbf{C}}_{y_i}\right|\right\} + \log|\mathbf{B}_{n\times n}|.$$

We can rearrange this by first calculating the values of the mean and covariance parameters:

$$\begin{aligned}\log L = &-\frac{N}{2}\log|\mathbf{B}_{n\times(n-p)}^{\top}\mathbf{C}^{(M)}\mathbf{B}_{n\times(n-p)}| - \sum_{k=1}^{K}\frac{N_k}{2}\log|\mathbf{B}_{n\times p}^{\top}\mathbf{C}_k\mathbf{B}_{n\times p}| - \frac{Nn}{2}\log 2\pi \\ &- \frac{1}{2}\sum_{k=1}^{K}\sum_{y_i=k}(\mathbf{x}_i-\boldsymbol{\mu}_k)^{\top}\mathbf{B}_{n\times(n-p)}(\mathbf{B}_{n\times(n-p)}^{\top}\mathbf{C}^{(M)}\mathbf{B}_{n\times(n-p)})^{-1}\mathbf{B}_{n\times(n-p)}^{\top}(\mathbf{x}_i-\boldsymbol{\mu}_k) \\ &- \frac{1}{2}\sum_{k=1}^{K}\sum_{y_i=k}(\mathbf{x}_i-\boldsymbol{\mu}_k)^{\top}\mathbf{B}_{n\times p}(\mathbf{B}_{n\times p}^{\top}\mathbf{C}_k\mathbf{B}_{n\times p})^{-1}\mathbf{B}_{n\times p}^{\top}(\mathbf{x}_i-\boldsymbol{\mu}_k) + N\log|\mathbf{B}_{n\times n}|\end{aligned}$$

Then, an HLDA objective function is derived as follows [19]:

$$J_{HLDA}(\mathbf{B}_{n\times n}) = \frac{|\mathbf{B}_{n\times n}|^{2N}}{\left|\mathbf{B}_{n\times(n-p)}^{\top}\mathbf{C}^{(M)}\mathbf{B}_{n\times(n-p)}\right|^{N}}\frac{1}{\prod_{k=1}^{K}\left|\mathbf{B}_{n\times p}^{\top}\mathbf{C}_k\mathbf{B}_{n\times p}\right|^{N_k}}, \tag{3.8}$$

The solution to maximize Equation (3.8) is not analytically obtained. Therefore, its maximization is performed using a numerical optimization technique. Alternatively, a computationally efficient scheme is given in [73].

Heteroscedastic Discriminant Analysis

HDA uses the following objective function which incorporates individual weighted contributions of the class variances [20]:

$$\begin{aligned}J_{HDA}(\mathbf{B}_{n\times p}) &= \prod_{k=1}^{K}\left(\frac{\left|\mathbf{B}_{n\times p}^{\top}\mathbf{C}^{(B)}\mathbf{B}_{n\times p}\right|}{\left|\mathbf{B}_{n\times p}^{\top}\mathbf{C}_k\mathbf{B}_{n\times p}\right|}\right)^{N_k} \\ &= \frac{\left|\mathbf{B}_{n\times p}^{\top}\mathbf{C}^{(B)}\mathbf{B}_{n\times p}\right|^{N}}{\prod_{k=1}^{K}\left|\mathbf{B}_{n\times p}^{\top}\mathbf{C}_k\mathbf{B}_{n\times p}\right|^{N_k}}.\end{aligned} \tag{3.9}$$

In contrast to HLDA, this function is not considered $(n-p)$ dimensions. Only a projection matrix $\mathbf{B}_{n\times p}$ is estimated. There is no closed-form solution to obtain projection matrix $\mathbf{B}_{n\times p}$ similar to HLDA.

3.2.4 Dependency on Data Set

In Figure 3.1, two-dimensional, two- or three-class data features are projected onto one-dimensional subspaces by LDA and HDA. Here, HLDA projections were omitted because they were close to

HDA projections. Figure 3.1(a) shows that HDA has higher separability than LDA for the data set used in [20]. On the other hand, as shown in Figure 3.1(b), LDA has higher separability than HDA for another data set. Figure 3.1(c) shows the case with another data set where both LDA and HDA have low separabilities. Thus, LDA and HDA do not always classify the given data set appropriately. All results show that the separabilities of LDA and HDA depend significantly on data sets.

3.3 Generalization of Discriminant Analyses

As shown above, it is difficult to separate appropriately every data set with one particular criterion such as LDA, HLDA or HDA. Here, we concentrate on providing a framework which integrates various criteria.

3.3.1 Relationship between HLDA and HDA

By using Equations (3.3), (3.6) and (3.7), let us rearrange $\mathbf{B}_{n\times n}^{\top}\mathbf{C}^{(M)}\mathbf{B}_{n\times n}$ as follows:

$$\begin{aligned}\mathbf{B}_{n\times n}^{\top}\mathbf{C}^{(M)}\mathbf{B}_{n\times n} &= \mathbf{B}_{n\times n}^{\top}\mathbf{C}^{(B)}\mathbf{B}_{n\times n} + \mathbf{B}_{n\times n}^{\top}\mathbf{C}^{(W)}\mathbf{B}_{n\times n} \\ &= \sum_{k} P_k(\hat{\boldsymbol{\mu}}_k - \hat{\boldsymbol{\mu}})(\hat{\boldsymbol{\mu}}_k - \hat{\boldsymbol{\mu}})^{\top} + \sum_{k} P_k\hat{\mathbf{C}}_k \\ &= \begin{bmatrix} \mathbf{B}_{n\times p}^{\top}\mathbf{C}^{(M)}\mathbf{B}_{n\times p} & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_{n\times(n-p)}^{\top}\mathbf{C}^{(M)}\mathbf{B}_{n\times(n-p)} \end{bmatrix}, \end{aligned} \tag{3.10}$$

where $\hat{\boldsymbol{\mu}} \equiv \mathbf{B}_{n\times n}^{\top}\boldsymbol{\mu}$.

The determinant of this is

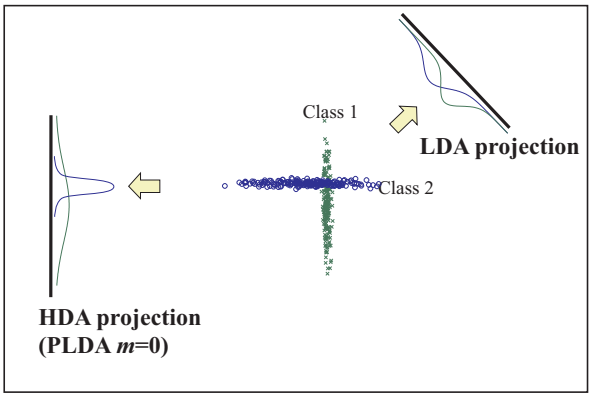
$$\left|\mathbf{B}_{n\times n}^{\top}\mathbf{C}^{(M)}\mathbf{B}_{n\times n}\right| = \left|\mathbf{B}_{n\times p}^{\top}\mathbf{C}^{(M)}\mathbf{B}_{n\times p}\right| \left|\mathbf{B}_{n\times(n-p)}^{\top}\mathbf{C}^{(M)}\mathbf{B}_{n\times(n-p)}\right|. \tag{3.11}$$

Hence, we have

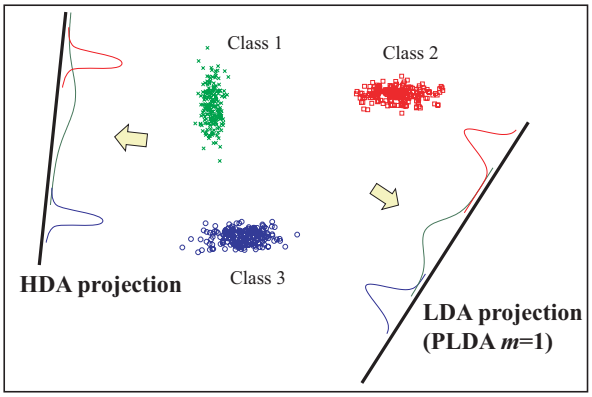
$$\frac{\left|\mathbf{C}^{(M)}\right|^N |\mathbf{B}_{n\times n}|^{2N}}{\left|\mathbf{B}_{n\times(n-p)}^{\top}\mathbf{C}^{(M)}\mathbf{B}_{n\times(n-p)}\right|^N} = \left|\mathbf{B}_{n\times p}^{\top}\mathbf{C}^{(M)}\mathbf{B}_{n\times p}\right|^N. \tag{3.12}$$

Inserting this in Equation (3.8) and removing a constant term yields

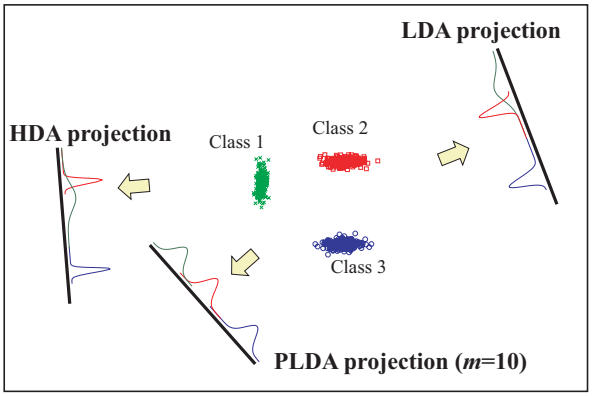
$$J_{HLDA}\left(\mathbf{B}_{n\times p}\right) \propto \frac{\left|\mathbf{B}_{n\times p}^{\top}\mathbf{C}^{(M)}\mathbf{B}_{n\times p}\right|^N}{\prod_{k=1}^{K}\left|\mathbf{B}_{n\times p}^{\top}\mathbf{C}_k\mathbf{B}_{n\times p}\right|^{N_k}}. \tag{3.13}$$



(a)



(b)



(c)

Figure 3.1: Examples of dimensionality reduction by LDA, HDA and PLDA.

From Equations (3.9) and (3.13), the difference between HLDA and HDA lies in their numerators, i.e., the mixture covariance matrix versus the between-class covariance matrix. This difference is the same as the difference between the two LDAs shown in (3.4) and (3.5). Thus, Equations (3.9) and (3.13) can be viewed as the same formulation except their numerators. Later, we write $\mathbf{B}$ to mean $\mathbf{B}_{n\times p}$. This will simplify the notation.

3.3.2 Relationship between LDA and HDA

The LDA and HDA objective functions can be rewritten as

$$J_{LDA}\left(\mathbf{B}\right) = \frac{\left|\mathbf{B}^\top \mathbf{C}^{(B)} \mathbf{B}\right|}{\left|\mathbf{B}^\top \mathbf{C}^{(W)} \mathbf{B}\right|} = \frac{\left|\tilde{\mathbf{C}}^{(B)}\right|}{\left|\sum_{k=1}^{K} P_k \tilde{\mathbf{C}}_k\right|}, \tag{3.14}$$

$$J_{HDA}\left(\mathbf{B}\right) = \frac{\left|\mathbf{B}^\top \mathbf{C}^{(B)} \mathbf{B}\right|^N}{\prod_{k=1}^{K} \left|\mathbf{B}^\top \mathbf{C}_k \mathbf{B}\right|^{N_k}} = \left(\frac{\left|\tilde{\mathbf{C}}^{(B)}\right|}{\prod_{k=1}^{K} \left|\tilde{\mathbf{C}}_k\right|^{P_k}} \right)^N, \tag{3.15}$$

where $\tilde{\mathbf{C}}^{(B)} \equiv \mathbf{B}^\top \mathbf{C}^{(B)} \mathbf{B}$ and $\tilde{\mathbf{C}}_k \equiv \mathbf{B}^\top \mathbf{C}_k \mathbf{B}$ are between-class and class k covariance matrices in the projected p-dimensional space, respectively. Here, we rewrite $J_{HDA}\left(\mathbf{B}\right)$ as follows:

$$J_{HDA}\left(\mathbf{B}\right) = \frac{\left|\tilde{\mathbf{C}}^{(B)}\right|}{\prod_{k=1}^{K} \left|\tilde{\mathbf{C}}_k\right|^{P_k}}. \tag{3.16}$$

Maximizations of Equations (3.15) and (3.16) result in the same transformation.

Both numerators denote determinants of the between-class covariance matrix. In Equation (3.5), the denominator can be viewed as a determinant of *the weighted arithmetic mean* of the class covariance matrices. Similarly, in Equation (3.15), the denominator can be viewed as a determinant of *the weighted geometric mean* of the class covariance matrices. Thus, the difference between LDA and HDA is the definitions of the mean of the class covariance matrices. Moreover, to replace their numerators with the determinants of the mixture covariance matrices, the difference between LDA and HLDA is the same as the difference between LDA and HDA.

3.3.3 Power Linear Discriminant Analysis

As described above, Equations (3.5) and (3.15) give us a new integrated interpretation of LDA and HDA. As an extension of this interpretation, their denominators can be replaced by a determinant of *the weighted harmonic mean*, or a determinant of *the root mean square*.

In the econometric literature, a more general definition of a mean is often used, called *the weighted mean of order m* [97]. We extend this notion to a determinant of a matrix mean and propose a new objective function as follows [1] :

$$J_{PLDA}\left(\mathbf{B}, m\right) = \frac{\left|\tilde{\mathbf{C}}_n\right|}{\left|\left(\sum_{k=1}^{K} P_k \tilde{\mathbf{C}}_k^m\right)^{1/m}\right|}, \tag{3.17}$$

where $\tilde{\mathbf{C}}_n \in \{\tilde{\mathbf{C}}^{(B)}, \tilde{\mathbf{C}}^{(M)}\}$, $\tilde{\mathbf{C}}^{(M)} \equiv \mathbf{B}^\top \mathbf{C}^{(M)} \mathbf{B}$, and m is a control parameter. By varying the control parameter m, the proposed objective function can represent various criteria. Some typical objective functions are enumerated below.

- $m = 2$ (root mean square)

$$J_{PLDA}\left(\mathbf{B}, 2\right) = \frac{\left|\tilde{\mathbf{C}}_n\right|}{\left|\left(\sum_{k=1}^{K} P_k \tilde{\mathbf{C}}_k^2\right)^{1/2}\right|}. \tag{3.18}$$

- $m = 1$ (arithmetic mean)

$$J_{PLDA}\left(\mathbf{B}, 1\right) = \frac{\left|\tilde{\mathbf{C}}_n\right|}{\left|\sum_{k=1}^{K} P_k \tilde{\mathbf{C}}_k\right|} = J_{LDA}\left(\mathbf{B}\right). \tag{3.19}$$

- $m \to 0$ (geometric mean)

$$J_{PLDA}\left(\mathbf{B}, 0\right) = \frac{\left|\tilde{\mathbf{C}}_n\right|}{\prod_{k=1}^{K} \left|\tilde{\mathbf{C}}_k\right|^{P_k}} = J_{HDA}\left(\mathbf{B}\right). \tag{3.20}$$

[1] We let the function f of a symmetric positive definite matrix $\mathbf{A}$ equal $\mathbf{U} diag(f(\lambda_1), \ldots, f(\lambda_n)) \mathbf{U}^T = \mathbf{U}(f(\mathbf{\Lambda}))\mathbf{U}^T$, where $\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$, $\mathbf{U}$ denotes the matrix of n eigenvectors, and $\mathbf{\Lambda}$ denotes the diagonal matrix of eigenvalues, λ_i's. We may define the function f as some power or the logarithm of $\mathbf{A}$.

- $m = -1$ (harmonic mean)

$$J_{PLDA}(\mathbf{B}, -1) = \frac{\left|\tilde{\mathbf{C}}_n\right|}{\left|\left(\sum_{k=1}^{K} P_k \tilde{\mathbf{C}}_k^{-1}\right)^{-1}\right|}. \tag{3.21}$$

The derivation of Equation (3.20) is given in Appendix A.1. The following equations are also obtained under a particular condition (see Appendix A.2).

- $m \to \infty$

$$J_{PLDA}(\mathbf{B}, \infty) = \frac{\left|\tilde{\mathbf{C}}_n\right|}{\max_k \left|\tilde{\mathbf{C}}_k\right|}. \tag{3.22}$$

- $m \to -\infty$

$$J_{PLDA}(\mathbf{B}, -\infty) = \frac{\left|\tilde{\mathbf{C}}_n\right|}{\min_k \left|\tilde{\mathbf{C}}_k\right|}. \tag{3.23}$$

Intuitively, as m becomes larger, the classes with larger variances become dominant in the denominator of Equation (3.17). Conversely, as m becomes smaller, the classes with smaller variances become dominant.

We call this new discriminant analysis formulation *Power Linear Discriminant Analysis* (PLDA). Figure 3.1(c) shows that PLDA can have a higher separability for a data set with which LDA and HDA have lower separability. To maximize the PLDA objective function with respect to $\mathbf{B}$, we can use numerical optimization techniques such as the Nelder-Mead method [98] or the SANN method [99]. These methods need no derivatives of the objective function. However, it is known that these methods converge slowly. In some special cases below, using a matrix differential calculus [100], the derivatives of the objective function are obtained. Hence, we can use some fast convergence methods, such as the quasi-Newton method and conjugate gradient method [101].

Order m Constrained to Be An Integer

Assuming that a control parameter m is constrained to be an integer, the derivatives of the PLDA objective function are formulated as follows:

$$\frac{\partial}{\partial \mathbf{B}} \log J_{PLDA}(\mathbf{B}, m) = 2\mathbf{C}_n \mathbf{B} \tilde{\mathbf{C}}_n^{-1} - 2\mathbf{D}_m, \tag{3.24}$$

where

$$\mathbf{D}_m = \begin{cases} \frac{1}{m}\sum_{k=1}^{K} P_k \mathbf{C}_k \mathbf{B} \sum_{j=1}^{m} \mathbf{X}_{m,j,k}, & \text{if } m > 0 \\ \sum_{k=1}^{K} P_k \mathbf{C}_k \mathbf{B} \tilde{\mathbf{C}}_k^{-1}, & \text{if } m = 0 \\ -\frac{1}{m}\sum_{k=1}^{K} P_k \mathbf{C}_k \mathbf{B} \sum_{j=1}^{|m|} \mathbf{Y}_{m,j,k}, & \text{otherwise} \end{cases}$$

$$\mathbf{X}_{m,j,k} = \tilde{\mathbf{C}}_k^{m-j} \left(\sum_{l=1}^{K} P_l \tilde{\mathbf{C}}_l^m \right)^{-1} \tilde{\mathbf{C}}_k^{j-1},$$

and

$$\mathbf{Y}_{m,j,k} = \tilde{\mathbf{C}}_k^{m+j-1} \left(\sum_{l=1}^{K} P_l \tilde{\mathbf{C}}_l^m \right)^{-1} \tilde{\mathbf{C}}_k^{-j}.$$

This equation is used for acoustic models with full covariance.

$\tilde{\mathbf{C}}_k$ Constrained to Be Diagonal

Because of computational simplicity, the covariance matrix in class k is often assumed to be diagonal [19, 20]. Since a diagonal matrix multiplication is commutative, the derivatives of the PLDA objective function are simplified as follows:

$$J_{PLDA}(\mathbf{B}, m) = \frac{\left|\tilde{\mathbf{C}}_n\right|}{\left|\left(\sum_{k=1}^{K} P_k diag(\tilde{\mathbf{C}}_k)^m\right)^{1/m}\right|}, \tag{3.25}$$

$$\frac{\partial}{\partial \mathbf{B}} \log J_{PLDA}(\mathbf{B}, m) = 2\mathbf{C}_n \mathbf{B} \tilde{\mathbf{C}}_n^{-1} - 2\mathbf{F}_m \mathbf{G}_m, \tag{3.26}$$

where

$$\mathbf{F}_m = \sum_{k=1}^{K} P_k \mathbf{C}_k \mathbf{B} diag\left(\tilde{\mathbf{C}}_k\right)^{m-1}, \tag{3.27}$$

$$\mathbf{G}_m = \left(\sum_{k=1}^{K} P_k diag\left(\tilde{\mathbf{C}}_k\right)^m\right)^{-1}, \tag{3.28}$$

and $diag(\mathbf{A})$ is an operator which sets zero for off-diagonal elements of $\mathbf{A}$. In Equation (3.25), the control parameter m can be any real number, unlike in Equation (3.24).

When m is equal to zero, the PLDA objective function corresponds to the diagonal HDA (DHDA) objective function introduced in [20].

3.3.4 Experiments

Speech recognition experiments on the CENSREC-3 database [102] are presented below. The database was designed as an evaluation framework of Japanese isolated word recognition in real driving car environments. Speech data was collected using 2 microphones, a close-talking (CT) microphone and a hands-free (HF) microphone. For training, driver's speech of phonetically-balanced sentences was recorded under two conditions: while idling and driving on a city street with normal in-car environment. A total of 14,050 utterances spoken by 293 drivers (202 males and 91 females) were recorded with each microphone. For evaluation, driver's speech of isolated words was recorded under 16 environmental conditions using combinations of three kinds of vehicle speeds (idling, low-speed driving on a city street, and high-speed driving on an express-way) and six kinds of in-car environments (normal, with hazard flasher on, with air-conditioner on (fan low/high), with audio CD player on, and with windows open). The speech signals for training and evaluation were both sampled at 16 kHz.

Baseline System

The acoustic models consisted of triphone HMMs. In order to train HMMs, all utterances recorded with CT and HF microphones were used. Each HMM had five states and three of them had output distributions. Each distribution was represented with 32 mixture diagonal Gaussians. The total number of states with the distributions were 2,000. The feature vector consisted of 12 MFCCs and log-energy with their corresponding delta and acceleration coefficients (39 dimensions). Frame length was 20 ms and frame shift was 10 ms. In the Mel-filter bank analysis, a cut-off was applied to frequency components lower than 250 Hz. The decoding process was performed without any language model. The vocabulary of the CENSREC-3 was 50 words [102], which is listed in Table 3.1. Fifty similar-sounding out-of-vocabulary words listed in Table 3.2 were appended to the vocabulary to make recognition tasks difficult. For evaluation, we used driver's speech recorded under three kinds of vehicle speeds in normal in-car environment. A total of 2,646 utterances spoken by 18 speakers (8 males and 10 females) were evaluated for each microphone.

Dimensionality Reduction Procedure

The dimensionality reduction was performed using PCA, LDA, (D)HDA, and PLDA for the spliced features. Eleven successive frames (143 dimensions) were reduced to 39 dimensions.

Table 3.1: A list of 50 words for evaluation.

digital_locker	ninsho_kaishi
2001/1/1	yamada_tarou
kensaku_shuryo	ansho_bango
0123	4567
8901	2345
6789	contents
eiga	hitsuji_tachino_chinmoku
sound_of_music	game
pack_man	ongaku
jpop	konsyu_no_top10
genre_betsu_kensaku	pops
rock	beatles
senkyoku	yesterday
let_it_be	haishin_kaishi
ferry_annai	jikoku_hyo
dai2bin_wo_yoyaku	net_news
topics	onsei_yomiage
tenki_yohou	koutsu_jouhou
kanagawa_ken	yokohama_shi
naka_ku	toukyou_to
setagaya_ku	syuto_kousoku
touhoku_jidoushadou	seven_eleven
uniqlo	star_bucks
hotel_ichiran	pacific_hotel
yoyaku_hyo	service_syuryo

Table 3.2: A list of appended 50 words for evaluation.

analog_locker	ninshiki_kaishi
2001/2/1	yamuda_tarou
kensaku_kaishi	ansho_ango
01223	45677
890	2335
6289	latent
keikaku	shitsuji_tachino_chinmoku
bound_of_music	aim
rock_man	hongaku
atop	honsyu_no_top10
genre_betsu_kenbetsuku	tops
look	fii_toruzu
wankyoku	iesta_wei
pet_it_be	henshin_kaishi
ferry_kannai	yokoku_hyo
kai2ben_wo_yoyaku	let_news
po_pics	onsei_momiage
tenki_gohou	koutsu_youhou
kanagawa_en	yokohama_ri
waka_ku	toukyou_ko
setagawa_ku	syuto_kyousoku
touhoku_jidouhadou	tebun_eleven
kunikuro	sujar_bucks
potel_ichiren	pacific_potel
kakaku_hyo	service_kyuryo

Table 3.3: Word error rates (%) by PLDA and conventional methods.

Method	m	CT	HF	Overall
MFCC + Δ +$\Delta\Delta$	–	7.45	15.04	11.24
PCA	–	10.58	19.39	14.98
LDA	–	8.78	15.80	12.28
HDA	–	7.94	17.16	12.55
PLDA	−3.0	6.73	15.04	10.88
PLDA	−2.0	7.29	12.32	9.81
PLDA	−1.5	6.27	**10.70**	**8.48**
PLDA	−1.0	6.92	11.49	9.20
PLDA	−0.5	**6.12**	12.51	9.32
DHDA	(0.0)	7.41	14.17	10.79
PLDA	0.5	7.29	13.53	10.41
PLDA	1.0	9.33	16.97	13.15
PLDA	1.5	8.96	17.31	13.13
PLDA	2.0	8.58	15.91	12.24
PLDA	3.0	9.41	16.36	12.89

In HDA and PLDA, to optimize their objective functions, we used the limited-memory BFGS algorithm as a numerical optimization technique [101]. Assuming that projected covariance matrices were diagonal, Equation (3.26) was used to compute a gradient. The LDA transformation matrix was used for the initial gradient. To assign one of the classes to every feature after dimensionality reduction, HMM state labels were generated for the training data by state-level forced alignment algorithm using a well-trained HMM system. The number of classes was 43 corresponding to the number of the monophones.

Experimental Results

Experimental results are summarized in Table 3.3. For the evaluation data recorded with a CT microphone, PLDA with $m = -0.5$ yielded the lowest word error rate (WER). On the other hand, for the evaluation data recorded with a HF microphone, the lowest WER was obtained by PLDA with a different control parameter ($m = -1.5$). Thus, these two data sets recorded with different microphones have different optimal control parameters. Experimental results demonstrated that PLDA with the optimal control parameters consistently outperform the other methods.

3.4 Selection of Sub-Optimal Control Parameter

As shown in the previous section, PLDA can describe various criteria by varying its control parameter m, and the effectiveness of PLDA with the optimal control parameter has been experimentally demonstrated. One way of obtaining an optimal control parameter m is to train HMMs and test recognition performance changing m, and then to choose the m with the smallest error. Unfortunately, this raises a considerable problem in a speech recognition task. In general, to train HMMs and to test recognition performance requires more than several dozen hours. Since it is able to choose a control parameter within a real number and the computational time is proportional to the number of candidate control parameters, PLDA incurs considerable time to select the optimal one.

This section provides a sub-optimal control parameter selection method without training of HMMs and test. To evaluate the relative performance among a number of control parameters, we focus on a class separability error of projected features and measure it on training data. We show that the proposed method can rapidly and accurately compare with the relative recognition performance.

3.4.1 Estimating Sub-Optimal Control Parameter without Testing

In this section we focus on a class separability error of the features in the projected space instead of using a recognition error. Better recognition performance can be obtained under the lower class separability error of projected features. Consequently, we measure the class separability error and use it as a criterion for the recognition performance comparison. We will define a class separability error of projected features.

Two-class Problem

This subsection focuses on the two-class case. We first consider the Bayes error of the projected features on an evaluation data as a class separability error:

$$\varepsilon = \int \min[P_1 p_1(\mathbf{x}), P_2 p_2(\mathbf{x})] d\mathbf{x}, \tag{3.29}$$

where P_i denotes a prior probability of the class i and $p_i(\mathbf{x})$ is a conditional density function of the class i. The Bayes error ε can represent a classification error, assuming that the training data and the evaluation data come from the same distributions. However, it is difficult to directly measure the Bayes error. Instead, we use the Chernoff bound between class 1 and class 2 as a class separability error [16]:

$$\varepsilon_u^{1,2} = P_1^s P_2^{1-s} \int p_1^s(\mathbf{x}) p_2^{1-s}(\mathbf{x}) d\mathbf{x} \quad \text{for} \quad 0 \leq s \leq 1 \tag{3.30}$$

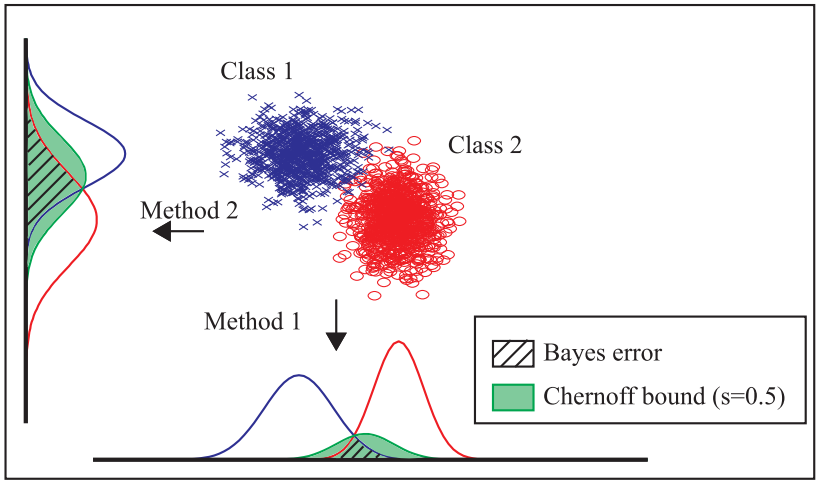


Figure 3.2: Examples of dimensionality reduction.

where $\varepsilon_u^{1,2}$ indicates an upper bound of ε. In addition, when the $p_i(\mathbf{x})$'s are normal with mean vectors $\boldsymbol{\mu}_i$ and covariance matrices $\mathbf{C}_i$, the Chernoff bound between class 1 and class 2 becomes

$$\varepsilon_u^{1,2} = P_1^s P_2^{1-s} \exp(-\eta^{1,2}(s)), \tag{3.31}$$

where

$$\begin{aligned}\eta^{1,2}(s) =& \frac{s(1-s)}{2}(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)^T \left(s\mathbf{C}_1 + (1-s)\mathbf{C}_2\right)^{-1} (\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1) \\ &+ \frac{1}{2} \ln \frac{|s\mathbf{C}_1 + (1-s)\mathbf{C}_2|}{|\mathbf{C}_1|^s |\mathbf{C}_2|^{1-s}}.\end{aligned} \tag{3.32}$$

In this case, ε_u can be obtained analytically and calculated rapidly.

In Figure 3.2, two-dimensional two-class data are projected onto a one-dimensional subspace by two methods. To compare with their Chernoff bounds, the lower class separability error is obtained from the projected features by Method 1 as compared with those by Method 2. In this case, Method 1 preserving the lower class separability error should be selected.

Extension to Multi-class Problem

In the previous subsection, we defined a class separability error for two-class data. Here, we extend a two-class case to a multi-class case. Unlike the two-class case, it is possible to define

several error functions for multi-class data. We define an error function as follows:

$$\tilde{\varepsilon}_u = \sum_{i=1}^{K} \sum_{j=1}^{K} I(i,j) \varepsilon_u^{i,j} \tag{3.33}$$

where $I(\cdot)$ denotes an indicator function. We consider the following three formulations as an indicator function.

Sum of Pairwise Approximated Errors The sum of all the pairwise Chernoff bounds is defined using the following indicator function:

$$I(i,j) = \begin{cases} 1, & \text{if } j > i, \\ 0, & \text{otherwise.} \end{cases} \tag{3.34}$$

Maximum Pairwise Approximated Error The maximum pairwise Chernoff bound is defined using the following indicator function:

$$I(i,j) = \begin{cases} 1, & \text{if } j > i \text{ and } (i,j) = (\hat{i},\hat{j}), \\ 0, & \text{otherwise,} \end{cases} \tag{3.35}$$

where $(\hat{i},\hat{j}) \equiv \arg\max_{i,j} \varepsilon_u^{i,j}$.

Sum of Maximum Approximated Errors in Each Class The sum of the maximum pairwise Chernoff bounds in each class is defined using the following indicator function:

$$I(i,j) = \begin{cases} 1, & \text{if } j = \hat{j}_i, \\ 0, & \text{otherwise,} \end{cases} \tag{3.36}$$

where $\hat{j}_i \equiv \arg\max_{j} \varepsilon_u^{i,j}$.

3.4.2 Parameter Selection Results

This section investigates the effectiveness of parameter selection methods under the same experimental condition presented in Section 3.3.4. In comparing dimensionality reduction criteria without training HMMs nor testing recognition performance, we used $s = 1/2$ for the Chernoff bound computation because there was no *a priori* information about weights of two class distributions. In the case of $s = 1/2$, Equation (3.30) is called the Bhattacharyya bound. Two covariance matrices in Equation (3.32) were treated as diagonal because diagonal Gaussians were used to model HMMs. The parameter selection was performed as follows: To select the

optimal control parameter for the data set recorded with a CT microphone, all the training data with a CT microphone were labeled with monophones using a forced alignment recognizer. Then, each monophone was modeled as a unimodal normal distribution, and the mean vector and covariance matrix of each class were calculated. Chernoff bounds were obtained using these mean vectors and covariance matrices. The optimal control parameter for the data set with an HF microphone was obtained using all of the training data with an HF microphone through the same process as a CT microphone. Both Tables 3.4 and 3.5 show that the results of the proposed method and relative recognition performance agree well. There was little difference in the parameter selection performances among Equations (3.34)-(3.36) in parameter selection accuracy. The proposed selection method yielded sub-optimal performance without training HMMs nor testing recognition performance on a development set, although it neglected time information of speech feature sequences to measure a class separability error and modeled a class distribution as a unimodal normal distribution. In addition, the optimal control parameter value can vary with different speech features, a different language, or a different noise environment. The proposed selection method can adapt to such variations.

3.4.3 Computational costs

The computational costs for the evaluation of recognition performance versus the proposed selection method are shown in Table 3.6. Here, the computational cost involves the optimization procedure of the control parameter. In this experiment, we evaluate the computational costs on the evaluation data set with a Pentium IV 2.8 GHz computer. For every dimensionality reduction criterion, the evaluation of recognition performance required 15 hours for training of HMMs and 5 hours for test. In total, 220 hours were required for comparing 11 dimensionality reduction criteria (PLDAs using 11 different control parameters). On the other hand, the proposed selection method only required approximately 30 minutes for calculating statistical values such as mean vectors and covariance matrices of each class in the original space. After this, 2 minutes were required to calculate Equations (3.34)-(3.36) for each dimensionality reduction criterion. In total, only 0.87 hour was required for predicting the optimal criterion among the 11 dimensionality reduction criteria described above. Thus, the proposed method could perform the prediction process much faster than a conventional procedure that included training of HMMs and test of recognition performance.

3.5 Combinational Use of Acoustic Feature Transformation and Discriminative Training

To improve speech recognition performance, the conventional and the proposed feature transformations such as LDA, HDA and PLDA were introduced in Sections 3.2 and 3.3. Recently, in

Table 3.4: Word error rates (%) and class separability errors according to Equations (3.34)-(3.36) for the evaluation set with CT microphone. The best results are highlighted in bold.

Method	WER	Eq. (3.34)	Eq. (3.35)	Eq. (3.36)
MFCC + Δ +$\Delta\Delta$	7.45	2.31	0.0322	0.575
PCA	10.58	3.36	0.0354	0.669
LDA	8.78	3.10	0.0354	0.641
HDA	7.94	2.99	0.0361	0.635
PLDA ($m = -3$)	6.73	2.02	0.0319	0.531
PLDA ($m = -2$)	7.29	2.07	0.0316	0.532
PLDA ($m = -1.5$)	6.27	**1.97**	0.0307	0.523
PLDA ($m = -1$)	6.92	1.99	0.0301	**0.521**
PLDA ($m = -0.5$)	**6.12**	2.01	**0.0292**	0.525
DHDA (PLDA m=0)	7.41	2.15	0.0296	0.541
PLDA ($m = 0.5$)	7.29	2.41	0.0306	0.560
PLDA ($m = 1$)	9.33	3.09	0.0354	0.641
PLDA ($m = 1.5$)	8.96	4.61	0.0394	0.742
PLDA ($m = 2$)	8.58	4.65	0.0404	0.745
PLDA ($m = 3$)	9.41	4.73	0.0413	0.756

Table 3.5: Word error rates (%) and class separability errors according to Equations (3.34)-(3.36) for the evaluation set with HF microphone. The best results are highlighted in bold.

Method	WER	Eq. (3.34)	Eq. (3.35)	Eq. (3.36)
MFCC + Δ +$\Delta\Delta$	15.04	2.56	0.0356	0.648
PCA	19.39	3.65	0.0377	0.738
LDA	15.80	3.38	0.0370	0.711
HDA	17.16	3.21	0.0371	0.697
PLDA ($m = -3$)	15.04	2.19	0.0338	0.600
PLDA ($m = -2$)	12.32	2.26	0.0339	0.602
PLDA ($m = -1.5$)	**10.70**	**2.18**	0.0332	**0.5921**
PLDA ($m = -1$)	11.49	2.23	**0.0327**	0.5922
PLDA ($m = -0.5$)	12.51	2.31	0.0329	0.598
DHDA (PLDA m=0)	14.17	2.50	0.0331	0.619
PLDA ($m = 0.5$)	13.53	2.81	0.0341	0.644
PLDA ($m = 1$)	16.97	3.38	0.0370	0.711
PLDA ($m = 1.5$)	17.31	5.13	0.0403	0.828
PLDA ($m = 2$)	15.91	5.22	0.0412	0.835
PLDA ($m = 3$)	16.36	5.36	0.0424	0.850

Table 3.6: Computational costs with the conventional and proposed method.

conventional	220 h = (15 h (training) + 5 h (test)) × 11 conditions
proposed	0.87 h = 30 min (mean and variance calculations) + 2 min (Chernoff bound calculation) × 11 conditions

machine learning/vision communities, other discriminant analyses have been proposed. Several researchers proposed other objective functions, such as oriented discriminant analysis (ODA) [27] and a heteroscedastic extension of LDA using Chernoff criterion [28]. All of these discriminant analyses transform features discriminatively in a feature space. On the other hand, various criteria for discriminative training of acoustic models have been studied. Maximum mutual information (MMI) and minimum phone error (MPE) criteria have been successfully applied to many speech recognition systems [46, 53, 56].

The feature transformation technique and the discriminative training technique aim to improve speech recognition performance at different levels. The combination of these two techniques can further improve speech recognition performance [92–95]. This section investigates combinations of discriminant analysis-based feature transformation and discriminative training through experiments using in-car speech [96]. We also investigate the robustness against mismatched noise conditions between training and evaluation environments.

3.5.1 Feature Transformation Based on Discriminant Analysis

This section briefly reviews five feature transformation techniques: LDA, HDA, PLDA, ODA and heteroscedastic extension of LDA using Chernoff distance. Let us recall that LDA, HDA, and PLDA objective functions are respectively defined as:

$$J_{LDA}(\mathbf{B}) = \frac{\left|\mathbf{B}^\top \mathbf{C}^{(B)} \mathbf{B}\right|}{\left|\mathbf{B}^\top \mathbf{C}^{(W)} \mathbf{B}\right|},$$

$$J_{HDA}(\mathbf{B}) = \prod_{k=1}^{K} \left(\frac{\left|\mathbf{B}^\top \mathbf{C}^{(B)} \mathbf{B}\right|}{\left|\mathbf{B}^\top \mathbf{C}_k \mathbf{B}\right|} \right)^{N_k},$$

$$J_{PLDA}(\mathbf{B}, m) = \frac{\left|\mathbf{B}^\top \mathbf{C}^{(B)} \mathbf{B}\right|}{\left| \left(\sum_{k=1}^{K} P_k (\mathbf{B}^\top \mathbf{C}_k \mathbf{B})^m \right)^{1/m} \right|},$$

where $\mathbf{C}^{(B)}$, $\mathbf{C}^{(W)}$ and MC_k denoted the between-class, the within-class, and the k-th class covariance matrices, respectively. The within-class covariance satisfied $\mathbf{C}^{(W)} = \sum_{k=1}^{K} P_k \mathbf{C}_k$, where P_k was the class weight, and K was the number of classes.

Oriented Discriminant Analysis (ODA)

ODA has adopted symmetric divergence as a measure of dissimilarity between two distributions [27]. Symmetric divergence between two Gaussian distributions $p_i(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_i \mathbf{C}_i)$ and $p_j(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_j \mathbf{C}_j)$ is given by:

$$\begin{aligned} KL_{ij} &= \int (p_i(\mathbf{x}) - p_j(\mathbf{x})) \log \frac{p_i(\mathbf{x})}{p_j(\mathbf{x})} d\mathbf{x} \\ &= tr\left(\mathbf{C}_i^{-1}\mathbf{C}_j + \mathbf{C}_j^{-1}\mathbf{C}_i - 2\mathbf{I}\right) + (\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)^\top (\mathbf{C}_i^{-1} + \mathbf{C}_j^{-1})(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j). \end{aligned}$$

The ODA objective function is defined as follows:

$$\begin{aligned} J_{ODA}(\mathbf{B}) &= \sum_{i=1}^{K} \sum_{j=1}^{K} KL_{ij} \\ &\propto \sum_{i=1}^{K} tr\left((\mathbf{B}^\top \mathbf{C}_i \mathbf{B})^{-1} \mathbf{B}^\top \mathbf{A}_i \mathbf{B}\right), \end{aligned}$$

where $\mathbf{A}_i = \sum_{j=1, j\neq i}^{K} (\mathbf{M}_{ij} + \mathbf{C}_j)$ and $\mathbf{M}_{ij} = (\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)^\top$. While the original ODA objective function in [27] appended a negative sign, we omitted it for convenience. To optimize the objective function, Torre et al. introduced an efficient computation scheme called bound optimization.

Heteroscedastic Linear Discriminant Analysis Using Chernoff Distance

A limitation of LDA is that it merely tries to separate class means as good as possible. The Chernoff distance considers mean differences as well as covariance differences. Loog et al. [28] proposed a heteroscedastic extension of LDA using the Chernoff criterion (HLDAC):

$$J_{HLDAC}(\mathbf{B}) = \sum_{i=1}^{K-1} \sum_{j=i+1}^{K} P_i P_j tr\left(\left(\mathbf{B}^\top \mathbf{B}\right)^{-1} \left(\mathbf{B}^\top \mathbf{C}_{C_{ij}} \mathbf{B}\right)\right),$$

where $\mathbf{C}_{C_{ij}}$ is the directed distance matrices capturing the Chernoff distance between class i and j, which is defined as:

$$\mathbf{C}_{C_{ij}} \equiv \mathbf{C}_{ij}^{-1/2} (\boldsymbol{\mu}_i - \boldsymbol{\mu}_j) (\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)^\top \mathbf{C}_{ij}^{-1/2} + \frac{1}{\pi_i \pi_j} (\log \mathbf{C}_{ij} - \pi_i \log \mathbf{C}_i - \pi_j \log \mathbf{C}_j).$$

Here, $\pi_i \equiv P_i/(P_i + P_j)$ and $\pi_j \equiv P_j/(P_i + P_j)$ are relative priors, i.e., only taking the two classes into account that define the particular pairwise term. Futhermore, $\mathbf{C}_{ij}$ is the average pairwise within-class scatter matrix, defined as $\pi_i \mathbf{C}_i + \pi_j \mathbf{C}_j$.

3.5.2 Discriminative Training

As described in Section 2.3.2, acoustic model parameters are typically estimated via maximum likelihood. Recall from Equation (2.15) that the objective function of the maximum likelihood estimation is given by:

$$\mathcal{F}_{ML}(\lambda) = \sum_{r=1}^{R} \log p_\lambda(\mathbf{O}_r|s_r),$$

where λ is the set of HMM parameters, $\mathbf{O}_r$ is the r-th training utterance (a word or a sentence), s_r is the r-th correct transcription, R denotes the number of training utterances, and $p_\lambda(\mathbf{O}_r|s)$ is the likelihood given transcription s. Many experimental results have shown that discriminative training techniques yield better performance than traditional maximum likelihood (ML) training. This section briefly reviews two discriminative training techniques introduced in Section 2.3.3: MMI [46, 56] and MPE [53].

Maximum Mutual Information (MMI)

Recall from Equation (2.16) that the MMI criterion is defined as follows [46, 56]:

$$\mathcal{F}_{MMI}(\lambda) = \sum_{r=1}^{R} \log \frac{p_\lambda(\mathbf{O}_r|s_r)^\kappa P(s_r)}{\sum_s p_\lambda(\mathbf{O}_r|s)^\kappa P(s)},$$

where κ is an acoustic de-weighting factor which can be adjusted to improve the test set performance, and $P(s)$ is the language model probability for sentence s. The MMI criterion equals the multiplication of the posterior probabilities of the correct sentences s_r.

Minimum Phone Error (MPE)

MPE training aims to minimize the phone classification error (or maximize the phone accuracy) [53]. Recall from Equation (2.17) that the objective function to be maximized by the MPE training is expressed as

$$\mathcal{F}_{MPE}(\lambda) = \sum_{r=1}^{R} \frac{\sum_s p_\lambda(\mathbf{O}_r|s)^\kappa P(s) \mathrm{A}(s, s_r)}{\sum_s p_\lambda(\mathbf{O}_r|s)^\kappa P(s)},$$

where $\mathrm{A}(s, s_r)$ represents the raw phone transcription accuracy of the sentence s given the correct sentence s_r, which equals the number of correct phones minus the number of errors.

3.5.3 Combination of Feature Transformation and Discriminative Training

Feature transformation aims to transform high dimensional features to low dimensional features in a feature space while separating different classes. Discriminative training estimates the

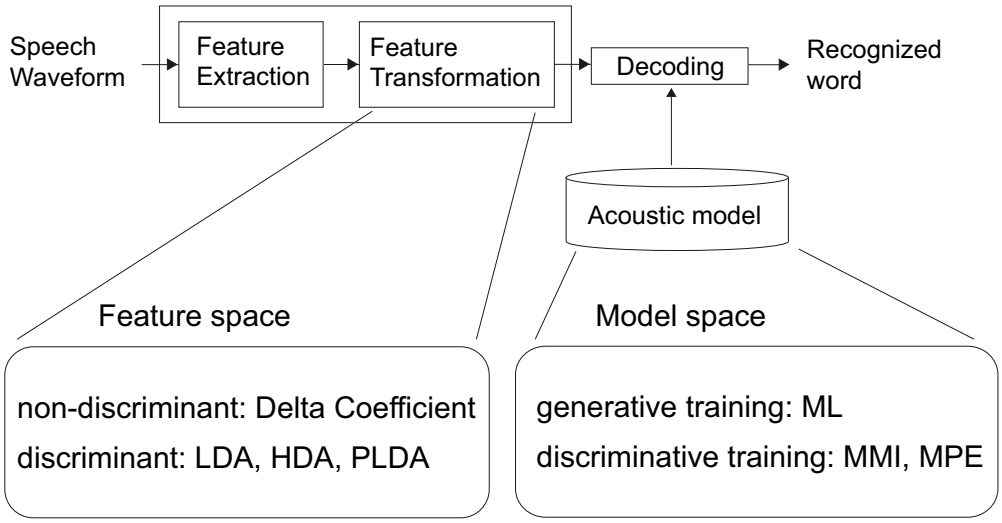


Figure 3.3: Feature transformation and discriminative training.

acoustic models discriminatively in a model space. These relations are illustrated in Figure 3.3. Because these two techniques are adopted at different levels, a combination of them is expected to have a complementary effect on speech recognition.

3.5.4 Experiments

We conducted experiments on the CENSREC-3 database [102]. Detailed descriptions of the database were given in Section 3.3.4. We used all utterances recorded with CT and HF microphones for training. For evaluation, we used drivers speech of isolated words recorded with CT and HF microphones under three different conditions: an in-car environment without A/C noise (*normal*), with low fan-speed noise (*fan-low*), and with high fan-speed noise (*fan-high*). Tables 3.7 and 3.8 show the amount of data for evaluation in each condition (total six conditions) and the average SNR (Signal to Noise Ratio) in each recording condition for evaluation data [102], respectively.

Experimental Setup

As for an evaluation procedure, we followed the CENSREC-3 baseline scripts except that fifty similar-sounding words presented in Table 3.2 were added to the vocabulary. The total vocabulary size became 100. The acoustic models in the speech recognizer consisted of triphone

Table 3.7: Amount of evaluation data.

Microphone	In-car condition	# Utterances
CT	A/C off (*normal*)	2646
CT	A/C on, low (*fan-low*)	2637
CT	A/C on, high (*fan-high*)	2695
HF	A/C off (*normal*)	2646
HF	A/C on, low (*fan-low*)	2637
HF	A/C on, high (*fan-high*)	2695

Table 3.8: Average SNR of evaluation data in each environment (dB) [102].

Condition	Normal		Fan (low)		Fan (high)	
Microphone	CT	HF	CT	HF	CT	HF
Idling	41.19	16.75	32.86	11.01	25.76	5.47
Low speed	38.39	10.96	32.11	8.67	22.64	2.75
High speed	30.11	5.89	28.58	3.59	21.65	1.46

HMMs. Each HMM had five states and three of them had output distributions. Each distribution was represented with 32 mixture diagonal Gaussians. The total number of states with the distributions was 2,000. The feature vector consisted of 12 MFCCs and log-energy with Δ and $\Delta\Delta$ (baseline). The frame length and the frame shift were 20 ms and 10 ms, respectively.

Feature Transformation Procedure

Feature transformation for concatenated features was performed by LDA, HDA, ODA, HLDAC, and PLDA . Eleven successive static frames (143 dimensions) were reduced to 39 dimensions, which are the same number of baseline feature dimensions. Although adding delta (and acceleration) coefficients to feature vectors to be processed may be regarded as finding a desirable projection, delta coefficients essentially have no additional information because they are a linear combination of static feature vectors around current time. Therefore, we did not add delta and acceleration to feature vectors. The number of classes was 43, corresponding to the number of monophones. MLLT [74] was applied after LDA, HDA, ODA and HLDAC. The optimal control parameter ($m = -1.5$) of PLDA was selected experimentally.

Discriminative Training Procedure

Discriminative training requires two lattices: one for the correct transcription of each training file and another derived from the recognition result of each training file. Having created these lattices using an initial set of models, the HMMs are re-estimated by 5 iterations of a parameter estimation procedure using the same set of lattices. Once these lattices were generated for each feature transformation technique, the same lattice was used to train HMMs with MMI and MPE criteria.

Experimental Results

The experimental results are presented in Tables 3.9 to 3.11. The noise condition for the evaluation data used in Table 3.9 matches that for training data. The evaluation data used in Table 3.10 contain low air-conditioner noise. The data used in Table 3.11 contain high air-conditioner noise. These noises are not contained in training data. The best overall performance is shown in bold.

These results showed that both of feature transformations and two discriminative training techniques worked well under a matched noise condition between training and evaluation. In particular, combinations of feature transformations and MPE evidenced outstanding performance. On the other hand, under a mismatched noise condition, the results under a *fan-low* noise condition and a *fan-high* noise condition had a different tendency. Under a *fan-low* condition, both of feature transformations and two discriminative training techniques also worked well. This result comes from the fact that the difference between a *normal* condition and a *fan-low* condition is slight because A/C noise with low fan-speed is small. Under a *fan-high* noise condition, neither feature transformations nor MPE worked well for the data recorded with an HF microphone, as shown in Table 3.11. When noise in training differs considerably from that in evaluation, the degree of confusability of acoustic features among different classes would change. Therefore, no feature transformations estimated under a *normal* noise environment in training worked well under a *fan-high* noise environment in evaluation. In terms of phone classification error among different classes, the data under a *normal* condition and the data under a *fan-high* condition would have different optimal boundaries to minimize phone classification errors. Therefore, MPE had worse recognition performance than the other training criteria.

3.6 Summary

In this chapter we propose a generalization framework for integrating various criteria to reduce dimensionality. The novel framework termed power linear discriminant analysis (PLDA) includes LDA, HLDA and HDA criteria as special cases. The experimental results on the CENSREC-3 database demonstrated that the PLDA with the optimal control parameters reduced word error rate from 11.24% to 8.48%.

Table 3.9: Word error rates (%) on the evaluation set recorded under a *normal* condition.

	CT			HF			Overall		
	ML	MMI	MPE	ML	MMI	MPE	ML	MMI	MPE
baseline	7.4	7.1	6.9	15.0	14.4	15.9	11.2	10.8	11.5
LDA	7.1	6.9	**3.9**	14.2	14.1	13.7	10.7	10.5	8.9
HDA	7.9	7.9	6.9	14.5	14.2	13.6	11.2	11.1	10.3
ODA	8.5	7.8	7.0	13.8	13.4	13.3	11.2	10.6	10.2
HLDAC	9.1	8.3	7.4	12.8	12.2	11.3	11.0	10.3	9.4
PLDA	6.2	6.0	5.0	10.7	10.3	**10.2**	8.5	8.2	**7.7**

Table 3.10: Word error rates (%) on the evaluation set recorded under a *fan-low* condition.

	CT			HF			Overall		
	ML	MMI	MPE	ML	MMI	MPE	ML	MMI	MPE
baseline	9.1	8.8	8.0	25.4	25.0	28.9	17.3	16.9	18.5
LDA	7.3	7.3	**4.4**	26.3	26.1	26.5	16.9	16.7	15.5
HDA	8.4	8.5	7.8	26.6	26.3	28.2	17.5	17.4	18.0
ODA	8.9	8.2	7.7	24.9	23.4	24.9	16.9	15.9	16.3
HLDAC	8.6	8.3	7.0	24.3	23.7	24.8	16.5	16.0	15.9
PLDA	6.4	6.1	4.9	19.7	19.7	**19.6**	13.1	12.9	**12.3**

Table 3.11: Word error rates (%) on the evaluation set recorded under a *fan-high* condition.

	CT			HF			Overall		
	ML	MMI	MPE	ML	MMI	MPE	ML	MMI	MPE
baseline	10.9	10.7	11.2	56.4	**55.9**	59.8	33.7	**33.3**	35.5
LDA	14.1	13.3	11.8	63.7	63.3	65.8	38.9	38.3	38.8
HDA	11.1	10.8	11.0	62.6	62.1	66.3	36.9	36.5	38.7
ODA	12.9	11.8	11.2	65.3	64.3	64.9	39.2	38.1	38.1
HLDAC	12.5	11.5	12.0	65.2	64.6	66.7	38.9	38.1	39.4
PLDA	11.3	11.0	**10.2**	61.4	63.2	62.4	36.4	37.1	36.3

Then, a sub-optimal control parameter selection method was proposed. The proposed method used the Chernoff bound as a measure of a class separability error which was an upper bound of the Bayes error. Experimental results showed that the proposed method could evaluate the relative recognition performance without training of HMMs and test on an evaluation set, and reduced a computational cost from 220 hours to less than one hour.

Finally, this chapter investigated the effectiveness of discriminant analysis-based feature transformation techniques and discriminative training techniques. Under a matched background noise condition between training and evaluation, both techniques achieved better results than the traditional one (MFCC+Δ+$\Delta\Delta$). In addition, a combination of these techniques obtained the best result. However, under a mismatched background noise condition, feature transformations, MPE and their combinations were not necessarily effective.

Chapter 4

Locality Preserving Extensions

This chapter extends HDA and PLDA introduced in the previous chapter to deal with data drawn from a multimodal distribution. This chapter is organized as follows. Conventional feature transformation methods are reviewed again in Section 4.2. Existing locality-preserving dimensionality reduction methods are reviewed in Section 4.3. Proposed methods are introduced in Section 4.4. An approximate calculation to obtain a sub-optimal projection is given in Section 4.5. Experimental results are presented in Section 4.6. Finally, summary is given in Section 4.7.

4.1 Introduction

In the previous chapter, we have reviewed LDA, also know as Fisher discriminant analysis (FDA), HLDA and HDA as acoustic feature transformation methods. This work has pointed out that the objective functions of HLDA and HDA can be viewed as the same formulation except their numerators, and the difference between LDA and HDA is the definitions of the mean of the class covariance matrices. Then, we have proposed a generalization framework including LDA and HDA, called power LDA (PLDA). Unfortunately, these methods may result in an unexpected dimensionality reduction if the data in a certain class consist of several clusters, i.e., multimodal, because they implicitly assume that data are generated from a single Gaussian distribution. In speech recognition, speech signals for acoustic model training tend to be multimodal distributed data because they are generally collected under various conditions, such as gender, age and noise environment. Therefore, each class such as a phone is generally represented as a Gaussian mixture model (GMM) or HMM whose states are represented by GMMs in a speech recognizer. Since dimensionality reduction methods without handling multimodality may give unsatisfactory performance, a dimensionality reduction method for multimodal data is desired to improve speech recognition performance.

Recently, several methods have been proposed to reduce the dimensionality of multimodal

data in the machine learning community [21–24]. It is important to preserve the local structure of data in reducing the dimensionality of multimodal data appropriately. Locality preserving projection (LPP) [22] finds a projection such that the data pairs close to each other in the original space remain close in the projected space. Thus, LPP reduces dimensionality without losing information on local structure. Local Fisher discriminant analysis (LFDA) [23] is also proposed as a supervised method for multimodal data, while LPP is an unsupervised method. To deal with multimodal data, LFDA combines the ideas of FDA and LPP, maximizes between-class separability and preserves within-class local structure. Thus, LFDA is an extension of LDA to reduce the dimensionality of multimodal data.

Since LFDA is based on LDA which assumes homoscedasticity, the effectiveness of LFDA may be limited. To reduce the dimensionality of multimodal data appropriately, we extend HDA which assumes heteroscedasticity. In order to deal with multimodal data using HDA, we combine the ideas of LPP and HDA, and propose locality-preserving HDA. In addition, we also propose locality-preserving PLDA. These extensions can be expected to yield better performance because they reduce the dimensionality of multimodal data appropriately.

Locality-preserving methods such as LFDA and the proposed methods incur considerable computational time to obtain optimal projections when there are many features. In order to slash time, we propose an approximate calculation scheme.

4.2 Linear Dimensionality Reduction Methods

Again, we formulate the problem of linear dimensionality reduction. Given n-dimensional features $\mathbf{x}_j \in \mathbb{R}^n$ where $j = 1, 2, \ldots, N$, e.g., concatenated speech frames, and associated class labels $y_j \in \{1, 2, \ldots, K\}$, e.g., phonemes, let us find a projection matrix $\mathbf{B} \in \mathbb{R}^{n \times p}$ that transforms these features to p-dimensional features $\mathbf{z}_j \in \mathbb{R}^p$, where $p < n$, $\mathbf{z}_j = \mathbf{B}^\top \mathbf{x}_j$, K denotes the number of classes, and N denotes the number of features. $\mathbf{X}^\top$ denotes the transpose of the matrix $\mathbf{X}$. Here, we briefly review existing dimensionality reduction methods. The aim of the techniques are to find a projection matrix $\mathbf{B}$.

4.2.1 Linear Discriminant Analysis

Recall that the LDA objective functions are given by the following:

$$J_{LDA}(\mathbf{B}) = \frac{\left|\mathbf{B}^\top \mathbf{C}^{(B)} \mathbf{B}\right|}{\left|\mathbf{B}^\top \mathbf{C}^{(W)} \mathbf{B}\right|}, \tag{4.1}$$

$$J_{LDA}(\mathbf{B}) = \frac{\left|\mathbf{B}^\top \mathbf{C}^{(M)} \mathbf{B}\right|}{\left|\mathbf{B}^\top \mathbf{C}^{(W)} \mathbf{B}\right|}. \tag{4.2}$$

where the three covariance matrices $\mathbf{C}^{(W)}$, $\mathbf{C}^{(B)}$ and $\mathbf{C}^{(M)}$ denoted the within-class, between-class and mixture covariance ones.

Added to these, the following function is also defined as an objective function of LDA [16]:

$$J_{LDA_3}(\mathbf{B}) = tr\left((\mathbf{B}^\top \mathbf{C}^{(W)} \mathbf{B})^{-1} \mathbf{B}^\top \mathbf{C}^{(B)} \mathbf{B}\right), \tag{4.3}$$

where $tr(\mathbf{X})$ is the trace of the matrix $\mathbf{X}$.

In Equations (4.1) to (4.3), within-class scatter, $\mathbf{S}^{(W)}$, between-class scatter, $\mathbf{S}^{(B)}$, and mixture scatter, $\mathbf{S}^{(M)}$, may be employed in place of $\mathbf{C}^{(W)}$, $\mathbf{C}^{(B)}$ and $\mathbf{C}^{(M)}$, respectively. These scatters are given by $\mathbf{S}^{(W)} = N\mathbf{C}^{(W)}$, $\mathbf{S}^{(B)} = N\mathbf{C}^{(B)}$, and $\mathbf{S}^{(M)} = N\mathbf{C}^{(M)}$. The same solution is obtained even if $\mathbf{C}^{(W)}$, $\mathbf{C}^{(B)}$ and $\mathbf{C}^{(M)}$ in Equations (4.1) to (4.3) are replaced with $\mathbf{S}^{(W)}$, $\mathbf{S}^{(B)}$ and $\mathbf{S}^{(M)}$, respectively.

4.2.2 Heteroscedastic Extensions

As described in Section 3.2.3, HDA and HLDA use the following objective function:

$$J_{HDA}(\mathbf{B}) = \frac{\left|\mathbf{B}^\top \mathbf{C}^{(B)} \mathbf{B}\right|}{\prod_{k=1}^{K} \left|\mathbf{B}^\top \mathbf{C}_k \mathbf{B}\right|^{P_k}}, \tag{4.4}$$

$$J_{HLDA}(\mathbf{B}) = \frac{\left|\mathbf{B}^\top \mathbf{C}^{(M)} \mathbf{B}\right|}{\prod_{k=1}^{K} \left|\mathbf{B}^\top \mathbf{C}_k \mathbf{B}\right|^{P_k}}, \tag{4.5}$$

where $\mathbf{C}_k$ is a class covariance matrix in class k. $\mathbf{C}_k$ and $\mathbf{C}^{(W)}$ satisfy $\mathbf{C}^{(W)} = \sum_{k=1}^{K} P_k \mathbf{C}_k$.

4.2.3 Power Linear Discriminant Analysis

In the previous chapter, we have proposed the following objective function, which integrates LDA and HDA [89, 90], called power LDA (PLDA) [1] :

$$J_{PLDA_1}(\mathbf{B}, m) = \frac{\left|\mathbf{B}^\top \mathbf{C}^{(B)} \mathbf{B}\right|}{\left|\left(\sum_{k=1}^{K} P_k (\mathbf{B}^\top \mathbf{C}_k \mathbf{B})^m\right)^{1/m}\right|}, \tag{4.6}$$

[1] Recall that we let a function f of a symmetric positive definite matrix $\mathbf{A}$ equal $\mathbf{U} diag(f(\lambda_1), \ldots, f(\lambda_n)) \mathbf{U}^T = \mathbf{U}(f(\mathbf{\Lambda}))\mathbf{U}^T$, where $\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$, $\mathbf{U}$ denotes the matrix of n eigenvectors, and $\mathbf{\Lambda}$ denotes the diagonal matrix of eigenvalues, λ_i's. We may define the function f as some power $\mathbf{A}$.

where m denotes a control parameter. Intuitively, as m becomes larger, the classes with larger variances become dominant in the denominator of Equation (4.6). Conversely, as m becomes smaller, the classes with smaller variances become dominant. Thus, by varying the control parameter m, the objective function can represent various objective functions. If m is set to one/zero, the objective function corresponds to the LDA/HDA objective function [90].

The following objective function is given as another definition of PLDA:

$$J_{PLDA_2}(\mathbf{B}, m) = \frac{\left|\mathbf{B}^\top \mathbf{C}^{(M)} \mathbf{B}\right|}{\left|\left(\sum_{k=1}^{K} P_k (\mathbf{B}^\top \mathbf{C}_k \mathbf{B})^m\right)^{1/m}\right|}, \tag{4.7}$$

If m is set to zero, the objective function corresponds to HLDA described in Section 3.2.3.

4.3 Existing Dimensionality Reduction Preserving Locality of Data Structure

Recently, several linear dimensionality reduction methods for multimodal data have been proposed in the machine learning community [21–24]. Here, we review two methods: locality preserving projection (LPP) [22] and local Fisher discriminant analysis (LFDA) [23].

4.3.1 Locality Preserving Projection

Let $\mathbf{A}$ be a symmetric $N \times N$ matrix, which represents an affinity between features [22]. The (i, j)-element A_{ij} of $\mathbf{A}$ is the affinity between $\mathbf{x}_i$ and $\mathbf{x}_j$. An affinity element A_{ij} becomes a large value if $\mathbf{x}_i$ and $\mathbf{x}_j$ are located close to each other. Contrarily, A_{ij} becomes a small value if $\mathbf{x}_i$ and $\mathbf{x}_j$ are located far from each other. There are several different definitions of $\mathbf{A}$, e.g., the nearest neighbor [103], the heat kernel [104] or the local scaling [105]. The objective function of LPP is defined as follows [22]:

$$\begin{aligned} J_{LPP}(\mathbf{B}) &= \frac{1}{2} \sum_{i,j=1}^{N} A_{ij} ||\mathbf{B}^\top \mathbf{x}_i - \mathbf{B}^\top \mathbf{x}_j||^2, \\ \text{s.t.} \quad & \mathbf{B}^\top \mathbf{X} \mathbf{D} \mathbf{X}^\top \mathbf{B} = \mathbf{I}, \end{aligned} \tag{4.8}$$

where $\mathbf{X} = [\mathbf{x}_1 \mathbf{x}_2 \cdots \mathbf{x}_N]$, $\mathbf{I}$ is the identity matrix, and $\mathbf{D}$ is a diagonal matrix whose (i, i)-element is given by $D_{i,i} = \sum_{j=1}^{N} A_{ij}$. Minimizing Equation (4.8) with respect to $\mathbf{B}$, LPP seeks for a projection matrix $\mathbf{B}$ such that nearby data pairs in the original space remain close in the projected space. To ignore a trivial solution, i.e., $\mathbf{B} = \mathbf{0}$, LPP imposes the constraint (4.8). Thus, LPP is an unsupervised dimensionality reduction method preserving locality of features in the original space.

4.3.2 Local Fisher Discriminant Analysis

A supervised dimensionality reduction method preserving locality of features has been proposed by Sugiyama [23, 106] and has been referred to as local Fisher discriminant analysis (LFDA). LFDA combines the ideas of LDA (FDA) and LPP.

Within-class scatter and between-class scatter explained in Section 4.2.1 can be rewritten in a pairwise manner:

$$\mathbf{S}^{(W)} = \frac{1}{2}\sum_{i,j=1}^{N} W_{ij}^{(W)}(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^\top, \tag{4.9}$$

$$\mathbf{S}^{(B)} = \frac{1}{2}\sum_{i,j=1}^{N} W_{ij}^{(B)}(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^\top, \tag{4.10}$$

where

$$W_{ij}^{(W)} = \begin{cases} 1/N_1 & \text{if } y_i = y_j = 1, \\ \vdots & \vdots \\ 1/N_K & \text{if } y_i = y_j = K, \\ 0 & \text{if } y_i \neq y_j, \end{cases} \tag{4.11}$$

$$W_{ij}^{(B)} = \begin{cases} 1/N - 1/N_1 & \text{if } y_i = y_j = 1, \\ \vdots & \vdots \\ 1/N - 1/N_K & \text{if } y_i = y_j = K, \\ 1/N & \text{if } y_i \neq y_j. \end{cases} \tag{4.12}$$

LDA searches for a projection matrix $\mathbf{B}$ such that data pairs in the same class are close to each other and data pairs in different classes are separate from each other. A more formal interpretation of this is given in [23]. Based on an affinity matrix $\mathbf{A}$ and the pairwise expressions of the between/within-class scatter, a *local* within-class scatter and a *local* between-class scatter are defined as follows [23]:

$$\mathbf{S}^{(LW)} = \frac{1}{2}\sum_{i,j=1}^{N} W_{ij}^{(LW)}(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^\top, \tag{4.13}$$

$$\mathbf{S}^{(LB)} = \frac{1}{2}\sum_{i,j=1}^{N} W_{ij}^{(LB)}(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^\top, \tag{4.14}$$

where

$$W_{ij}^{(LW)} = \begin{cases} A_{ij}/N_1 & \text{if } y_i = y_j = 1, \\ \vdots & \vdots \\ A_{ij}/N_K & \text{if } y_i = y_j = K, \\ 0 & \text{if } y_i \neq y_j, \end{cases} \tag{4.15}$$

$$W_{ij}^{(LB)} = \begin{cases} A_{ij}(1/N - 1/N_1) & \text{if } y_i = y_j = 1, \\ \vdots & \vdots \\ A_{ij}(1/N - 1/N_K) & \text{if } y_i = y_j = K, \\ 1/N & \text{if } y_i \neq y_j. \end{cases} \tag{4.16}$$

Both $\mathbf{S}^{(LW)}$ and $\mathbf{S}^{(LB)}$ put a weight on data pairs in the same class, which is proportional to their affinity. The objective function of LFDA corresponding to Equation (4.3) is defined as follows [23, 106]:

$$J_{LFDA_3}(\mathbf{B}) = tr\left(\left(\mathbf{B}^\top \mathbf{S}^{(LW)} \mathbf{B}\right)^{-1} \mathbf{B}^\top \mathbf{S}^{(LB)} \mathbf{B}\right). \tag{4.17}$$

LFDA searches for a projection matrix $\mathbf{B}$ such that nearby data pairs in the same class remain close and the data pairs in different classes are separate from each other; far-apart data pairs in the same class are not forced to be close. Thus, LFDA is a supervised dimensionality reduction method preserving locality. If A_{ij} is taken to be one for all in-class pairs, LFDA corresponds exactly to LDA because $\mathbf{S}^{(LW)}$ and $\mathbf{S}^{(LB)}$ agree with $\mathbf{S}^{(W)}$ and $\mathbf{S}^{(B)}$, respectively. Thus, LFDA is an extension of LDA to deal with multimodal data.

In the same fashion as the definition of LDA objective functions, the following function could be defined as other objective functions of LFDA:

$$J_{LFDA_1}(\mathbf{B}) = \frac{\left|\mathbf{B}^\top \mathbf{S}^{(LB)} \mathbf{B}\right|}{\left|\mathbf{B}^\top \mathbf{S}^{(LW)} \mathbf{B}\right|}, \tag{4.18}$$

$$J_{LFDA_2}(\mathbf{B}) = \frac{\left|\mathbf{B}^\top \mathbf{S}^{(LM)} \mathbf{B}\right|}{\left|\mathbf{B}^\top \mathbf{S}^{(LW)} \mathbf{B}\right|}, \tag{4.19}$$

where a *local* mixture scatter $\mathbf{S}^{(LM)}$ is given by

$$\mathbf{S}^{(LM)} = \frac{1}{2} \sum_{i,j=1}^{N} W_{ij}^{(LM)} (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^\top \tag{4.20}$$

and $W_{ij}^{(LM)}$ is given by

$$W_{ij}^{(LM)} = W_{ij}^{(LW)} + W_{ij}^{(LB)} = \begin{cases} A_{ij}/N & \text{if } y_i = y_j, \\ 1/N & \text{if } y_i \neq y_j. \end{cases} \tag{4.21}$$

The optimizations of Equations (4.17) to (4.19) result in the same projection.

Local within-class covariance, $\mathbf{C}^{(LW)}$, local between-class covariance, $\mathbf{C}^{(LB)}$, and local mixture covariance, $\mathbf{C}^{(LM)}$, can be defined as $\mathbf{C}^{(LW)} = \frac{1}{N}\mathbf{S}^{(LW)}$, $\mathbf{C}^{(LB)} = \frac{1}{N}\mathbf{S}^{(LB)}$ and $\mathbf{C}^{(LM)} = \frac{1}{N}\mathbf{S}^{(LM)}$, respectively. The same solution is obtained when $\mathbf{S}^{(LW)}$, $\mathbf{S}^{(LB)}$ and $\mathbf{S}^{(LM)}$ in Equations (4.17) to (4.19) are replaced with $\mathbf{C}^{(LW)}$, $\mathbf{C}^{(LB)}$ and $\mathbf{C}^{(LM)}$, respectively.

4.4 Extensions of HDA and PLDA to Deal with Multimodality

We first describe limitations facing the existing methods: LDA, HDA, PLDA and LFDA. Next, in order to ease the limitations, we propose two methods that extend HDA and PLDA.

4.4.1 Limitations of Existing Methods

While LDA is widely used to reduce dimensionality because of its simplicity and effectiveness, it assumes that each class shares common class covariance (i.e., homoscedasticity) [18]. Therefore, if this assumption is far from the real data, LDA sometimes does not work well. In order to overcome the limitation, HDA has been proposed, which can deal with unequal class covariances (i.e., heteroscedasticity). These two methods, however, sometimes does not work well because the fixed weight of each class covariance in the two methods cannot be necessarily suitable for any kind of data [90]. So we previously proposed PLDA to generalize LDA and HDA to control the class weights. Unfortunately, all these methods implicitly assume that data are generated from a single Gaussian distribution. Therefore, they cannot deal with multimodal data appropriately. To deal with multimodal data, LFDA has been proposed as explained in Section 4.3.2. It extends the between-class covariance and the within-class covariance to preserve locality of data structure. Nevertheless, since LFDA is based on LDA that assumes homoscedasticity, the effectiveness of LFDA may be limited.

In the following sections, we extend HDA that assumes heteroscedasticity using locality-preserving class covariances that can deal with multimodal data. We also propose locality-preserving PLDA. These extensions can be expected to yield better performance because they do not assume homoscedasticity and can reduce dimensionality of multimodal data appropriately.

4.4.2 Local Heteroscedastic Discriminant Analysis

To deal with multimodality using LDA, LFDA extends the within-class and between-class covariances in the LDA objective function to the *local* within-class and between-class covariances,

respectively. The HDA objective function uses class covariances instead of a within-class covariance. Therefore, we will extend class covariances, similar to the *local* within-class and *local* between-class covariances. We first rearrange a class covariance matrix in a pairwise manner:

$$\mathbf{C}_k = \frac{1}{2N_k} \sum_{i,j=1}^{N} W_{k,ij} (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^\top,$$

where

$$W_{k,ij} = \begin{cases} 1/N_k & \text{if } y_i = y_j = k, \\ 0 & \text{otherwise.} \end{cases} \tag{4.22}$$

$W_{ij}^{(W)}$ and $W_{k,ij}$ satisfy $W_{ij}^{(W)} = \sum_{k=1}^{K} W_{k,ij}$. Similar to LDA, HDA also searches for a projection matrix $\mathbf{B}$ so that data pairs in the same class are close to each other and data pairs in different classes are separate from each other. A more formal interpretation is given in Appendix A.3.

A class covariance matrix can extend to preserve locality of the data structure, similar to the extensions of $\mathbf{S}^{(W)}$ and $\mathbf{S}^{(B)}$. Let us define a *local* class covariance matrix $\mathbf{C}_k^{(L)}$ as follows:

$$\mathbf{C}_k^{(L)} = \frac{1}{2N_k} \sum_{i,j=1}^{N} W_{k,ij}^{(L)} (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^\top, \tag{4.23}$$

where

$$W_{k,ij}^{(L)} = \begin{cases} A_{ij}/N_k & \text{if } y_i = y_j = k, \\ 0 & \text{otherwise.} \end{cases} \tag{4.24}$$

From Equations (4.15) and (4.24), $W_{ij}^{(LW)} = \sum_{k=1}^{K} W_{k,ij}^{(L)}$. In addition, $\mathbf{C}_k^{(L)}$ and $\mathbf{C}^{(LW)}$ satisfy $\mathbf{C}^{(LW)} = \sum_{k=1}^{K} P_k \mathbf{C}_k^{(L)}$. Replacing class and the between-class covariance matrices with local class and the local between-class ones, the objective function of HDA preserving locality is defined as follows:

$$J_{LHDA}(\mathbf{B}) = \frac{\left| \mathbf{B}^\top \mathbf{C}^{(LB)} \mathbf{B} \right|}{\prod_{k=1}^{K} \left| \mathbf{B}^\top \mathbf{C}_k^{(L)} \mathbf{B} \right|^{P_k}}. \tag{4.25}$$

We call it local HDA. If A_{ij} is taken to be one for all in-class pairs, LHDA is proportionate to HDA because $\mathbf{C}_k^{(L)}$ corresponds to $\mathbf{C}_k$. Since the only difference between Equations (4.4) and (4.25) is the definitions of their covariance matrices, the solution to maximize Equation (4.25) with respect to $\mathbf{B}$ is obtained through the same numerical optimization procedure of HDA.

4.4.3 Local Power Linear Discriminant Analysis

As in the case of LHDA, using *local* class covariances $\mathbf{C}_k^{(L)}$, we extend a PLDA objective function as follows:

$$J_{LPLDA_1}(\mathbf{B}, m) = \frac{\left|\mathbf{B}^\top \mathbf{C}^{(LB)} \mathbf{B}\right|}{\left|\left(\sum_{k=1}^{K} P_k (\mathbf{B}^\top \mathbf{C}_k^{(L)} \mathbf{B})^m\right)^{1/m}\right|}. \tag{4.26}$$

We call it local PLDA (LPLDA). From Equations (4.18) and (4.25), LPLDA corresponds exactly to LFDA when m=1 and LPLDA corresponds exactly to LHDA when $m \to 0$. Since the only difference between Equations (4.6) and (4.26) is the definitions of their covariance matrices, the solution to maximize Equation (4.26) with respect to $\mathbf{B}$ is obtained through the same numerical optimization procedure of PLDA [89, 90]. We can also extend the other definition of PLDA as follows:

$$J_{LPLDA_2}(\mathbf{B}, m) = \frac{\left|\mathbf{B}^\top \mathbf{C}^{(LM)} \mathbf{B}\right|}{\left|\left(\sum_{k=1}^{K} P_k (\mathbf{B}^\top \mathbf{C}_k^{(L)} \mathbf{B})^m\right)^{1/m}\right|}.$$

LPLDA corresponds exactly to PLDA when A_{ij} is taken to be one for all in-class pairs.

4.5 Approximate Computations of Local Covariances

To obtain the optimal projections by LFDA, LHDA and LPLDA, $\mathbf{C}_k^{(L)}$, $\mathbf{C}^{(LW)}$, $\mathbf{C}^{(LM)}$ and $\mathbf{C}^{(LB)}$ must be calculated in advance. Throughout the thesis, these covariance matrices are called *local* covariance matrices. Each local covariance matrix requires N^2 times calculations from their definitions. Therefore, their computational complexities are proportional to N^2. Since acoustic models in a speech recognition system are generally trained using a large amount of speech data, the value of N tends to become large, e.g., 10^6 to 10^9. Hence, the computational costs of local covariance matrices tend to be high.

4.5.1 Approximation of Local Class Covariances

For rapid calculation of local covariances, we first consider an approximate computation of local class covariances. In general, each class is represented as GMMs or HMMs in a speech recognizer. Therefore, we assume that the distribution of each class is constructed from several separate clusters. In addition, we approximate a local class covariance by the average of covariances of the clusters. The relation between a local class covariance and covariances of clusters is similar

to that between the within-class covariance and class covariances. Then, we have

$$\mathbf{C}_k^{(L)} \approx \sum_{m=1}^{M_k} P_{k,m}\mathbf{C}_{k,m} \equiv \tilde{\mathbf{C}}_k^{(L)}, \tag{4.27}$$

where M_k is the number of clusters in class k, $P_{k,m}$ is the weight of the m-th cluster in class k, and $\mathbf{C}_{k,m}$ is an m-th cluster covariance in class k. $\tilde{\mathbf{C}}_k^{(L)}$ denotes an approximated local class covariance matrix. $\tilde{\mathbf{C}}_k^{(L)}$ agrees with $\mathbf{C}_k^{(L)}$ when the affinity matrix is defined as follows: $A_{ij} = 1/P_{k,m}$, if $\mathbf{x}_i$ and $\mathbf{x}_j$ are assigned to the same cluster m in a class k, otherwise $A_{ij} = 0$. If the number of clusters equals one, $\tilde{\mathbf{C}}_k^{(L)}$ corresponds to $\mathbf{C}_k$. To obtain $P_{k,m}$ and $\mathbf{C}_{k,m}$, we employ the Expectation-Maximization (EM) algorithm. Since the computational complexities of the E-step and the M-step in the EM algorithm are proportional to the number of data, we can rapidly calculate $\mathbf{C}_k^{(L)}$ by using Equation (4.27).

4.5.2 Approximation of Other Local Covariances

$\mathbf{C}^{(LW)}$, $\mathbf{C}^{(LM)}$ and $\mathbf{C}^{(LB)}$ can be rewritten using $\mathbf{C}_k^{(L)}$ as follows:

$$\mathbf{C}^{(LW)} = \sum_{k=1}^{K} P_k\mathbf{C}_k^{(L)}, \tag{4.28}$$

$$\mathbf{C}^{(LM)} = \mathbf{C}^{(M)} - \sum_{k=1}^{K} P_k^2(\mathbf{C}_k - \mathbf{C}_k^{(L)}), \tag{4.29}$$

$$\mathbf{C}^{(LB)} = \mathbf{C}^{(LM)} - \mathbf{C}^{(LW)}. \tag{4.30}$$

The derivation of Equation (4.29) is given in Appendix A.4. Since the computational cost of $\mathbf{C}_k^{(L)}$ is proportional to N^2, these covariances involve considerable computational costs.

To calculate these covariances rapidly, we replace all $\mathbf{C}_k^{(L)}$ in Equations (4.28)-(4.30) by $\tilde{\mathbf{C}}_k^{(L)}$:

$$\mathbf{C}^{(LW)} \approx \sum_{k=1}^{K} P_k\tilde{\mathbf{C}}_k^{(L)} \equiv \tilde{\mathbf{C}}^{(LW)}, \tag{4.31}$$

$$\mathbf{C}^{(LM)} \approx \mathbf{C}^{(M)} - \sum_{k=1}^{K} P_k^2(\mathbf{C}_k - \tilde{\mathbf{C}}_k^{(L)}) \equiv \tilde{\mathbf{C}}^{(LM)}, \tag{4.32}$$

$$\mathbf{C}^{(LB)} \approx \tilde{\mathbf{C}}^{(LM)} - \tilde{\mathbf{C}}^{(LW)} \equiv \tilde{\mathbf{C}}^{(LB)}. \tag{4.33}$$

$\tilde{\mathbf{C}}^{(LW)}$, $\tilde{\mathbf{C}}^{(LM)}$ and $\tilde{\mathbf{C}}^{(LB)}$ denote approximated $\mathbf{C}^{(LW)}$, $\mathbf{C}^{(LM)}$ and $\mathbf{C}^{(LB)}$, respectively. Since the computational costs of $\mathbf{C}^{(M)}$ and $\mathbf{C}_k$ are proportional to the number of data, there are no N^2 times calculations in Equations (4.31)-(4.33). Once we calculate $\mathbf{C}^{(M)}$ and $\mathbf{C}_k$, and estimate $P_{k,m}$ and $\mathbf{C}_{k,m}$ for $\tilde{\mathbf{C}}_k^{(L)}$ using the EM algorithm, we can calculate $\tilde{\mathbf{C}}^{(LW)}$, $\tilde{\mathbf{C}}^{(LB)}$ and $\tilde{\mathbf{C}}^{(LM)}$ immediately. Thus, the computational costs are significantly reduced.

4.6 Experiments

We conducted experiments on the CENSREC-3 database [102]. Detailed descriptions of the database were given in Section 3.3.4. We only used the speech data collected using a CT microphone for training of HMMs. For evaluation, we used driver's speech of isolated words recorded with a CT microphone under three different conditions: an in-car environment without A/C noise (*normal*), with low fan-speed noise (*fan-low*), and with high fan-speed noise (*fan-high*). Originally, the aim of feature transformation is to reduce redundant information and not to treat mismatched conditions explicitly. However, the transformations should not compromise the system's robustness and so we also investigate robustness under different noise conditions. Although one can use various noise conditions, to make the problem simple, we selected fan noise for the investigation. There are 2,646, 2,637 and 2,695 speech utterances for *normal*, *fan-low* and *fan-high* conditions, respectively.

4.6.1 Experimental setup

For an evaluation procedure, we followed the CENSREC-3 baseline scripts except that fifty similar-sounding words listed in Table 3.2 were appended to the vocabulary to make the recognition task difficult. The acoustic models consisted of triphone HMMs. Each HMM had five states, and three of them had output distributions. Each distribution was represented with 32 mixture diagonal Gaussians. The total number of states with the distributions was 2,000. The baseline performance was calculated with 39 dimensional feature vectors that consist of 12 MFCCs and log-energy with their corresponding delta and acceleration coefficients. Eleven successive frames, whose center was the current frame, were used to obtain dynamic coefficients because delta and acceleration window sizes were three and two, respectively. At the beginning and end of the speech, the first or last vector was replicated five-fold. Frame length was 20 ms and frame shift was 10 ms. In the Mel-filter bank analysis, a cut-off was applied to frequency components lower than 250 Hz. Throughout the experiments, cepstral mean normalization was not applied to the features because there was no difference in the recording conditions between the training data and the evaluation data from the standpoint of convolutional noises such as reverberation.

4.6.2 Feature Transformation Procedure

Feature transformation was performed using LDA, HDA [20], PLDA [89], LFDA [23], LHDA and LPLDA for spliced features. Eleven successive frames (143 dimensions), whose center was the current frame, were reduced to 20, 30 and 39 to investigate the effectiveness of the feature transformation methods. At the beginning and end of the speech, the first or last vector is replicated five-fold. In PLDA and LPLDA, we used the limited-memory BFGS algorithm as a

numerical optimization technique, and their control parameters (m=−0.1) were experimentally selected. The LDA transformation matrix was used as the initial gradient. In LFDA, LHDA and LPLDA, the number of mixtures was four for each class, while the number of mixtures was one for the classes that have training data of less than one percent of the total. In addition, to obtain projection matrices by LFDA, LHDA and LPLDA, we employed an approximate computation scheme for calculating covariances. To assign one of the classes to every feature vector, HMM state labels were generated for the training data by a state-level forced alignment algorithm using a well-trained HMM system. Although a total of 43 monophone labels were used in the CENSREC-3, the number of classes was grouped into 40 to reduce phonetic confusion.

4.6.3 Results

Experimental results are presented in Tables 4.1 to 4.3. The noise condition for the evaluation data used in Table 4.1 matches that for training data. The evaluation data used in Table 4.2 and the data used in Table 4.3 contain low air-conditioner noise and high air-conditioner noise, respectively. These noises are not contained in training data.

We first discuss the results of the feature transformation methods when the size of a reduced space is 39 (i.e., $p = 39$). The size is equal to that of baseline. Table 4.1 showed that the locality-preserving dimensionality reduction methods consistently yielded better performance than the traditional methods. This result suggests that projected features using the locality-preserving methods have higher separability among acoustic classes than those using the traditional methods because the locality-preserving methods can consider multimodality of data. Especially, LPLDA yielded the lowest word error rate (WER) among all dimensionality reduction methods. Table 4.2 showed a similar tendency to Table 4.1. The locality-preserving dimensionality reduction methods also yielded better performance. These results were obtained from the fact that the difference between a *normal* condition and a *fan-low* condition is slight because A/C noise with a low fan-speed is small. In addition, the combinations of heteroscedasticity and locality-preservation worked well. On the other hand, Table 4.3 showed a different tendency from the others. The feature transformation methods excluding LPLDA gave worse performance than at baseline (MFCC+Δ+$\Delta\Delta$). In general, the degree of confusability of acoustic features among different classes would change when the noise in training differs considerably from that in evaluation. Therefore, a feature transformation estimated under a *normal* noise environment in training did not necessarily work well under a *fan-high* noise environment in evaluation. Nevertheless, LPLDA kept comparable performance with the baseline whether or not the noise condition in evaluation matches when training because it would transform features that have sufficiently high separability among different classes even in a mismatch noise condition.

Table 4.1: Word error rates (%) under a *normal* condition.

Method	Size of reduced space (p)		
	39	30	20
baseline	6.50	-	-
LDA	6.50	6.00	6.87
HDA	7.33	5.85	5.14
PLDA	5.40	6.08	6.84
LFDA	6.00	5.93	5.44
LHDA	6.46	5.32	5.29
LPLDA	4.83	5.89	5.17

Table 4.2: Word error rates (%) under a *fan-low* condition.

Method	Size of reduced space (p)		
	39	30	20
baseline	8.00	-	-
LDA	8.22	7.24	8.49
HDA	7.73	6.40	6.52
PLDA	6.29	6.75	7.58
LFDA	6.97	7.05	5.95
LHDA	6.94	6.29	6.90
LPLDA	5.46	6.14	6.90

Table 4.3: Word error rates (%) under a *fan-high* condition.

Method	Size of reduced space (p)		
	39	30	20
baseline	10.72	-	-
LDA	12.05	12.39	16.99
HDA	13.21	14.62	15.91
PLDA	11.42	14.21	16.10
LFDA	11.50	12.02	12.80
LHDA	10.98	13.02	14.91
LPLDA	10.64	11.42	15.17

Next, we discuss the results of the feature transformation methods when $p = 20$ and $p = 30$[2]. As shown in Tables 4.1 and 4.2, under matched and almost matched noise conditions between training and evaluation, the optimal dimensions of most feature transformation methods are lower than 39. On the other hand, Table 4.3 showed that all methods degraded recognition performance under a mismatched noise condition when the dimensions were relatively small. These results imply that feature transformation methods might obtain lower dimensions in matched conditions, whereas in mismatched conditions, redundant information can contribute to the improvement of recognition performance. Tables 4.1 to 4.3 also showed that while the proposed methods did not necessarily yield comparable performance of the other methods when $p = 20$, they consistently yielded the lowest word error rate when $p \geq 30$.

4.7 Summary

In this chapter, two dimensionality reduction methods were proposed; HDA preserving the local structure of the data (LHDA) and PLDA preserving the local structure (LPLDA), to reduce dimensionality of multimodal data appropriately. The best performance, 4.83%, was obtained by LPLDA, while the word error rate of the baseline system was 6.50%. Hence, LPLDA provided a relative word error rate reduction of 25% compared to the baseline system. Moreover, the locality-preserving dimensionality reduction methods yielded better performance than traditional ones, especially under matched noise conditions. In particular, LPLDA outperformed the others whether or not the noise condition in evaluation matched that in training. Finally, to obtain the optimal projections by the locality-preserving methods rapidly, we proposed an approximate calculation scheme.

[2]In some preliminary experiments, the degradation of recognition performance was found when $p > 39$ and $p < 20$ with a few exceptions. While the best performances of PLDA and LPLDA were found at $p = 50$ under a *fan-high* condition, the differences of the performances between $p = 50$ and $p = 39$ were not so large. Although the best results using different methods under the different environments are obtained with a few variety of p as explained here, these facts does not affect the overall conclusion of this section.

Chapter 5

Minimization of Classification Error

This chapter focuses attention on acoustic feature transformations which minimize a kind of classification error between different classes. This chapter starts with an introduction and a review of conventional feature transformations. Then, a problem of the conventional feature transformations is pointed out in Section 5.3. Minimization criteria of the maximum classification error among different phonetic classes are given in Section 5.4. Experimental results are presented in Section 5.5. Finally, summary is given in Section 5.6.

5.1 Introduction

This chapter focuses attention on acoustic feature transformation methods that minimize misclassification in the sense of the Bayes classification error [25, 26, 107] between different phonetic classes. As the performance of speech recognition systems generally correlates strongly with the classification accuracy of phonetic features, the features should have the power to discriminate between different classes. We show that the purpose of the existing methods can be regarded as minimization of the average classification error (AveCE) between different classes. While minimizing the AveCE suppresses total classification error, it cannot prevent the occurrence of considerable overlaps between distributions of some different classes. Therefore, there may be class pairs that have little or no discriminative information on each other. Hence, the AveCE does not necessarily find a suitable transformation for speech recognition. To avoid this, an alternative dimensionality reduction method is proposed, which minimizes the maximum classification error (MaxCE) among all class pairs. The proposed method can avoid considerable error between different classes. Moreover, interpolated methods between AveCE and MaxCE are proposed.

5.2 Minimization of Approximated Bayes Error

In this section the Bayes error [26, 107] is briefly reviewed. Then, other criteria for estimating the classification error are presented.

.

5.2.1 Bayes Error

Let us consider the discrimination problem of classifying an observation as coming from one of K possible classes $k \in \{1, 2, \cdots, K\}$. Let $\mathbf{x}$ be an n-dimensional feature vector such as a concatenated speech frame. The error probability P_e of the optimal Bayes rule for the classification into K classes becomes [16, 108]

$$P_e = 1 - \int \max_k \left[\lambda_k p_k(\mathbf{x})\right] d\mathbf{x},$$

where λ_k and p_k denote a prior probability and a probability density function (pdf) for class k, respectively. We assume that the λ_k and p_k for $k = 1, \cdots, K$ are entirely known.

The number of the dimension of a feature vector $\mathbf{x}$ can be reduced to $p < n$ by a transformation $\mathbf{z} = \mathbf{B}^\top \mathbf{x}$ with a transformation matrix $\mathbf{B} \in \mathbb{R}^{n \times p}$ of rank p as described in Section 2.5.1. Then, the error probability in the range space of $\mathbf{B}^\top$, $P_e^{\mathbf{B}}$, becomes:

$$P_e^{\mathbf{B}} = 1 - \int \max_k \left[\lambda_k p_k^{\mathbf{B}}(\mathbf{z})\right] d\mathbf{z},$$

where $p_k^{\mathbf{B}}$ denotes the pdf for class k in the projected space spanned by the column vectors of $\mathbf{B}$. Since the transformation $\mathbf{z} = \mathbf{B}^\top \mathbf{x}$ produces a linear combination of the components of the feature vector $\mathbf{x}$, discriminative information is generally lost and $P_e^{\mathbf{B}} \geq P_e$ [25].

The feature transformation problem could be stated as a selection of an n by p matrix $\hat{\mathbf{B}}$ from all n by p matrices of rank p such that

$$\hat{\mathbf{B}} = \underset{\mathbf{B} \in \mathbb{R}^{n \times p}, rank(\mathbf{B}) = p}{\operatorname{argmin}} P_e^{\mathbf{B}}. \tag{5.1}$$

Unfortunately, it is generally difficult to calculate $P_e^{\mathbf{B}}$ directly.

5.2.2 Other Criteria for Estimating Error Probability

Instead of minimizing the Bayes error P_e directly, the following affinity between two pdfs are often used:

$$\rho_{i,j} = \int \sqrt{p_i(\mathbf{x}) p_j(\mathbf{x})} d\mathbf{x}. \tag{5.2}$$

The term $\rho_{i,j}$ is called the Bhattacharyya coefficient and is an upper bound on the Bayes error [107]. This coefficient can be regarded as a classification error between two pdfs. Clearly, $\rho_{i,j}$ lies between zero and one.

The Bhattacharyya coefficient in the range space of $\mathbf{B}^\top$ becomes:

$$\rho_{i,j}^{\mathbf{B}} \equiv \int \sqrt{p_i^{\mathbf{B}}(\mathbf{z}) p_j^{\mathbf{B}}(\mathbf{z})} d\mathbf{z}. \tag{5.3}$$

If we assume that the p_k is a Gaussian distribution with a mean vector $\boldsymbol{\mu}_k$ and a covariance matrix $\mathbf{C}_k$, Equation (5.3) has the closed form expression:

$$\rho_{i,j}^{\mathbf{B}} = \exp(-\eta_{i,j}^{\mathbf{B}}). \tag{5.4}$$

where we let

$$\eta_{i,j}^{\mathbf{B}} \equiv \frac{1}{8} tr\left(\left(\mathbf{B}^\top \mathbf{C}_{ij} \mathbf{B}\right)^{-1} \mathbf{B}^\top \mathbf{M}_{ij} \mathbf{B}\right) + \frac{1}{2} \log \frac{\left|\mathbf{B}^\top \mathbf{C}_{ij} \mathbf{B}\right|}{\sqrt{\left|\mathbf{B}^\top \mathbf{C}_i \mathbf{B}\right| \left|\mathbf{B}^\top \mathbf{C}_j \mathbf{B}\right|}}, \tag{5.5}$$

$\mathbf{C}_{ij} \equiv \frac{\mathbf{C}_i + \mathbf{C}_j}{2}$, and $\mathbf{M}_{ij} \equiv (\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)^\top$. The term $\eta_{ij}^{\mathbf{B}}$ is called the Bhattacharyya distance [16]. The goal of acoustic feature transformation by meas of the Bhattacharyya coefficient is to find the optimal transformation matrix $\hat{\mathbf{B}}$ so that

$$\hat{\mathbf{B}} = \underset{\mathbf{B} \in \mathbb{R}^{n \times p}, rank(\mathbf{B}) = p}{\operatorname{argmin}} \rho_{i,j}^{\mathbf{B}}. \tag{5.6}$$

To obtain the optimal matrix $\hat{\mathbf{B}}$, there is no closed-form solution. Instead, we can obtain it numerically. Taking the derivative of logarithm of Equation (5.4) with respect to $\mathbf{B}$ yields,

$$\begin{aligned} \frac{d}{d\mathbf{B}} \log \rho_{ij}^{\mathbf{B}} =& \frac{1}{4} \mathbf{C}_{ij} \mathbf{B} \tilde{\mathbf{C}}_{ij}^{-1} \tilde{\mathbf{M}}_{ij} \tilde{\mathbf{C}}_{ij}^{-1} - \frac{1}{4} \mathbf{M}_{ij} \mathbf{B} \tilde{\mathbf{C}}_{ij}^{-1} \\ & - \mathbf{C}_{ij} \mathbf{B} \tilde{\mathbf{C}}_{ij}^{-1} + \frac{1}{2} \mathbf{C}_i \mathbf{B} \tilde{\mathbf{C}}_i^{-1} + \frac{1}{2} \mathbf{C}_j \mathbf{B} \tilde{\mathbf{C}}_j^{-1}. \end{aligned} \tag{5.7}$$

Hence, we can employ some fast convergence numerical optimization methods such as the quasi-Newton and conjugate gradient [101].

Several extensions of Eq. (5.2) to handle multi-class problems have been proposed. Two techniques are briefly reviewed below.

Upper Bound on Bayes Error

The Bayes error for multi-class data is redefined as follows:

$$P_e = \int \min_{1 \le i \le K} \sum_{j \neq i} \lambda_j p_j(\mathbf{x}) d\mathbf{x}, \tag{5.8}$$

where K denotes the number of classes. To represent the Bayes error using the Bhattacharyya distance, Saon et al. introduced a permutation function [26], $\sigma_{\mathbf{x}} : \{1, \ldots, K\} \rightarrow \{1, \ldots, K\}$, such that the terms $\lambda_1 p_1(\mathbf{x}), \ldots, \lambda_K p_K(\mathbf{x})$ are sorted in ascending order, i.e., $\lambda_{\sigma_{\mathbf{x}}(1)} p_{\sigma_{\mathbf{x}}(1)}(\mathbf{x}) \leq \ldots \leq \lambda_{\sigma_{\mathbf{x}}(K)} p_{\sigma_{\mathbf{x}}(K)}(\mathbf{x})$.

For $1 \leq k \leq K-1$, the following inequality holds:

$$\lambda_{\sigma_{\mathbf{k}}(K)} p_{\sigma_{\mathbf{x}}(k)}(\mathbf{x}) \leq \sqrt{\lambda_{\sigma_{\mathbf{x}}(k)} p_{\sigma_{\mathbf{x}}(k)}(\mathbf{x}) \lambda_{\sigma_{\mathbf{x}}(k+1)} p_{\sigma_{\mathbf{x}}(k+1)}(\mathbf{x})}$$

from which it follows that

$$\begin{aligned} \min_{1 \leq i \leq K} \sum_{j \neq i} \lambda_j p_j(\mathbf{x}) &= \sum_{k=1}^{K-1} \lambda_{\sigma_{\mathbf{x}}(k)} p_{\sigma_{\mathbf{x}}(k)}(\mathbf{x}) \\ &\leq \sum_{k=1}^{K-1} \sqrt{\lambda_{\sigma_{\mathbf{x}}(k)} p_{\sigma_{\mathbf{x}}(k)}(\mathbf{x}) \lambda_{\sigma_{\mathbf{x}}(k+1)} p_{\sigma_{\mathbf{x}}(k+1)}(\mathbf{x})} \\ &\leq \sum_{1 \leq i < j \leq K} \sqrt{\lambda_i p_i(\mathbf{x}) \lambda_j p_j(\mathbf{x})}. \end{aligned} \tag{5.9}$$

Inserting this in (5.8), an upper bound of the Bayes error for multi-class data is expressed as follows:

$$P_e \leq \sum_{1 \leq i < j \leq K} \sqrt{\lambda_i \lambda_j} \int \sqrt{p_i(\mathbf{x}) p_j(\mathbf{x})} d\mathbf{x}. \tag{5.10}$$

That is, the Bayes error is bounded from above by the following expression [26, 109]:

$$\sum_{i,j>i} \sqrt{\lambda_i \lambda_j} \rho_{i,j}. \tag{5.11}$$

Saon et al. [26] proposed the following objective function based on Eq. (5.11):

$$J_{bound}(\mathbf{B}) = \sum_{i,j>i} \sqrt{\lambda_i \lambda_j} \rho_{i,j}^{\mathbf{B}}. \tag{5.12}$$

Average Bhattacharyya Coefficient

Another natural extension to treat multi-class problems is the average Bhattacharyya coefficient as follows [107]:

$$\sum_{i,j} \lambda_i \lambda_j \rho_{i,j} \tag{5.13}$$

Based on the average Bhattacharyya coefficient, we can define the following objective function to reduce dimensionality:

$$J_{ave}(\mathbf{B}) = \sum_{i,j} \lambda_i \lambda_j \rho_{i,j}^{\mathbf{B}}. \tag{5.14}$$

5.3 Issue about Existing Methods

From $\rho^{\mathbf{B}}_{i,i} = 1$, $\rho^{\mathbf{B}}_{i,j} = \rho^{\mathbf{B}}_{j,i}$, and $\sum_i \lambda_i = 1$, we have

$$\sum_{i,j>i} \sqrt{\lambda_i \lambda_j} \rho^{\mathbf{B}}_{i,j} = \frac{1}{2} \left(\sum_{i,j} \sqrt{\lambda_i \lambda_j} \rho^{\mathbf{B}}_{i,j} - 1 \right). \tag{5.15}$$

Using this, Eq. (5.12) can be rewritten as follows:

$$\begin{aligned} J_{bound}(\mathbf{B}) &\propto \sum_{i,j} \sqrt{\lambda_i \lambda_j} \rho^{\mathbf{B}}_{i,j} \\ &\propto \sum_{i,j} \frac{\sqrt{\lambda_i}}{Z} \frac{\sqrt{\lambda_j}}{Z} \rho^{\mathbf{B}}_{i,j} \\ &= \sum_{i,j} \lambda'_i \lambda'_j \rho^{\mathbf{B}}_{i,j}, \end{aligned} \tag{5.16}$$

where $Z \equiv \sum_k \sqrt{\lambda_k}$ is a normalizing constant, and $\lambda'_k \equiv \sqrt{\lambda_k}/Z$. Eqs. (5.14) and (5.16) are essentially the same objective function, and the only difference between them is their priors. Hence, both functions can be regarded as the average of Bhattacharyya coefficient $\rho^{\mathbf{B}}_{i,j}$. That is, both objective functions search for a projection matrix $\mathbf{B}$ so that the average classification error (AveCE) is minimized. Although minimizing the AveCE suppresses total classification error between different classes, it cannot prevent the occurrence of considerable overlaps between distributions of some classes, which is critical for speech recognition because there may be class pairs that have little or no discriminative information on each other.

Figure 5.1 shows that two-dimensional three-class samples are projected onto a one-dimensional subspace. Each class sample is synthetic data drawn from different Gaussians. The priors of classes 1 to 3 were 0.75, 0.125 and 0.125, respectively. The projection by J_{ave} gave high separabilities between classes 1 and 2, and between classes 1 and 3. On the other hand, there was a considerable overlap between classes 2 and 3. Here, let us regard the situation in Figure 5.1 as a phone classification task. Suppose that classes 1 to 3 represent some phones (ex. /sil/, /a/, /o/, etc.). When we transform features by J_{ave}, classification becomes difficult between two phones associated with classes 2 and 3.

5.4 Minimization of Maximum Bhattacharyya Coefficient

To overcome the drawback of the AveCE described in the previous section, we propose an alternative objective function that minimizes the maximum classification error (MaxCE) among all class pairs. The proposed objective function can avoid considerable error between different classes. Moreover, we propose interpolated objective functions between two criteria.

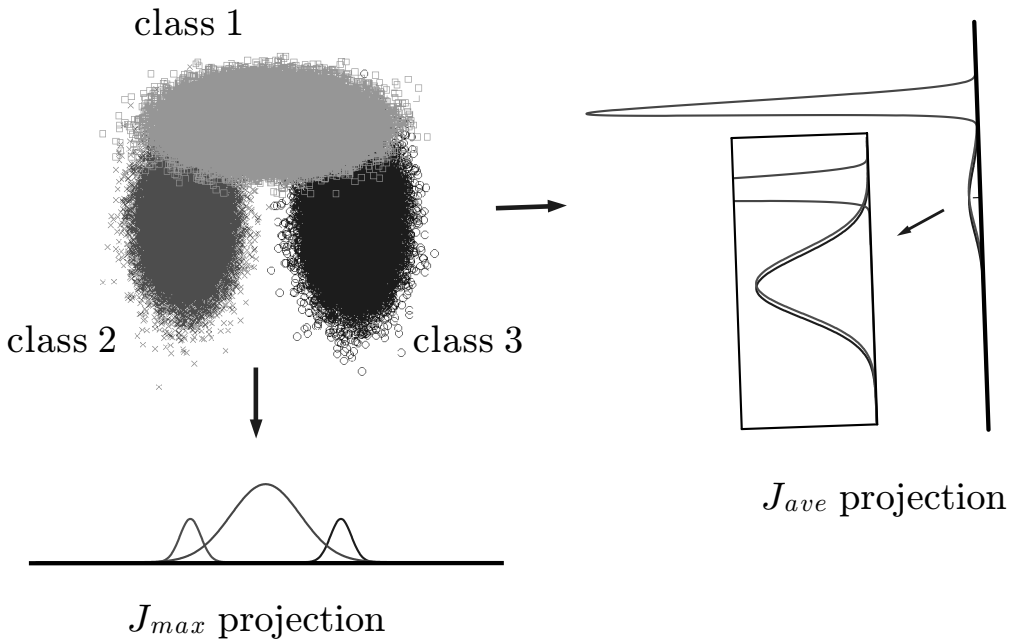


Figure 5.1: Example of a synthetic data set comprising three classes. Two lines are the one-dimensional subspaces. The vertical line and the horizontal line are obtained using Eqs. (5.14) and (5.17), respectively.

5.4.1 Approximated Maximum Classification Error

To prevent less discrimination power of some class pairs, we define the alternative objective function that minimizes the maximum overlap among classes regardless of their priors, instead of AveCE, as follows:

$$J_{max}(\mathbf{B}) \equiv \max_{i,j} \rho_{i,j}^{\mathbf{B}}. \tag{5.17}$$

Unfortunately, minimization of Eq. (5.17) with respect to $\mathbf{B}$ is not feasible. Instead, we approximate Eq. (5.17). Let $\mathbf{y}$ be an $n \times 1$ vector with positive components $\{y_i\}_{i=1}^n$, and let $\boldsymbol{\alpha}$ be an $n \times 1$ vector of positive weights $\{\alpha_i\}_{i=1}^n$, so that $0 < \alpha_i < 1$ and $\sum_{i=1}^n \alpha_i = 1$. To approximate Eq. (5.17), we focus on the generalized mean, also known as the weighted mean of order m. The generalized mean is given by [97]:

$$M(\mathbf{y}, \boldsymbol{\alpha}, m) = \left(\sum_{i=1}^{n} \alpha_i y_i^m \right)^{1/m}, \tag{5.18}$$

for any real m. Eq. (5.18) can describe several means by changing m. For example, Eq. (5.18) with $m = 1$ corresponds to the arithmetic mean of $\{y_i\}_{i=1}^n$, and Eq. (5.18) with $m \to 0$ converges to the geometric mean of $\{y_i\}_{i=1}^n$. We especially focus on the following special case of the generalized mean:

$$\lim_{m\to\infty} M(\mathbf{y}, \boldsymbol{\alpha}, m) = \max_i y_i. \tag{5.19}$$

We approximate Eq. (5.17) using the generalized mean and sufficiently large value $\hat{m}$ as follows:

$$J_{max}(\mathbf{B}) = \lim_{m\to\infty} \left(\sum_{i,j} \lambda_i \lambda_j \left(\rho_{i,j}^{\mathbf{B}}\right)^m \right)^{1/m} \tag{5.20}$$

$$\approx \left(\sum_{i,j} \lambda_i \lambda_j \left(\rho_{i,j}^{\mathbf{B}}\right)^{\hat{m}} \right)^{1/\hat{m}}. \tag{5.21}$$

Taking the derivative of logarithm of Equation (5.21) with respect to $\mathbf{B}$ yields,

$$\frac{\partial}{\partial \mathbf{B}} J_{max}(\mathbf{B}) = \sum_{i,j} \frac{\lambda_i \lambda_j \left(\rho_{ij}^{\mathbf{B}}\right)^{\hat{m}}}{\sum_{k,l} \lambda_k \lambda_l \left(\rho_{kl}^{\mathbf{B}}\right)^{\hat{m}}} \frac{\partial}{\partial \mathbf{B}} \log \rho_{ij}^{\mathbf{B}}. \tag{5.22}$$

Eq. (5.21) with $\hat{m} = 100$ was applied in Fig. 5.1. The figure showed that the projection by J_{max} gave higher separability between class 2 and class 3 than that by J_{ave}. That is, J_{max} can offer greatly improved classification power between class 2 and class 3.

5.4.2 Interpolation between Two Criteria

In Fig. 5.1, the projection by J_{max} gave a more desirable result than by J_{ave}. However, similar to J_{ave}, J_{max} also does not necessarily find a suitable projection. If a number of class pairs have an overlap comparable to the maximum one, the total error increases significantly. In such a situation, speech recognition performance will deteriorate because most class pairs have only small discrimination power. Therefore, an interpolated criterion that minimizes MaxCE while minimizing AveCE would be effective. Here, we propose two interpolated functions between MaxCE and AveCE.

$$J_{interp1}(\mathbf{B}, \alpha) = (1-\alpha) J_{ave}(\mathbf{B}) + \alpha J_{max}(\mathbf{B}),$$

$$J_{interp2}(\mathbf{B}, m) = \left(\sum_{i,j} \lambda_i \lambda_j \left(\rho_{i,j}^{\mathbf{B}}\right)^m \right)^{1/m},$$

where α and m denote control parameters so that $\alpha \in [0, 1]$ and $m \geq 1$, respectively. $J_{interp1}$ corresponds to J_{ave} when $\alpha = 0$ and to J_{max} when $\alpha = 1$. From Eq. (5.14), $J_{interp2}$ corresponds to J_{ave} when $m = 1$. Similarly, from Eq. (5.20), $J_{interp2}$ converges to J_{max} when $m \to \infty$. As α becomes larger, only one class pair with the maximum overlap between class distributions becomes dominant in $J_{interp1}$. On the other hand, as m becomes larger, several class pairs with large overlaps become dominant in $J_{interp2}$.

5.5 Experiments

We conducted experiments on the CENSREC-3 database [102]. Detailed descriptions of the database were given in Section 3.3.4. For training of HMMs, we used drivers speech of phonetically-balanced sentences recorded under two conditions: while idling and driving on city streets under a normal in-car environment. There were 14,050 utterances by 293 drivers (202 males and 91 females), which were collected using a CT microphone. For evaluation, we used driver's speech of isolated words recorded with a CT microphone under three different conditions: an in-car environment without A/C noise (*normal*), with low fan-speed noise (*fan-low*), and with high fan-speed noise (*fan-high*). There are 2,646, 2,637 and 2,695 speech utterances for *normal*, *fan-low* and *fan-high* conditions, respectively.

We followed the CENSREC-3 baseline scripts as the evaluation procedure except that fifty similar-sounding words listed in Table 3.2 were appended to the vocabulary. Hence, the total vocabulary size became 100. The acoustic models consisted of triphone HMMs. Each HMM had five states three of which had output distributions. Each distribution was represented with a 32 mixture of diagonal Gaussians. The total number of states with the distributions was 2,000. The baseline performance was evaluated with 39 dimensional feature vectors that consist of 12 MFCCs and log-energy, and their delta and delta-delta coefficients. A delta coefficient was calculated from seven successive frames of MFCC, and a delta-delta from five successive frames of delta. Consequently, a feature vector was calculated using eleven successive MFCC vectors. The frame length and the frame shift were 20 ms and 10 ms, respectively.

5.5.1 Feature Transformation Procedure

Eleven successive frames were concatenated into one feature vector (143 dimensions), which is the same number of frames used for calculating delta and delta-delta coefficients. Feature transformation was performed by LDA, J_{ave}, J_{max}, $J_{interp1}$ and $J_{interp2}$ for the concatenated features. The concatenated vectors were reduced to 39, which are the same number of dimensions of the baseline feature vectors, and then MLLT was applied. The number of classes was 40.

5.5.2 Experimental Results

The experimental results for *normal*, *fan-low* and *fan-high* conditions are respectively summarized in Tables 5.1, 5.4 and 5.7, where optimal control parameters of $J_{interp1}$ and $J_{interp2}$ were experimentally selected. The results showed that J_{max} consistently yielded better performance than J_{ave}. The performances of both interpolated methods $J_{interp1}$ and $J_{interp2}$ were superior or comparable to those of J_{max}. These results suggests that $J_{interp1}$ and $J_{interp2}$ could play a complementary role between J_{ave} and J_{max}.

Tables 5.2, 5.5 and 5.8 showed WERs under *normal*, *fan-low* and *fan-high* conditions for different control parameters of $J_{interp1}$, respectively. In a similar way, Tables 5.3, 5.6 and 5.9 showed WERs for different control parameters of $J_{interp2}$, respectively. The control parameters, α for $J_{interp1}$ and m for $J_{interp2}$, varied between 0 and 1, and 1 and 100, respectively. The results showed that $J_{interp2}$ gave better performance than that of $J_{interp1}$. This is because that $J_{interp2}$ can reduce classification error of several class pairs with large overlaps, as m is a large value, while $J_{interp1}$ reduces that of only one class pair with the maximum overlap between class distributions.

5.6 Summary

To improve speech recognition performance, this chapter focuses attention on acoustic feature transformations of speech features, which minimize a classification error between different phonetic classes. The recognition performance of speech recognition systems generally correlates strongly with the classification accuracy of different phonetic features. Therefore, speech recognition performance would improve when classification error becomes small.

We first showed that the purpose of the conventional methods for this approach could be regarded as minimization of the average classification error between different classes. Although minimizing the average classification error can suppress total classification error among classes, it cannot prevent the occurrence of considerable overlaps between distributions of some classes. Then, instead of the average classification error, minimization methods of maximum classification error are proposed herewith so as to avoid considerable error between different classes. The proposed method achieved word error rate of 5.36%. The relative improvement was 17.5% compared to the baseline system. In addition, interpolation methods that minimize the maximization error while minimizing the average classification error are also proposed. The best result of the proposed methods was 3.32%, which was a relative word error rate reduction of 48.9% compared to the baseline system.

Table 5.1: Word error rates (%) under a *normal* condition

	WER
MFCC + Δ + $\Delta\Delta$	6.50
LDA	6.12
J_{ave}	5.85
J_{max}	5.36
$J_{interp1}$ ($\alpha = 0.6$)	4.72
$J_{interp2}$ ($m = 16$)	3.32

Table 5.2: Word error rates (%) for $J_{interp1}$ under a *normal* condition versus value of control parameter α

α	0	0.2	0.4	0.6	0.8	1.0
$J_{interp1}$	5.85	5.78	5.74	4.72	5.10	5.36

Table 5.3: Word error rates (%) for $J_{interp2}$ under a *normal* condition versus value of control parameter m

m	1	2.5	6	16	30	100
$J_{interp2}$	5.85	4.57	4.00	3.32	4.19	5.36

Table 5.4: Word error rates (%) under a *fan-low* condition

	WER
MFCC + Δ + $\Delta\Delta$	8.00
LDA	7.01
J_{ave}	6.82
J_{max}	4.43
$J_{interp1}$ ($\alpha = 1.0$)	4.43
$J_{interp2}$ ($m = 16$)	3.90

Table 5.5: Word error rates (%) for $J_{interp1}$ under a *fan-low* condition versus value of control parameter α

α	0	0.2	0.4	0.6	0.8	1.0
$J_{interp1}$	6.82	6.40	6.75	5.34	6.56	4.43

Table 5.6: Word error rates (%) for $J_{interp2}$ under a *fan-low* condition versus value of control parameter m

m	1	2.5	6	16	30	100
$J_{interp2}$	6.83	6.67	4.77	3.90	5.49	4.43

Table 5.7: Word error rates (%) under a *fan-high* condition

	WER
MFCC + Δ + $\Delta\Delta$	10.72
LDA	10.64
J_{ave}	9.64
J_{max}	7.53
$J_{interp1}$ ($\alpha = 1.0$)	7.53
$J_{interp2}$ ($m = 16$)	6.53

Table 5.8: Word error rates (%) for $J_{interp1}$ under a *fan-high* condition versus value of control parameter α

α	0	0.2	0.4	0.6	0.8	1.0
$J_{interp1}$	9.64	9.20	9.05	8.60	9.94	7.53

Table 5.9: Word error rates (%) for $J_{interp2}$ under a *fan-high* condition versus value of control parameter m

m	1	2.5	6	16	30	100
$J_{interp2}$	9.64	11.39	7.49	6.53	8.23	7.53

Chapter 6

Conclusions

In this chapter a comprehensive summary of the thesis is given. Then, the chapter concludes by reviewing some directions for the future.

6.1 Review of Work

In this thesis acoustic feature transformations with dimensionality reduction were studied to improve basic recognition performance of a speech recognizer. The aim of acoustic feature transformations is to reduce dimensionality of long-term speech features without losing discriminative information between different classes. Acoustic feature transformations with dimensionality reduction could be divided into two groups: One maximizes the ratio of between-class scatter to within-class scatter and the other minimizes a kind of classification error. The former and latter approaches were studied in Chapters 3 and 4, and in Chapter 5, respectively.

Chapter 3 studied the interrelationship between several linear transformations which have been commonly used in state-of-the-art speech recognition systems. In the chapter close relationships were proven to exist between linear discriminant analysis (LDA), heteroscedastic linear discriminant analysis (HLDA), and heteroscedastic discriminant analysis (HDA). This work has pointed out that the objective functions of HLDA and HDA can be viewed as the same formulation except their numerators, and the difference between LDA and HDA is the definitions of the mean of the class covariance matrices. Then, a common framework was proposed for integrating various criteria, which includes LDA, HLDA and HDA. The framework termed power linear discriminant analysis (PLDA) could describe various criteria by varying its control parameter. The experimental results on the CENSREC-3 database showed that the PLDA with the optimal control parameters reduced word error rate from 11.24% to 8.48%. Then, a sub-optimal control parameter selection method was given. The proposed selection method used the Chernoff bound as a measure of a class separability error, which was an upper bound of the Bayes error. It could evaluate the relative recognition performance without training HMMs and testing an evalua-

tion set, and reduced a computational cost from 220 hours to less than one hour. In addition, it yielded accurate performance comparison with a drastic reduction of computational costs. The effectiveness of the method was experimentally demonstrated. Finally, the effectiveness of discriminant analysis-based feature transformation techniques and discriminative training techniques was investigated. Under a matched background noise condition between training and evaluation, both techniques achieved better results than the traditional one. In addition, a combination of these techniques obtained the best result. However, in a mismatched background noise condition, the combinations of acoustic feature transformations and discriminative training techniques were not necessarily effective.

In Chapter 4, two dimensionality reduction methods described in Chapter 3, HDA and PLDA, were extended to deal with multimodal data. The dimensionality reduction methods reviewed and proposed in Chapter 3, however, may result in an unexpected dimensionality reduction if the data in a certain class consist of several clusters, i.e., multimodal, because they implicitly assume that data are generated from a single Gaussian distribution. In order to deal with multimodal data using HDA, we combine the ideas of LPP and HDA, and propose locality-preserving HDA. In addition, we propose locality-preserving PLDA. These extensions can be expected to yield better performance because they reduce the dimensionality of multimodal data appropriately. In general, considerable computational time is required to obtain the optimal projections by locality-preserving methods. To overcome this problem, we proposed an approximate calculation scheme. Experimental results showed that the best performance, 4.83%, was obtained by locality-preserving PLDA, while the word error rate of the baseline system was 6.50%. Hence, locality-preserving PLDA provided a relative word error rate reduction of 25% compared to the baseline system. Moreover, the locality-preserving dimensionality reduction methods yielded better performance than the traditional ones, especially under matched noise conditions. In particular, LPLDA outperformed the others whether or not the noise condition in evaluation matched that in training.

Chapter 5 investigated linear feature transformation methods that minimizes a classification error between different classes. As the performance of speech recognition systems generally has a close correlation with the classification accuracy of features, the features should have the power to discriminate between different classes. The existing methods for this approach served to minimize the average classification error between different classes. Although minimizing the average classification error suppresses total classification error, it cannot prevent the occurrence of considerable overlaps between distributions of some different classes, which is critical for speech recognition because there may be class pairs that have little or no discriminative information on each other. Instead of the average classification error, minimization methods of the maximum classification error was proposed in the thesis, which could avoid considerable error between different classes. The proposed method achieved word error rate of 5.36%, which was a relative word error rate reduction of 17.5% compared to the baseline system. In addition, interpolation

methods that minimized the maximization error while minimizing the average classification error were proposed. Instead of the average classification error, minimization methods of the maximum classification error are proposed so as to avoid considerable error between different classes. In addition, interpolation methods that minimize the maximization error while minimizing the average classification error are also proposed. The best result of the proposed methods was 3.32%. The relative improvement was 48.9% compared to the baseline system.

In summary, acoustic feature transformations for speech recognition were studied. First, acoustic feature transformations using criteria with which to maximize the ratio of between-class scatter to within-class scatter were developed. Second, acoustic feature transformations which minimize a kind of classification error between different phonetic classes were developed. Both approaches evidenced significant enhancement of the basic performance of a speech recognizer.

6.2 Future Work

Several acoustic feature transformations were studied to improve basic performance of a speech recognizer. There are many possible directions in which future work could proceed. The following may serve as starting point for further research:

- The control parameter selection method introduced in Chapter 3 and the estimation of the classification error proposed in Chapter 5 are needed to calculate the Bhattacharyya coefficient. A single Gaussian assumption for each class distribution was imposed for calculating it. Since a single Gaussian was too simple to represent a class distribution, their effectiveness might be limited. Recently, reasonable approximations to the Bhattacharyya coefficient under a Gaussian mixture model (GMM) assumption were derived [33]. It is worthwhile to try to extend the control parameter selection methods and the estimation of the classification error, which use the Bhattacharyya coefficient under a GMM assumption.

- The combinational use of acoustic feature transformations and discriminative training techniques was investigated in Chapter 3. Several techniques based on margin maximization and Bayesian learning have recently been proposed as other techniques [110–113]. The combinations of acoustic feature transformations and the other discriminative training techniques might further improve speech recognition performance.

- This work only attempted to carry out an isolated word recognition to measure the effectiveness of the proposed methods. Acoustic feature transformations proposed in this thesis will be effective not only in an isolated word recognition task but also in a continuous speech recognition one. It should assess the effectiveness of application of proposed acoustic feature transformations to continuous speech recognition.

While human speech is a natural interface that links a person and a computer, speech recognizers are required to achieve human-like recognition performance. Recognition errors, however, are difficult to eradicate even with the state-of-the-art speech recognition technology. Hence, as well as improving speech recognition performance, it is important to develop a recognition error robust interface of an ASR system. We believe that further improvements of speech recognition performance and interface lead to a widespread use of ASR systems.

Appendix A

Mathematical Appendices

A.1 Derivation of Equation (3.20)

Let $\tilde{\mathbf{C}}_k (1 \leq k \leq K)$ be symmetric positive definite matrices. Then,

$$\lim_{m \to 0} J_{PLDA}(\mathbf{B}, m) = \frac{\left|\tilde{\mathbf{C}}_n\right|}{\prod_{k=1}^{K} \left|\tilde{\mathbf{C}}_k\right|^{P_k}}. \tag{A.1}$$

Proof. Here, we focus on the denominator of a PLDA objective function. We let

$$f(m) = \log \left| \sum_{i=1}^{K} P_i \tilde{\mathbf{C}}_i^m \right| \tag{A.2}$$

and $g(m) = m$, so that

$$\log \left| \left(\sum_{i=1}^{K} P_i \tilde{\mathbf{C}}_i^m \right)^{1/m} \right| = \frac{1}{m} \log \left| \sum_{i=1}^{K} P_i \tilde{\mathbf{C}}_i^m \right| = \frac{f(m)}{g(m)}. \tag{A.3}$$

Then $f(0) = g(0) = 0$, and

$$\frac{\partial f(m)}{\partial m} = tr \left(\mathbf{Z}_m \sum_i P_i \frac{\partial}{\partial m} \tilde{\mathbf{C}}_i^m \right) \tag{A.4}$$

$$= tr \left(\mathbf{Z}_m \sum_i P_i \mathbf{U}_i \left(\frac{\partial}{\partial m} \mathbf{\Lambda}_i^m \right) \mathbf{U}_i^\top \right) \tag{A.5}$$

$$= tr \left(\mathbf{Z}_m \sum_i P_i \mathbf{U}_i \mathbf{\Lambda}_i^m \left(\log \mathbf{\Lambda}_i \right) \mathbf{U}_i^\top \right), \tag{A.6}$$

$$\frac{\partial g(m)}{\partial m} = 1, \tag{A.7}$$

where $\mathbf{Z}_m = \left(\sum_j P_j \tilde{\mathbf{C}}_j^m\right)^{-1}$, $\mathbf{U}_i$ denotes the matrix of eigenvectors of $\tilde{\mathbf{C}}_i^m$, and $\mathbf{\Lambda}_i$ denotes the diagonal matrix of eigenvalues of $\tilde{\mathbf{C}}_i^m$.

By l'Hôpital's rule,

$$\lim_{m\to 0} \frac{f(m)}{g(m)} = \lim_{m\to 0} \frac{f'(m)}{g'(m)} = \frac{f'(0)}{g'(0)} \tag{A.8}$$

$$= \sum_i P_i tr(\log \mathbf{\Lambda}_i) \tag{A.9}$$

$$= \log \prod_i \left|\tilde{\mathbf{C}}_i\right|^{P_i}, \tag{A.10}$$

and (A.1) follows. □

A.2 Derivations of Equations (3.22) and (3.23)

Let $|\tilde{\mathbf{C}}_k| = \max |\tilde{\mathbf{C}}_i|$ (k is not necessarily unique). If $\tilde{\mathbf{C}}_k$ satisfies $\tilde{\mathbf{C}}_k^m \succeq \sum_i P_i \tilde{\mathbf{C}}_i^m$, then

$$J_{PLDA}\left(\mathbf{B}_{[p]}, \infty\right) = \frac{\left|\tilde{\mathbf{C}}_n\right|}{\max_i \left|\tilde{\mathbf{C}}_i\right|}, \tag{A.11}$$

$$J_{PLDA}\left(\mathbf{B}_{[p]}, -\infty\right) = \frac{\left|\tilde{\mathbf{C}}_n\right|}{\min_i \left|\tilde{\mathbf{C}}_i\right|}, \tag{A.12}$$

where $\mathbf{X} \succeq \mathbf{Y}$ denotes that $(\mathbf{X} - \mathbf{Y})$ is a positive semidefinite matrix. Equations (3.22) and (3.23) are Equations (A.11) and (A.12), respectively.

Proof. To prove (A.11), let

$$\phi(m) = \left| \left(\sum_i P_i \tilde{\mathbf{C}}_i^m \right)^{1/m} \right|. \tag{A.13}$$

We have the following inequality[1]:

$$\left| \sum_i P_i \tilde{\mathbf{C}}_i^m \right| \geq \left| P_k \tilde{\mathbf{C}}_k^m \right|. \tag{A.14}$$

[1]The inequality follows from observing that $|\mathbf{X} + \mathbf{Y}| \geq |\mathbf{X}|$ for symmetric positive definite matrices $\mathbf{X}$ and $\mathbf{Y}$.

Using this, for $m > 0$, we obtain

$$\phi(m) \geq \left| P_k \tilde{\mathbf{C}}_k^m \right|^{1/m} \tag{A.15}$$

$$= \left| P_k^{1/m} \tilde{\mathbf{C}}_k \right| \tag{A.16}$$

$$= \left| \tilde{\mathbf{C}}_k \right| \ (m \to \infty). \tag{A.17}$$

Added to this, since we assume that $\tilde{\mathbf{C}}_k$ satisfies $\tilde{\mathbf{C}}_k^m \succeq \sum_i P_i \tilde{\mathbf{C}}_i^m$, we also obtain

$$\left| \tilde{\mathbf{C}}_k^m \right| = \left| \sum_i P_i \tilde{\mathbf{C}}_i^m + \tilde{\mathbf{C}}_k^m - \sum_i P_i \tilde{\mathbf{C}}_i^m \right| \tag{A.18}$$

$$\geq \left| \sum_i P_i \tilde{\mathbf{C}}_i^m \right|. \tag{A.19}$$

Hence,

$$\left| \tilde{\mathbf{C}}_k \right| \geq \left| \sum_i P_i \tilde{\mathbf{C}}_i^m \right|^{1/m} = \phi(m). \tag{A.20}$$

(A.17) and (A.20) imply

$$\lim_{m \to \infty} \phi(m) = \left| \tilde{\mathbf{C}}_k \right| = \max_i \left| \tilde{\mathbf{C}}_i \right|. \tag{A.21}$$

(A.11) then follows.

To prove (A.12), let $n = -m$ and $\check{\mathbf{C}}_i = \tilde{\mathbf{C}}_i^{-1}$. Then

$$\phi(m) = \left(\left| \sum_i P_i \check{\mathbf{C}}_i^n \right|^{1/n} \right)^{-1} \tag{A.22}$$

and hence

$$\lim_{m \to -\infty} \phi(m) = \lim_{n \to \infty} \left(\left| \sum_i P_i \check{\mathbf{C}}_i^n \right|^{1/n} \right)^{-1} \tag{A.23}$$

$$= \left(\max_i \left| \check{\mathbf{C}}_i \right| \right)^{-1} \tag{A.24}$$

$$= \min_i |\tilde{\mathbf{C}}_i|, \tag{A.25}$$

and (A.12) follows. □

A.3 Interpretation of HDA

Similar to LDA, HDA searches for a projection matrix $\mathbf{B}$ such that data pairs in the same class are close to each other and data pairs in different classes are separated from each other. We give a formal interpretation of this. In [23], for $\mathbf{v}_{ij} \equiv \mathbf{B}^\top(\mathbf{x}_i - \mathbf{x}_j)$, the change in LDA objective function when only $\mathbf{v}_{ab}$ becomes $\alpha\mathbf{v}_{ab}$ with $\alpha > 0$ was investigated, where $a, b \in \{1 \cdots N\}$ and $a \neq b$. The result has showed the following: the value of the LDA objective function becomes large when a data pair in the same class is close to each other, i.e., $y_a = y_b$ and $0 < \alpha < 1$, or when a data pair in different classes is separated from each other, i.e., $y_a \neq y_b$ and $\alpha > 1$. Through a similar approach in [23], we will investigate the change in HDA objective function when only $\mathbf{v}_{ab}$ becomes $\alpha\mathbf{v}_{ab}$ with $\alpha > 0$.

To simplify the notations of projected covariance matrices, we rewrite $\mathbf{B}^\top\mathbf{C}^{(B)}\mathbf{B}$ and $\mathbf{B}^\top\mathbf{C}_k\mathbf{B}$ as $\bar{\mathbf{C}}^{(B)}$ and $\bar{\mathbf{C}}_k$, respectively. From here on, we denote covariance matrices in the projected space by symbols with a bar. Let us rewrite an HDA objective function as follows:

$$J_{HDA}(\mathbf{B}) = \frac{\left|\bar{\mathbf{C}}^{(B)}\right|}{\prod_{k=1}^{K}\left|\bar{\mathbf{C}}_k\right|^{P_k}}. \tag{A.26}$$

$\bar{\mathbf{C}}^{(\alpha B)}$ and $\bar{\mathbf{C}}_k^{(\alpha)}$ denote the between-class and class covariance matrices for $\alpha\mathbf{v}_{ab}$ defined by

$$\begin{aligned}
\bar{\mathbf{C}}^{(\alpha B)} &\equiv \bar{\mathbf{C}}^{(B)} - (\beta/N)W_{ab}^{(B)}\mathbf{v}_{ab}\mathbf{v}_{ab}^\top, \\
\bar{\mathbf{C}}_k^{(\alpha)} &\equiv \bar{\mathbf{C}}_k - (\beta/N_k)W_{k,ab}\mathbf{v}_{ab}\mathbf{v}_{ab}^\top, \\
\beta &\equiv \frac{1-\alpha^2}{2},
\end{aligned}$$

assuming that $\bar{\mathbf{C}}^{(B)}$, $\bar{\mathbf{C}}^{(\alpha B)}$, $\bar{\mathbf{C}}_k$ and $\bar{\mathbf{C}}_k^{(\alpha)}$ are positive definite matrices. The objective function of HDA for $\alpha\mathbf{v}_{ab}$ is given by

$$J_{HDA}^{(\alpha)}(\mathbf{B}) \equiv \frac{\left|\bar{\mathbf{C}}^{(\alpha B)}\right|}{\prod_{k=1}^{K}\left|\bar{\mathbf{C}}_k^{(\alpha)}\right|^{P_k}}. \tag{A.27}$$

From the definition of $\bar{\mathbf{C}}^{(\alpha B)}$, the determinant of $\bar{\mathbf{C}}^{(B)}$ becomes

$$\begin{aligned}\left|\bar{\mathbf{C}}^{(B)}\right| &= \left|\bar{\mathbf{C}}^{(\alpha B)} + (\beta/N) W_{ab}^{(B)} \mathbf{v}_{ab}\mathbf{v}_{ab}^{\top}\right| \\ &= \left|\bar{\mathbf{C}}^{(\alpha B)}\left(\mathbf{I} + (\beta/N) W_{ab}^{(B)} \mathbf{P}^{(\alpha B)}\mathbf{v}_{ab}\mathbf{v}_{ab}^{\top}\right)\right| \\ &= \left|\bar{\mathbf{C}}^{(\alpha B)}\right|\left|\mathbf{I} + (\beta/N) W_{ab}^{(B)} \mathbf{P}^{(\alpha B)}\mathbf{v}_{ab}\mathbf{v}_{ab}^{\top}\right| \\ &= \left(1 + (\beta/N) W_{ab}^{(B)} \mathbf{v}_{ab}^{\top}\mathbf{P}^{(\alpha B)}\mathbf{v}_{ab}\right)\left|\bar{\mathbf{C}}^{(\alpha B)}\right|,\end{aligned}$$

where we let $\mathbf{P}^{(\alpha B)} = (\bar{\mathbf{C}}^{(\alpha B)})^{-1}$. We have made use of the following formula [15]: $|\mathbf{I} + \mathbf{a}\mathbf{b}^{\top}| = 1 + \mathbf{a}^{\top}\mathbf{b}$, where $\mathbf{a}$ and $\mathbf{b}$ are arbitrary vectors and $\mathbf{I}$ is an identity matrix.

We first consider the case when both $\mathbf{x}_i$ and $\mathbf{x}_j$ are in the same class, i.e., $y_i = y_j$. In addition, without loss of generality, suppose that both y_i and y_j are equal to $l \in \{1, \cdots, K\}$. From Equation (4.12), we have $W_{ij}^{(B)} < 0$. We also have $W_{l,ij} > 0$ if $y_i = y_j = l$ and $W_{k,ij} = 0$ if $y_i = y_j = k(\neq l)$. Similar to the case of the determinant of $\bar{\mathbf{C}}^{(B)}$, the determinant of $\bar{\mathbf{C}}_l$ becomes

$$\left|\bar{\mathbf{C}}_l\right| = \left(1 + (\beta/N_l) W_{l,ab}\mathbf{v}_{ab}^{\top}\left(\bar{\mathbf{C}}_l^{(\alpha)}\right)^{-1}\mathbf{v}_{ab}\right)\left|\bar{\mathbf{C}}_l^{(\alpha)}\right|$$

and the determinant of $\bar{\mathbf{C}}_k$ becomes $|\bar{\mathbf{C}}_k| = |\bar{\mathbf{C}}_k^{(\alpha)}|$ for $k \neq l$. Equation (A.26) can be rewritten as

$$\begin{aligned}J_{HDA}(\mathbf{B}) &= \frac{\zeta\left|\bar{\mathbf{C}}^{(\alpha B)}\right|}{\eta^{P_l}\prod_{k=1}^{K}\left|\bar{\mathbf{C}}_k^{(\alpha)}\right|^{P_k}} \\ &= \frac{\zeta}{\eta^{P_l}} J_{HDA}^{(\alpha)}(\mathbf{B}),\end{aligned} \tag{A.28}$$

where $\zeta \equiv 1 + (\beta/N) W_{ab}^{(B)}\mathbf{v}_{ab}^{\top}\left(\bar{\mathbf{C}}^{(\alpha B)}\right)^{-1}\mathbf{v}_{ab}$ and $\eta \equiv 1 + (\beta/N_l) W_{l,ab}\mathbf{v}_{ab}^{\top}\left(\bar{\mathbf{C}}_l^{(\alpha)}\right)^{-1}\mathbf{v}_{ab}$.

If $0 < \alpha < 1$, then $\beta > 0$. Since we assume that $\bar{\mathbf{C}}^{(\alpha B)}$ and $\bar{\mathbf{C}}_k^{(\alpha)}$ are positive definite, $\mathbf{v}_{ab}^{\top}\left(\bar{\mathbf{C}}_k^{(\alpha B)}\right)^{-1}\mathbf{v}_{ab}$ and $\mathbf{v}_{ab}^{\top}\left(\bar{\mathbf{C}}_k^{(\alpha)}\right)^{-1}\mathbf{v}_{ab}$ satisfy $\mathbf{v}_{ab}^{\top}\left(\bar{\mathbf{C}}_k^{(\alpha B)}\right)^{-1}\mathbf{v}_{ab} > 0$ and $\mathbf{v}_{ab}^{\top}\left(\bar{\mathbf{C}}_k^{(\alpha)}\right)^{-1}\mathbf{v}_{ab} > 0$, respectively. Hence, Equation (A.28) yields $J_{HDA}(\mathbf{B}) < J_{HDA}^{(\alpha)}(\mathbf{B})$ because $\zeta/\eta^{P_l} < 1$. In other words, the value of the objective function for $\alpha\mathbf{v}_{ab}$ is always greater than that of Equation (A.26) if $y_i = y_j$ and $0 < \alpha < 1$, i.e., a data pair in the same class is made close.

Similarly, if $y_i \neq y_j$, then we have $W_{k,ij} = 0$ and $W_{ij}^{(B)} > 0$. From $\eta = 1$, Equation (A.26)

becomes

$$J_{HDA}(\mathbf{B}) = \frac{\zeta \left|\bar{\mathbf{C}}^{(\alpha B)}\right|}{\prod_{k=1}^{K} \left|\bar{\mathbf{C}}_k^{(\alpha)}\right|^{P_k}} = \zeta\, J_{HDA}^{(\alpha)}(\mathbf{B}). \tag{A.29}$$

If $\alpha > 1$, then $\beta < 0$. Hence, Equation (A.29) yields $J_{HDA}(\mathbf{B}) < J_{HDA}^{(\alpha)}(\mathbf{B})$ because $\zeta < 1$. In other words, the value of the objective function for $\alpha \mathbf{v}_{ab}$ is greater than that of Equation (A.26) if $y_i \neq y_j$ and $\alpha > 1$, i.e., a data pair in different classes is separated from each other.

A.4 Derivation of Equation (4.29)

$W_{ij}^{(LM)}$ in Equation (4.21) can be decomposed as

$$W_{ij}^{(LM)} = \frac{1}{N} - W_{ij}^{(LM1)} + W_{ij}^{(LM2)},$$

where

$$W_{ij}^{(LM1)} \equiv \begin{cases} 1/N & \text{if } y_i = y_j, \\ 0 & \text{if } y_i \neq y_j, \end{cases}$$

$$W_{ij}^{(LM2)} \equiv \begin{cases} A_{ij}/N & \text{if } y_i = y_j, \\ 0 & \text{if } y_i \neq y_j. \end{cases}$$

From the definitions of $W_{ij}^{(LM1)}$ and $W_{ij}^{(LM2)}$, we have

$$W_{ij}^{(LM1)} = \sum_{k=1}^{K} P_k W_{k,ij},$$

$$W_{ij}^{(LM2)} = \sum_{k=1}^{K} P_k W_{k,ij}^{(L)}.$$

Hence,

$$W_{ij}^{(LM)} = \frac{1}{N} - \sum_{k=1}^{K} P_k W_{k,ij} + \sum_{k=1}^{K} P_k W_{k,ij}^{(L)}.$$

Then, we have

$$\begin{aligned}
\mathbf{C}^{(LM)} &= \frac{1}{2N}\sum_{i,j}^{N} W_{ij}^{(LM)}\mathbf{X}_{ij} \\
&= \frac{1}{2N}\sum_{i,j=1}^{N}\left(\frac{1}{N} - \sum_{k=1}^{K} P_k(W_{k,ij} - W_{k,ij}^{(L)})\right)\mathbf{X}_{ij} \\
&= \mathbf{C}^{(M)} - \sum_{k=1}^{K} P_k^2\left(\frac{1}{2N_k}\sum_{i,j=1}^{N}(W_{k,ij} - W_{k,ij}^{(L)})\mathbf{X}_{ij}\right) \\
&= \mathbf{C}^{(M)} - \sum_{k=1}^{K} P_k^2\left(\mathbf{C}_k - \mathbf{C}_k^{(L)}\right),
\end{aligned}$$

where we let $\mathbf{X}_{ij} \equiv (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^\top$.

Bibliography

[1] Jelinek, F.: Continuous speech recognition by statistical methods, *IEEE Proceedings*, Vol. 64, No. 4, pp. 532–556 (1976).

[2] Rabiner, L.: A tutorial on hidden Markov models and selected applications in speech recognition, *IEEE Proceedings*, Vol. 77, No. 2, pp. 257–285 (1989).

[3] Durbin, R., Eddy, S., Krogh, A. and Mitchison, G.: *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*, Cambridge University Press (1998).

[4] Ostendorf, M. and Roukos, S.: A stochastic segment model for phoneme-based continuous speech recognition, *IEEE Trans. Acoust., Speech and Signal Processing*, Vol. 37, No. 12, pp. 1857–1869 (1989).

[5] Haeb-Umbach, R. and Ney, H.: Linear discriminant analysis for improved large vocabulary continuous speech recognition, *Proc. ICASSP*, pp. 13–16 (1992).

[6] Gish, H. and Russell, M.: Parametric trajectory models for speech recognition, *Proc. ICSLP*, pp. 466–469 (1996).

[7] Tokuda, K., Zen, H. and Kitamura, T.: Trajectory modeling based on HMMs with the explicit relationship between static and dynamic features, *Proc. of Eurospeech*, pp. 865–868 (2003).

[8] Gupta, V. N., Lenning, M. and Mermelstein, P.: Integration of acoustic information in a large vocabulary word recognizer, *Proc. ICASSP*, pp. 697–700 (1987).

[9] Deng, L., Aksmanoric, M., Sun, X. and Wu, C. F. J.: Speech recognition using hidden Markov models with polynomial regression functions as non stationary states, *IEEE Trans. Speech & Audio Processing*, Vol. 2, No. 4, pp. 507–520 (1994).

[10] Wellekens, C. J.: Explicit correlation in hidden Markov model for speech recognition, *Proc. ICASSP*, pp. 383–386 (1987).

[11] Ming, J. and Smith, F. J.: Modelling of the interframe dependence in an HMM using conditional Gaussian mixtures, *Computer Speech and Language*, Vol. 10, No. 4, pp. 229–247 (1996).

[12] Ostendorf, M. and Roukos, S.: A stochastic segment model for phoneme-based continuous speech recognition, *IEEE Trans. Acoust., Speech and Signal Processing*, Vol. 37, No. 12, pp. 1857–1869 (1989).

[13] Nakagawa, S. and Yamamoto, K.: Evaluation of segmental unit input HMM, *Proc. ICASSP*, pp. 439–442 (1996).

[14] Bellman, R.: *Adaptive Control Processes: A Guided Tour*, Princeton University Press (1961).

[15] Bishop, C. M.: *Pattern Recognition and Machine Learning*, Springer (2006).

[16] Fukunaga, K.: *Introduction to Statistical Pattern Recognition*, Academic Press, New York, second edition (1990).

[17] Duda, R. O., Hart, P. B. and Stork, D. G.: *Pattern Classification*, John Wiley & Sons, New York (2001).

[18] Campbell, N. A.: Canonical variate analysis – a general model formulation, *Australian Journal of Statistics*, Vol. 4, pp. 86–96 (1984).

[19] Kumar, N. and Andreou, A. G.: Heteroscedastic discriminant analysis and reduced rank HMMs for improved speech recognition, *Speech Communication*, Vol. 26, No. 4, pp. 283–297 (1998).

[20] Saon, G., Padmanabhan, M., Gopinath, R. and Chen, S.: Maximum likelihood discriminant feature spaces, *Proc. ICASSP*, pp. 129–132 (2000).

[21] Hastie, T. and Tibshirani, R.: Discriminant analysis by Gaussian mixtures, *Journal of the Royal Statistical Society*, Vol. 58, No. 1, pp. 155–176 (1996).

[22] He, X. and Niyogi, P.: Locality preserving projections, *Advances in Neural Information Processing Systems* (2004).

[23] Sugiyama, M.: Dimensionality Reduction of Multimodal Labeled Data by Local Fisher Discriminant Analysis, *Journal of Machine Learning Research*, Vol. 8, pp. 1027–1061 (2007).

[24] de la Torre, F. and Kanade, T.: Multimodal Oriented Discriminant Analysis, *International Conference on Machine Learning (ICML)* (2005).

[25] Decell, H. P. and Quirein, J. A.: An Iterative Approach to the Feature Selection Problem, *Conf. Machine Processing of Remotely Sensed Data*, pp. 3B1–3B12 (1973).

[26] Saon, G. and Padmanabhan, M.: Minimum Bayes Error Feature Selection for Continuous Speech Recognition, *Advances in Neural Information Processing Systems*, pp. 800–806 (2001).

[27] de la Torre, F. and Kanade, T.: Oriented discriminant analysis, *British Machine Vision Conference*, pp. 132–141 (2004).

[28] Loog, M. and Duin, R. P. W.: Linear Dimensionality Reduction via a Heteroscedastic Extension of LDA: The Chernoff Criterion, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 26, No. 6, pp. 732–739 (2004).

[29] Nenadic, Z.: Information Discriminant Analysis: Feature Extraction with an Information-Theoretic Objective, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 29, No. 8, pp. 1394–1407 (2007).

[30] Torkkola, K.: On Feature Extraction by Mutual Information Maximization, *Proc. ICASSP*, pp. 821–824 (2002).

[31] Renyi, A.: On measures of information and entropy, *Proceedings of the 4th Berkeley Symposium on Mathematics, Statistics and Probability*, pp. 547–561 (1960).

[32] Principe, J. C., III, J. W. F. and Xu, D.: *Information Theoretic Learning*, in Unsupervised Adaptive Filtering, John Wiley & Sons (2000).

[33] Olsen, P. A. and Hershey, J. R.: Bhattacharyya Error and Divergence using Variational Importance Sampling, *Proc. Interspeech*, pp. 46–49 (2007).

[34] Smith, N. and Gales, M. J. F.: Speech Recognition using SVMs, *Advances in Neural Information Processing Systems*, Vol. 14, pp. 1197–1204 (2002).

[35] Ganapathiraju, A., Hamaker, J. E. and Picone, J.: Applications of support vector machines to speech recognition, *IEEE Transactions on Signal Processing*, Vol. 52, No. 8, pp. 2348–2355 (2004).

[36] Kuo, H. K. J. and Gao, Y.: Maximum entropy direct models for speech recognition, *IEEE Transactions on Audio, Speech, and Language Processing*, Vol. 14, No. 3, pp. 873–881 (2006).

[37] Gunawardana, A., Mahajan, M., Acero, A. and Platt, J. C.: Hidden conditional random fields for phone classification, *Proc. Interspeech*, pp. 1117–1120 (2005).

[38] Sung, Y.-H. and Jurafsky, D.: Hidden Conditional Random Fields for Phone Recognition, *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 107–112 (2009).

[39] Lafferty, J., McCallum, A. and Pereira, F.: Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data, *Proc. 18th International Conf. on Machine Learning*, pp. 282–289 (2001).

[40] Deller, J. R., Hansen, J. H. L. and Proakis, J. G.: *Discrete-Time Processing of Speech Signals*, IEEE PRESS (2000).

[41] Davis, S. B. and Mermelstein, P.: Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences, *IEEE Trans. Speech & Audio Processing*, Vol. 28, pp. 357–366 (1980).

[42] Hermansky, H.: Perceptual linear prediction (PLP) analysis of speech, *Journal of Acoustic Society of America*, Vol. 87, No. 4, pp. 1738–1752 (1990).

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

[44] Rabiner, L. and Juang, B.-H.: *Fundamentals of Speech Recognition*, Prrentice Hall (1993).

[45] Dempster, A., Laird, N. and Rubin, D.: Maximum likelihood from incomplete data via the EM algorithm, *Journal of the Royal Statistical Society*, Vol. 39, pp. 1–38 (1977).

[46] Bahl, L., Brown, P., de Sousa, P. and Mercer, R.: Maximum mutual information estimation of hidden Markov model parameters for speech recognition, *Proc. ICASSP*, Vol. 49–52, pp. 49–52 (1986).

[47] Juang, B.-H. and Katagiri, S.: Discriminative Learning for Minimum Error Classification, *IEEE Transactions on Signal Processing*, Vol. 40, No. 12, pp. 3043–3054 (1992).

[48] Kapadia, S.: Discriminative Training of Hidden Markov Models, PhD Thesis, University of Cambridge (1998).

[49] Povey, D. and Woodland, P. C.: Frame Discrimination Training Of HMMs For Large Vocabulary Speech Recognition, *Proc. ICASSP*, pp. 333–336 (1999).

[50] Byrne, W.: Minimum Bayes risk estimation and decoding in large vocabulary continuous speech recognition, *IEICE Special Issue on Statistical Modelling for Speech Recognition* (2006).

[51] Na, K., Jeon, B., Chang, D., Chae, S. and Ann, S.: Discriminative training of hidden Markov models using overall risk criterion and reduced gradient method, *Proc. Eurospeech*, pp. 97–100 (1995).

[52] Kaiser, J., Horvat, B. and Kacic, Z.: A novel loss function for the overall risk criterion based discriminative training of HMM models, *International Conference on Spoken Language Processing*, pp. 887–890 (2000).

[53] Povey, D. and Woodland, P.: Minimum phone error and I-smoothing for improved discriminative training, *Proc. ICASSP*, pp. 105–108 (2002).

[54] Shannon, C. E. and Weaver, W.: *The mathematical theory of communication*, University of Illinois Press (1949).

[55] Cover, T. M. and Thomas, J. A.: *Elements of information theory*, John Wiley & Sons (1991).

[56] Woodland, P. and Povey, D.: Large scale MMIE training for conversational telephone speech recognition, *NIST 2000 Speech Transcription Workshop* (2000).

[57] Povey, D., Kanevsky, D., Kingsbury, B., Rambhadran, B., Saon, G. and Visweswariah, K.: Boosted MMI for model and feature-space discriminative training, *Proc. ICASSP*, pp. 2398–2401 (2008).

[58] Nakamura, A., McDermott, E., Watanabe, S. and Katagiri, S.: A UNIFIED VIEW FOR DISCRIMINATIVE OBJECTIVE FUNCTIONS BASED ON NEGATIVE EXPONENTIAL OF DIFFERENCE MEASURE BETWEEN STRINGS, *Proc. ICASSP*, pp. 1633–1636 (2009).

[59] Macherey, W., Haferkamp, L., Schluter, R. and Ney, H.: Investigations on Error Minimizing Training Criteria for Discriminative Training in Automatic Speech Recognition, *Proc. Interspeech*, pp. 2133–2136 (2005).

[60] Levenshtein, V. I.: Binary codes capable of correcting deletions, insertions, and reversals, *Soviet physics doklady*, Vol. 10, No. 8, pp. 707–710 (1966).

[61] Viterbi, A.: Error bounds for convolutional codes and an asymmetrically optimum decoding algorithm, *IEEE Transactions on Information Theory*, Vol. IT-13, pp. 260–267 (1967).

[62] *HTK Web site*. `http://htk.eng.cam.ac.uk/`.

[63] Furui, S.: Speaker independent isolated word recognition using dynamic features of speech spectrum, *IEEE Trans. Acoust., Speech and Signal Processing*, Vol. 34, No. 1, pp. 52–59 (1986).

[64] Lima, A., Zen, H., Nankaku, Y., Miyajima, C., Tokuda, K. and Kitamura, T.: On the Use of Kernel PCA for Feature Extraction in Speech Recognition, *IEICE Transactions on Information and Systems*, Vol. E87-D(12), pp. 2802–2811 (2004).

[65] Schölkopf, B., Smola, A. and Müller, K.-R.: Nonlinear component analysis as a kernel eigenvalue problem, *Neural Computation*, Vol. 10, No. 5, pp. 1299–1319 (1998).

[66] Schölkopf, B., Smola, A. and M′uller, K. R.: Kernel principal component analysis, *Advances in Neural Information Processing Systems*, pp. 327–352 (1998).

[67] Roth, V. and Steinhage, V.: Nonlinear discriminant analysis using kernel functions, *Advances in Neural Information Processing Systems* (1998).

[68] Mika, S., Ratsch, G., Weston, J., Schölkopf, B. and Müller, K.-R.: Fisher discriminant analysis with kernels, *Neural Networks for Signal Processing*, Vol. IX, pp. 41–48 (1999).

[69] Kocsor, A., Toth, L. and Paczolay, D.: A Nonlinearized Discriminant Analysis and its Application to Speech Impediment Therapy, *Proceedings of Text, Speech and Dialogue*, pp. 249–457 (2001).

[70] Tipping, M. E.: Sparse kernel principal component analysis, *Advances in Neural Information Processing Systems*, pp. 633–639 (2001).

[71] Erdogan, H.: Subspace kernel discriminant analysis for speech recognition, *Robust 2004 Workshop* (2004).

[72] Gales, M. J. F.: Semi-Tied Full-Covariance Matrices for Hidden Markov Models, Technical report cued/f-infeng/tr287, University of Cambridge (1997).

[73] Gales, M. J. F.: Semi-tied covariance matrices for hidden Markov models, *IEEE Trans. Speech Audio Processing*, Vol. 7, No. 3, pp. 272–281 (1999).

[74] Gopinath, R. A.: Maximum likelihood modeling with Gaussian distributions for classification, *Proc. ICASSP* (1998).

[75] Olsen, P. A. and Gopinath, R. A.: Modeling inverse covariance matrices by basis expansion, *IEEE Transactions on Speech and Audio Processing*, Vol. 12, No. 1, pp. 37–46 (2004).

[76] Vanhoucke, V. and Sankar, A.: Mixtures of inverse covariances, *IEEE Transactions on Speech and Audio Processing*, Vol. 12, No. 3, pp. 250–264 (2004).

[77] Axelrod, S., Goel, V., Gopinath, R. A., Olsen, P. and Visweswaria, K.: Subspace constrained Gaussian mixture models for speech recognition, *IEEE Transactions on Speech and Audio Processing*, Vol. 13, No. 6, pp. 1144–1160 (2005).

[78] Wegmann, S., McAllaster, D., Orloff, J. and Peskin, B.: Speaker Normalization on Conversational Telephone Speech, *Proc. ICASSP*, pp. 339–341 (1996).

[79] Eide, E. and Gish, H.: A Parametric Approach to Vocal Tract Length Normalization, *Proc. ICASSP*, pp. 346–349 (1996).

[80] Lee, L. and Rose, R.: Speaker Normalization Using Efficient Frequency Warping Procedures, *Proc. ICASSP*, pp. 353–356 (1996).

[81] Zhan, P. and Westohal, M.: Speaker Normalization Based on Frequency Warping, *Proc. ICASSP*, pp. 1039–1042 (1997).

[82] Emori, T. and Shinoda, K.: Rapid Vocal Tract Length Normalization using Maximum Likelihood Estimation, *Proc. EuroSpeech*, pp. 1649–1652 (2001).

[83] Jaschul, J.: Speaker Adaptation by a linear transformation with optimised parameters, *Proc. ICASSP*, pp. 1657–1670 (1982).

[84] Class, F., Kaltenmeier, A., Regel, P. and Troller, K.: Fast speaker adaptation for speech recognition systems, *Proc. ICASSP*, pp. 133–136 (1990).

[85] Leggetter, C. J. and Woodland, P. C.: Maximum Likelihood Linear Regression for Speaker Adaptation of Continuous Density Hidden Markov Models, *Computer Speech and Language*, Vol. 9, p. 2 (1995).

[86] Gales, M. J. F. and Woodland, P. C.: Mean and Variance Adaptation within the MLLR framework, *Computer Speech and Language*, Vol. 10, No. 4, pp. 249–264 (1996).

[87] Digalakis, V., Rtischev, D. and Neumeyer, L.: Speaker Adaptation Using Constrained Estimation of Gaussian Mixtures, *IEEE Transaction on Speech and Audio Processing*, Vol. 3, No. 5, pp. 357–366 (1995).

[88] Gales, M. J. F.: Maximum Likelihood Linear Transformations for HMM-based Speech Recognition, *Computer Speech and Language*, Vol. 12, No. 2, pp. 75–98 (1998).

[89] Sakai, M., Kitaoka, N. and Nakagawa, S.: Generalization of linear discriminant analysis used in segmental unit input HMM for speech recognition, *Proc. ICASSP*, pp. 333–336 (2007).

[90] Sakai, M., Kitaoka, N. and Nakagawa, S.: Linear Discriminant Analysis Using a Generalized Mean of Class Covariances and Its Application to Speech Recognition, *IEICE Transactions on Information and Systems*, Vol. E91-D, No. 3, pp. 478–487 (2008).

[91] Sakai, M., Kitaoka, N. and Nakagawa, S.: Selection of optimal dimensionality reduction method using Chernoff bound for segmental unit input HMM, *Proc. Interspeech*, pp. 1110–1113 (2007).

[92] Oja, E.: *Subspace Methods of Pattern Recognition*, Letchworth: Research Studies Press (1983).

[93] Biem, A., Katagiri, S. and Juang, B.-H.: Pattern recognition using discriminative feature extraction, *IEEE Transactions on Signal Processing*, Vol. 45, No. 2, pp. 500–504 (1997).

[94] Soltau, H., Kingsbury, B., Mangu, L., Povey, D., Saon, G. and Zweig, G.: The IBM conversational telephony system for rich transcription, *Proc. ICASSP*, pp. 205–208 (2005).

[95] Ma, J. Z. and Matsoukas, S.: Improvements to the BBN RT04 Mandarin conversational telephone speech recognition system, *Proc. Interspeech*, pp. 1625–1628 (2005).

[96] Kitaoka, N., Sakai, M., Hattori, Y., Nakagawa, S. and Takeda, K.: Evaluation of Discriminant Analysis-based Feature Transformation and Discriminative Training for Speech Recognition, *SPECOM*, pp. 47–50 (2009).

[97] Magnus, J. R. and Neudecker, H.: *Matrix Differential Calculus with Applications in Statistics and Econometrics*, John Wiley & Sons (1999).

[98] Nelder, J. A. and Mead, R.: A Simplex Method for Function Minimization, *Computer Journal*, Vol. 7, pp. 308–313 (1965).

[99] Belisle, C. J. P.: Convergence theorems for a class of simulated annealing algorithms, *Journal of Applied Probability*, Vol. 29, pp. 885–892 (1992).

[100] Searle, S. R.: *Matrix Algebra Useful for Statistics*, Wiley Series in Probability and Mathematical Statistics, New York (1982).

[101] Nocedal, J. and Wright, S. J.: *Numerical Optimization*, Springer-Verlag (1999).

[102] Fujimoto, M., Takeda, K. and Nakamura, S.: CENSREC-3: An Evaluation Framework for Japanese Speech Recognition in Real Driving-Car Environments, *IEICE Transactions on Information and Systems*, Vol. E89-D, No. 11, pp. 2783–2793 (2006).

[103] Roweis, S. T. and Saul, L. K.: Nonlinear Dimensionality Reduction by Locally Linear Embedding, *Science*, Vol. 290, No. 5500, pp. 2323–2326 (2000).

[104] Belkin, M. and Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation, *Neural Computation*, Vol. 15, No. 6, pp. 1373–1396 (2003).

[105] Zelnik-Manor, L. and Perona, P.: Self-Tuning Spectral Clustering, *Advances in Neural Information Processing Systems*, pp. 1601–1608 (2005).

[106] Sugiyama, M.: Local Fisher discriminant analysis for supervised dimensionality reduction, *International Conference on Machine Learning (ICML)*, pp. 905–912 (2006).

[107] Kailath, T.: The Divergence and Bhattacharyya Distance Measures in Signal Selection, *IEEE Transactions on Communication Technology*, Vol. 15, No. 1, pp. 52–60 (1967).

[108] Basseville, M.: Distance Measures for Signal Processing and Pattern Recognition, *Signal Processing*, Vol. 18, No. 4, pp. 349–369 (1989).

[109] Boekee, D. E. and der Lubbe, J. C. A. V.: Some Aspects of Error Bounds in Feature Selection, *Pattern Recognition*, Vol. 11, pp. 353–360 (1979).

[110] Jiang, H.: Large margin hidden Markov models for speech recognition, *IEEE Transactions on Audio, Speech, and Language Processing*, Vol. 14, pp. 1584–1595 (2006).

[111] Sha, F. and Saul, L. K.: Large margin hidden Markov models for automatic speech recognition, *Advances in Neural Information Processing Systems*, pp. 1249–1256 (2007).

[112] Beal, M. J.: Variational Algorithms for Approximate Bayesian Inference, PhD Thesis, University of London (2003).

[113] Watanabe, S., Minami, Y., Nakamura, A. and Ueda, N.: Application of Variational Bayesian Approach to Speech Recognition, *Advances in Neural Information Processing Systems*, pp. 1261–1268 (2002).

List of Publications

Major Publications

[1] M. Sakai, N. Kitaoka and S. Nakagawa. Linear Discriminant Analysis Using a Generalized Mean of Class Covariances and its Application to Speech Recognition. IEICE Transactions on Information and Systems, pp. 478–487, E91-D, No. 3, 2008.

[2] M. Sakai, N. Kitaoka, Y. Hattori, S. Nakagawa and K. Takeda. Evaluation fo Combinational Use of Discriminant Analysis-based Acoustic Feature Transformation and Discriminative Training. IEICE Transactions on Information and Systems, pp. 395–398, E93-D, No. 2, 2010.

[3] M. Sakai, N. Kitaoka, and K. Takeda. Acoustic Feature Transformation Based on Discriminant Analysis Preserving Local Structure for Speech Recognition. IEICE Transactions on Information and Systems, pp. 1244–1252, E93-D, No. 5, 2010.

[4] M. Sakai, N. Kitaoka, and K. Takeda. Acoustic Feature Transformation Combining Average and Maximum Classification Error Minimization Criteria. IEICE Transactions on Information and Systems, E93-D, No. 7, 2010 (to appear).

International Conferences

[1] M. Sakai, N. Kitaoka and S. Nakagawa. Power Linear Discriminant Analysis. International Symposium on Signal Processing and its Applications, , 2007.

[2] M. Sakai, N. Kitaoka and S. Nakagawa. Generalization of Linear Discriminant Analysis Used in Segmental Unit Input HMM for Speech Recognition. International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Vol. 4, pp. 333–336, April, 2007.

[3] M. Sakai, N. Kitaoka, S. Nakagawa. Selection of Optimal Dimensionality Reduction Method Using Chernoff Bound for Segmental Unit Input HMM. INTERSPEECH-EUROSPEECH, pp. 1110–1113, Aug., 2007.

[4] M. Sakai, N. Kitaoka, K. Takeda. Feature Transformation Based on Discriminant Analysis Preserving Local Structure for Speech Recognition. International Conference on Acoustics, Speech, and Signal Processing (ICASSP), pp. 3813–2816, Apr. 2009.

[5] N. Kitaoka, M. Sakai, Y. Hattori, S. Nakagawa, K. Takeda. Evaluation of Discriminant Analysis-based Feature Transformation and Discriminative Training for Speech Recognition. SPECOM2009, pp. 47–50, June, 2009.

Domestic Conferences

[1] M. Sakai, N. Kitaoka and S. Nakagawa. Generalization of Linear Discriminant Analysis Used in Segmental Unit Input HMM for Speech Recognition (in Japanese). Proc. Acoustic Society Japan Spring Meeting, 3-10-10, 2007.

[2] M. Sakai, N. Kitaoka, Y. Hattori, S. Nakagawa, K. Takeda. A combination of feature transformation based on discriminant analysis and discriminative training for speech recognition (in Japanese). Proc. Acoustic Society Japan Spring Meeting, 1-Q-16, 2008.

[3] M. Sakai, N. Kitaoka, and K. Takeda. Acoustic Feature Transformation Combining Average Error and Maximum Error Minimization Criteria (in Japanese). Proc. Acoustic Society Japan Spring Meeting, 1-P-18, 2009.

Book

[1] F. Mihelic and J. Zibert, (Eds.), Speech Recognition, IN-TECH, ISBN 978-953-7619-29-9 Hard cover, 550 pages, Nov. 2008.