



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

공학박사 학위논문

상대 엔트로피를 이용한 음성 특징벡터의  
변별적 변환에 관한 연구

A Study on Discriminative Transformation of  
Speech Feature Vector based on Relative Entropy

지도교수 신 욱 근



2009년 2월

한국해양대학교 대학원

제어계측공학과

유 강 주

本 論文을 兪鋼柱의 工學博士 學位論文으로 認准함.

위원장 공학박사 진 강 규 (인)

위 원 공학박사 류 길 수 (인)

위 원 공학박사 김 재 훈 (인)



위 원 공학박사 김 형 순 (인)

위 원 공학박사 신 옥 근 (인)

2008 년 12 월 26 일

한국해양대학교 대학원

# 목차

제 1 장 서론.....	1
1.1 연구의 배경.....	1
1.2 연구 방법 및 구성.....	5
제 2 장 음성 인식과정.....	7
2.1 음성 인식을 위한 특징벡터.....	7
2.1.1 선형 예측 계수.....	8
2.1.2 MFCC.....	10
2.1.3 미분 계수.....	14
2.2 은닉 마르코프 모델을 이용한 음성인식.....	15
2.2.1 마르코프 프로세서.....	15
2.2.2 은닉 마르코프 모델.....	17
제 3 장 특징벡터의 변별적 변환.....	26
3.1 특징벡터의 변별적 변환을 이용한 음성 인식과정.....	26
3.2 주요 성분분석.....	27
3.3 선형 판별 분석.....	29
3.4 Li의 방법.....	31
제 4 장 상대 엔트로피에 기반한 특징벡터의 변별적 변환.....	36
4.1 상대 엔트로피.....	36
4.2 상대 엔트로피를 이용한 특징벡터의 변별적 변환.....	38

제 5 장 실험 및 고찰.....	45
5.1 음성 데이터.....	45
5.2 특징벡터의 변별적 변환 및 클러스터링 실험.....	47
5.3 음소 단위의 인식 실험.....	62
제 6 장 결론.....	74
참고 문헌.....	77



## 그림 목차

그림 2.1	음성 인식 과정.....	7
그림 2.2	멜 스케일.....	11
그림 2.3	멜 스케일 필터 뱅크.....	12
그림 2.4	MFCC 추출 과정.....	12
그림 2.5	마르코프 체인의 상태 천이도.....	17
그림 2.6	이산형 은닉 마르코프 모델.....	19
그림 2.7	연속형 은닉 마르코프 모델.....	19
그림 2.8	LEFT-TO-RIGHT 은닉 마르코프 모델.....	20
그림 2.9	ERGODIC 은닉 마르코프 모델.....	20
그림 2.10	전향 확률 및 후향 확률.....	24
그림 3.1	특징벡터의 변별적 변환에 의한 음성 인식과정.....	26
그림 3.2	주요 성분분석.....	28
그림 4.1	클래스 $C_i$ 와 $C_j$ 의 데이터 분포.....	36
그림 5.1	판별 분석 및 클러스터링의 블록 다이어그램.....	48
그림 5.2	모음 상호간의 특징벡터의 분포.....	51
그림 5.3	모음과 자음의 특징벡터의 분포.....	52
그림 5.4	PCA를 적용한 모음 상호간의 특징벡터의 분포.....	53
그림 5.5	PCA를 적용한 모음과 자음의 특징벡터의 분포.....	54
그림 5.6	LDA를 적용한 모음 상호간의 특징벡터의 분포.....	55
그림 5.7	LDA를 적용한 모음과 자음의 특징벡터의 분포.....	56
그림 5.8	LI 방법을 적용한 모음 상호간의 특징벡터의 분포.....	57
그림 5.9	LI의 방법을 적용한 모음과 자음의 특징벡터의 분포.....	58
그림 5.10	제안한 방법을 적용한 모음 상호간의 특징벡터의 분포.....	59
그림 5.11	제안한 방법을 적용한 모음과 자음의 특징벡터의 분포.....	60
그림 5.12	음소 인식 과정.....	63
그림 5.13	13차원의 특징벡터에 대한 음소 각각의 인식률.....	68
그림 5.14	26차원의 특징벡터에 대한 음소 각각의 인식률.....	69
그림 5.15	13차원 특징벡터에 대한 음소 인식률.....	70

그림 5.16 26차원 특징벡터에 대한 음소 인식률..... 71



## 표 목차

표 5.1 TIMIT의 음소 테이블..... 46  
표 5.2 그림 5.2-5.11에서 분포도에 대한 음소 정보..... 49

표 5.3	음소 단위의 클러스터링 결과.....	61
표 5.4	13차원의 특징벡터에 대한 음소 각각의 인식률 .....	64
표 5.5	26차원의 특징벡터에 대한 음소 각각의 인식률 .....	66
표 5.6	13차원 특징벡터에 대한 음소의 인식률 .....	70
표 5.7	26차원 특징벡터에 대한 음소 인식률.....	71



## A Study on Discriminative Transformation of Speech Feature Vector based on Relative Entropy

Yu, Gang-Ju



Department of Control and Instrumentation Engineering,  
Graduate School, Korea Maritime University

Advised by  
Prof. Shin, Ok-Keun

## Abstract

Generally, the recognition rate of an automatic speech recognition (ASR) system depends largely on the discriminability of the feature vectors representing the input speech signal. To improve the recognition rate, it is therefore, desirable to increase the discriminating power of the feature vectors.

In this thesis, we propose a linear transformation of the feature vector which aims to augment the recognition rate of the ASR by increasing the discriminating power of the feature vectors. By making use of the relative entropy of each phoneme (the unit of recognition), the proposed method tries to shorten the distances between within-class feature vectors, while lengthening the inter-class distances of the feature vectors. The method is based on the observation that as the relative entropy between two classes of feature vectors becomes larger, the dissimilarity increases, and so does the discriminating power between the classes. The proposed transformation matrix of the feature vector is derived as follows: Firstly, the objective function is defined as a function of the

divergence which is the average of relative entropy between classes. Then, the objective function is maximized to give the optimal linear transformation matrix by an iterative learning algorithm, the natural gradient ascent method.

To examine the effect on the discriminating power of the proposed method, two sets of experiments are performed using the TIMIT corpus: a simple phoneme classification experiment using Euclidian distance measure and a recognition experiment by an ASR system. The results are compared with those of the well known methods, such as PCA, LDA and Li's method and shown at least 0.28% of improvement.



# 제 1 장 서론

## 1.1 연구의 배경

음성은 가장 쉽고 자연스러운 의사 전달의 수단인 동시에 음성의 입력 및 전달 과정에 고가의 장치가 필요 없으므로, 인간과 기계 사이의 의사 소통에 있어서 그 효용성을 인정받고 있다. 이를 가능하게 하는 기술이 패턴 인식 기술의 일종인 음성 인식이다[1]. 최근에는 전화망이나 이동통신 환경에서의 많은 응용 시스템들이 인간과 기계 사이의 편리한 상호 작용을 가능하게 하는 음성 인식 기술을 사용하고 있다. 음성 인식 기술을 사용하는 분야는 주식 조회, 자동 교환 서비스, 보이스 포털, 자동차, 장난감, 음성 다이얼링, 자동 예약, 음성 정보 서비스, 콜 센터, PC 명령장치, 자동 타이프 라이터, 그리고 텔레매틱스 등이 있다[1].

음성 인식 시스템은 크게 전처리 부분과 인식 부분으로 나눌 수 있다. 전처리 부분에서는 사용자가 발성한 음성 신호로부터 음성이 가지는 고유한 정보인 특징들의 집합 즉, 특징벡터를 추출하고 인식 부분에서는 음성 데이터베이스로부터 훈련한 인식기의 음성 기준 패턴들과 입력 패턴을 비교하여 인식 결과를 얻게 된다.

인식 부분에서의 인식기 구현에 많이 사용되는 알고리즘으로는 DTW(Dynamic Time Warping) [2,14,15], 신경망(Neural Network) [3, 15]에 의한 방법과 은닉 마르코프 모델(Hidden Markov Model: HMM) [2,4-6,10,11,14,15]을 이용한 방법 등이 있다. DTW는 패턴을 비교하는 알고리즘으로서 인식기의 인식 단위에 대한 표준적인 특징을 가지는 시계열 패턴을 기준 패턴으로 설정한 다음 입력된 시계열 패턴을 비선형적으로 신축해가며 기준패턴과 비교하는 방법이다. 은닉 마르코프 모델은 화자의 개인차에 의한 음성 패턴의 변동을 통계적으로 처리한 후,

그 통계량을 확률적인 형태의 모델에 적용하여 음성을 인식하는 방법이다. 신경망은 은닉 마르코프 모델과 같이 화자의 개인차에 따른 스펙트럼의 변화를 망의 가중치로 변환해서 음성을 인식하는 방법이다.

전처리 부분에서 추출되는 특징벡터는 앞서 언급한 알고리즘들에 의해서 생성된 음성 인식기의 입력으로 사용되어, 인식기의 복잡도는 감소시키고 성능은 향상시키는 역할을 한다. 주로 사용되는 벡터에는 인간의 성도 특성을 모델링한 선형 예측 계수 (Linear Prediction Coefficient: LPC) [2,7,8,10,11]와 인간의 청각 특성을 모델링한 켈스트럼 계수 [2,7,8,10,11]와 MFCC (Mel Frequency Cepstral Coefficient) [9,10,11,12] 등이 있다.

음성 인식기나 영상 인식기 등의 패턴 인식기는 인식기를 사용하기 전에 반드시 인식기의 기준 패턴이나 인식기의 구동에 필요한 파라미터를 생성하는 훈련 과정이 필요한데, 기준 패턴과 구동용 파라미터는 인식기의 학습 때 인가되는 특징 벡터에 매우 의존적이다. 또한 인식기를 사용할 때에도 입력되는 특징 벡터에 많은 영향을 받는다. 그러므로 인식기를 학습하거나 사용할 때, 변별력이 있는 특징 벡터를 생성하여 입력으로 사용하는 것이 필요하고, 이는 인식기의 복잡도 감소와 성능에 많은 도움이 된다. 이러한 이유로, 보다 나은 인식기의 성능을 위해 특징 벡터의 변별력을 향상시키는 많은 연구가 이루어져 왔다. 이러한 연구들은 원시 특징벡터 (raw feature vector)  $\mathbf{x}$ 를 임의의 변환 함수  $g(\mathbf{x})$ 에 적용하여 변별력이 개선된 특징벡터  $\mathbf{y}$ 를 구하는 것으로, 변환 함수에 따라 비선형적인 방법과 선형적인 방법으로 나눌 수 있다. 비선형적인 방법에는 KPCA (Kernel Principal Component Analysis) [19,21], KFDA (Kernel Fisher Discriminant Analysis) [18], 그리고 NDA (Nonlinear Discriminant Analysis) [17] 등이 있는데, 이

논문에서는 선형적인 방법에 관하여 논하므로 이들에 관해서는 더 이상 언급하지 않는다.

선형적인 방법에서는 변환 함수  $g(\mathbf{x})$ 를 변환 행렬  $\mathbf{W}$ 와 원시 특징벡터  $\mathbf{x}$ 의 선형 조합(linear combination) 즉,  $g(\mathbf{x}) = \mathbf{W}\mathbf{x}$ 로 설정하여, 변별력이 개선된 특징벡터  $\mathbf{y} = g(\mathbf{x})$ 를 구한다. 이 방법에서 특징벡터  $\mathbf{y}$ 의 변별력은 변환 행렬  $\mathbf{W}$ 에 의해서 결정된다. 선형적인 방법은 변환 행렬  $\mathbf{W}$ 를 구하는 방법에 따라, 해석적인(analytic) 방법과 반복적 학습(iterative learning)에 의한 방법으로 나눌 수 있다. 해석적인 방법의 대표적인 것으로는 특징벡터에 대한 2차의 통계학적 특성인 평균과 공분산을 이용하는 주요 성분분석(Principal Component Analysis: PCA) [16,23,24,30,32]과 선형 판별분석(Linear Discriminant Analysis: LDA) [25-27,29]이 있다.

주요 성분분석은 특징벡터의 분산이 큰 몇 개의 축 방향으로 특징벡터를 투사하여 특징벡터의 차원을 감소시키는 방법이다. 주요 성분분석에서는 원시 특징벡터에 대한 공분산 행렬의 고유 벡터를 고유치 크기에 따라 정렬한 다음, 정렬된 고유 벡터로부터 변환 행렬을 구한다. 그리고 원시 특징벡터를 변환 행렬에 적용해서 새로운 특징벡터를 생성한다. 주요 성분분석은 변환이 단지 원시 특징벡터와 변환된 특징벡터 사이의 오차를 최소화하는 방향으로만 이루어지므로, 특징벡터의 변별력을 최대로 하지는 못한다. 그러나 특징벡터의 차원감소에는 효과적인 방법이다.

선형 판별분석은 특징벡터의 클래스 정보를 이용하여 변별력은 증가시키고, 차원은 감소시키는 선형 변환이다. 이 방법에서는 클래스 상호간의 공분산 행렬과 클래스 내부의 공분산 행렬의 비로 표현되는 Fisher ratio를 최대화하는 변환 행렬을 구한 후에, 이 행렬과 입력 특징벡터를 곱하여 새로운 특징벡터를 생성한다. 그러나 선형 판별분석은 모든 클래스에서 2차의 통계적 정보인 공분산이 동일한 값을 가진다는 가정하에

특징벡터를 변환하므로 평균의 분류가 용이하고, 가우시안 분포를 가지는 데이터에서 잘 동작한다.

반복적 학습에 의해서 변환 행렬  $\mathbf{W}$ 를 구하는 방법의 대표적인 것으로는 Torkkola의 방법 [21], Li의 방법 [20], HDA(Heteroscedastic Discriminant Analysis) [29,38], 그리고 MLLT(Maximum Likelihood Linear Transformation) [39]가 있다. Torkkola는 클래스의 확률밀도 함수에 관한 아무런 사전 지식 없이 변별력이 있는 특징벡터를 생성하기 위해 특징벡터와 클래스 사이의 MI(Mutual Information) [23,24,28,30,31,32]를 이용하는 방법을 제안했다. 그는 Renyi 엔트로피를 토대로 한 이차 엔트로피(quadratic entropy) [22]와 Parzen 밀도 추정(Parzen density estimation) 방법 [22]을 이용하여 특징벡터와 클래스 사이의 MI를 구한 후에, 이를 최대화해서 변환 행렬을 찾았다. 그리고 이 변환 행렬을 특징벡터에 적용하여 변별력을 향상시켰다.

HDA는 선형 판별분석법에서 모든 클래스가 동일한 공분산을 가진다는 가정의 단점을 보완하기 위해, 각 클래스 내부의 공분산을 구할 때 각각의 클래스마다 다른 가중치를 부여하여 특징벡터의 변별력을 개선하는 방법이다. 또한 선형 판별분석법을 일반화시켜 특징벡터의 차원을 감소시키는 방법이기도 하다.

MLLT는 각 클래스의 분포를 가우시안으로 가정하여, 클래스들이 공분산 행렬을 가지는 분포일 때와 분산으로 이루어진 대각 행렬을 가지는 분포일 때의 ML(Maximum Likelihood)차를 최소가 되도록 하는 변환 행렬을 구하여, 특징벡터의 변별력을 개선하는 방법이다.

Li는 음성인식기의 인식률을 향상시키기 위하여 특징벡터의 변별력을 개선하는 방법을 제안하였다. 그가 제안한 방법에서는 특징벡터의 클래스 정보를 토대로 클래스  $c$ 와 특징벡터  $\mathbf{y}$ 사이의 조건부확률 밀도 함수

$p(\mathbf{y}|\mathbf{c})$ 를 정의한 다음, 이를 Bayes 법칙[13,20,25-28]에 적용하여 정규화된 우도 함수(normalized likelihood function)를 만들었다. 그런 다음 이 함수를 최대화하는 변환 행렬을 구해서 특징벡터에 적용함으로써 변별력을 개선하였다. 그러나 이 방법은 클래스 상호간의 어떠한 정보도 고려하지 않아, 최적화된 변환 행렬을 구할 수 없다.

## 1.2 연구 방법 및 구성

이 논문에서는 인식단위인 음소(phoneme)의 클래스 정보를 이용하여 인식기에 인가되는 특징벡터의 변별력을 개선함과 동시에 인식률은 증가시키고, 인식기의 계산량은 감소시키는 특징벡터의 선형 변환 방법을 제안한다. 제안하는 방법은 인식기의 클래스 정보에 기반한 상대 엔트로피(relative entropy)[26-28,30-32]를 토대로, 통계적인 측면에서 클래스 내부의 거리는 가깝게 하고, 클래스 상호간의 거리는 멀게 하는 특징벡터의 선형 변환 방법이다. 이 방법은 상대 엔트로피가 클수록 클래스 상호간의 유사도가 작아지므로, 클래스 상호간의 변별력이 증가한다는 점에 착안한 것이다. 특징벡터의 변별력을 개선하는 변환 행렬은 클래스 상호간의 상대 엔트로피에 대한 평균인 divergence[26-28]를 이용하여 목적 함수를 정의한 다음, 이 목적 함수를 반복 학습의 일종인 natural gradient ascent 방법[30-33]으로 최대화하여, 최적화된 선형 변환 행렬을 유도한다. 그리고 목적 함수를 최대화하여 추정된 선형 변환 행렬을 원시 특징벡터에 적용하여, 특징벡터의 변별력을 개선한다.

제안한 방법이 특징벡터의 변별력 개선과 음성 인식기의 성능 향상에 효과가 있는지를 검증하기 위해서, TIMIT 음성 데이터베이스[36]를 이용하여 음소에 대한 클러스터링 실험과 인식 실험을 수행한다. 그리고 실험 결과를 기존의 특징벡터 변환 방법인 주요 성분분석법, 선형 판별

분석법, Li의 방법과 비교 분석하여, 제안한 방법이 음성 인식기의 성능을 향상시킬 수 있음을 보인다.

본 논문은 다음과 같이 6장으로 구성 된다. 2장에서는 음성 인식에 사용되는 특징벡터 중에서 음성의 생성 기관인 성도 특성을 모델링한 선형 예측계수, 인간의 청각 특성을 모델링한 MFCC, 그리고 특징벡터의 시간적 변화량을 나타내는 미분계수의 추출방법에 관하여 설명한 후, 음성 인식기의 구현에 사용되는 알고리즘인 은닉 마르코프 모델에 관하여 서술한다. 또한 은닉 마르코프 모델을 이용하는 음성 인식 모델의 파라미터 추정방법에 관하여 논한다. 3 장에서는 특징 벡터의 변별력을 개선하기 위한 기존의 방법인 주요 성분분석, 선형 판별분석, 그리고 Li의 방법에 관하여 설명한다. 4 장에서는 이 논문에서 제안하는 방법인 상대 엔트로피를 이용한 특징벡터의 변별력 개선 방법에 관하여 서술한다. 5 장에서는 제안한 방법을 클러스터링 실험과 음성 인식 실험을 통해서, 기존의 방법과 비교 분석하고, 특징벡터의 변별력 개선과 음성 인식기의 성능 개선에 효과가 있는지를 검증한다. 마지막으로 6장에서는 이 논문을 정리하고 결론을 맺는다.

## 제 2 장 음성 인식과정

음성 인식과정은 일련의 음성 신호(speech signal)로부터 음성의 특



정을 나타내는 특징벡터를 추출하고 특징벡터를 이용하여 음성을 인식하는 과정으로 나눌 수 있으며, 그림 2.1과 같다.

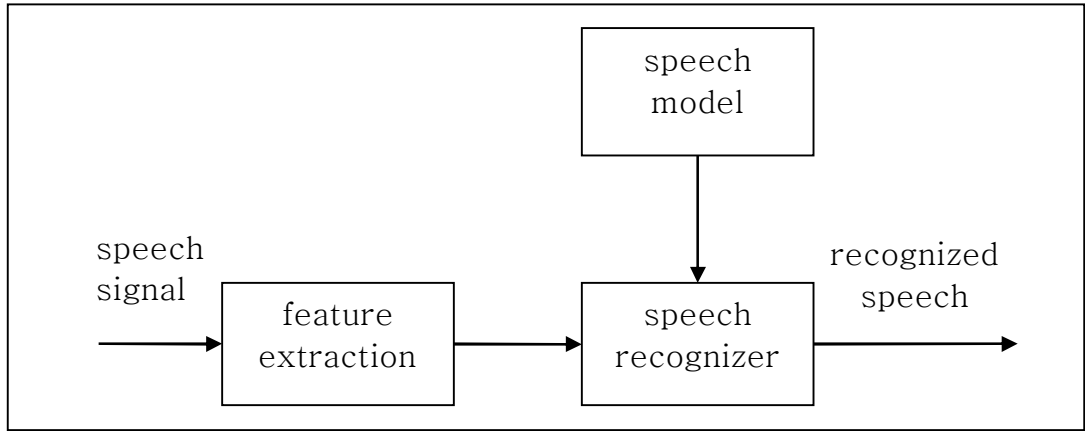


그림 2.1 음성 인식 과정  
Figure 2.1 Procedure of speech recognition

본 장에서는 먼저, 음성 인식에 주로 사용되는 특징벡터에 관하여 서술한 다음, 은닉 마르코프 모델을 이용하는 음성 인식 방법에 관하여 논한다.

## 2.1 음성 인식을 위한 특징벡터

음성 인식 분야에서 특징벡터는 일련의 음성 신호로부터 음향학적 특징을 추출한 것으로, 음성 인식기에 입력으로 사용되어 인식기의 성능은 향상시키고, 복잡도는 감소시키는 역할을 한다. 음성 인식기에 사용되는 특징벡터에는 정적인 특징벡터와 동적인 특징벡터가 있다.

정적인 특징벡터에서 대표적인 것으로는 음성 생성 기관을 모델링하여

추출한 것과, 음성 인지 기관을 모델링하여 추출한 것이 있다. 음성 생성 기관을 모델링하여 추출한 특징벡터에는 인간의 성도(vocal tract) 특성을 모델링한 선형 예측 계수가 있고, 음성 인지 기관을 모델링하여 추출한 특징벡터에는 인간의 청각 특성을 이용한 MFCC가 있다.

동적인 특징벡터는 임의의 시간에서 그 순간의 특성만을 나타내는 정적인 특징벡터와는 달리, 인접한 특징벡터 사이의 시간적 특성 변화를 반영한 벡터로, 대표적인 것에는 미분 계수[2,10,11]가 있다.

### 2.1.1 선형 예측 계수

선형 예측 계수는 음성 신호의 시간에 따른 변화의 특성을 나타내는 계수로서, 시간에 따른 성도의 변화 특성을 나타낸다. 이 계수는 음성 생성 시스템에서 현재의 음성 신호는 과거의 음성 신호들의 선형 조합에 의해서 근사화 할 수 있다는데 기반을 두고 있으며, 현재의 음성신호  $x(n)$ 은 식 (2.1)과 같다.

$$x(n) = - \sum_{k=1}^P a(k)x(n-k) + Gu(n) \quad (2.1)$$


여기서  $a(k)$ 는 선형 예측 계수이고,  $u(n)$ 은 현재의 여기신호(excitation signal)이고,  $G$ 는 여기신호의 이득이다. 그리고  $P$ 는 선형 예측 계수의 차수이다. 음성 생성 시스템의 전달 함수  $H(z)$ 는 식 (2.1)로부터 식 (2.2)와 같이 쓸 수 있다.

$$H(z) = \frac{X(z)}{U(z)} = \frac{1}{1 + \sum_{k=1}^P a(k)z^{-k}} \quad (2.2)$$

식 (2.1)의 음성 신호  $x(n)$ 은 과거의 음성 신호들의 선형 조합에 의해서 추정할 수 있으며, 추정 신호  $\hat{x}(n)$ 은 식 (2.3)과 같이 나타낼 수 있다.

$$\hat{x}(n) = - \sum_{k=1}^P a(k)x(n-k) \quad (2.3)$$

추정 신호  $\hat{x}(n)$ 을 실제 음성 신호  $x(n)$ 으로 근사화하기 위해서는 이들의 평균 자승 오차를 최소로 하는 선형 예측 계수  $a(k)$ 를 구하면 된다. 따라서 오차  $e(n)$ 은 식 (2.4)와 같이 나타낼 수 있고, 평균 자승 오차  $E$ 는 식 (2.5)와 같이 나타낼 수 있다.



$$e(n) = x(n) - \hat{x}(n) = x(n) + \sum_{k=1}^P a(k)x(n-k) \quad (2.4)$$

$$E = \sum_n e^2(n) = \sum_n \left[ x(n) + \sum_{k=1}^P a(k)x(n-k) \right]^2 \quad (2.5)$$

식 (2.5)의 평균 자승 오차를 최소 자승법을 이용하여 풀면 식 (2.6)와 같은 관계식이 유도되는데, 이 식을 자기 상관 계수(autocorrelation coefficient)를 이용해서 표현하면 식 (2.7)과 같이 표현된다.

$$\sum_{k=1}^P a(k) \sum_n x(n-k)x(n-i) = - \sum_n x(n)x(n-i), 1 \leq i \leq P \quad (2.6)$$

$$\sum_{k=1}^P a(k)R(i-k) = -R(i), 1 \leq i \leq P \quad (2.7)$$

식 (2.7)의 관계식에서 선형 예측 계수  $a(k)$ 는 Levinson-Durbin 알고리즘[7]으로 구할 수 있다.

### 2.1.2 MFCC

MFCC는 인간의 귀가 주파수 변화에 반응하는 양상이 선형적이지 않고, 로그 스케일과 비슷한 멜 스케일(mel scale)을 따른다는 청각적 특성을 반영한 계수이다. 멜 스케일은 Stevens에 의해서 연구되었으며, 1kHz의 60dB 기준 음을 1000 멜로 정의한다[37]. 그림 2.2의 멜 스케일에 따르면 인간의 귀가 낮은 주파수에서는 작은 변화에도 민감하게 반응하지만, 높은 주파수로 갈수록 민감도가 작아진다는 것을 알 수 있다. 또한 인간의 귀가 1kHz 이하의 주파수에서는 주파수 변화에 거의 선형적으로 반응하지만, 그 이상의 주파수에서는 비선형적으로 반응한다는 것을 알 수 있다.

Davis 등[12]은 멜 스케일을 이용하여 MFCC를 추출하는 과정에 앞서 언급한 멜 스케일의 특성을 반영하기 위해서 그림 2.3와 같은 삼각 필터들로 구성된 멜 스케일 필터 뱅크를 이용하였다. 이들은 그림 2.3에서는 보이는 바와 같이 1kHz 이하의 주파수에서는 선형적인 특성을 보

존하기 위해 대역폭이 균일한 필터들로 필터 뱅크를 구성하였고, 그 이상의 주파수에서는 비선형적인 특성을 반영하기 위해 대역폭이 넓어지는 필터들로 필터 뱅크를 구성하였다.

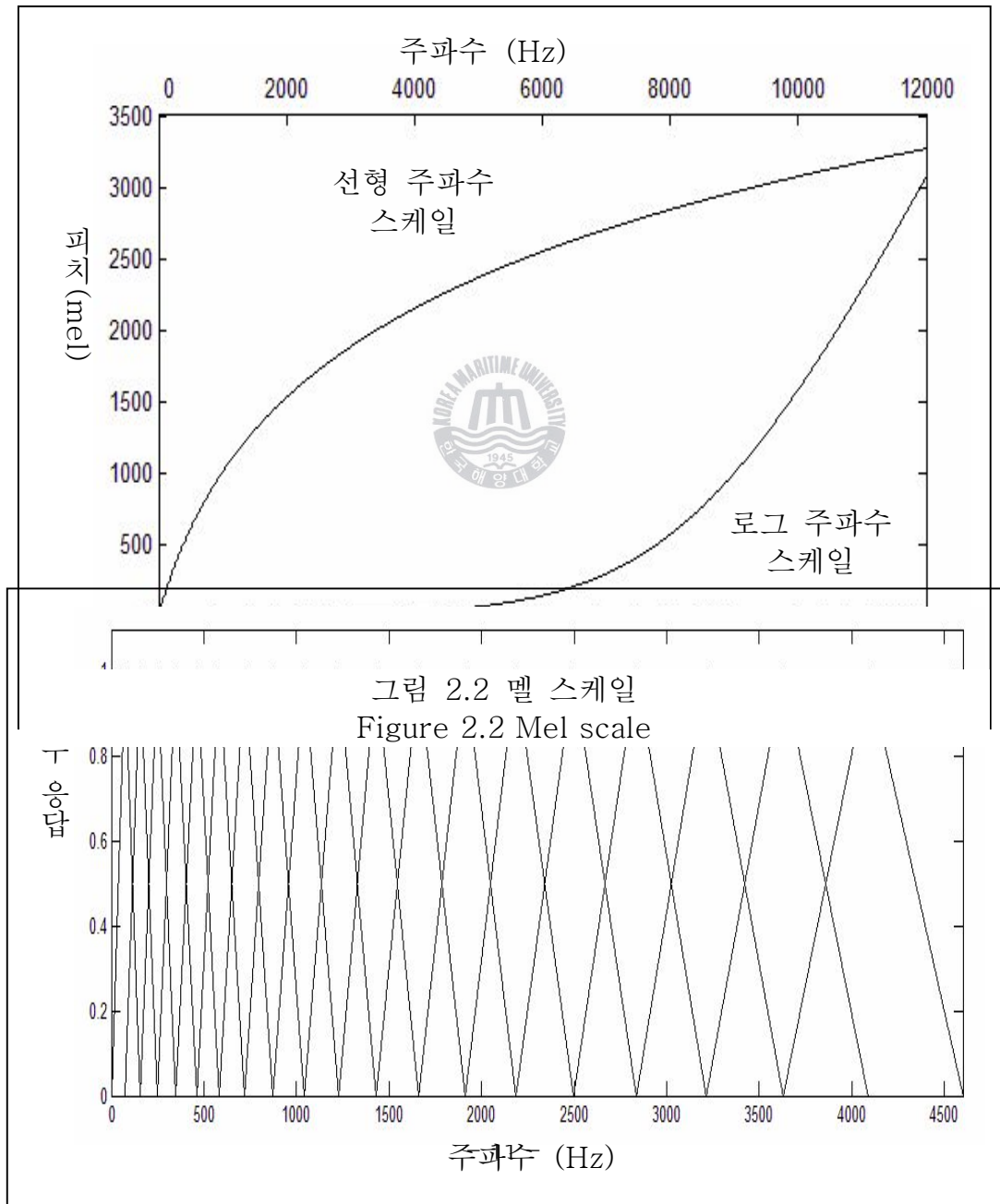


그림 2.3 멜 스케일 필터 뱅크  
Figure 2.3 Mel scale filter bank

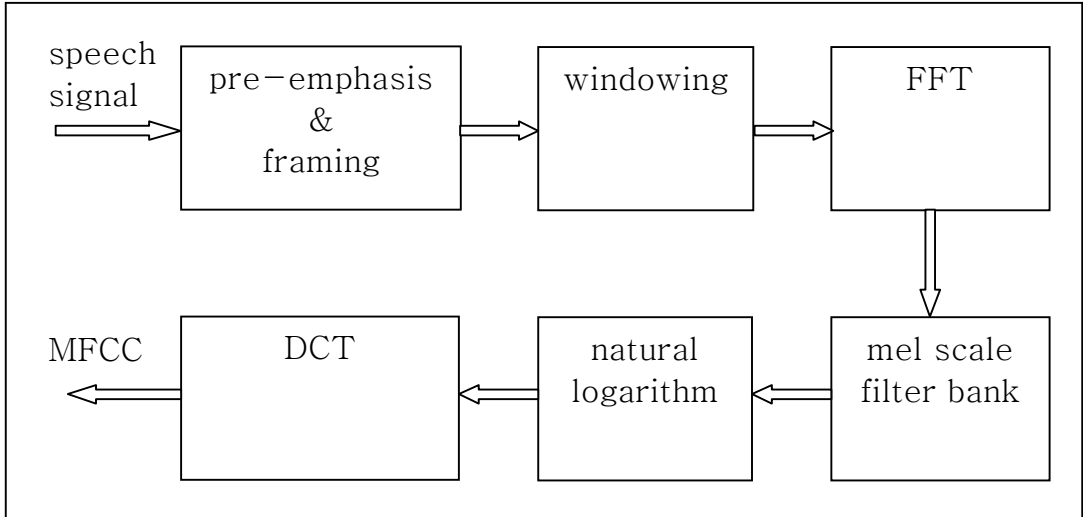


그림 2.4 MFCC 추출 과정  
Figure 2.4 Procedure of MFCC extraction

MFCC를 추출하는 일반적인 과정은 그림 2.4에 보이는 것과 같이 여섯 단계로 구성된다. 첫 번째 과정에서는 입력된 음성 신호(speech signal)를 식 (2.8)과 같은 고대역 통과 특성을 갖는 pre-emphasis 필터에 적용한 다음, 필터링된 일련의 음성 신호들을 일정한 길이를 갖는 프레임 단위의 신호들로 나눈다. 이 과정에서 사용하는 pre-emphasis 필터는 낮은 주파수의 에너지는 감쇠시키고, 높은 주파수의 에너지는 상대적으로 증폭시키는 역할을 한다.

$$H(z) = 1 - \alpha z^{-1}, \quad 0.9 \leq \alpha \leq 1.0 \quad (2.8)$$

MFCC추출을 위한 두 번째 과정에서는 프레임 단위의 음성 신호들에 창 함수(window function)를 적용하여, 프레임의 시작과 끝 부분에서 발생

하는 신호의 불연속성을 최소화한다. 이 과정에서 일반적으로 사용되는 창 함수는 Hamming 창 함수로, 식 (2.9)와 같다.

$$w(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right), \quad 0 \leq n \leq N-1 \quad (2.9)$$

여기서  $N$ 은 프레임 길이를 나타낸다. 세 번째 과정은 두 번째 과정을 거친 프레임 단위의 음성 신호들을 FFT(Fast Fourier Transform)하여, 주파수 영역의 신호로 변환하는 과정이다. 네 번째 과정에서는 FFT로부터 얻은 주파수 영역의 신호에서 파워를 구한 다음, 이 파워에 그림 2.3에 보인 멜 스케일 필터 बैं크를 적용하여, 멜 스케일 파워 스펙트럼을 구하며, 멜 스케일 파워 스펙트럼  $Y(m)$ 은 식 (2.10)와 같다.



$$Y(m) = \sum_{k=0}^{N/2} |X(k)|^2 H_m(k) \quad (2.10)$$

여기서  $N$ 은 FFT 개수이고,  $X(k)$ 는 FFT로부터 얻은 주파수 영역의 신호이고,  $H_m(k)$ 는 필터 बैं크 중에서  $m$ 번째 필터를 나타낸다. 다섯 번째 과정은 멜 스케일 파워 스펙트럼에 자연 로그(natural log)를 취하는 과정이다. 마지막으로 여섯 번째 과정에서는 로그화된 멜 스케일 파워 스펙트럼  $Y(m)$ 을 DCT(Discrete Cosine Transform)하여 MFCC를 추출하며, MFCC  $c(n)$ 은 식 (2.11)과 같다.

$$c(n) = \sum_{m=1}^M (\log Y(m)) \cos \left[ n(m - 0.5) \frac{\pi}{M} \right], 1 \leq n \leq L \quad (2.11)$$

여기서  $M$ 은 필터 बैं크에서 총 필터의 개수를 나타내고,  $L$ 은 MFCC의 차원을 나타낸다.

### 2.1.3 미분 계수

앞서 언급한 선형 예측 계수나 MFCC와 같은 정적인 특징벡터는 임의의 시간에서 그 순간의 특성만을 나타내지만, 미분 계수는 인접 특징벡터들 사이의 시간적인 특성 변화를 잘 나타내는 계수로 식 (2.12)를 이용해서 구한다.

$$\nabla c(t) = \frac{\sum_{k=1}^K k[c(t+k) - c(t-k)]}{2 \sum_{k=1}^K k^2} \quad (2.12)$$

식 (2.12)에서  $K$ 는 미분할 구간의 길이를 나타내고,  $c(t)$ 는 정적인 특징벡터를 나타낸다.

## 2.2 은닉 마르코프 모델을 이용한 음성인식

이 절에서는 먼저 마르코프 프로세스(Markov process)에 관하여 간단히 서술한 후에, 은닉 마르코프 모델(Hidden Markov Model: HMM)에 관하여 서술한다.



## 2.2.1 마르코프 프로세스

마르코프 프로세스이란 시간 영역의 순차적인 사건들에 대해 과거와 현재의 사건들이 주어졌을 때, 현재 사건의 조건부 확률(conditional probability)이 가장 최근  $N$ 개의 사건들에 영향을 받는다는 조건을 만족하는 확률 프로세스를 말하며, 이를  $N$ 차 마르코프 프로세스라 부른다.

확률 프로세스  $\{X(t)\}$ 가 있을 때, 시간  $t_1, t_2, \dots, t_N$ 에서의 관측치를  $x_1, x_2, \dots, x_N$ 이라 하면, 1차 마르코프 프로세스의 조건부 확률은 체인 법칙에 의해 식 (2.13)과 같이 정의된다[2,4,5].

$$P(x_N | x_{N-1}, x_{N-2}, \dots, x_1) = P(x_N | x_{N-1}) \quad (2.13)$$

그리고 마르코프 프로세스에서 시간  $t$ 와 상태 공간이 이산적일 때, 이 프로세스를 마르코프 체인(Markov chain)이라 한다. 확률 프로세스  $\{X[n]\}$ 이 있을 때 시간  $1, 2, \dots, N$ 에서의 상태가  $s_1, s_2, \dots, s_N$ 이라 하면, 1차 마르코프 체인의 조건부 확률은 식 (2.14)와 같이 나타낼 수 있다.

$$P(s_N | s_{N-1}, s_{N-2}, \dots, s_1) = P(s_N | s_{N-1}) \quad (2.14)$$

식 (2.14)에서  $P(s_N | s_{N-1})$ 은  $N-1$ 시간에  $s_{N-1}$ 이라는 상태에 있다가  $N$

시간에  $s_N$ 이라는 상태로 천이할 확률이고, 이는 현재의 상태가 바로 이전의 상태에만 의존한다는 것을 의미한다. 그림 2.5는 3개의 상태  $\{s_1 = 1, s_2 = 2, s_3 = 3\}$ 를 갖는 1차 마르코프 체인의 상태 천이도를 나타낸 것이다. 이 그림에서  $a_{ij}$ 는  $i$ 라는 상태에서  $j$ 라는 상태로 천이할 확률이다. 그리고 상태 천이 확률  $a_{ij}, 1 \leq i, j \leq 3$ 로 구성된 상태 천이행렬  $\mathbf{A}$ 는 식 (2.15)과 같다.

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \quad (2.15)$$

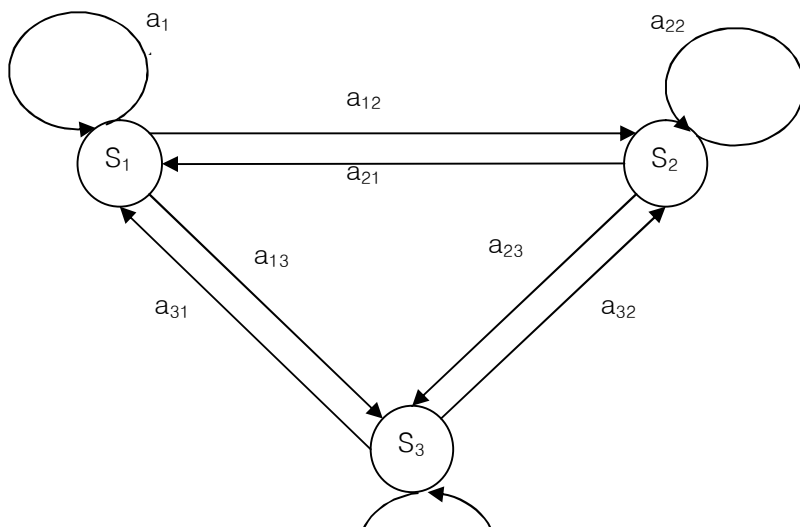


그림 2.5 마르코프 체인의 상태 천이도

Figure 2.5 State transition diagram of Markov chain

## 2.2.2 은닉 마르코프 모델

마르코프 모델은 시간에 따른 상태 천이 정보와 각 상태에서 출력되는 심볼을 알 수 있는 마르코프 프로세스이다. 그리고 마르코프 모델에서 단지 시간에 관계하는 출력만을 알 수 있을 때, 이를 은닉 마르코프 모델이라 한다. 이 모델에서는 관측 확률에 의해서 발생하는 심볼(출력)만 알 수 있고, 상태 천이에 관한 정보는 은닉되어 있어, 상태 천이는 출력 심볼을 통해서 간접적으로 추정하여야만 한다.

은닉 마르코프 모델은 관측 확률의 형태에 따라 그림 2.6의 이산형 은닉 마르코프 모델(Discrete Hidden Markov Model: DHMM)과 그림 2.7의 연속형 은닉 마르코프 모델(Continuous Hidden Markov Model: CHMM)로 나눌 수 있다. 이산형 은닉 마르코프 모델에서는 먼저 벡터 양자화를 통해 특징벡터를 양자화한 다음, 특징벡터의 양자화 코드를 모델의 입력으로 사용한다. 따라서 이산형 은닉 마르코프 모델의 관측 확률은 학습 시 사용된 코드들의 빈도를 나타내는 히스토그램으로 나타나게 되어, 학습 시 한번도 관측되지 않은 코드에 대해서는 관측 확률이 '0' 이 되거나 아주 작은 기본 값으로 되는 문제를 야기시킨다.

연속형 은닉 마르코프 모델에서는 각 상태에서 관측되는 값이 다차원의 연속된 벡터이므로, 관측 확률이 정규분포의 형태를 갖는다. 따라서 각 모델은 평균과 공분산 등의 많은 파라미터를 포함하게 되어, 학습 시 많은 데이터가 필요하고, 계산량이 많아지는 단점이 있다. 그러나 모델을 보다 정확하게 추정할 수 있어, 높은 성능을 기대할 수 있다.

또한 은닉 마르코프 모델은 상태 천이의 구조에 따라 Left-To-

Right 모델과 Ergodic 모델로 나누어 진다. Left-To-Right 모델은 시간에 따른 상태 천이가 오른쪽에서 왼쪽으로만 이루어지는 모델이고, Ergodic 모델은 모든 상태가 완전히 연결되어 있어 모든 상태로 천이가 가능한 모델이다. 그림 2.8과 2.9는 Left-To-Right 모델과 Ergodic 모델을 나타낸다.

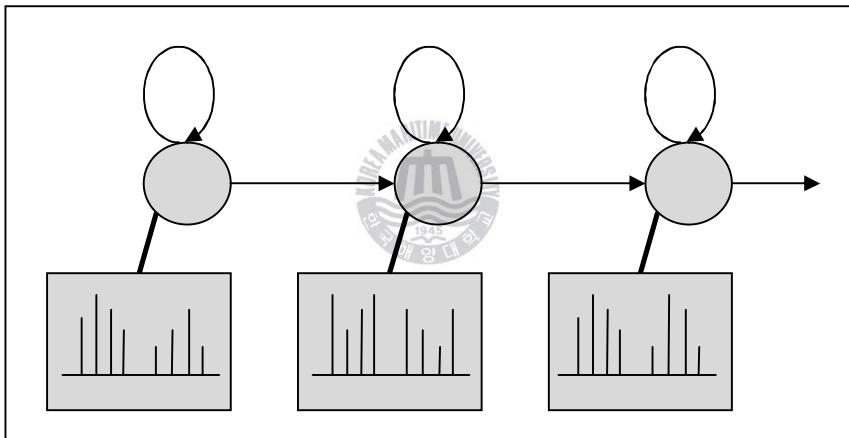


그림 2.6 이산형 은닉 마르코프 모델  
Figure 2.6 Discrete hidden Markov model

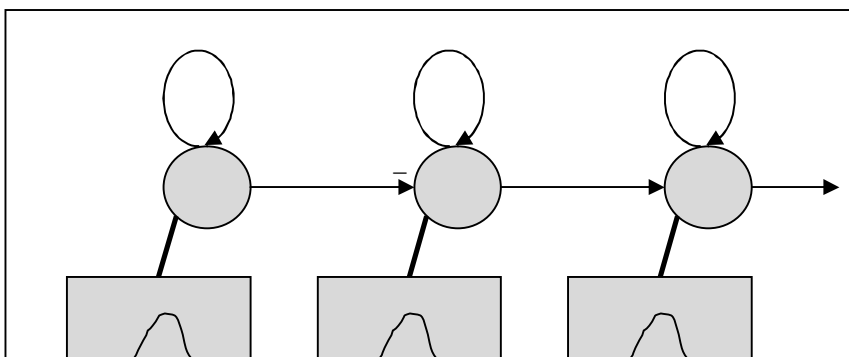


그림 2.7 연속형 은닉 마르코프 모델  
Figure 2.7 Continuous hidden Markov model

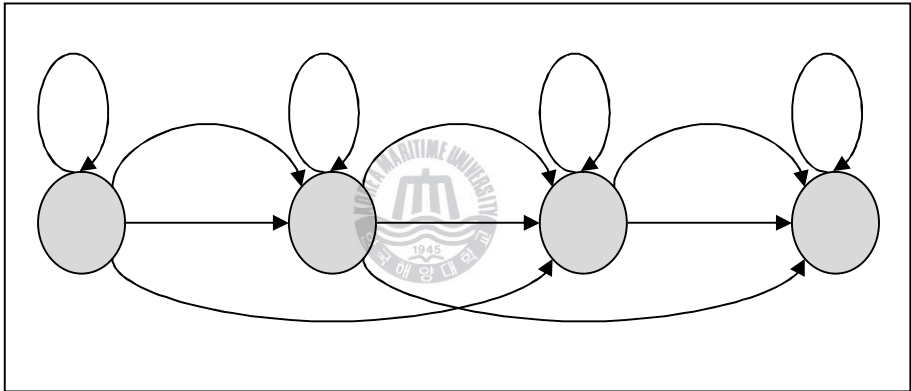


그림 2.8 Left-To-Right 은닉 마르코프 모델  
Figure 2.8 Left-To-Right hidden Markov model

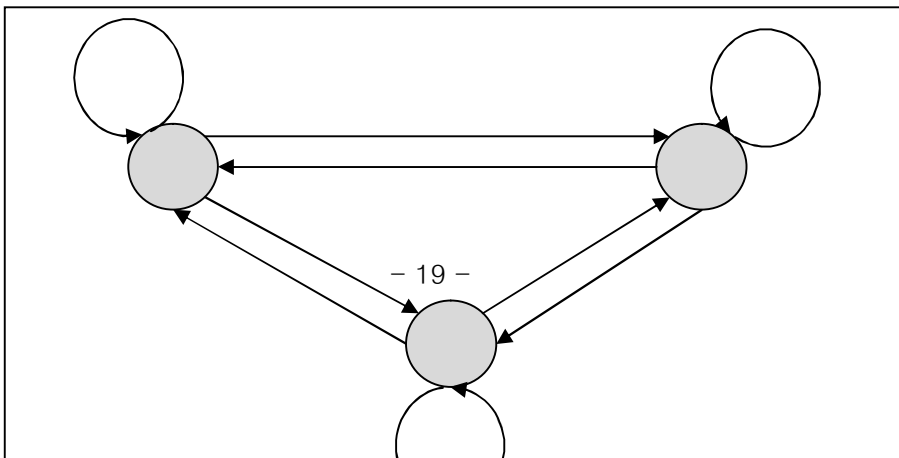


그림 2.9 Ergodic 은닉 마르코프 모델  
Figure 2.9 Ergodic hidden Markov model

음성 인식에서는 음성 신호가 발성구조의 시간적 변화에 의해서 발생된 신호이므로, 1차 마르코프 모델에 의해서 발생한다고 가정하여, 상태 천이가 한쪽 방향으로만 이루어지는 Left-To-Right형의 은닉 마르코프 모델을 주로 사용한다. 그리고 은닉 마르코프 모델을 음성 인식에 사용하기 위해서는 학습 과정과 인식 과정이 필요하다. 학습 과정에서는 모델의 파라미터를 추정하고, 인식 과정에서는 학습 과정에서 추정한 파라미터를 이용하여, 미지의 입력 음성에 가장 적합한 모델을 찾는다. 즉, 학습 과정은 성도나 청각의 특성이 반영된 특징벡터를 이용하여 각 인식 단위에 해당하는 모델을 구축하는 것을 말하고, 인식 과정은 입력된 특징벡터에 가장 적합한 인식 모델을 찾는 것을 말한다.

은닉 마르코프 모델  $\lambda$ 는 식 (2.16)과 같이 초기 상태를 나타내는 확률 벡터  $\pi$ 와 상태 천이를 나타내는 천이 행렬  $A$ , 그리고 각 상태의 관측 확률 밀도를 구성하는 요소를 나타내는  $B$ 로 구성된다.

$$\lambda = \{\pi, A, B\} \tag{2.16a}$$

$$\boldsymbol{\pi} = \{\pi_i\}, \quad \pi_i = \begin{cases} 1 & i = 1 \\ 0 & 2 \leq i \leq N \end{cases} \quad (2.16b)$$

$$\mathbf{A} = \{a_{ij}\}, \quad 1 \leq i, j \leq N, \quad \sum_{j=1}^N a_{ij} = 1 \quad (2.16c)$$


$$\mathbf{B} = \{b_i\}, \quad b_i = \{\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i\}, \quad 1 \leq i \leq N \quad (2.16d)$$

식 (2.16)에서  $N$ 은 모든 상태의 개수이고, 식(2.16b)의  $\pi_i$ 는  $i$  상태에서의 초기 확률이다. 식 (2.16c)의  $a_{ij}$ 는 현재 상태  $i$ 에서 다음 상태  $j$ 로 천이할 확률이며,  $\sum_{j=1}^N a_{ij} = 1$ 은 상태  $i$ 에서 모든  $j$ 의 상태로 천이할 확률의 합이 '1'이 되어야 함을 의미한다. 식 (2.16d)의  $b_i$ 는 상태  $i$ 에서의 관측 확률 밀도를 구성하는 요소로, 이것은 상태  $i$ 에서의 평균  $\boldsymbol{\mu}_i$ 와 공분산  $\boldsymbol{\Sigma}_i$ 의 집합이다. 그리고 시간  $t$ 에서 관측된 특징벡터가  $\mathbf{o}_t$ 일 때, 상태  $i$ 에서의 관측 확률 밀도  $b_i(\mathbf{o}_t)$ 는 식 (2.17)과 같다.

$$b_i(\mathbf{o}_t) = \frac{1}{\sqrt{(2\pi)^M |\boldsymbol{\Sigma}_i|}} \exp \left[ -\frac{1}{2} (\mathbf{o}_t - \boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1} (\mathbf{o}_t - \boldsymbol{\mu}_i) \right] \quad (2.17)$$

여기서  $M$ 은 특징벡터의 차원이다.

식 (2.16)로 정의되는 인식 단위에 대한 은닉 마르코프 모델  $\lambda$ 는 학습 데이터로부터 Baum-Welch 알고리즘 [2,4,5]을 이용해서 추정할 수 있다. Baum-Welch 알고리즘은 모델  $\lambda$ 를 추정하기 위해 전향 확률  $\alpha_t(i)$ 와 후향 확률  $\beta_t(i)$ 를 이용하는데, 추정 방법은 다음과 같다. 상태의 개수가  $N$ 인 모델  $\lambda$ 와  $T$ 개의 심볼(특징벡터)로 구성된 심볼열  $\mathbf{O} = \{\mathbf{o}_1, \mathbf{o}_2, \dots, \mathbf{o}_T\}$ 가 주어졌을 때, 전향 확률  $\alpha_t(i)$ 와 후향 확률  $\beta_t(i)$ 는 각각 식 (2.18)과 식 (2.19)와 같다.

$$\begin{aligned} \alpha_t(i) &= P(\mathbf{o}_1, \mathbf{o}_2, \dots, \mathbf{o}_t, q_t = i \mid \lambda) \\ &= \left[ \sum_{j=1}^N \alpha_{t-1}(j) a_{ji} \right] b_i(\mathbf{o}_t) \end{aligned} \quad (2.18)$$


$$\begin{aligned} \beta_t(i) &= P(\mathbf{o}_{t+1}, \mathbf{o}_{t+2}, \dots, \mathbf{o}_T \mid q_t = i, \lambda) \\ &= \sum_{j=1}^N a_{ij} b_j(\mathbf{o}_{t+1}) \beta_{t+1}(j) \end{aligned} \quad (2.19)$$

여기서  $\alpha_t(i)$ 는 심볼열  $\mathbf{o}_1, \mathbf{o}_2, \dots, \mathbf{o}_t$ 를 시간에 따라 생성하고  $t$ 라는 시간에 상태  $i$ 에 도달하는 확률이고,  $\beta_t(i)$ 는  $t+1$  시간에 상태  $j$ 에서 시작해



서 시간의 흐름에 따른 상태 천이에 의해서 심볼열  $\mathbf{o}_{t+1}, \mathbf{o}_{t+2}, \dots, \mathbf{o}_T$ 를 생성하는 확률이며, 이들을 그림 2.10에 도시한다. 모델  $\lambda$ 에서 심볼열  $\mathbf{O} = \{\mathbf{o}_1, \mathbf{o}_2, \dots, \mathbf{o}_T\}$ 가 관측될 확률  $P(\mathbf{O} | \lambda)$ 와 전향 확률  $\alpha_t(i)$  그리고 후향 확률  $\beta_t(i)$ 를 이용해서,  $t$  시간에 상태  $i$ 에 있고  $t+1$  시간에는 상태  $j$ 에 있을 확률  $\xi_t(i, j)$ 를 식 (2.20)과 같이 정의하고,  $t$  시간에 상태  $i$ 에 있을 확률  $\gamma_t(i)$ 를 식 (2.21)이라 하면, 상태 천이 확률  $a_{ij}$ 와 관측 확률을 구성하는 요소  $b_i = \{\mu_i, \Sigma_i\}$ 의 재추정 식은 식 (2.22) - (2.24)와 같이 된다.

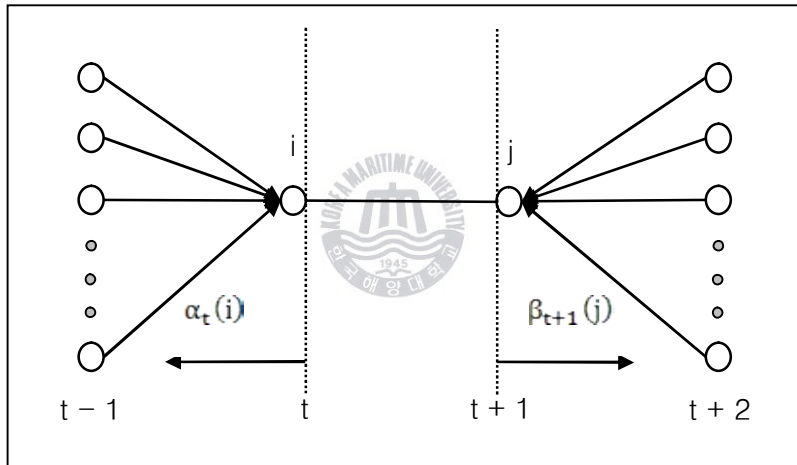


그림 2.10 전향 확률 및 후향 확률

Figure 2.10 Forward probability and backward probability

$$\begin{aligned} \xi_t(i, j) &= \frac{P(q_t = i, q_{t+1} = j, \mathbf{O} | \lambda)}{P(\mathbf{O} | \lambda)} \\ &= \frac{\alpha_t(i) a_{ij} b_j(\mathbf{o}_{t+1}) \beta_{t+1}(j)}{P(\mathbf{O} | \lambda)} \end{aligned} \tag{2.20}$$

$$\gamma_t(i) = \sum_{j=1}^N \xi_t(i, j) \quad (2.21)$$

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \quad (2.22)$$

$$\bar{\boldsymbol{\mu}}_j = \frac{\sum_{t=1}^T \gamma_t(j) \mathbf{o}_t}{\sum_{t=1}^T \gamma_t(j)} \quad (2.23)$$

$$\bar{\boldsymbol{\Sigma}}_j = \frac{\sum_{t=1}^T \gamma_t(j) (\mathbf{o}_t - \boldsymbol{\mu}_j)(\mathbf{o}_t - \boldsymbol{\mu}_j)^T}{\sum_{t=1}^T \gamma_t(j)} \quad (2.24)$$

식 (2.22) - (2.24)의 재추정 식을 반복해서 수행하면,  $P(\mathbf{O} | \boldsymbol{\lambda})$ 가 최대 값을 가지는  $\mathbf{a}_{ij}$ 와  $b_i = \{\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i\}$ 에 도달한다.

## 제 3 장 특징벡터의 변별적 변환

이 장에서는 먼저 특징벡터의 변별적 변환을 이용하여 음성을 인식하는 과정을 설명한 후, 기존의 변별적 변환방법인 주요 성분분석과 선형 판별분석, 그리고 Li의 방법에 대하여 서술한다.

### 3.1 특징벡터의 변별적 변환을 이용한 음성 인식과정

특징벡터의 변별적 변환을 이용하여 음성을 인식하는 과정은 그림 3.1과 같다.

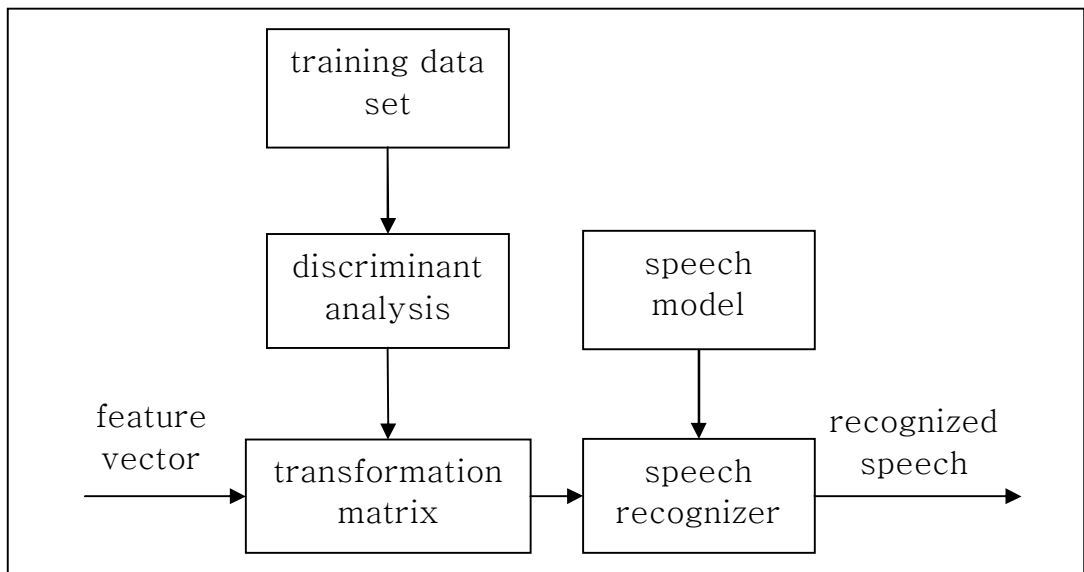


그림 3.1 특징벡터의 변별적 변환에 의한 음성 인식과정  
Figure 3.1 Procedure of speech recognition by discriminative transformation of feature vectors

그림 3.1에서는 먼저, 학습용 특징벡터의 집합인 training data set을 선형 판별분석 방법, 주요 성분분석 방법, Li의 방법, 그리고 본 논문에서 제안하는 방법등과 같은 특징벡터의 변별력 개선 방법(discriminant analysis)에 적용하여 변환 행렬(transformation matrix)을 구한다. 그리고 앞서 구한 변환 행렬에 음성 신호로부터 추출한 특징벡터(feature vector)를 적용하여 특징벡터의 변별력을 개선한다. 마지막으로 변별력이 개선된 특징벡터를 음성 인식기에 인가하여, 음성을 인식한다.

### 3.2 주요 성분분석



주요 성분분석은 데이터의 차원을 감소시킬 수 있는 선형 변환의 일종으로 원시 데이터와 변환된 데이터 사이의 오차를 최소화하는 주요 성분 벡터들로 이루어진 변환 행렬을 찾는 방법이다. 여기서 각 주요 성분 벡터들은 크기가 '1' 이고 서로 직교해야 한다는 속성을 만족해야 한다. 주요 성분분석에서 사용하는 목적 함수  $J(\mathbf{w})$ 는 식 (3.1)과 같다.

$$\begin{aligned}
J(\mathbf{W}) &= E \left\{ \left\| \mathbf{x} - \sum_{i=1}^M \mathbf{w}_i^T \mathbf{x} \mathbf{w}_i \right\|^2 \right\} \\
&= E\{\|\mathbf{x}\|^2\} - E \left\{ \sum_{i=1}^M (\mathbf{w}_i^T \mathbf{x})^2 \right\} \\
&= \text{trace}(\mathbf{C}_x) - \sum_{i=1}^M \mathbf{w}_i^T \mathbf{C}_x \mathbf{w}_i
\end{aligned} \tag{3.1}$$

식 (3.1)에서  $\mathbf{x}$ 는 원시 데이터를,  $M$ 은 찾고자 하는 주성분 벡터의 개수를,  $\mathbf{W}$ 는 주요 성분벡터들로 이루어진 변환 행렬을,  $\mathbf{w}_i$ 는 변환 행렬  $\mathbf{W}$ 의  $i$ 번째 주요 성분벡터를,  $\mathbf{w}_i^T \mathbf{x} \mathbf{w}_i$ 는 원시 데이터  $\mathbf{x}$ 를  $i$ 번째 주요 성분벡터를 이용해서 변환한 데이터를,  $\mathbf{C}_x$ 는 원시 데이터  $\mathbf{x}$ 의 공분산 행렬을 나타낸다. 식 (3.1)에서 오차를 최소화하기 위해서는  $\sum_{i=1}^M \mathbf{w}_i^T \mathbf{C}_x \mathbf{w}_i$ 를 최대화하면서 각각의 크기가 '1'이고 서로 직교하는 주요 성분벡터  $\mathbf{w}_i, 1 \leq i \leq M$ 를 구하면 된다. 즉, 이는 원시 데이터  $\mathbf{x}$ 의 공분산 행렬  $\mathbf{C}_x$ 에서 분산이 가장 큰 것부터 순차적으로  $M$ 개의 큰 분산을 추출할 수 있는 주요성분 벡터  $\mathbf{w}_i$ 를 찾는 것을 의미하며, 주요 성분벡터가 고유벡터임을 의미한다.

그림 3.2는 2차원의 데이터 집합에서의 주요 성분벡터를 도시한 예로,  $\mathbf{u}_1$ 은 첫 번째 주요 성분 벡터이고,  $\mathbf{u}_2$ 는 두 번째 주요 성분벡터이며,  $\bar{\mathbf{x}}$ 는 데이터의 평균 벡터이다.

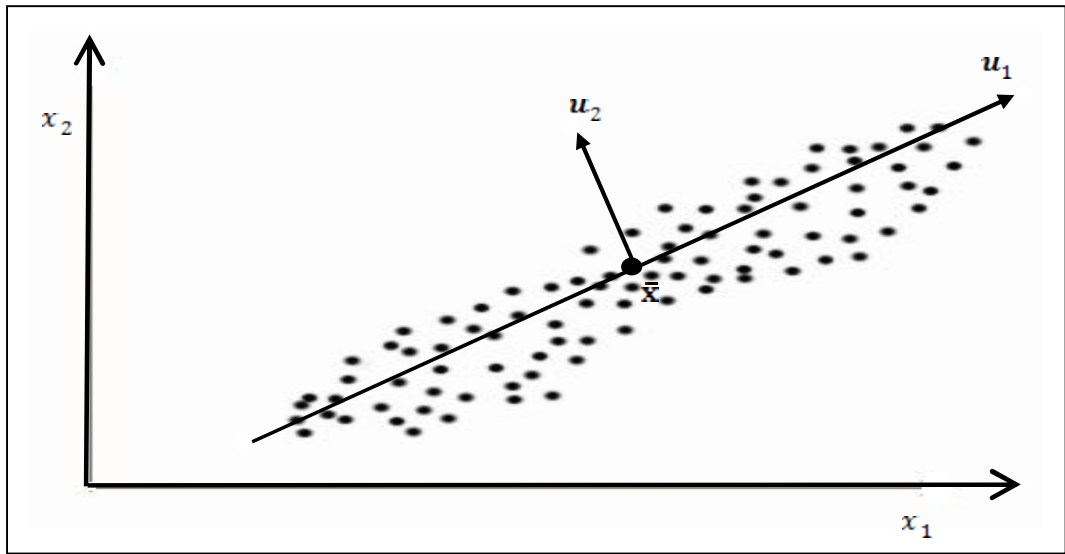


그림 3.2 주요 성분분석  
 Figure 3.2 Principal component analysis

이 논문에서는 식(3.1)을 최소화하는 변환 행렬  $W$ 을 구하기 위해 데이터의 공분산 행렬로부터 고유치와 고유벡터들을 구한 후에, 고유치들을 내림차순으로 정렬하여 각 고유치에 해당하는 고유벡터들로 변환 행렬  $W$ 를 구성하였다. 그리고 변환 행렬  $W$ 를 원시 데이터에 적용하여, 변별력이 향상된 데이터를 구했다.

### 3.3 선형 판별 분석

선형 판별분석은 주요 성분분석과 달리 데이터의 클래스 정보를 토대로, 클래스와 클래스 사이의 거리는 멀게 하고, 클래스 내부 거리는 가깝게 하여, 데이터의 변별력을 증가시킴과 동시에 자신의 클래스를 오인하게 하는 데이터 성분의 양을 최소화하는 변환 행렬을 찾는 방법이다. 또한 주요성분 분석과 마찬가지로 데이터의 차원을 감소시킬 수 있는 방

법이다. 이 방법에서는 클래스와 클래스 사이의 공분산 행렬(covariance matrix of between class)과 클래스 자신에 속한 모든 데이터에 대한 공분산 행렬(covariance matrix of within class)의 비인 Fisher ratio를 목적 함수로 사용하며, 이 함수를 최대화해서 변환 행렬을 구한다. 선형 변환된 데이터에 대한 Fisher ratio는 다음의 정의들을 사용하여 정의한다.

원시 데이터 집합  $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ 와 원시 데이터에 대한 클래스 집합  $\mathbf{c} = \{c_1, c_2, \dots, c_M\}$ 가 주어졌을 때, 변환 행렬  $\mathbf{W}$ 에 의해서 선형 변환된 데이터 집합이  $\mathbf{Y} = \mathbf{W}^T \mathbf{X} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N\}$ 이고, 변환된 데이터의 클래스 정보가 원시 데이터의 클래스 정보와 동일하다고 하자. 이때 원시 데이터의 클래스 자신에 대한 공분산 행렬  $\mathbf{S}_W$ 는 식 (3.2)와 같이 정의되고, 클래스 상호간에 대한 공분산 행렬  $\mathbf{S}_B$ 는 식 (3.3)과 같이 정의된다.



$$\mathbf{S}_W = \sum_{i=1}^M \sum_{\mathbf{x}_i \in c_i} (\mathbf{x}_i - \boldsymbol{\mu}_{c_i})(\mathbf{x}_i - \boldsymbol{\mu}_{c_i})^T \quad (3.2)$$

$$\mathbf{S}_B = \sum_i^M N_{c_i} (\boldsymbol{\mu}_{c_i} - \boldsymbol{\mu})(\boldsymbol{\mu}_{c_i} - \boldsymbol{\mu})^T \quad (3.3)$$

또한 선형 변환된 데이터의 클래스 자신에 대한 공분산 행렬  $\overline{\mathbf{S}}_W$ 는 식 (3.4)와 같이 정의되고, 클래스 상호간에 대한 공분산 행렬  $\overline{\mathbf{S}}_B$ 는 식

(3.5)와 같이 정의 된다.

$$\begin{aligned}
 \overline{S}_W &= \sum_{i=1}^M \sum_{\mathbf{x}_i \in c_i} \mathbf{W}^T (\mathbf{x}_i - \boldsymbol{\mu}_{c_i}) (\mathbf{x}_i - \boldsymbol{\mu}_{c_i})^T \mathbf{W} \\
 &= \mathbf{W}^T \left[ \sum_{i=1}^M \sum_{\mathbf{x}_i \in c_i} (\mathbf{x}_i - \boldsymbol{\mu}_{c_i}) (\mathbf{x}_i - \boldsymbol{\mu}_{c_i})^T \right] \mathbf{W} \\
 &= \mathbf{W}^T \mathbf{S}_W \mathbf{W}
 \end{aligned} \tag{3.4}$$

$$\begin{aligned}
 \overline{S}_B &= \sum_i^M N_{c_i} \mathbf{W}^T (\boldsymbol{\mu}_{c_i} - \boldsymbol{\mu}) (\boldsymbol{\mu}_{c_i} - \boldsymbol{\mu})^T \mathbf{W} \\
 &= \mathbf{W}^T \left[ \sum_i^M N_{c_i} (\boldsymbol{\mu}_{c_i} - \boldsymbol{\mu}) (\boldsymbol{\mu}_{c_i} - \boldsymbol{\mu})^T \right] \mathbf{W} \\
 &= \mathbf{W}^T \mathbf{S}_B \mathbf{W}
 \end{aligned} \tag{3.5}$$

식 (3.2) - (3.5)에서  $\boldsymbol{\mu}_{c_i}$ 는  $i$ 번째 클래스에 속한 원시 데이터의 평균이고,  $\boldsymbol{\mu}$ 는 모든 원시 데이터의 평균이며,  $N_{c_i}$ 는  $i$ 번째 클래스에 속한 원시 데이터의 개수이다.

선형 변환된 데이터에 대한 Fisher ratio는 식 (3.5)를 식(3.4)로 나눈 것으로, 식 (3.6)과 같이 정의 된다.



$$J(W) = \frac{|\overline{S_B}|}{|\overline{S_W}|} = \frac{|W^T S_B W|}{|W^T S_W W|} \quad (3.6)$$

이 논문에서는 식 (3.6)의 Fisher ratio를 최대화하는 변환 행렬  $W$ 를  $S_W^{-1} S_B$ 의 고유치와 고유벡터를 이용하여 구한다.

### 3.4 Li의 방법

Li는 음성 인식기의 인식률을 향상시키기 위하여 인식기의 클래스 정보(인식 단위 또는 HMM의 상태)를 이용하여 인식기에 인가되는 특징 벡터의 변별력을 개선하는 선형 변환 방법을 제안하였다. 그는 제안한 방법에서 특징벡터의 클래스 정보에 기반한 클래스와 특징벡터 사이의 조건부 확률 밀도  $P(\mathbf{x}_i | c_j)$ 가 식 (3.7)과 같이 가우시안 분포를 가진다고 가정하여, 정규화된 우도 함수(normalized likelihood function)  $P_c(\mathbf{x}_i)$ 를 식 (3.8)로 정의한 다음, 이 식을 특징벡터의 변별력 개선에 이용하였다.

$$P(\mathbf{x}_i | c_j) = \frac{1}{\sqrt{(2\pi)^N |\Sigma_j^x|}} \exp\left[-\frac{1}{2}(\mathbf{x}_i - \mu_j^x)^T (\Sigma_j^x)^{-1} (\mathbf{x}_i - \mu_j^x)\right] \quad (3.7)$$

$$P_c(\mathbf{x}_i) = \frac{P(\mathbf{x}_i | c_{f(\mathbf{x}_i)})}{\sum_{j=1}^M P(\mathbf{x}_i | c_j)} \quad (3.8)$$

여기서  $\mathbf{x}_i$ 는 특징 벡터를,  $N$ 은 특징벡터의 차원을,  $M$ 은 총 클래스의 개

수를,  $c_j$ 는  $j$ 번째 클래스를,  $\Sigma_j^x$ 는  $j$ 번째 클래스의 공분산 행렬을,  $\mu_j^x$ 는  $j$ 번째 클래스의 평균을,  $f(\mathbf{x}_i)$ 는 특징 벡터  $\mathbf{x}_i$ 가 속한 클래스의 색인을 찾는 함수를 나타낸다.

원시 특징벡터의 집합  $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_T\}$ 와 원시 특징벡터에 대한 클래스 정보  $\mathbf{c} = \{c_1, c_2, \dots, c_M\}$  그리고 변환 행렬  $\mathbf{W}$ 가 주어지고, 선형 변환된 특징 벡터의 집합이  $\mathbf{Y} = \mathbf{W}\mathbf{X} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_T\}$ 라 하면,  $\mathbf{Y}$ 의 변별력을 향상시킬 수 있는 변환 행렬  $\mathbf{W}$ 는 Li가 제안한 식 (3.9)의 목적 함수  $J(\mathbf{W})$ 를  $\mathbf{W}$ 에 대하여 최대화함으로써 구할 수 있다.

$$\begin{aligned}
 J(\mathbf{W}) &= \prod_{i=1}^T P_c(\mathbf{y}_i) \\
 &= \prod_{i=1}^T \frac{P(\mathbf{y}_i | c_{f(\mathbf{y}_i)})}{\sum_{j=1}^M P(\mathbf{y}_i | c_j)}
 \end{aligned} \tag{3.9}$$

식 (3.9)의  $P(\mathbf{y}_i | c_j)$ 는 식 (3.7)을  $\mathbf{y}$ 의 함수로 변형한 것으로 식 (3.10)과 같고,

$$P(\mathbf{y}_i | c_j) = \frac{1}{\sqrt{(2\pi)^N |\Sigma_j^y|}} \exp\left[-\frac{1}{2}(\mathbf{y}_i - \mu_j^y)^T (\Sigma_j^y)^{-1} (\mathbf{y}_i - \mu_j^y)\right] \tag{3.10}$$

$\frac{P(\mathbf{y}_i | c_{f(\mathbf{y}_i)})}{\sum_{j=1}^M P(\mathbf{y}_i | c_j)}$ 는 식 (3.8)을  $\mathbf{y}$ 의 함수로 변형한 것으로 식 (3.11)과 같다.

$$P_c(\mathbf{y}_i) = \frac{P(\mathbf{y}_i | c_{f(\mathbf{y}_i)})}{\sum_{j=1}^M P(\mathbf{y}_i | c_j)} \tag{3.11}$$

식 (3.9) - (3.11)에서  $\mu_j^y = \mathbf{W} \mu_j^x$  와  $\Sigma_j^y = \mathbf{W} \Sigma_j^x \mathbf{W}^T$ 는  $j$ 번째 클래스의 평균과 공분산 행렬을 나타낸다.

Li는 급등반법(steepest gradient ascent)을 사용하여, 식 (3.9)의 목적함수를 변환 행렬  $\mathbf{W}$ 에 대하여 최대화하였으며, 최적화 과정에서 필요한 목적함수의  $\mathbf{W}$ 에 대한 변화 량  $\nabla \mathbf{W}$ 는 식 (3.12)로 근사화하였다.

$$\begin{aligned}
 \nabla \mathbf{W} &\cong \frac{\partial \ln(J(\mathbf{W}))}{\partial \mathbf{W}} = \sum_{i=1}^T \frac{\partial \ln P_c(\mathbf{y}_i)}{\partial \mathbf{W}} \\
 &= \sum_{i=1}^T \left\{ \frac{\partial \ln P(\mathbf{y}_i | c_{f(\mathbf{y}_i)})}{\partial \mathbf{W}} - \frac{\partial \ln [\sum_{j=1}^M P(\mathbf{y}_i | c_j)]}{\partial \mathbf{W}} \right\} \quad (3.12) \\
 &= \sum_{i=1}^T \left\{ \nabla \ln P(\mathbf{y}_i | c_{f(\mathbf{y}_i)}) - \nabla \ln \left[ \sum_{j=1}^M P(\mathbf{y}_i | c_j) \right] \right\}
 \end{aligned}$$

여기서  $\nabla \ln [\sum_{j=1}^M P(\mathbf{y}_i | c_j)]$ 는 식 (3.13)과 같다.

$$\begin{aligned}
 \nabla \ln \left[ \sum_{j=1}^M P(\mathbf{y}_i | c_j) \right] &= \frac{\sum_{j=1}^M \nabla P(\mathbf{y}_i | c_j)}{\sum_{j=1}^M P(\mathbf{y}_i | c_j)} \\
 &= \frac{\sum_{j=1}^M P(\mathbf{y}_i | c_j) \nabla \ln P(\mathbf{y}_i | c_j)}{\sum_{j=1}^M P(\mathbf{y}_i | c_j)} \quad (3.13)
 \end{aligned}$$

그리고 식 (3.12)와 (3.13)을 완전히 계산하기 위해서는  $\nabla \ln P(\mathbf{y}_i | \mathbf{c}_j)$ 의 계산이 필요한데,  $L_i$ 는 특징벡터  $\mathbf{y}_i = \{y_{i1}, y_{i2}, \dots, y_{iN}\}$ 가 서로 상관관계가 없는 요소로 이루어져 있는 벡터이고  $P(\mathbf{y}_i | \mathbf{c}_j)$ 가 대각 공분산 행렬 (diagonal covariance matrix)을 가지는 가우시안 밀도라고 가정하여,  $\nabla \ln P(\mathbf{y}_i | \mathbf{c}_j)$ 를 식 (3.14)와 같이 전개하였다.

$$\nabla \ln P(\mathbf{y}_i | \mathbf{c}_j) = - \sum_{k=1}^N \nabla \left[ \frac{(y_{ik} - \mu_{jk}^y)^2}{2(\sigma_{jk}^y)^2} + \frac{\ln (\sigma_{jk}^y)^2}{2} \right] \quad (3.14)$$

여기서  $y_{ik}$ 는 특징벡터  $\mathbf{y}_i$ 의  $k$ 번째 차원의 요소를,  $\mu_{jk}^y$ 는 클래스  $j$ 의 평균에서  $k$ 번째 차원의 요소를,  $\sigma_{jk}^y$ 는 클래스  $j$ 의 분산에서  $k$ 번째 차원의 요소를,  $N$ 은 특징벡터의 차원을 나타낸다.

마지막으로 목적함수  $J(\mathbf{W})$ 의  $\mathbf{W}$ 에 대한 변화 량  $\nabla \mathbf{W}$ 는 식 (3.14)를  $\mathbf{W}$ 의 각 요소에 대하여 미분한 다음, 그 결과를 식 (3.12)와 (3.13)에 대입하여 구한다.

## 제 4 장 상대 엔트로피에 기반한 특징벡터의 변별적 변환

이 장에서는 먼저 상대 엔트로피에 관하여 서술한 후에, 이 논문에서 제안하는 특징 벡터의 변별력 개선 방법에 대하여 논한다.



### 4.1 상대 엔트로피

상대 엔트로피는 두 클래스 상호간의 거리나 비유사도(dissimilarity)의 정도를 나타내는 척도로, 데이터의 클러스터링(clustering)에 있어서 유용한 정보로 이용되며, Kullback-Leibler distance라고도 한다.

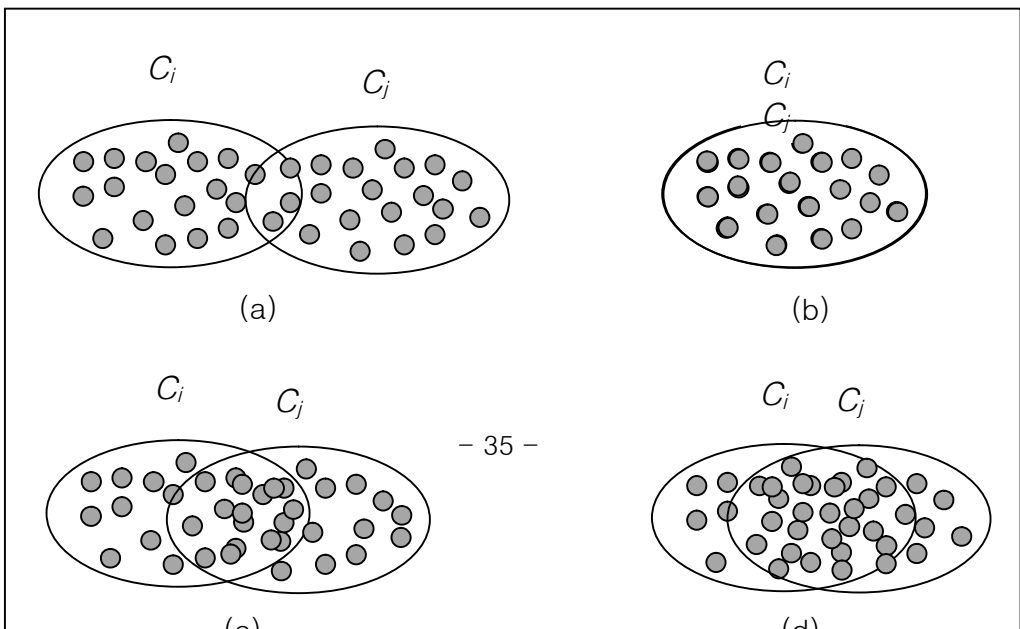


그림 4.1은 두 개의 클래스  $C_i$  와  $C_j$  의 데이터 분포에 관한 4 가지의 경우를 나타낸 것인데, 이 그림에서는 상대 엔트로피가 가장 큰 것은 두 클래스 사이의 겹친 부분이 가장 작은 (a)의 경우고, 상대 엔트로피가 가장 작은 것은 두 클래스가 서로 완전히 겹쳐져있는 (b)의 경우라는 것을 쉽게 알 수 있다.

데이터 집합  $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ 와 두 개의 클래스  $c_i, c_j$ 가 주어지고, 각 클래스가 임의 조건부 확률 밀도  $P(\mathbf{x}_k | c_i)$ 와  $P(\mathbf{x}_k | c_j)$ 를 가진다고 할 때,  $\mathbf{X}$ 의 클래스  $c_i$ 에 대한  $c_j$ 의 상대 엔트로피는 식 (4.1)과 같고,

$$I(i, j) = \sum_{k=1}^N P(\mathbf{x}_k | c_i) \ln \frac{P(\mathbf{x}_k | c_i)}{P(\mathbf{x}_k | c_j)} \quad (4.1)$$

$\mathbf{X}$ 의 클래스  $c_j$ 에 대한  $c_i$ 의 상대 엔트로피는 식 (4.2)와 같다.

$$I(j, i) = \sum_{k=1}^N P(\mathbf{x}_k | c_j) \ln \frac{P(\mathbf{x}_k | c_j)}{P(\mathbf{x}_k | c_i)} \quad (4.2)$$

여기서  $N$ 은 데이터 집합  $\mathbf{X}$ 에서의 총 데이터의 개수를 나타낸다. 식 (4.1)과 (4.2)에서  $\ln \frac{P(\mathbf{x}_k | c_i)}{P(\mathbf{x}_k | c_j)}$ 이 0보다 크고,  $\ln \frac{P(\mathbf{x}_k | c_j)}{P(\mathbf{x}_k | c_i)}$ 이 0보다 작다는 것은 데이터  $\mathbf{x}_k$ 가 클래스  $c_i$ 에 속할 가능성이 높다는 것을,  $\ln \frac{P(\mathbf{x}_k | c_i)}{P(\mathbf{x}_k | c_j)}$ 이 0보다 작고  $\ln \frac{P(\mathbf{x}_k | c_j)}{P(\mathbf{x}_k | c_i)}$ 이 0보다 크다는 것은 데이터  $\mathbf{x}_k$ 가 클래스  $c_j$ 에 속할 가능성이 높다는 것을, 그리고  $\ln \frac{P(\mathbf{x}_k | c_i)}{P(\mathbf{x}_k | c_j)}$ 과  $\ln \frac{P(\mathbf{x}_k | c_j)}{P(\mathbf{x}_k | c_i)}$ 이 0이라는 것은 데이터  $\mathbf{x}_k$ 가 어느 클래스에 속하는지를 구별할 수 없음을 의미한다. 이는 식 (4.1)과 (4.2)의 상대 엔트로피가 클래스 상호간의 거리나 비유사도의 정도를 측정하는데 사용할 수 있음을 의미한다.

일반적으로 식 (4.1)의 상대 엔트로피와 (4.2)의 상대 엔트로피는 비대칭적인 것으로 알려져 있으며, 이들의 합을 식 (4.3)과 같이 정의하여, divergence라 부른다. 이 논문에서는 클래스 상호간의 거리(비유사도)에 대한 평균 값을 나타내는 식 (4.3)의 divergence를 이용하여 특징벡터의 변별력을 개선한다.

$$I = I(i, j) + I(j, i) = \sum_{k=1}^N [P(\mathbf{x}_k | c_i) - P(\mathbf{x}_k | c_j)] \ln \frac{P(\mathbf{x}_k | c_i)}{P(\mathbf{x}_k | c_j)} \quad (4.3)$$

## 4.2 상대 엔트로피를 이용한 특징벡터의 변별적 변환

이 논문에서는 음소의 클래스 정보를 이용하여 인식기에 인가되는 특징벡터의 변별력을 개선함과 동시에 인식률을 향상 시킬 수 있는 특징벡터의 선형 변환 방법을 제안한다. 제안하는 방법은 인식기의 클래스 정보에 기반한 상대 엔트로피를 이용하여 클래스 내부의 거리는 가깝게 하

고, 클래스 상호간의 거리는 멀게 하는 특징벡터의 선형 변환 방법으로, 이 방법은 상대 엔트로피가 클수록 클래스 상호간의 유사도가 작아지므로, 클래스에 대한 분류가 용이해진다는 점에 착안한 것이다. 그리고 특징벡터의 변별력을 개선하는 변환 행렬은 클래스 상호간의 상대 엔트로피에 대한 평균인 식 (4.3)의 divergence를 이용하여 목적함수를 정의한 다음, 이 목적함수를 최대화하여 구하는데, 그 과정은 다음과 같다.

원시 특징벡터의 집합  $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_T\}$ 와 특징벡터의 클래스 정보  $\mathbf{c} = \{c_1, c_2, \dots, c_M\}$  그리고 변환 행렬  $\mathbf{W}$ 가 주어지고, 선형 변환된 특징벡터의 집합이  $\mathbf{Y} = \mathbf{W}\mathbf{X} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_T\}$ 라 하고,  $\mathbf{Y}$ 의 클래스 정보가  $\mathbf{X}$ 의 클래스 정보와 동일하다고 하자. 또한 임의의 클래스  $c_i$ 와 임의의 특징벡터  $\mathbf{x}_t$ 의 조건부 확률밀도와 임의의 클래스  $c_i$ 와 임의의 특징벡터  $\mathbf{y}_t$ 의 조건부 확률밀도가 각각 식(4.4)와 식 (4.5)와 같이 가우시안 분포를 가진다고 하자. 이때 divergence에 기반한 목적함수  $J(\mathbf{W})$ 는 식 (4.6)과 같다.



$$P(\mathbf{x}_t | c_i) = \frac{1}{\sqrt{(2\pi)^N |\boldsymbol{\Sigma}_i^x|}} \exp\left[-\frac{1}{2}(\mathbf{x}_t - \boldsymbol{\mu}_i^x)^T (\boldsymbol{\Sigma}_i^x)^{-1} (\mathbf{x}_t - \boldsymbol{\mu}_i^x)\right] \quad (4.4)$$

$$P(\mathbf{y}_t | c_i) = \frac{1}{\sqrt{(2\pi)^N |\boldsymbol{\Sigma}_i^y|}} \exp\left[-\frac{1}{2}(\mathbf{y}_t - \boldsymbol{\mu}_i^y)^T (\boldsymbol{\Sigma}_i^y)^{-1} (\mathbf{y}_t - \boldsymbol{\mu}_i^y)\right] \quad (4.5)$$

$$J(\mathbf{W}) = \sum_{t=1}^T \sum_{j=1}^M [P(\mathbf{y}_t | c_{f(\mathbf{y}_t)}) - P(\mathbf{y}_t | c_j)] \ln \frac{P(\mathbf{y}_t | c_{f(\mathbf{y}_t)})}{P(\mathbf{y}_t | c_j)} \quad (4.6)$$



여기서  $N$ 은 특징벡터의 차원을,  $M$ 은 클래스의 개수를,  $f(\mathbf{y}_t)$ 는  $\mathbf{y}_t$ 가 속한 클래스의 색인을 구하는 함수를,  $\boldsymbol{\Sigma}_i^x$ 는 원시 특징벡터에 대한 클래스  $i$ 의 공분산 행렬을,  $\boldsymbol{\mu}_i^x$ 는 원시 특징벡터에 대한 클래스  $i$ 의 평균을,  $\boldsymbol{\Sigma}_i^y$ 는 선형 변환된 특징벡터에 대한 클래스  $i$ 의 공분산 행렬을,  $\boldsymbol{\mu}_i^y$ 는 선형 변환된 특징벡터에 대한 클래스  $i$ 의 평균을 나타낸다.

이 논문에서는 natural gradient ascent 방법을 사용하여 식 (4.6)의 목적함수를 변환 행렬  $\mathbf{W}$  대하여 최대화하는데, 최적화 과정에서 요구되는 변환 행렬  $\mathbf{W}$ 에 대한 목적함수의 변화량  $\nabla \mathbf{W}$ 는 식 (4.7)과 같이 근사화하였다.

$$\begin{aligned}
 \nabla \mathbf{W} &\cong \frac{\partial J(\mathbf{W})}{\partial \mathbf{W}} && (4.7) \\
 &\cong \sum_{t=1}^T \sum_{j=1}^M \frac{\partial [P(\mathbf{y}_t | c_{f(\mathbf{y}_t)}) - P(\mathbf{y}_t | c_j)] [\ln P(\mathbf{y}_t | c_{f(\mathbf{y}_t)}) - \ln P(\mathbf{y}_t | c_j)]}{\partial \mathbf{W}} \\
 &\cong \sum_{t=1}^T \sum_{j=1}^M \left\{ \left[ P(\mathbf{y}_t | c_{f(\mathbf{y}_t)}) \frac{\partial \ln P(\mathbf{y}_t | c_{f(\mathbf{y}_t)})}{\partial \mathbf{W}} - P(\mathbf{y}_t | c_j) \frac{\partial \ln P(\mathbf{y}_t | c_j)}{\partial \mathbf{W}} \right] \right. \\
 &\quad \times [\ln P(\mathbf{y}_t | c_{f(\mathbf{y}_t)}) - \ln P(\mathbf{y}_t | c_j)] + [P(\mathbf{y}_t | c_{f(\mathbf{y}_t)}) - P(\mathbf{y}_t | c_j)] \\
 &\quad \left. \times \left[ \frac{\partial \ln P(\mathbf{y}_t | c_{f(\mathbf{y}_t)})}{\partial \mathbf{W}} - \frac{\partial \ln P(\mathbf{y}_t | c_j)}{\partial \mathbf{W}} \right] \right\}
 \end{aligned}$$

여기서  $\frac{\partial \ln P(\mathbf{y}_t | c_j)}{\partial \mathbf{W}}$ 는 랜덤 변수의 선형 변환에 대한 확률 밀도함수의 변

환 이론에 의해,  $P(\mathbf{y}_t | c_j)$ 와  $P(\mathbf{x}_t | c_j)$ 의 관계가 식 (4.8)을 만족하므로, 식 (4.9)와 같이 전개된다.

$$P(\mathbf{y}_t | c_j) = \frac{1}{|J_j|} P(\mathbf{x}_t | c_j) \quad (4.8)$$

$$\begin{aligned} \frac{\partial \ln P(\mathbf{y}_t | c_j)}{\partial \mathbf{W}} &= \frac{\partial \ln P(\mathbf{x}_t | c_j)}{\partial \mathbf{W}} - \frac{\partial \ln |J_j|}{\partial \mathbf{W}} \\ &= - \frac{\partial \ln |J_j|}{\partial \mathbf{W}} \end{aligned} \quad (4.9)$$



그리고 식 (4.7)과 식 (4.9)를 완전히 전개하기 위해서는 클래스  $j$ 의 Jacobian 행렬  $J_j$ 의 행렬 값을 미분하는 수식인  $\frac{\partial \ln |J_j|}{\partial \mathbf{W}}$ 의 계산이 필요하다.

이 논문에서는 식 (4.9)의  $\frac{\partial \ln |J_j|}{\partial \mathbf{W}}$ 를 전개하기 위해 선형 변환된 특징벡터  $\mathbf{y}_t$ 를 식 (4.10)과 같이 원시 특징벡터  $\mathbf{x}_t = \{x_{t1}, x_{t2}, \dots, x_{tN}\}$ 와 변환 행렬  $\mathbf{W}$ 의 곱으로 정의하였다. 그리고  $\mathbf{y}_t$ 가 서로 독립적인 요소로 이루어져 있는 벡터이고,  $P(\mathbf{y}_t | c_j)$ 가 대각 공분산 행렬(diagonal covariance matrix)을 가지는 가우시안 밀도라고 가정하여,  $P(\mathbf{y}_t | c_j)$ 의 누적 분포  $f_j$ 와 Jacobian 행렬  $J_j$ 의 행렬식  $|J_j|$ 를 식 (4.11)과 식 (4.12)로 정의

하였다. 위의 가정과 정의로부터  $\frac{\partial \ln|J_j|}{\partial \mathbf{w}}$  는 식 (4.13)의 체인 법칙과 식 (4.14)를 식 (4.12)에 적용해서 식 (4.15)와 같이 유도하였다.

$$\mathbf{y}_t = \mathbf{W} \mathbf{x}_t = \{y_{t1}, y_{t2}, \dots, y_{tN}\} \quad (4.10)$$

$$f_j = \int_{-\infty}^{y_t} P(\mathbf{z} | c_j) d\mathbf{z} = \{f_{j1}, f_{j2}, \dots, f_{jN}\} \quad (4.11a)$$

$$f_{jk} = \int_{-\infty}^{y_{tk}} P(\mathbf{z} | c_j) d\mathbf{z}, \quad 1 \leq k \leq N \quad (4.11b)$$

$$|J_j| = \begin{vmatrix} \frac{\partial f_{j1}}{\partial x_{t1}} & \dots & \frac{\partial f_{j1}}{\partial x_{tN}} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_{jN}}{\partial x_{t1}} & \dots & \frac{\partial f_{jN}}{\partial x_{tN}} \end{vmatrix} \quad (4.12)$$



$$= |\mathbf{W}| \prod_{k=1}^N \left| \frac{\partial f_{jk}}{\partial y_{tk}} \right|$$

$$\frac{\partial f_{jk}}{\partial x_{tr}} = \frac{\partial y_{tk}}{\partial x_{tr}} \frac{\partial f_{jk}}{\partial y_{tk}} = w_{kr} \frac{\partial f_{jk}}{\partial y_{tk}} \quad (4.13a)$$

$$\frac{\partial f_{jk}}{\partial w_{kr}} = \frac{\partial^2 f_{jk}}{\partial^2 y_{tk}} \frac{\partial y_{tk}}{\partial w_{kr}} = \frac{\partial^2 f_{jk}}{\partial^2 y_{tk}} x_{tr} \quad (4.13b)$$

$$\frac{\partial f_{jk}}{\partial y_{tk}} = P(y_{tk} | c_j) \quad (4.14)$$

$$\begin{aligned} \frac{\partial \ln |J_j|}{\partial \mathbf{W}} &= \frac{\partial \ln |\mathbf{W}|}{\partial \mathbf{W}} + \sum_{k=1}^N \frac{\partial \ln \left| \frac{\partial f_{jk}}{\partial y_{tk}} \right|}{\partial \mathbf{W}} \\ &= (\mathbf{W}^T)^{-1} + \sum_{k=1}^N \frac{\partial \ln \left| \frac{\partial f_{jk}}{\partial y_{tk}} \right|}{\partial \mathbf{W}} \\ &= (\mathbf{W}^T)^{-1} + \frac{\partial P(\mathbf{y}_t | c_j)}{\partial \mathbf{y}_t} \mathbf{x}_t^T \\ &= (\mathbf{W}^T)^{-1} + \frac{\partial \ln P(\mathbf{y}_t | c_j)}{\partial \mathbf{y}_t} \mathbf{x}_t^T \\ &= (\mathbf{W}^T)^{-1} - (\boldsymbol{\Sigma}_j^y)^{-1} (\mathbf{y}_t - \boldsymbol{\mu}_j^y) \mathbf{x}_t^T \end{aligned} \quad (4.15)$$

식 (4.13)의  $w_{kr}$ 는 변환 행렬  $\mathbf{W}$ 의  $k$ 행  $r$ 열의 요소를 나타내고, 식 (4.10) - (4.15)의  $N$ 은 특징벡터의 차원을 나타내며, 식 (4.14)는  $y_{tk}$ 에 대한 식 (4.11)의  $f_{jk}$ 의 변화량을 의미한다.

목적함수  $l(\mathbf{W})$ 의  $\mathbf{W}$ 에 대한 변화량  $\nabla \mathbf{W}$ 는 식 (4.15)를 식 (4.7)과 식 (4.9)에 대입하여 구했으며, 식 (4.16)과 같다. 그리고 변환 행렬  $\mathbf{W}$ 의 재추정 식은 식 (4.17)과 같고 선형 변환된 특징벡터의 공분산 행렬  $\boldsymbol{\Sigma}_j^y$ 와 평균  $\boldsymbol{\mu}_j^y$ 의 재추정 식은 식 (4.18)과 식 (4.19)와 같다.

$$\begin{aligned}
\nabla W \cong & \sum_{t=1}^T \sum_{j=1}^M \left\{ \left[ P(\mathbf{y}_t | c_j) \left[ (\mathbf{W}^T)^{-1} - (\boldsymbol{\Sigma}_j^y)^{-1} (\mathbf{y}_t - \boldsymbol{\mu}_j^y) \mathbf{x}_t^T \right] \right. \right. & (4.16) \\
& - P(\mathbf{y}_t | c_h) \left[ (\mathbf{W}^T)^{-1} - (\boldsymbol{\Sigma}_h^y)^{-1} (\mathbf{y}_t - \boldsymbol{\mu}_h^y) \mathbf{x}_t^T \right] \\
& \times \left[ \ln P(\mathbf{y}_t | c_h) - \ln P(\mathbf{y}_t | c_j) \right] + \left[ P(\mathbf{y}_t | c_h) - P(\mathbf{y}_t | c_j) \right] \\
& \times \left[ \left[ (\mathbf{W}^T)^{-1} - (\boldsymbol{\Sigma}_j^y)^{-1} (\mathbf{y}_t - \boldsymbol{\mu}_j^y) \mathbf{x}_t^T \right] \right. \\
& \left. \left. - \left[ (\mathbf{W}^T)^{-1} - (\boldsymbol{\Sigma}_h^y)^{-1} (\mathbf{y}_t - \boldsymbol{\mu}_h^y) \mathbf{x}_t^T \right] \right] \right\}, \quad \text{where } h = f(\mathbf{y}_t)
\end{aligned}$$



식 (4.17)–(4.19)의 재추정 식을 natural gradient ascent 방법으로 반복 수행하면, 목적함수  $J(\mathbf{W})$ 가 최대값을 가지는  $\mathbf{W}$ ,  $\boldsymbol{\Sigma}_j^y$  그리고  $\boldsymbol{\mu}_j^y$ 에 도달한다.

$$\overline{\mathbf{W}} = \frac{\mathbf{W} + \alpha \nabla W \mathbf{W}^T \mathbf{W}}{|\mathbf{W} + \alpha \nabla W \mathbf{W}^T \mathbf{W}|} \quad (4.17)$$

$$\overline{\boldsymbol{\Sigma}_j^y} = \overline{\mathbf{W}} \boldsymbol{\Sigma}_j^x \overline{\mathbf{W}}^T, \quad 1 \leq j \leq M \quad (4.18)$$

$$\overline{\boldsymbol{\mu}_j^y} = \overline{\mathbf{W}} \boldsymbol{\mu}_j^x, \quad 1 \leq j \leq M \quad (4.19)$$

식 (4.17) - (4.19)의  $\alpha$ 는 학습률을 나타내고,  $M$ 은 총 클래스의 개수를 나타낸다.

## 제 5 장 실험 및 고찰

이 장에서는 제안한 특징벡터의 변별적 변환방법을 검증하기 위해 수행한 음성 데이터의 클러스터링 실험, 그리고 인식 실험의 실험 환경과 실험 결과에 대하여 소개한다. 첫 번째 절에서는 실험에 사용한 음성 코퍼스에 관하여 서술하고, 두 번째 절에서는 특징벡터의 변별적 변환과 클러스터링 실험에 관하여 서술한 다음, 마지막 절에서 음성 인식 실험에 관하여 논한다.

### 5.1 음성 데이터



이 절에서는 기존의 변별적 변환방법인 주요 성분분석, 선형 판별분석, Li의 방법과 제안한 변별적 변환방법을 이용하여, 클러스터링 실험과 음소 단위의 인식 실험을 수행하였다. 실험에 사용한 음성 코퍼스는 미국 8개 지역의 남녀 630명의 화자로부터 10개의 문장을 16kHz, 16Bit로 녹음한 영어 음성 데이터베이스인 TIMIT이다. TIMIT는 학습용 데이터 집합과 테스트용 데이터 집합으로 이루어져 있고, 64개의 음소를 가지고 있는데, 이 논문에서는 표 5.1에 나타낸 48개의 음소를 선택하여 실험에 사용하였다[34,35]. 그리고 실험에 사용한 특징 벡터는 2장에서 언급한 MFCC 계수 추출 방법과 미분 계수 추출 방법으로 13차원과 26차원의 벡터를 구한 다음, 이 벡터들을 식 (5.1)에 적용하여 65 차원과 130 차원의 새로운 벡터로 변환한 것이다.

표 5.1 TIMIT의 음소 테이블  
Table 5.1 TIMIT phone table

Phone	Example	Phone	Example
iy	<u>beat</u>	en	<u>button</u>
ih	<u>bit</u>	ng	<u>sing</u>
eh	<u>bet</u>	ch	<u>church</u>
ae	<u>Bat</u>	jh	<u>judge</u>
ix	ros <u>es</u>	dh	<u>they</u>
ax	th <u>e</u>	b	<u>bob</u>
ah	<u>butt</u>	d	<u>dad</u>
uw	<u>boot</u>	dx	<u>butter</u>
uh	<u>book</u>	g	<u>gag</u>
ao	<u>about</u>	p	<u>pop</u>
aa	<u>cot</u>	t	<u>tot</u>
ey	<u>bait</u>	k	<u>kick</u>
ay	<u>bite</u>	z	<u>zoo</u>
oy	<u>boy</u>	zh	<u>measure</u>
aw	<u>bough</u>	v	<u>very</u>
ow	<u>boat</u>	f	<u>fief</u>
l	<u>led</u>	th	<u>thief</u>
el	<u>bottle</u>	s	<u>sis</u>
r	<u>red</u>	sh	<u>shoe</u>
y	<u>yet</u>	hh	<u>hay</u>
w	<u>wet</u>	cl	(unvoiced closure)
er	<u>bird</u>	vcl	(voiced closure)
m	<u>mom</u>	epi	(epinthetic closure)
n	<u>non</u>	sil	(silence)

$$\mathbf{o}(t)_{new} = \{\mathbf{o}(t-2), \mathbf{o}(t-1), \mathbf{o}(t), \mathbf{o}(t+1), \mathbf{o}(t+2)\}, \quad 0 \leq t \leq T \quad (5.1)$$

식(5.1)의  $\mathbf{o}(t)$ 는  $t$ 번째 프레임의 13 차원 또는 26 차원의 MFCC 계수이고,  $\mathbf{o}(t)_{new}$ 는  $t$ 번째 프레임과 좌/우 두 프레임씩의 특징벡터를 모아서 구한 65차원 또는 130차원의 특징벡터이고,  $T$ 는 총 프레임 개수이다.

변별적 변환방법과 클러스터링 실험에서는 TIMIT 학습용 데이터의 각 음소에 대한 특징 벡터의 약 15% 정도를 선형 변환 행렬  $W$ 의 훈련에 사용하고, 나머지 85%와 TIMIT 테스트용 데이터의 특징벡터를 테스트 데이터로 사용하였다. 이 실험에서 15%의 학습용 특징벡터는 TIMIT의 음성 데이터 각각의 음소 전사 파일을 토대로 하여, 각 음소의 중앙에 있는 단일 프레임 또는 두 프레임의 특징 벡터들을 모아서 만들었다.

인식 실험에서는 변별적 변환방법과 클러스터링 실험에서 구한 변환 행렬  $W$ 를 이용하여 TIMIT 학습용 데이터와 테스트용 데이터의 특징 벡터들을 변환한 후에, 변환된 학습용 특징 벡터는 음소 단위의 인식기 학습에 사용하고, 변환된 테스트용 특징 벡터는 인식기의 성능 평가를 위해 사용하였다.

## 5.2 특징벡터의 변별적 변환 및 클러스터링 실험

변별적 변환방법과 클러스터링 실험은 그림 5.1과 같이 학습(training) 과정과 테스트(test) 과정으로 나눌 수 있다. 학습 과정은 학습용 특징벡터의 집합으로부터 선형 변환 행렬  $W$ 를 훈련하는 과정이고, 테스트 과정은 학습 과정에서 구한 선형 변환 행렬을 테스트용 특징 벡터의 집합에 적용하여 클러스터링하는 과정을 말한다. 클러스터링은 임의의 테스트용 특징벡터가 클러스터링 엔진에 입력되었을 때, 그것과 48개의 음소 각각에 대한 중심(centroid)과의 유클리디언 거리(Euclidean



distance)를 구한 다음, 거리가 가장 가까운 음소로 입력 데이터를 결정하는 것을 말하며, 식 (5.2)와 같다.

$$k = \mathit{arg} \min_{i=1, \dots, 48} \|\mathbf{x}_t - \boldsymbol{\mu}_i\| \quad (5.2)$$

식 (5.2)의  $\mathbf{x}_t$ 는  $t$ 번째 테스트용 특징벡터를,  $\boldsymbol{\mu}_i$ 는  $i$ 번째 음소의 중심을,  $k$ 는  $\mathbf{x}_t$ 와 거리가 가장 가까운 음소  $\boldsymbol{\mu}_i$ 의 색인을 나타낸다.

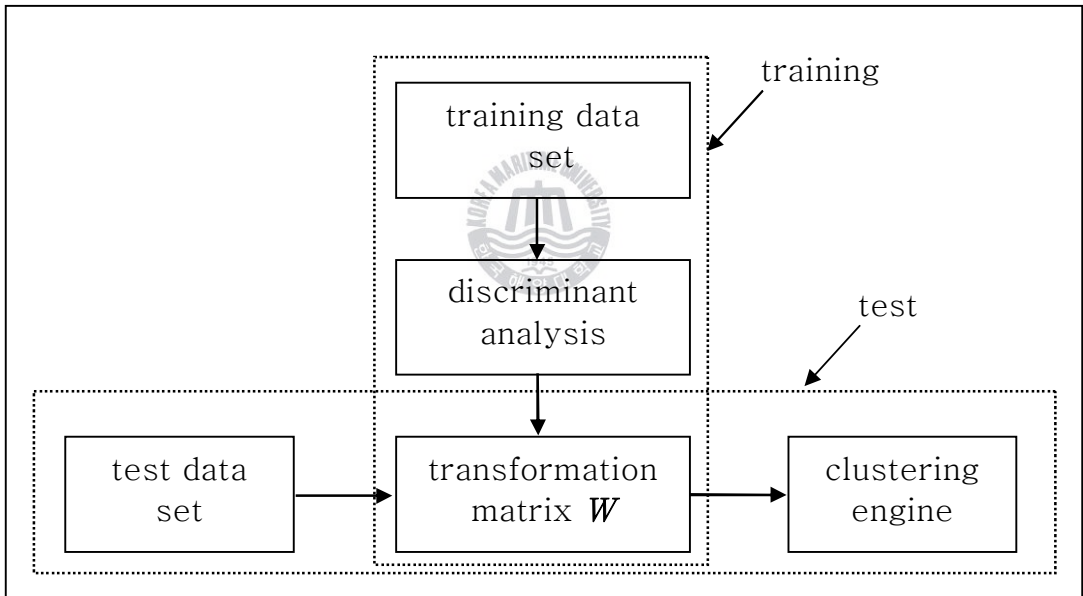


그림 5.1 변별적 변환 및 클러스터링의 블록 다이어그램  
 Figure 5.1 Block diagram of discriminative transformation and clustering

그림 5.2에서 5.11은 2 장에서 언급한 MFCC 계수 추출 방법으로 구한 원시 특징벡터(13차)와 변별적 변환방법을 통하여 구한 특징벡터(13

차)의 분포를 나타낸 것으로, 분포도의 각 축은 특징벡터의 1 - 3차원을 의미한다. 또한 표 5.2는 각 그림 내부에 존재하는 분포도에 대한 음소 정보를 나낸 것이다.

표 5.2 그림 5.2-5.11에서 분포도에 대한 음소 정보  
Table 5.2 Phone information of distributions in figures 5.2-5.11

종류 색인	그림 5.2, 5.4, 5.6, 5.8, 5.10	그림 5.3, 5.5, 5.7, 5.9, 5.11
(a)	“aa” 와 “ae” 의 분포도	“iy” 와 “m” 의 분포도
(b)	“aa” 와 “ae” 의 분포도	“eh” 와 “ng” 의 분포도
(c)	“aa” 와 “uh” 의 분포도	“aa” 와 “s” 의 분포도
(d)	“aa” 와 “ix” 의 분포도	“ow” 와 “f” 의 분포도
(e)	“aa” 와 “ax” 의 분포도	“uw” 와 “z” 의 분포도
(f)	“aa” 와 “ow” 의 분포도	“ax” 와 “i” 의 분포도
(g)	“aa” 와 “eh” 의 분포도	“ao” 와 “b” 의 분포도
(h)	“aa” 와 “er” 의 분포도	“aw” 와 “p” 의 분포도
(i)	“aa” 와 “ey” 의 분포도	“er” 와 “ch” 의 분포도

그림 5.2와 5.3은 변별적 변환방법을 적용하지 않은 특징벡터에 대해서 모음 상호간의 분포와 모음과 자음 상호간의 분포를 나타낸 것이다. 그림 5.2에서는 “aa” 와 “er” 의 변별력이 다른 것들에 비해 낮음을 알 수 있고, 그림 5.3에서는 “ow” 와 “f” 의 변별력이 다른 것들에

비해 현저하게 떨어짐을 알 수 있다.

그림 5.4와 5.5는 65차원의 학습용 특징벡터 집합에 주요 성분분석법을 적용하여 선형 변환 행렬을 구한 다음, 이 변환 행렬로 테스트용 특징벡터를 13차원의 벡터로 선형 변환하여 모음 상호간의 분포(그림 5.4)와 모음과 자음 상호간의 분포(그림 5.5)를 그린 것이다. 이 그림들에서는 그림 5.2와 5.3에 비해, 각 음소의 분산이 커짐과 동시에 중심이 이동한 것을 알 수 있다. 특히 그림 5.5에서의 (f)에서는 “ow”와 “f”의 변별력이 그림 5.3의 (f)에서 보다 많이 증가하였음을 확인할 수 있다.

그림 5.6과 5.7은 선형 판별분석법을 이용하여 선형 변환 행렬 구한 다음, 변환 행렬에 테스트용 특징벡터의 집합을 적용해서 추출한 13차원의 벡터들에 대한 모음 상호간의 분포와 모음과 자음 상호간의 분포를 나타낸 것이다. 그림 5.6과 5.7에서는 그림 5.4와 5.5에 비해서 각 음소의 분산은 감소하고, 중심이 이동하여 변별력이 개선되었음을 알 수 있다. 특히 각 음소의 분산이 많이 감소한 것을 알 수 있다.

그림 5.8과 5.9는 Li가 제안한 방법으로 특징벡터를 선형 변환하여, 모음 상호간의 특징벡터의 분포와 모음과 자음 상호간의 특징벡터의 분포를 나타낸 것이다. 특징벡터를 선형 변환하는 행렬은 먼저, 학습용 특징벡터들을 선형 판별분석법에 적용하여 초기화한 다음, 3.4절에 언급한 반복 학습 방법에 의해서 최적화하여 구했다. 그림 5.8과 5.9에서는 그림 5.6과 5.7의 선형 판별분석법에 비해 각 음소 사이의 거리가 멀어졌음을 알 수 있다. 또한 그림 5.4와 5.6의 주요 성분분석법에 비해 음소 사이의 거리는 멀어지고, 음소 내부의 분산은 감소한 것을 알 수 있다.

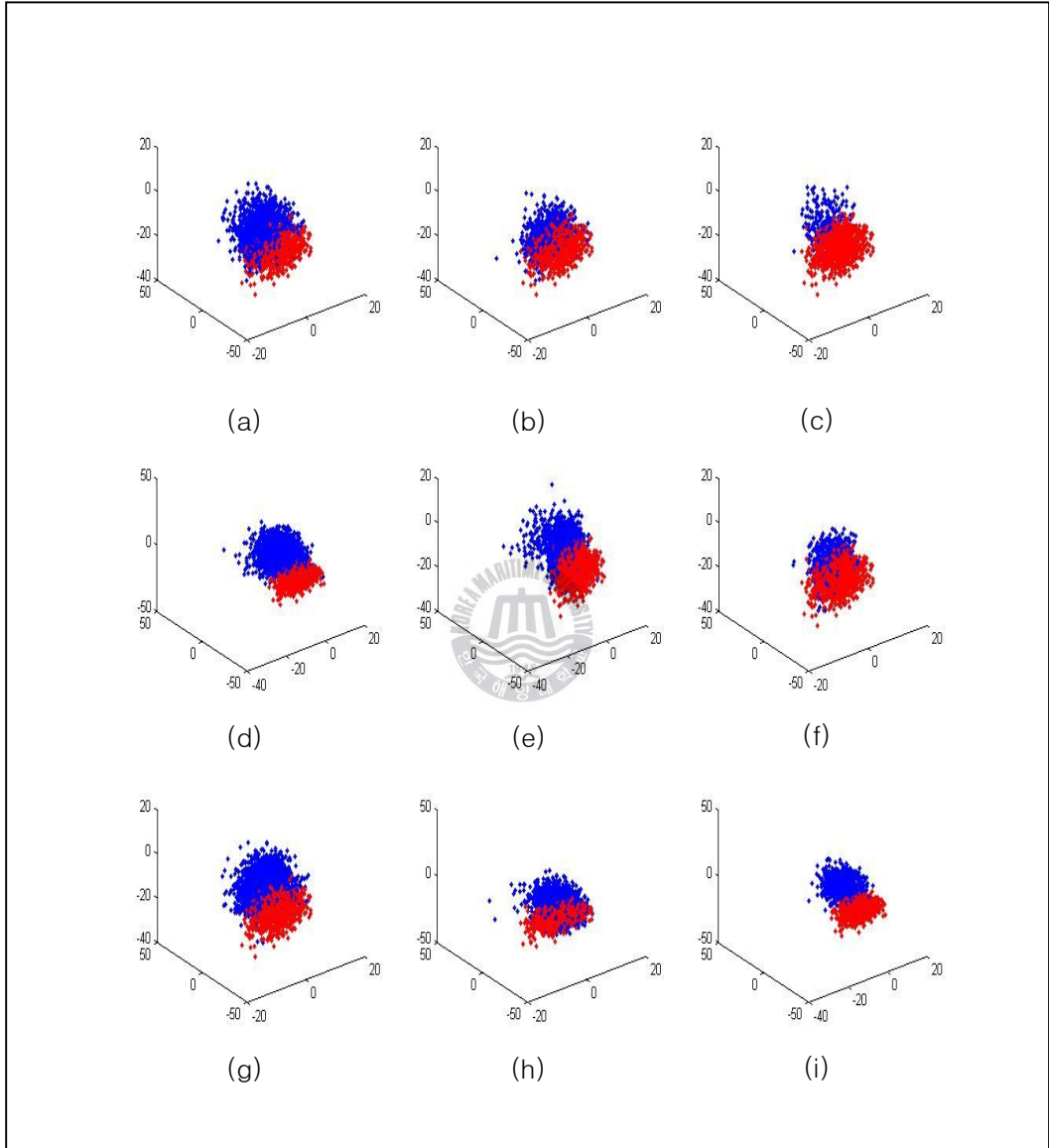


그림 5.2 모음 상호간의 특징벡터의 분포  
 Figure 5.2 Distribution of feature vectors of vowels

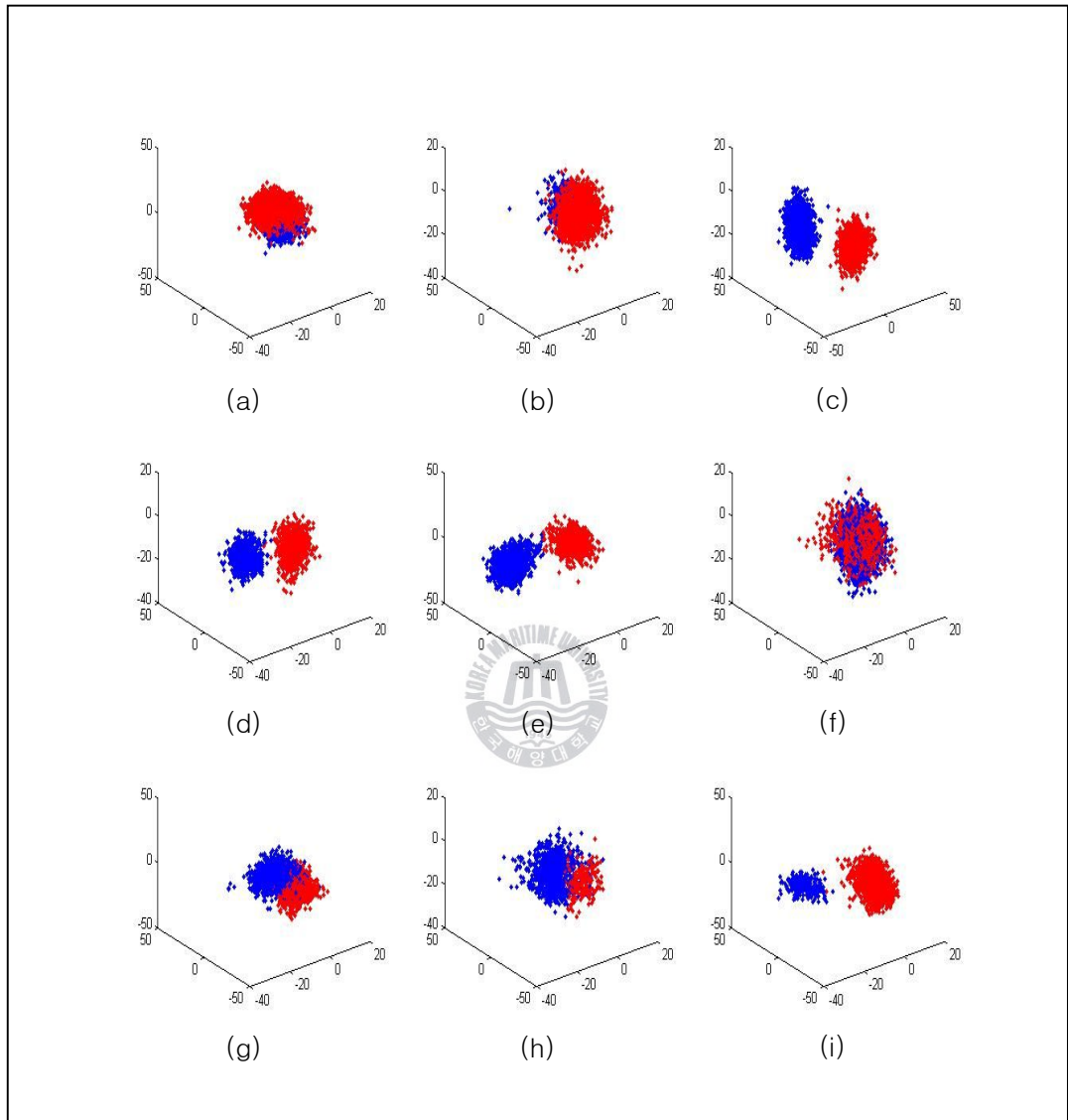


그림 5.3 모음과 자음의 특징벡터의 분포

Figure 5.3 Distribution of feature vectors of vowel and consonant

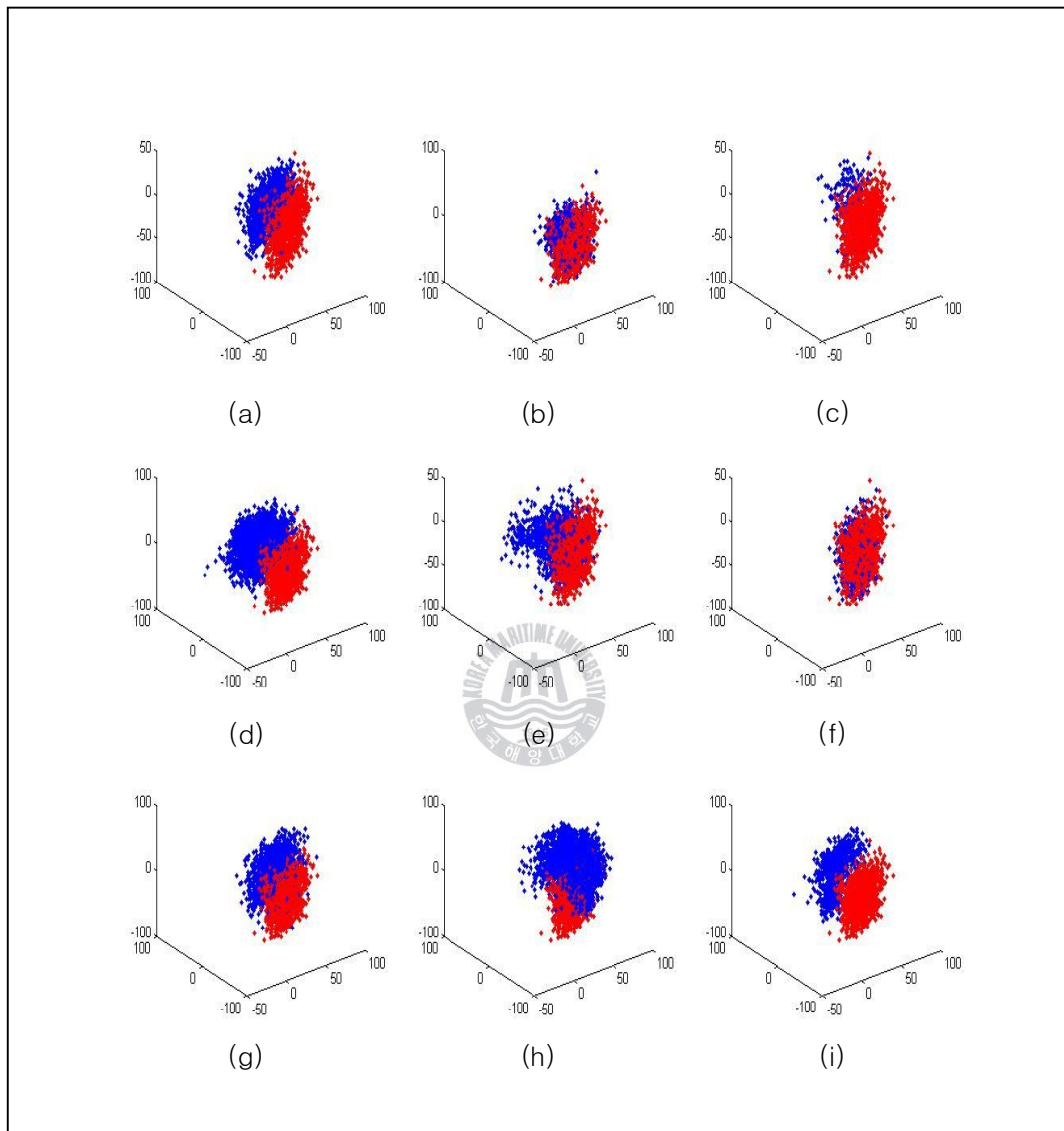


그림 5.4 PCA를 적용한 모음 상호간의 특징벡터의 분포  
 Figure 5.4 Distribution of feature vectors of vowels after transformation by the PCA

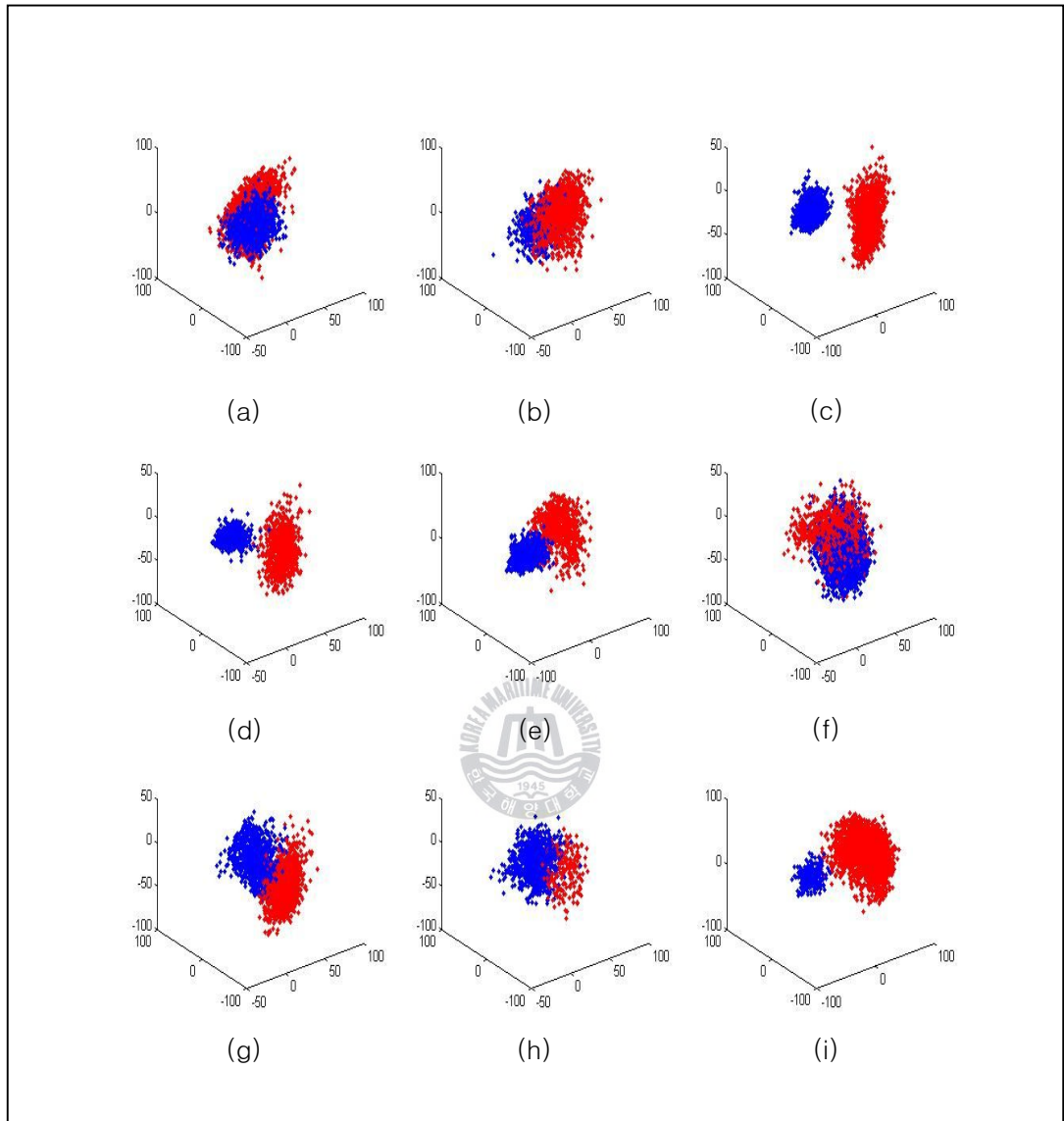


그림 5.5 PCA를 적용한 모음과 자음의 특징벡터의 분포  
 Figure 5.5 Distribution of feature vectors of vowel and consonant after transformation by the PCA

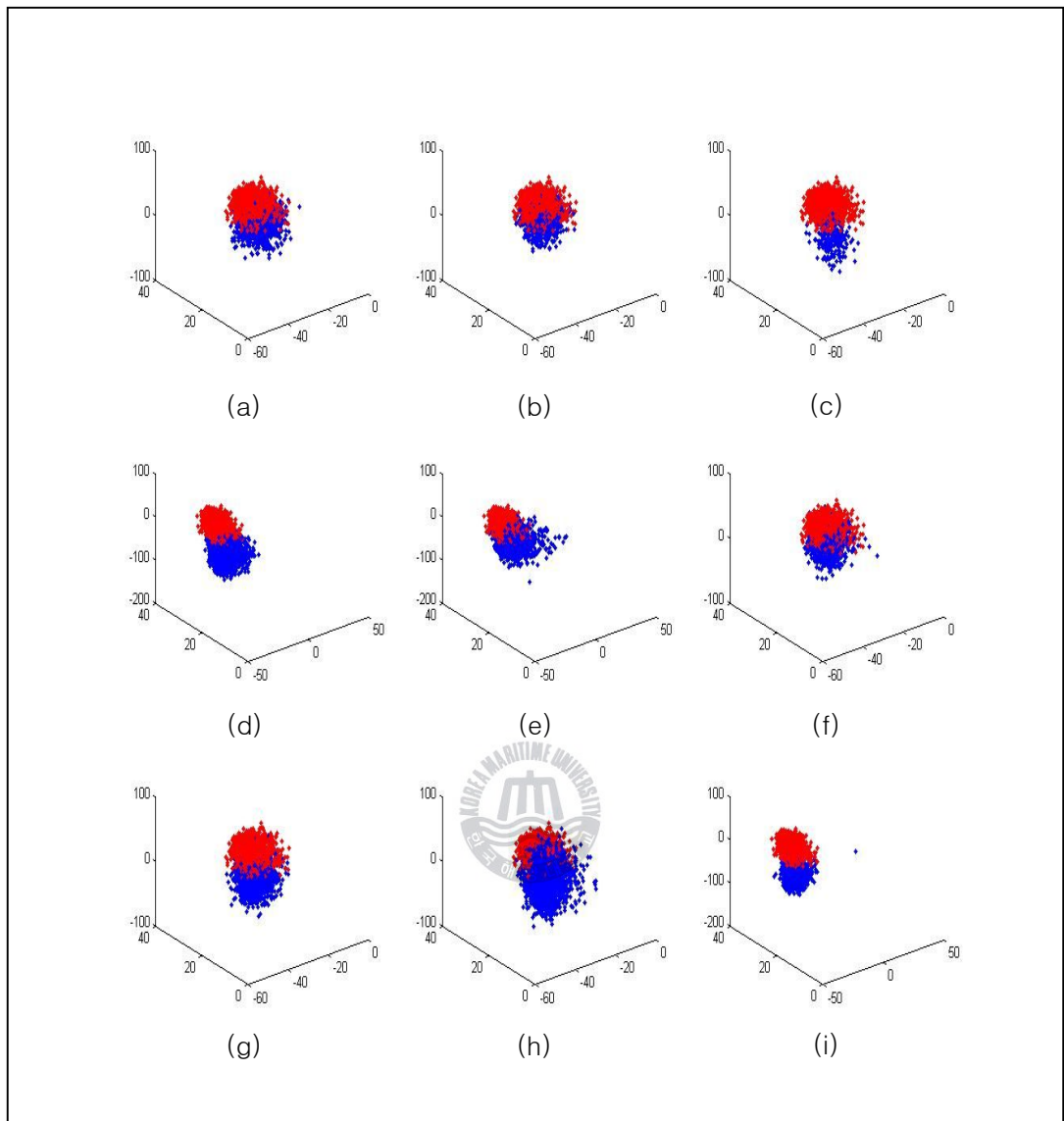


그림 5.6 LDA를 적용한 모음 상호간의 특징벡터의 분포  
 Figure 5.6 Distribution of feature vectors of vowels after transformation by the LDA



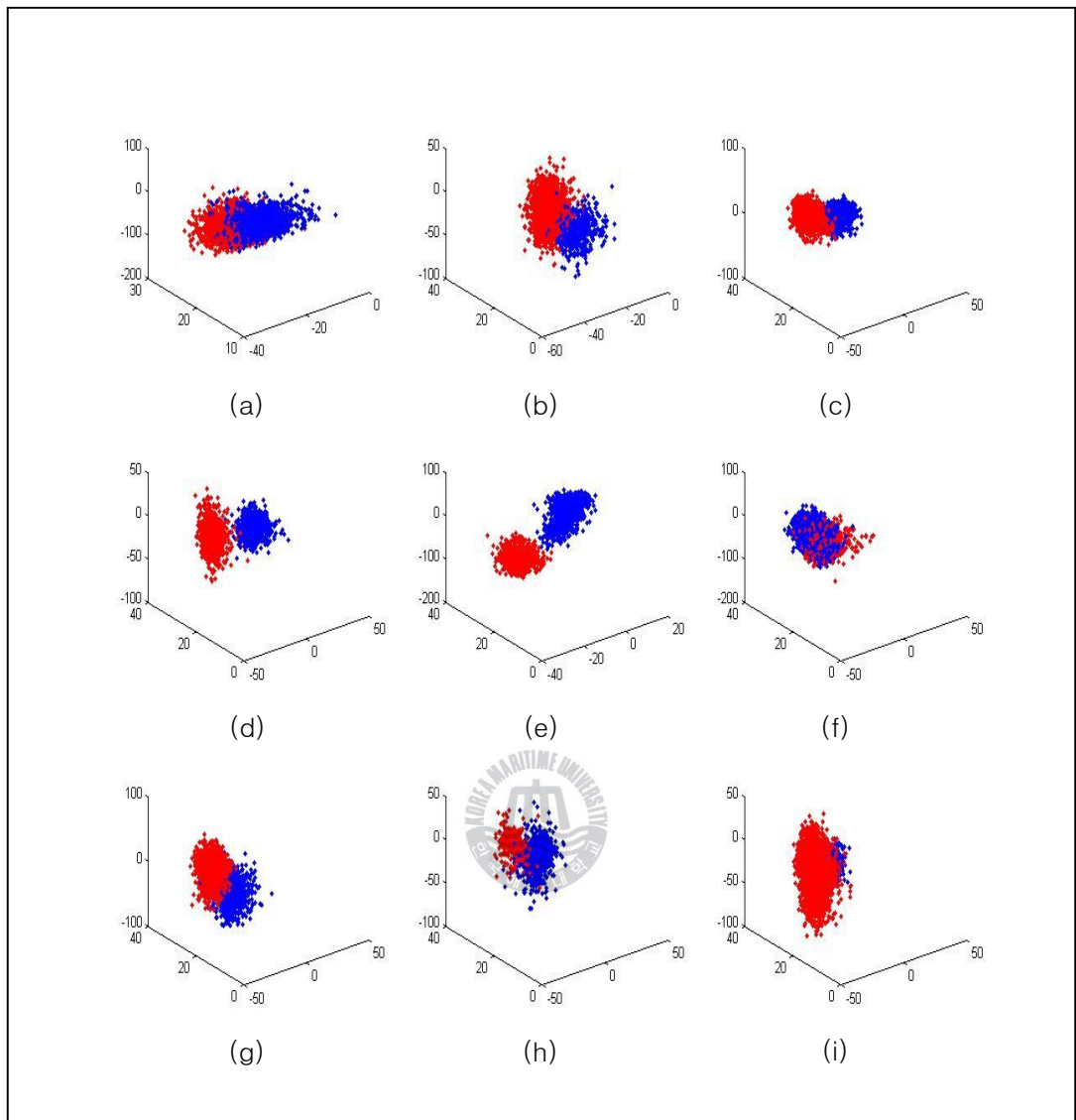


그림 5.7 LDA를 적용한 모음과 자음의 특징벡터의 분포  
 Figure 5.7 Distribution of feature vectors of vowel and consonant after transformation by the LDA

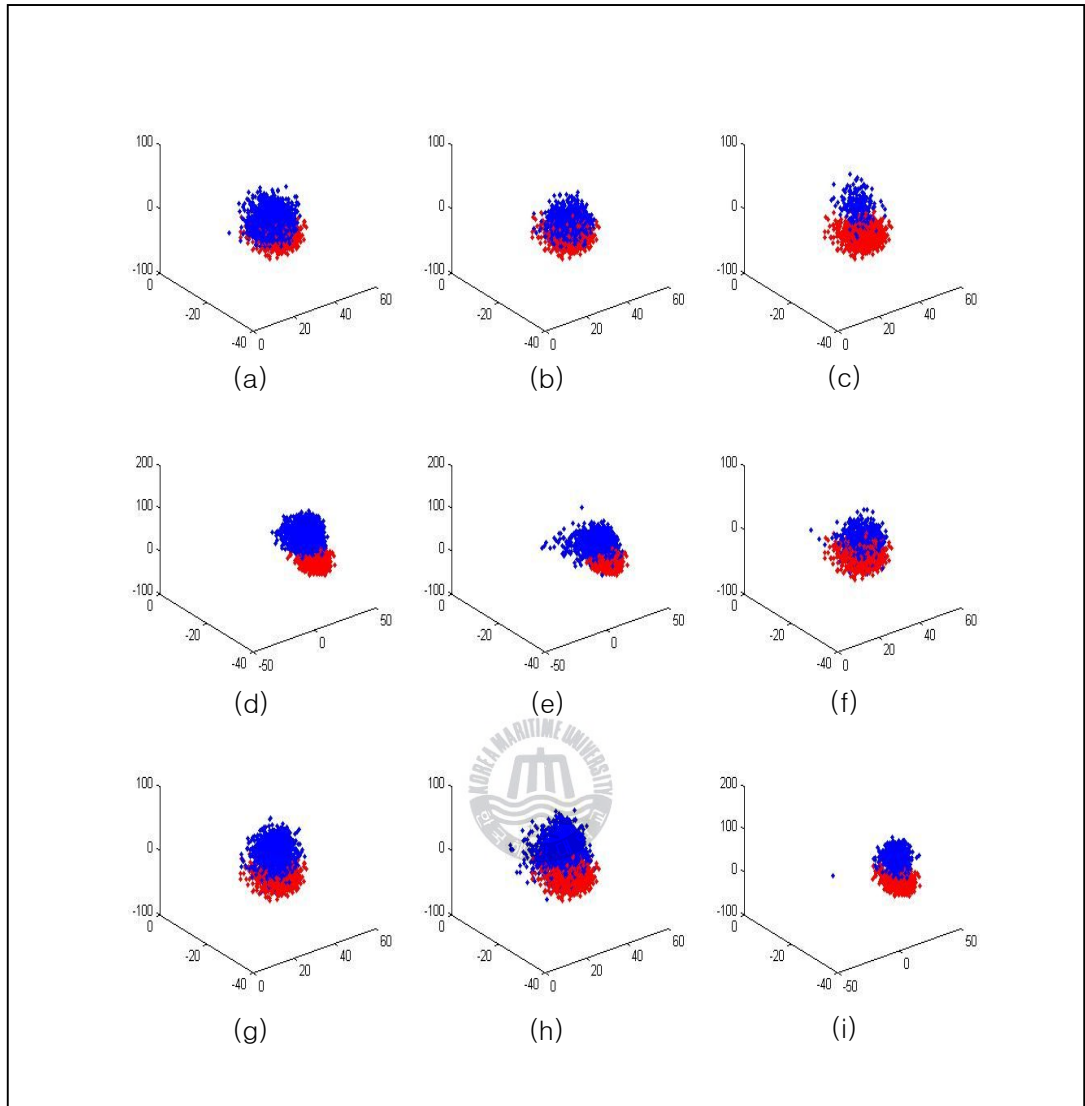


그림 5.8 Li 방법을 적용한 모음 상호간의 특징벡터의 분포  
 Figure 5.8 Distribution of feature vectors of vowels after transformation by the Li's method

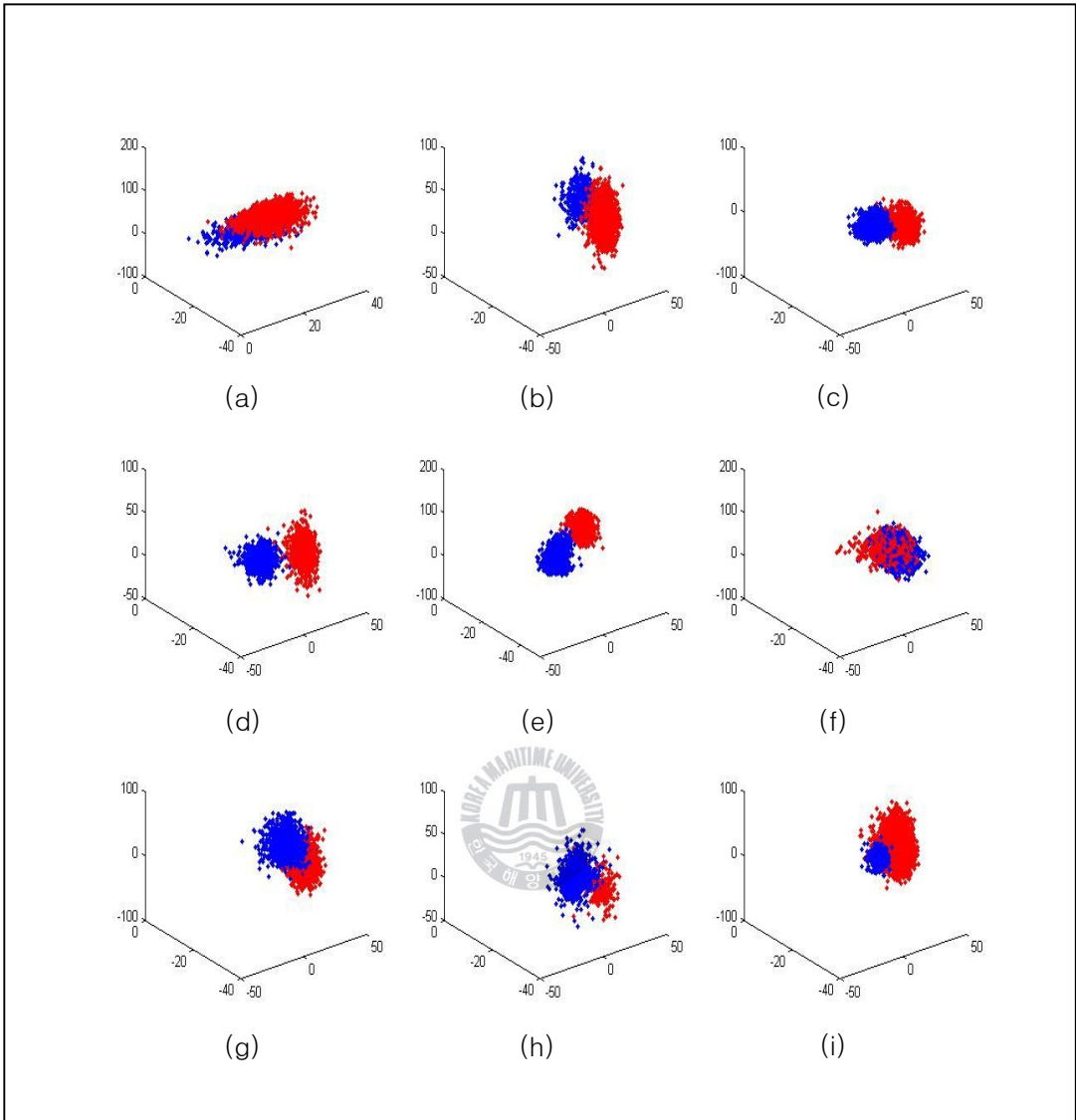


그림 5.9 Li의 방법을 적용한 모음과 자음의 특징벡터의 분포  
 Figure 5.9 Distribution of feature vectors of vowel and consonant after transformation by the Li's method

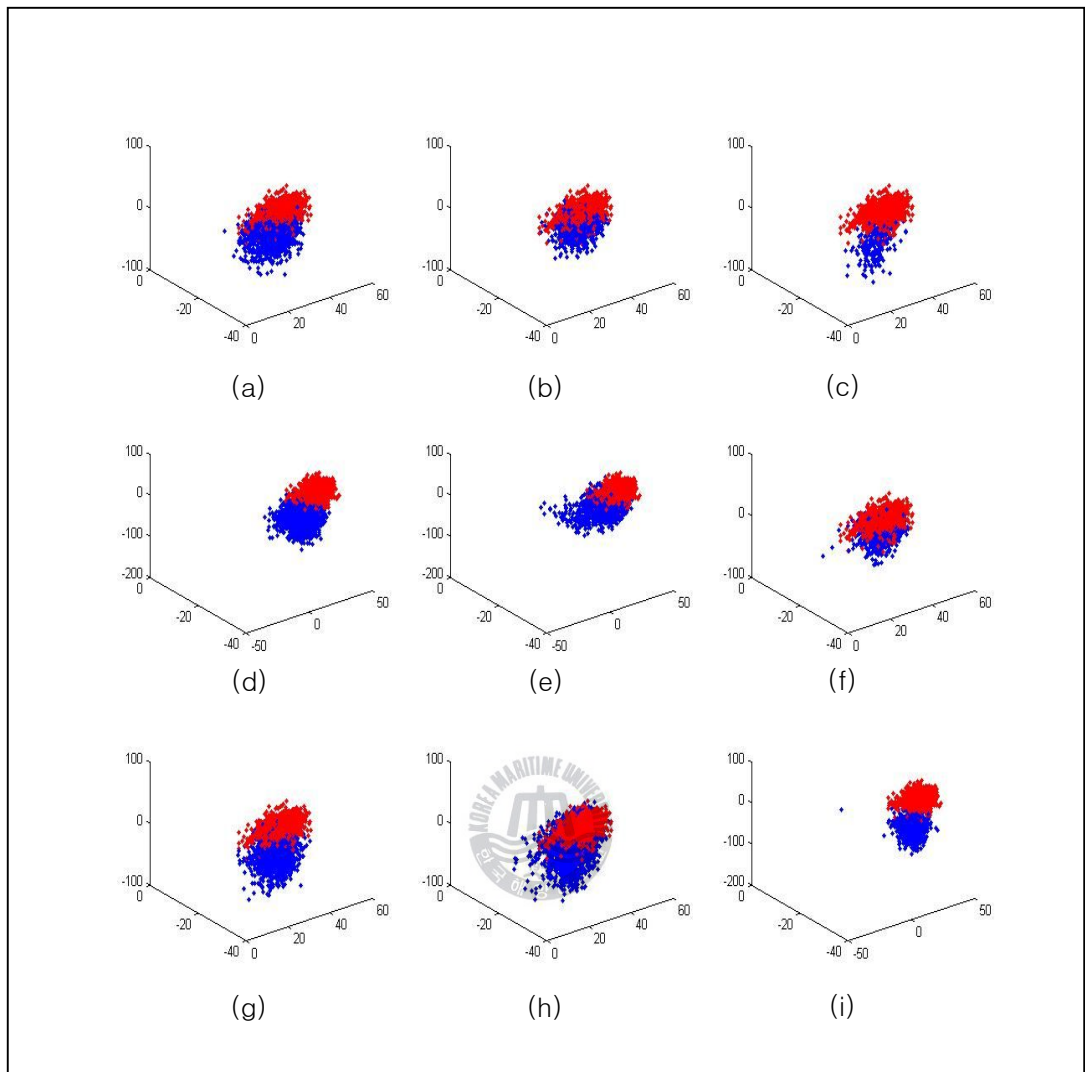


그림 5.10 제안한 방법을 적용한 모음 상호간의 특징벡터의 분포  
 Figure 5.10 Distribution of feature vectors of vowels after transformation by the proposed method

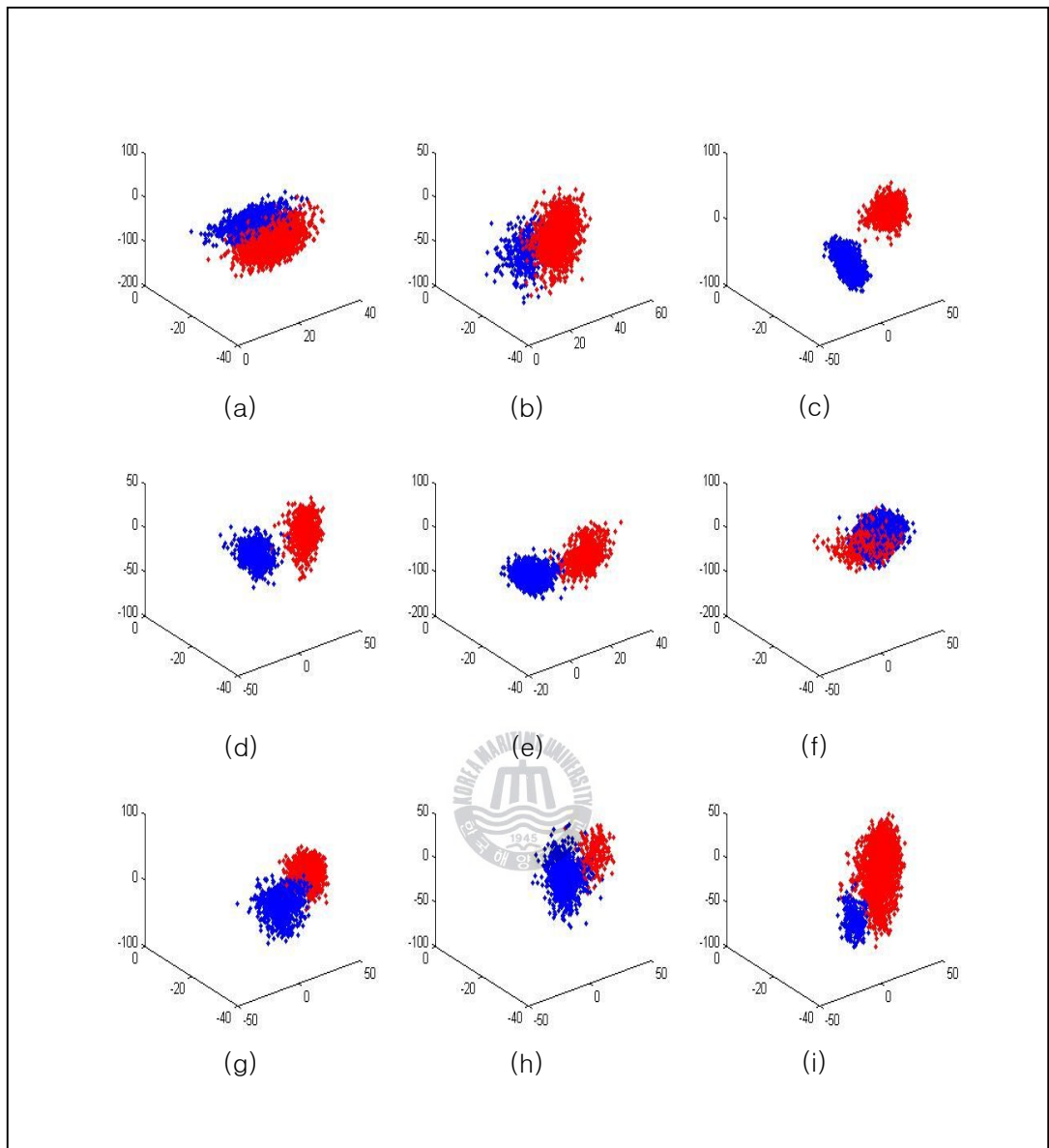


그림 5.11 제안한 방법을 적용한 모음과 자음의 특징벡터의 분포  
 Figure 5.11 Distribution of feature vectors of vowel and consonant after transformation by the proposed method

그림 5.10과 5.11은 이 논문에서 제안한 방법으로 특징벡터를 선형 변환하여, 모음 상호간의 특징 벡터의 분포와 모음과 자음 상호간의 분포를 도식한 것이고, 선형 변환 행렬은 Li의 방법과 동일하게 선형 판별

분석법을 이용하여 초기화 한 후에, 4.2절에 소개한 반복 학습 방법에 의해 최적화하였다. 이 그림들에서는 그림 5.4와 5.5의 주요 성분분석법보다 음소 사이의 거리는 멀어지고, 각 음소 내부의 분산은 감소하였음을 알 수 있고, 그림 5.6과 5.7의 선형 판별분석법과 그림 5.9와 5.10의 Li의 방법보다는 음소 사이의 거리가 멀어 졌음을 직관적으로 알 수 있다. 특히 그림 5.11의 (a), (c), (d), (e), (i)에서는 기타의 다른 방법들에 비해 음소의 변별력이 많이 개선되었음을 알 수 있다.

표 5.3 음소 단위의 클러스터링 결과  
Table 5.3 Clustering result of phone unit

차원 \ 종류	13 차원			26 차원		
	Fisher ratio	Hit ratio(%)	Error ratio(%)	Fisher ratio	Hit ratio(%)	Error ratio(%)
MFCC	1.55	2.36	97.64	1.46	1.74	98.26
PCA	1.58	2.17	97.83	1.47	1.86	98.14
LDA	2.30	3.92	96.08	1.47	1.82	98.18
Li	2.30	2.99	97.01	1.47	4.63	95.37
Proposed	2.31	5.20	94.80	1.47	5.00	95.00

표 5.3은 변별적 변환방법을 적용하지 않은 특징벡터, 주요 성분분석법, 선형 판별분석법과 같이 반복 학습을 통하지 않은 변별적 변환방법을 적용한 특징벡터, 그리고 Li의 방법과 이 논문에서 제안한 방법과 같은 반복 학습에 의한 변별적 변환방법을 적용한 특징벡터를 식 (5.2)를 이용하여 음소 단위로 클러스터링한 다음, 그 결과를 13차원과 26차원의 특징벡터 각각에 대하여 표시한 것이다. 이 표에서 Hit ratio는 올바르게 클러스터링한 결과를 나타내고, Error ratio는 오인한 결과를 나타낸다. 표 5.3에서는 13차원과 26차원의 특징벡터 모두에서 본 논문에서 제안한 방법이 다른 방법에 비해 Fisher ratio와 Hit ratio 모두가 다른

방법에 비해 높음을 알 수 있다. 그러나 이 표에 의하면, 모든 방법에서 Hit ratio가 전체적으로 낮게 나타나는데, 이는 음성 데이터가 시간을 동반한 데이터인데, 클러스터링 과정에서 시간을 고려하지 않아 발생한 현상으로 사료된다.

### 5.3 음소 단위의 인식 실험

제안 방법의 성능을 평가하기 위해 음소 단위의 음성 인식 실험을 수행하는 과정은 그림 5.12와 같고, 이 과정은 변별적 변환방법에 의해서 특징벡터를 선형 변환하는 행렬을 구하는 과정, 음소 단위의 인식기를 학습하는 과정, 그리고 입력되는 특징 벡터들로부터 음소를 인식하는 과정으로 구성된다. 변별적 변환과정은 먼저, training data set I을 이용하여 특징 벡터를 선형 변환하는 행렬을 주요 성분분석법, 선형 판별분석법, Li의 방법, 또는 이 논문에서 제안한 방법으로 구하는 과정을 말한다. 그리고 음소 단위의 인식기를 학습하는 과정은 training data set II를 이용하여 각 음소의 phoneme model을 생성하는 과정을 말하며, 인식 과정은 test data set을 phoneme recognizer에 입력하여, 입력된 데이터와 가장 잘 매칭이 되는 음소 모델을 찾는 과정이다.

이 논문에서는 그림 5.12의 음소 단위의 인식기 학습 과정과 인식 과정에 HTK(Hidden Markov Model Tool Kit) [11]를 사용하여 인식기의 학습과 인식 실험을 하였으며, 실험에 사용한 HMM은 상태 개수가 5개인 triphone model이다. 그리고 5.1절에 언급한 48개의 음소 집합을 Lee 등 [34,35]이 사용한 39개 음소 집합으로 변환하여 인식 실험을 수행하였고, 인식 실험에 사용한 특징 벡터는 13차원의 MFCC 계수와 26차원의 MFCC 계수이다.

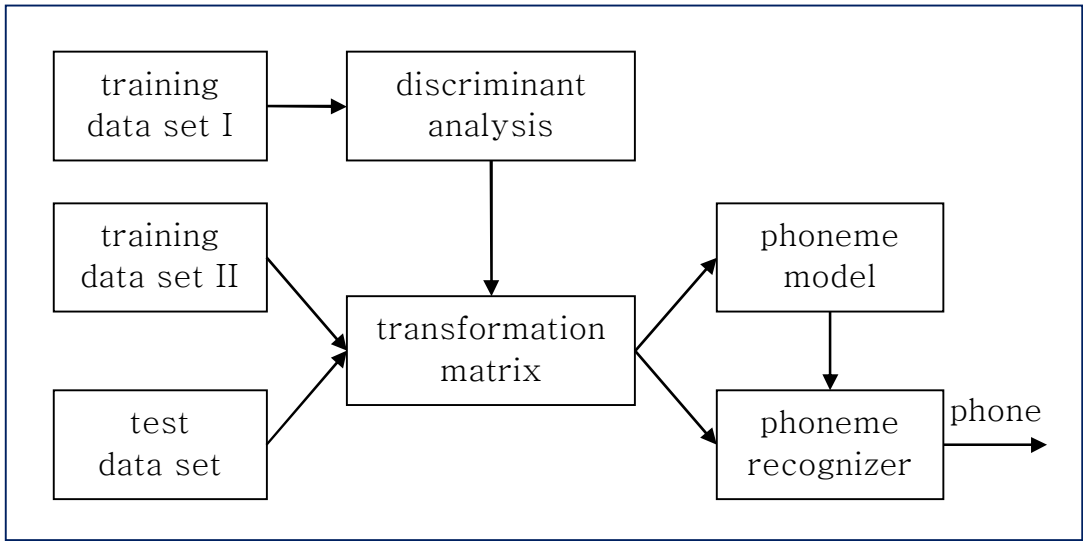


그림 5.12 음소 인식 과정  
Figure 5.12 Phone recognition procedure

표 5.4와 5.5는 변별적 변환방법을 적용하지 않은 원시 MFCC 특징 벡터와 변별적 변환방법을 적용하여 구한 MFCC 특징 벡터를 이용하여 음소 단위의 인식 실험 결과를 각 음소 별로 나타낸 것으로, insertion error와 deletion error를 고려하지 않은 것이다. 그림 5.13과 5.14는 표 5.4과 5.5를 그래프로 도시한 것이다. 그리고 표 5.4과 그림 5.13은 13차원의 특징벡터를 사용하여 얻은 결과이고, 표 5.5와 그림 5.14는 26차원의 특징벡터를 이용하여 얻은 결과이다.

표 5.4 13차원의 특징벡터에 대한 음소 각각의 인식률  
Table 5.4 Recognition rate of each phone on 13 dimensional feature vectors

종류 Phone	MFCC (%)	PCA (%)	LDA (%)	Li' s method (%)	Proposed method (%)
-------------	----------	---------	---------	---------------------	------------------------



ae	60.6	61.2	62.9	63.3	63.2
ah	49.6	49.6	49.1	50.1	49.2
ao	62.4	61.9	69.8	70.5	70.8
aw	45.9	46.3	51.5	50.5	53.3
ay	49.4	57.4	66.1	65.2	66.5
b	67.5	59.7	72.8	73.6	73.4
ch	71.0	66.2	75.7	78.2	78.7
d	54.3	48.7	64.6	65.1	63.0
dh	60.2	60.5	51.9	53.9	53.4
dx	43.8	51.2	75.3	74.1	76.7
eh	36.8	40.6	45.5	46.0	43.6
el	58.8	56.4	61.8	62.5	63.8
en	57.1	48.7	59.1	59.2	59.2
er	60.9	60.8	64.0	64.2	64.5
ey	58.6	64.2	65.6	66.8	67.5
f	81.2	79	83.4	84.1	83.0
g	55.6	54.9	64.1	64.1	66.4
hh	64.5	65.1	76.1	75.0	75.0
ih	40.7	42.8	53.2	53.6	52.9
iy	50.9	54.9	64.4	64.4	64.5
jh	59.8	58.7	63.0	64.3	60.9
k	76.2	78.6	71.8	73.9	76.0
m	63.9	57.5	65.9	66.8	68.0
ng	64.7	60.3	64.0	64.4	64.4
ow	23.8	33.5	33.3	33.6	34.1
oy	47.2	47.0	63.9	63.1	62.5
p	69.2	75.0	69.5	71.4	71.4
r	64.3	61.9	67.7	68.5	69.2
s	74.4	79.3	80.9	81.3	82.7
sh	72.4	75.7	85.5	86.2	84.9
sil	86.6	83.9	93.8	93.7	93.6
t	56.8	58.0	64.8	60.2	61.8
th	62.9	57.5	47.7	46.9	48.6
uh	27.9	36.4	43	43.5	43.3
uw	47.6	47.3	56.7	57.4	58.5

v	70.0	57.2	57.5	57.1	56.0
w	76.4	72.2	80.7	81.0	80.9
y	71.2	69.4	74.5	74.2	76.2
z	67.7	63.8	63.4	63.5	63.3
<b>평균</b>	<b>59.30</b>	<b>59.05</b>	<b>64.73</b>	<b>65.01</b>	<b>65.25</b>

표 5.4와 5.5, 그림 5.14와 5.15에서 보이는 것과 같이 주요 성분분석법은 음성 인식을 위한 특징 벡터의 변별력 향상에는 도움이 되지 않음을 알 수 있고, 선형 판별분석법은 변별력 향상에 도움이 되는 것을 알 수 있는 데, 이는 주요 성분분석법이 선형 판별분석법과 달리 인식 단위인 클래스 정보를 이용하지 않고, 단지 원시 특징벡터와 선형 변환된 특징 벡터 사이의 오차만 최소화 했기 때문에 일어나는 현상이다. 그리고 Li의 방법과 본 논문에서 제안한 방법의 인식률이 선형 판별분석법 보다 높은 데, 이는 선형 판별분석법이 특징 벡터의 변별력을 최대화하지 못함을 보여주는 것이다. 또한 Li의 방법이 이 논문에서 제안한 방법보다 인식률이 낮은 이유는 클래스 상호간의 정보를 고려하지 않은 것이 원인으로 사료된다.

표 5.5 26차원의 특징벡터에 대한 음소 각각의 인식률  
 Table 5.5 Recognition rate of each phone on 26 dimensional feature vectors

종류 Phone	MFCC (%)	PCA (%)	LDA (%)	Li' s method (%)	Proposed method (%)
ae	67.5	66.8	70.6	71.3	72.3
ah	53.4	56.7	53.0	53.2	54.0
ao	75.7	71.3	74.2	74.5	75.6
aw	69.2	63.4	62.9	64.5	63.6
ay	74.8	74.6	78.4	79.6	79.7
b	76.2	68.9	78.6	78.4	78.2
ch	76.1	70.6	75.7	76.1	73.6
d	70.9	64.3	64.5	65.7	68.2
dh	53.9	60.3	57.3	58.2	58.9
dx	67.5	77.7	83.3	82.5	82.2
eh	40.7	41.8	46.2	46.0	46.5
el	67.1	66.3	65.9	67.0	66.7
en	62.3	64.9	63.9	64.6	64.8
er	67.3	64.9	61.0	62.5	63.3
ey	74.9	76.0	77.4	77.9	77.2
f	85.3	84.4	82.8	82.9	83.7
g	75.1	72.2	72.6	73.2	74.3
hh	81.9	72.7	83.9	85.0	84.4
ih	50.5	50.3	61.2	60.0	61.4
iy	75.8	70.9	73.9	74.3	74.8
jh	66.7	59.9	71.1	68.8	70.2
k	75.6	82.0	74.1	76.0	77.6
m	73.4	70.6	72.1	72.2	74.3
ng	74.3	72.0	76.8	75.8	75.4
ow	49.9	45.1	47.1	48.8	48.1
oy	78.5	66.7	71.7	71.4	73.5
p	76.3	74.7	71.5	71.9	72.6
r	72.7	67.6	75.0	73.6	72.8
s	82.5	81.8	78.6	80.7	81.3
sh	85.8	85.7	85.9	86.5	85.1
sil	93.6	90.5	94.8	95.0	94.6

t	64.7	63.7	68.0	65.1	65.9
th	58.4	57.1	48.5	53.1	48.8
uh	31.4	41.5	42.8	44.2	45.0
uw	57.3	55.8	56.2	57.6	58.2
v	68.8	64.9	62.0	62.0	62.4
w	81.9	79.6	77.3	78.0	80.3
y	80.9	78.7	78.0	78.5	78.8
z	65.2	67.5	71.7	65.2	66.2
<b>평균</b>	<b>69.33</b>	<b>67.80</b>	<b>69.50</b>	<b>69.78</b>	<b>70.11</b>

표 5.5와 그림 5.14에서는 인식률이 표 5.4와 그림 5.13의 인식률 보다 전체적으로 높은 것을 알 수 있는 데, 이는 앞선 연구들에서 알려진 바와 같이 정적이 특징벡터를 사용하는 것보다 정적인 특징 벡터와 미분 계수와 같은 동적인 특징벡터를 함께 사용하는 것이 인식률 향상에 도움이 된다는 것을 보여주는 것이다[2-6,11].



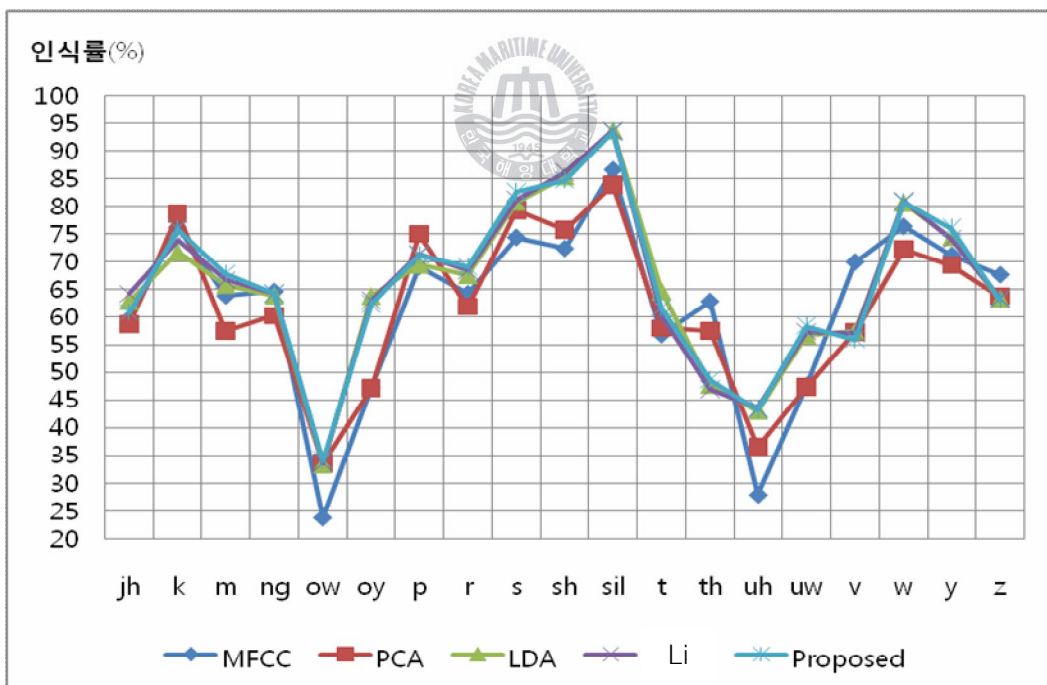
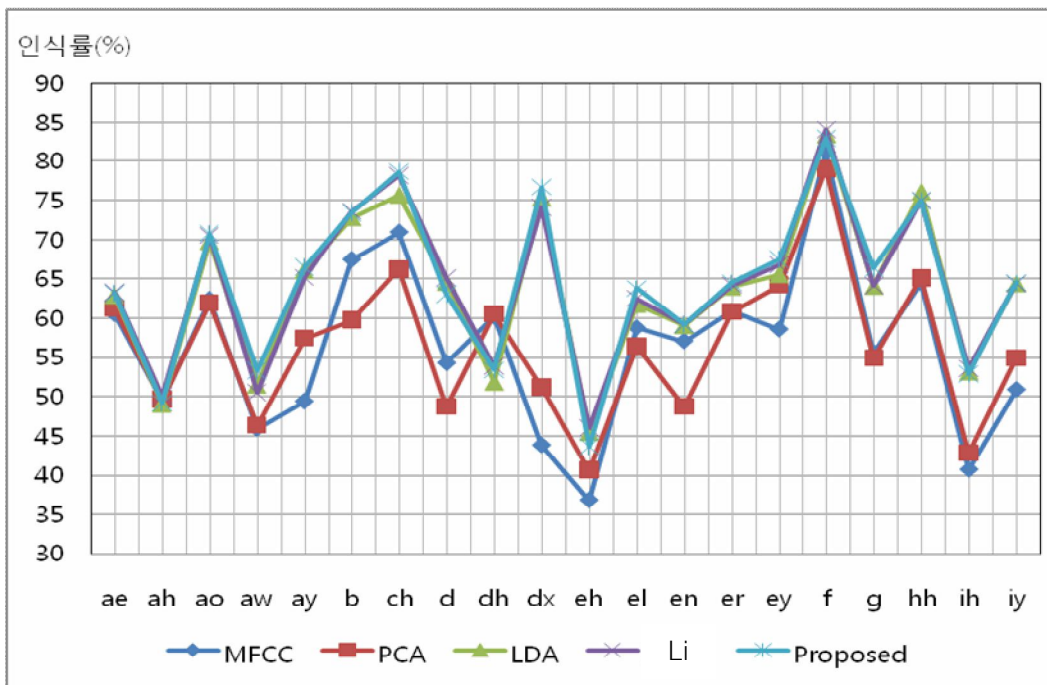


그림 5.13 13차원의 특징벡터에 대한 음소 각각의 인식률  
 Figure 5.13 Recognition rate of each phone on 13 dimensional feature vectors

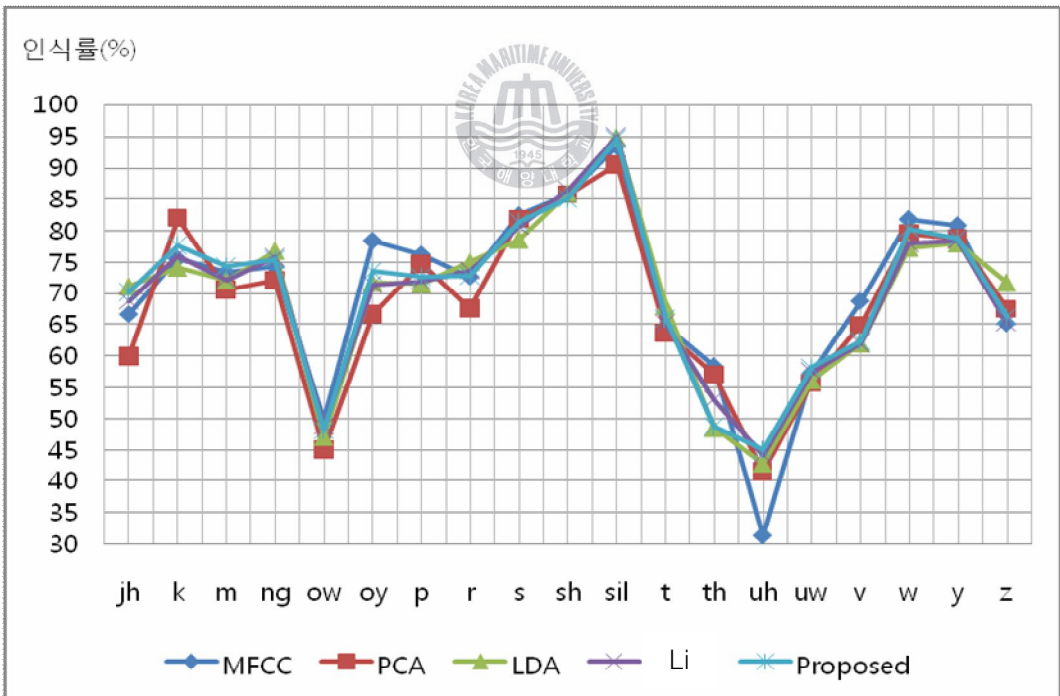
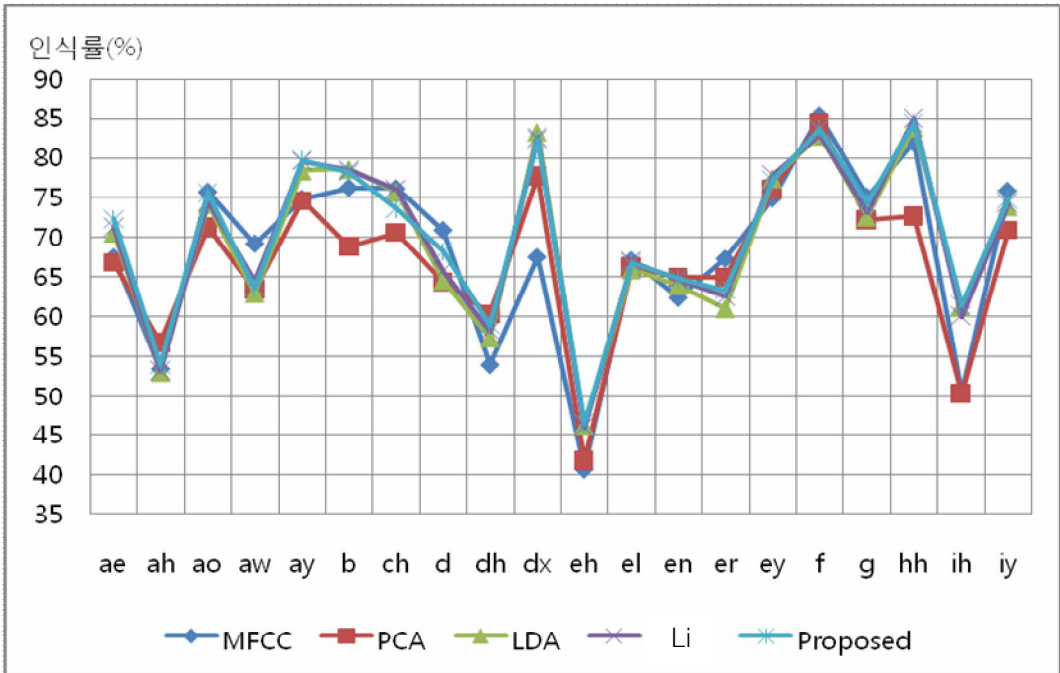


그림 5.14 26차원의 특징벡터에 대한 음소 각각의 인식률  
 Figure 5.14 Recognition rate of each phone on 26 dimensional feature vectors

표 5.6 13차원 특정벡터에 대한 음소의 인식률

Table 5.6 Recognition rate of phone on 13 dimensional feature vectors

	Correct (%)	Accuracy (%)	Hit	Insertion	Total
MFCC	56.05	45.77	35259	6472	62901
PCA	53.83	44.64	33857	5781	62901
LDA	63.59	54.41	40000	5774	62901
Li	63.71	54.73	40074	5650	62901
Proposed	63.83	55.14	40147	5464	62901

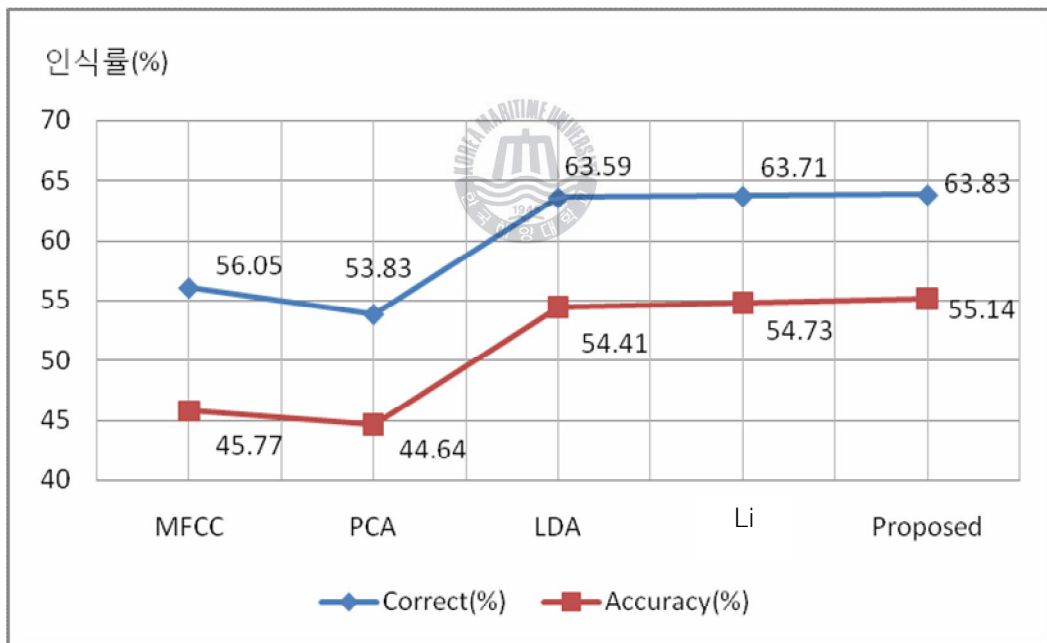


그림 5.15 13차원 특징벡터에 대한 음소 인식률

Figure 5.15 Recognition rate of phone on 13 dimensional feature vectors

표 5.7 26차원 특징벡터에 대한 음소 인식률

Table 5.7 Recognition rate of phone on 26 dimensional feature vectors

	Correct (%)	Accuracy (%)	Hit	Insertion	Total
MFCC	66.65	56.95	41922	6101	62901
PCA	63.80	54.51	40132	5843	62901
LDA	67.85	59.28	42679	5389	62901
Li	67.92	59.43	42720	5338	62901
Proposed	68.01	59.71	42779	5219	62901

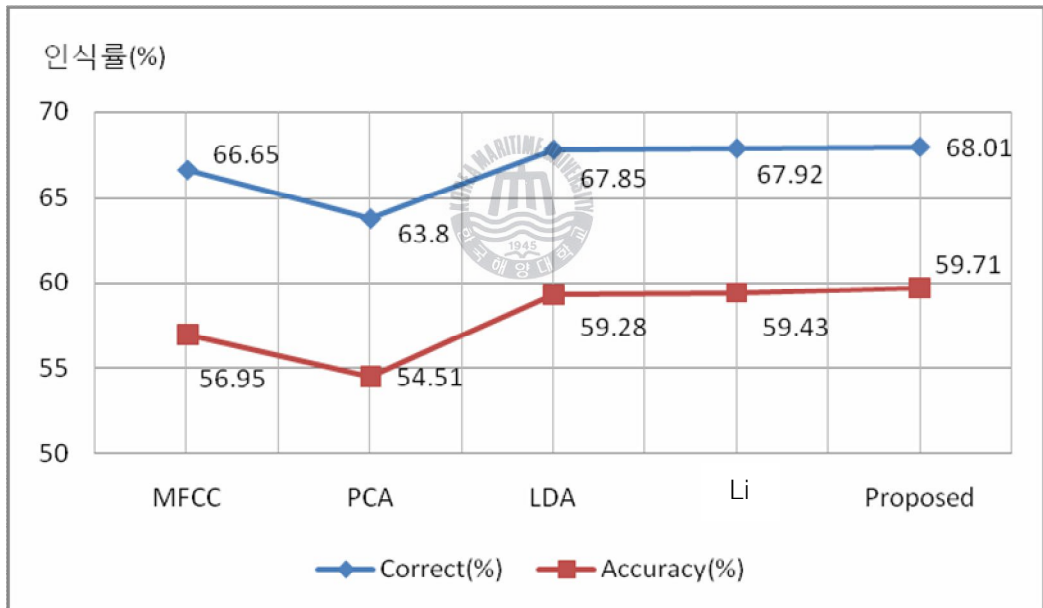


그림 5.16 26차원 특징벡터에 대한 음소 인식률

Figure 5.16 Recognition rate of phone on 26 dimensional feature vectors

표 5.6과 5.7은 음소 단위의 인식 실험 결과를 나타낸 것이고, 그림 5.15와 그림 5.16은 표 5.6과 5.7의 결과를 그래프로 표현한 것이다.



표 5.6과 5.7의 **Correct** 인식률과 **Accuracy** 인식률은 각각 식 (5.3)과 (5.4)에 의해서 구할 수 있다.

$$Correct = \frac{Hit}{Total} \times 100 \quad (5.3)$$

$$Accuracy = \frac{Hit - Insertion}{Total} \times 100 \quad (5.4)$$

식 (5.3)과 (5.4)에서 **Total**은 총 인식해야 할 음소 발화의 총 개수를, **Hit**는 올바르게 인식한 음소 발화의 개수를, **Insertion**은 실제로 존재하지 않은 음소 발화를 존재하는 것으로 인식한 발화의 개수를 말하며, 표 5.6과 5.7에서도 같은 의미를 가진다.

표 5.6과 5.7, 그림 5.15와 5.16의 결과에서는 앞서 언급한 음소 각각의 실험 결과와 마찬가지로 주요 성분분석법은 **Correct** 인식률과 **Accuracy** 인식률 모두에서 원시 특징벡터(MFCC)를 사용한 경우의 인식률보다 낮게 나타나, 인식률 개선에 도움이 되지 않는 것으로 보인다. 선형 판별분석법을 이용한 실험 결과에서는 13차원의 특징벡터인 경우, 원시 특징벡터를 사용한 경우의 인식률보다 **Correct** 인식률은 7.54%가 개선되고, **Accuracy** 인식률은 8.64%가 개선됨을, 26차원의 특징벡터의 경우에는 각각 1.2%와 2.33%가 개선 됨을 보였다. Li의 방법을 이용한 실험 결과에서는 13차원의 특징벡터인 경우, 원시 특징벡터를 사용한 경우의 인식률보다 **Correct** 인식률은 7.66%가 개선되고, **Accuracy** 인식률

은 8.96%가 개선됨을, 26차원의 특징벡터의 경우에는 각각 1.27%와 2.48%가 개선 됨을 보여, 선형 판별분석법 보다 성능이 좋음을 알 수 있었다. 본 논문에서 제안한 방법을 이용한 실험에서는 13차원의 특징벡터인 경우, 원시 특징벡터를 사용한 경우의 인식률보다 **Correct** 인식률은 7.78%가 개선되었고, **Accuracy** 인식률은 9.37%가 개선되었으며, 26 차원의 특징벡터의 경우에는 각각 1.36%와 2.76%가 개선되어, 제안한 방법이 특징벡터의 변별력과 인식기의 향상에 효과가 있음을 알 수 있었다.



## 제 6 장 결론

이 논문에서는 음성 인식기의 클래스 정보(인식 단위 또는 HMM의

상태)를 이용하여 인식기에 인가되는 특징벡터의 변별력을 개선함과 동시에 인식률을 향상시킬 수 있는 특징벡터의 선형 변환 방법을 제안하였다. 제안한 방법은 인식기의 클래스 정보에 기반한 상대 엔트로피를 이용하여 클래스 내부의 거리는 가깝게 하고, 클래스 상호간의 거리는 멀게 하는 특징벡터의 선형 변환 방법으로, 이 방법은 상대 엔트로피가 클수록 클래스 상호간의 유사도가 작아지므로, 클래스에 대한 분류가 용이해진다는 점에 착안한 것이다. 특징벡터의 변별력을 개선하는 변환 행렬은 클래스 상호간의 상대 엔트로피에 대한 평균인 divergence를 이용하여 목적 함수를 정의한 다음, 이 목적 함수를 natural gradient ascent 방법으로 최대화하여, 최적화된 선형 변환 행렬을 구했다. 그리고 목적 함수를 최대화하여 구한 선형 변환 행렬을 원시 특징벡터에 적용하여, 변별력이 향상된 새로운 특징벡터를 유도했다.

제안한 방법이 특징벡터의 변별력 개선과 음성 인식기의 성능 개선에 효과가 있는지를 검증하기 위해서, TIMIT 음성 데이터베이스를 이용하여 두 가지 실험을 하였으며, 실험 결과를 기존의 변별적 변환방법인 주요 성분분석법, 선형 판별분석법, Li의 방법과 비교 분석하였다. 첫 번째 실험은 음소 단위의 클러스터링 실험이고, 두 번째 실험은 음소의 인식 실험이다. 두 실험에 사용한 음소는 TIMIT 음성 데이터베이스의 64개 음소를 축소한 48개의 음소이고, 실험에 사용한 특징벡터는 13차원의 MFCC 계수와 13차원의 MFCC 계수에 미분 계수를 합한 26차원의 MFCC(13) + Delta MFCC(13) 계수에 프레임 길이가 5인 윈도우를 적용하여 구한 65차원과 130차원의 MFCC 계수와 MFCC + Delta MFCC 계수이다. 그리고 두 실험에서는 변별적 변환방법에 의해서 구한 선형 변환 행렬에 65차원의 MFCC 계수와 130차원의 MFCC 계수를 적용하여, 13차원과 26차원의 특징벡터로 변환한 다음, 이들 벡터를 각각의 실

험에 사용하였다.

실험 결과, 클러스터링 실험에서는 이 논문에서 제안한 방법의 Fisher ratio와 Hit ratio가 13차원의 특징벡터 경우, 변별적 변환방법을 적용하지 않은 특징벡터의 경우 보다 각각 0.77%와 2.85%가 증가하였고, 26차원의 특징벡터의 경우, 각각 0.01%과 3.27%가 증가하였다. 또한 제안한 방법의 성능이 기존의 변별적 변환방법인 주요 성분분석법, 선형 판별분석법, Li의 방법보다 우수함을 알 수 있었다. 음소 단위의 인식 실험에서는 제안한 방법의 인식률이 변별적 변환방법을 적용하지 않은 특징벡터를 사용한 경우 보다, 13차원의 특징벡터에서는 9.37%가 향상되고, 26차원의 특징벡터에서는 2.76%가 향상되어, 제안한 방법이 음성 인식기의 성능을 개선할 수 있음을 확인하였다. 또한 제안한 방법의 성능이 클러스터링 실험에서의 결과와 마찬가지로, 앞서 언급한 기존의 변별적 변환방법들보다 우수하다는 것도 알 수 있었다. 마지막으로, 두 실험 결과로 미루어 볼 때, 제안 방법을 음성 인식 이외의 영상 인식, 패턴 분류, 데이터 클러스터링, 데이터 압축 등의 분야에 적용할 수 있을 것으로 예상된다.

이 논문에서 제안한 방법은 음성 인식기의 인식 단위를 클래스 정보로 이용하여 특징벡터의 변별력과 인식률을 개선하였다. 그러나 실제 음성 인식기에서는 음성이 시간적인 정보를 동반하고 있다는 점에 착안하여, 인식 단위에 대한 정보뿐만 아니라, 음성의 시간적인 변화를 상태 정보로 모델링하여 사용하고 있다. 제안한 방법에 이러한 음성의 시간적인 변화를 나타내는 상태 정보를 포함하여 특징벡터의 변별력을 개선한다면, 음성 인식기의 모델을 보다 정확하게 모델링 할 수 있어, 보다 나은 결과가 나올 것으로 기대된다.



## 참고 문헌

- [1] 오영환, *음성언어정보처리*, 홍릉과학, 1997

- [2] L. Rabiner and B. H. Juang, *Fundamentals of Speech Recognition*, Prentice Hall, 1993.
- [3] R. P. Ramachandran and R. J. Mammone, *Modern Method of Speech Recognition*, Kluwer Academic, pp.159–183, 1995.
- [4] L. Rabiner, “A tutorial on hidden markov models and selected application in speech recognition” , *Proceeding of IEEE*, Vol. 77, No. 2, pp. 257–286, 1989.
- [5] S. E. Levinson, L. Rabiner and Sondhi, “An introduction to the application of the theory of probabilistic function of a markov process to automatic speech recognition” , *Bell System Technical Journal*, Vol. 62, No. 4, pp. 1035–1074, 1983.
- [6] S.E. Levinson, “Structural method in automatic speech recognition” , *Proceeding of IEEE*, Vol. 73, No. 11, pp. 1625–1650, 1985.
- [7] J. Makhoul, “Linear prediction : a tutorial review” , *Proceeding of IEEE*, Vol. 63, No. 4, pp. 561–580, 1975.
- [8] J. W. Picone, “Signal modeling techniques in speech recognition” , *Proceeding of IEEE*, Vol. 81, No. 9, pp. 1215–1247, 1993.
- [9] N. R. French and J. C. Steinberg, “Factors governing the intelligibility of speech sounds” , *Journal of the Acoustical Society of America*, Vol. 19, pp. 90–119, 1947.
- [10] 유 강주, “DHMM을 이용한 숫자음 인식의 Data fusion에 관한

- 연구” , 한국 해양대학교 석사 학위논문, 1998.
- [11] S. Young, D. Ollason and P. Woodland et al., *The HTK Book*, Microsoft Corporation, 1995.
- [12] S. B. Davis and P. Mermelstein, “Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences” , *IEEE Transactions on Acoustics, Speech and Signal Processing*, Vol. 28, No. 4, pp. 357 - 366, 1980.
- [13] J. L. Melsa and D. L. Cohn, *Decision and Estimation Theory*, McGraw-Hill, 1978.
- [14] D. O. Shayghnessy, *Speech Communication*, Addison-Wesely, 1990.
- [15] J. R. Deller, H. L. Hansen and J. G. Proakis, *Discrete-Time Processing of Speech Signals*, IEEE Press, 2000.
- [16] I. T. Jolliffe, *Principal Component Analysis*, Springer-Verlag, New York, 1986.
- [17] L. Breimam and R. Ihaka, “Nonlinear discriminant analysis via scaling and ACE” , *Technical Report, University of California*, Berkeley, 1984.
- [18] S. Mika, G. Ratsch and J. Weston et al., “Fisher discriminant analysis with kernels” , *IEEE Neural Networks for Signal Processing Workshop*, pp. 41-48, 1999.
- [19] S. Bermhard and Alexandors, “Nonlinear component analysis as kernel eigenvalue problem” , *Neural*

- Computation*, Vol. 10, No. 5, pp. 1299–1319, 1998.
- [20] X. Li and R. M. Stern, “Feature generation based on maximum classification probability for improved speech recognition” , *Proceeding of Eurospeech, Geneva*, pp. 845–848, 2003.
- [21] K. Torkkola, “Feature extraction by non-parametric Mutual Information maximization” , *Journal of Machine Learning Research*, Vol. 3, pp. 1415–1438, 2003.
- [22] E. Parzen, “On the estimation of probability density function and mode” , *The Annals of Mathematical Statistics*, Vol. 33, No. 3, pp. 1065–1076, 1962.
- [23] S. Roberts and R. Everson, *Independent Component Analysis Principles and Practice*, Cambridge University Press, 2001.
- [24] A. Cichocki and S. I. Amari, *Adaptive Blind Signal and Image Processing*, John Wiley & Sons, 2002.
- [25] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*, John Wiley & Sons, 1973.
- [26] S. Theodoridis and Konstantinos, *Pattern Recognition*, Academic Press, 1999.
- [27] J. T. Tou and R. C. Gonzalez, *Pattern Recognition Principles*, Addison–Wesley, 1974.
- [28] A. Papoulis, *Probability, Random Variables, and Stochastic*



*Processes*, McGraw–Hill, 1991.

- [29] N. Kumar and A. G. Andreou, “Heteroscedastic discriminant analysis and reduced rank HMMs for improved speech recognition” , *Speech Communication*, Vol. 25, No. 4, pp. 283–297, 1998.
- [30] T. W. Lee, *Independent Component Analysis Theory and Applications*, Kluwer Academic, 1998.
- [31] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, John Wiley & Sons, 1991.
- [32] A. Hyvarinen, J. Karhunen and E. Oja, *Independent Component Analysis*, John Wiley & Sons, 2001.
- [33] A. J. Bell and T. J. Sejnowski, “An information–maximization approach to blind source separation and blind deconvolution” , *Neural Computation*, Vol. 7, pp. 1129–1159, 1995.
- [34] K. F. Lee and H. W. Hon, “Speaker–independent phone recognition using hidden markov models” , *IEEE Transactions on Acoustics, Speech and Signal Processing*, Vol. 37, No. 11, pp. 1641–1648, 1989.
- [35] M. Antal, “Speaker independent phoneme classification in continuous speech” , *Studia University, Bsbes–Bolyai, Informatica*, Vol. 49, No. 2, pp. 55–64, 2004.
- [36] W. M. Fisher, G. R. Doddington and K. M. Goudie–Marshall, “The DARPA speech recognition research database:

pecifications and status” , *Proceedings of DARPA Workshop on Speech Recognition*, pp. 93–99, 1986.

- [37] S. S. Stevens and J. Volkman, “The relation of pitch of frequency: A revised scale” , *American Journal of Psychology*, Vol. 53, pp. 329–353, 1940.
- [38] G. Saon, M. Padmanabhan, R. Gopinath and S. Chen, “Maximum likelihood discriminant feature spaces” , *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 1129–1132, 2000.
- [39] R. A. Gopinath, “Maximum likelihood modeling with Gaussian distributions for classification” , *IEEE International Conference on Acoustics, Speech, and Signal Processing II*, pp. 661–664, 1998.

