HMM과 연결 숫자음의 후처리를 이용한 음성 다이얼링에 관한 연구

A Study on the Voice Dialing using HMM and Post Processing of the Connected Digits

양 진 우*, 김 순 협*
(Jin-Woo Yang*, Soon-Hyob Kim*)

요 약

본 논문은 HMM과 연결 숫자음의 후처리를 이용한 음성 다이얼링에 관한 연구이다. HMM(Hidden Markov Model)은 좋은 결과를 보이면서 현재 음성 인식 분야에서 널리 사용되는 알고리즘이다. 그러나, HMM의 학습 방법인 maximum likelihood estimation은 인식률을 극대화하는 모델의 파라메터 값을 생성하지 못하는 단점이 있다. 이러한 문제점을 보완하기 위하여 Segmental K-means 학습 과정에 후처리를 이용하여 인식 실험을 하였다.

한국어 연속 숫자음은 영어 연속 숫자음과 달리 연음 현상의 영향을 많이 받는다. Level Building 과정에서 연음에 의한 오류를 감소시키기 위해 연음에 의해 발생할 수 있는 단어를 별도의 모델로 추가하였다. 이렇게 추가된 단어 모델들에 대한 몇 가지 규칙을 인식 결과에 적용하여 출력을 다시 조정한다.

본 시스템은 TMS320C30 프로세서를 내장한 DSP 보드와 IBM PC 상에서 구현되었고, 표준 패턴은 실험실 잡음 환경에서 남성 화자 3명을 대상으로 작성하였다.

인식 실험 결과 21종 전화 번호 252개 데이타에 대하여 화자 종속의 경우 91.6%, 화자 독립의 경우 80.5%의 인식률을 나타내었다.

ABSTRACT

This paper is study on the voice dialing using HMM and post processing of the connected digits.

HMM algorithm is widely used in the speech recognition with a good result. But, the maximum likelihood estimation of HMM(Hidden Markov Model) training in the speech recognition does not lead to values which maximize recognition rate. To solve the problem, we applied the post processing to segmental K-means procedure are in the recognition experiment.

Korea connected digits are influenced by the prolongation more than English connected digits. To decrease the segmentation error in the level building algorithm, some word models which can be produced by the prolongation are added. Some rules for the added models are applied to the recognition result and it is updated.

The recognition system was implemented with DSP board having a TMS320C30 processor and IBM PC. The reference patterns were made by 3 male speakers in the noisy laboratory.

The recognition experiment was performed for 21 sort of telephone number, 252 data. The recognition rate was 6% in the speaker dependent, and 80.5% in the speaker independent recognition test.

I. 서 론

본 연구의 목적은 차세대 맨-머신 인터페이스로 각광받고 있는 음성 인식 분야에서 대표적인 응용이라고 할 수 있는 음성 다이얼링 시스템을 실용화할 수 있도록 인식률을 향상시키는데 있다.

몇 년전 까지 음성 인식 분야에서 주도적인 역할을 해오던 알고리즘은 Myers와 Rabiner가 제안한 DTW(Dynamic Time Warping)였다[1]. 그러나, 많은 계산량과

*광운대학교 전자계산기공학과, 신기술 연구소
Dept. of Computer Engineering & Institute of New Technology, Kwang Woon Univ.
접수일자 : 1995년 7월 28일

연속 음성 인식에 적용이 곤란하다는 단점 때문에 현재는 이러한 단점을 보완할 수 있는 HMM(Hidden Markov Model)이 음성 인식 시스템에서 주류를 이루고 있다. 이와 함께, 음성 인식에 적당한 신경망(Neural Network)을 개발하려는 연구도 활발히 진행 중이다.

음성 인식에서 HMM 알고리즘은 양자화된 데이타열의 확률 모델을 만들고, 이 모델들을 표준 패턴으로 하여 미지의 데이타에 대한 최대 확률값을 나타내는 모델을 찾는 역할을 한다. 그런데 HMM 알고리즘은 모델을 작성할 때 같은 카테고리(category)에 속하는 데이타들만을 사용하고 다른 카테고리의 모델과의 상대성은 전혀 고려하지 않기 때문에 비슷한 확률값을 출력하는 모델을 생성할 수가 있다.

본 논문에서는 이러한 단점을 보완하기 위하여, 선형 분류의 학습 절차와 유사한 후처리를 이용하였다. 이것은 이미 생성된 모델을 선정된 데이타들의 상대적인 값이 고려되도록 다시 학습시키는 과정이다. 이 방법을 연결어 인식 알고리즘인 Segmental K-means 알고리즘은 학습 알고리즘 과정에서 새로이 발생하는 모델과 단어 토큰에 적용하여 각 모델들을 재조정하였다. 학습 대상은 HMM 파라메터중 인식률에 가장 큰 영향을 미치는 출력 확률값이다. 학습시 출력 확률값은 선정된 학습 데이타를 고려하여 재조정되는 것이므로 학습데이타의 선정에 따라 인식률이 오히려 저하되는 경우가 있다. 따라서 학습 데이타 선정은 매우 중요한 의미를 가지며, 신중한 데이타 수집을 필요로 한다.

한국어 숫자음은 대부분 단음절이고 초성이 'ㅇ' 인 단어가 많아 연속 발음시 영어 숫자음에 비해 연음 현상을 많이 받는다. 연결어 인식 알고리즘인 Segmental K-means 과정 초기에 단독 숫자음을 표준 패턴으로 하여 연결 숫자음을 각각의 숫자음으로 분리 해내는 Level Building 과정을 거치게 되는데, 한국어 연결 숫자음의 경우 연음 현상 때문에 단독 숫자음 표준 패턴을 그대로 사용하는 것은 불합리하다.

이러한 점을 개선하기 위하여 숫자음 조합시에 발생할 수 있는 새로운 단어들에 대한 모델을 별도로 만들고, 이들의 결합 규칙을 적용하는 후처리 과정을 첨가시켜 인식된 결과를 규칙에 맞게 바꾸어 인식률을 향상시켰다.

Ⅱ. 인식 알고리즘

HMM 알고리즘에는 인식과 관련된 전향 알고리즘, 후향 알고리즘, Viterbi 알고리즘, 학습과 관련된 Baum-Welch 재추정 알고리즘이 있다[2].

(1) 전향 알고리즘 (2-1)

①$\alpha_1(i) = \pi_i b_i(O_1), \quad 1 \le i \le N$

②For $t = 1, 2, \ldots, T-1, \quad 1 \le j \le N$

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i)\, a_{ij} \right] b_j(O_t)$$

③$$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i)$$

(2) 후향 알고리즘 (2-2)

①$\beta_T(i) = 1, \quad 1 \le i \le N$

②For $t = T-1, T-2, \ldots, 1 \quad 1 \le i \le N$

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

③$$P(O|\lambda) = \sum_{j=1}^{N} \pi_j b_j(O_1)\, \beta_1(j)$$

(3) Viterbi 알고리즘 : 최적 상태열 (state sequence) 계산 (2-3)

①초기화

$\delta_1(i) = \pi_i b_i(O_1), \ 1 \le i \le N$

$\Psi_1(i) = 0$

②재귀적 반복

For $2 \le t \le T, \ 1 \le j \le N$

$\delta_t(j) = \max_i [\delta_{t-1}(i)\, a_{ij}]\, b_j(O_t)$

$\Psi_t(j) = \operatorname{argmax}_i [\delta_{t-1}(i)\, a_{ij}]$

③종료

$P^* = \max_i [\delta_T(i)]$

$i^*_T = \operatorname{argmax}_i [\delta_T(i)]$

④경로 역추적

For $t = T-1, T-2, \ldots, 1$

$i^*_t = \Psi_{t+1}(i^*_{t+1})$

(Ψ : backtracking factor)

(4) Baum-Welch 재추정 알고리즘 : $P(O|\lambda)$의 최대화를 위한 A, B, π의 조정 (2-4)

①$\pi^-_i = \gamma_1(i), \quad 1 \le i \le N$

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

$$= \frac{\text{상태 } i\text{에서 상태 } j\text{로 전이되는 수의 기대값}}{\text{상태 } i\text{로 부터 전이될 수 있는 수의 기대값}}$$

③$$b^-_j(k) = \frac{\sum_{t=1,\, O_t = v_k}^{T} \gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)}$$

$$= \frac{\text{상태 } j\text{에서 심볼 } v_k\text{의 발생할 횟수의 기대값}}{\text{상태 } j\text{에 존재하는 횟수의 기대값}}$$

여기서, $\gamma_t(i) = \dfrac{\alpha_t(i)\, \beta_t(i)}{P(O|\lambda)}$

$$\xi_t(i, j) = \frac{\alpha_t(i)\, a_{ij}\, b_j(O_{t+1})\, \beta_{t+1}(j)}{P(O|\lambda)}$$

2.1 HMM을 이용한 음성 인식

HMM을 이용한 음성 인식 시스템에서 각 대상 어휘들

에 대한 HMM들이 블랙 박스와 같은 역할을 하여 미지의 데이타를 각 모델에 적용했을 때 가장 큰 값의 확률을 나타내는 것을 인식된 것으로 한다. HMM의 성능에 영향을 미치는 것은 상태 전이 (state transition) 모델, 상태 수, 코드 워드 (code word) 수이다. 본 논문에서 상태 전이는 상태 자체에 시간적 순서를 부여할 수 있는 left-to-right 모델을 사용하였고, 상태수와 코드 워드 수는 대상어와 어휘수, 계산량을 고려하여 선정하였다.

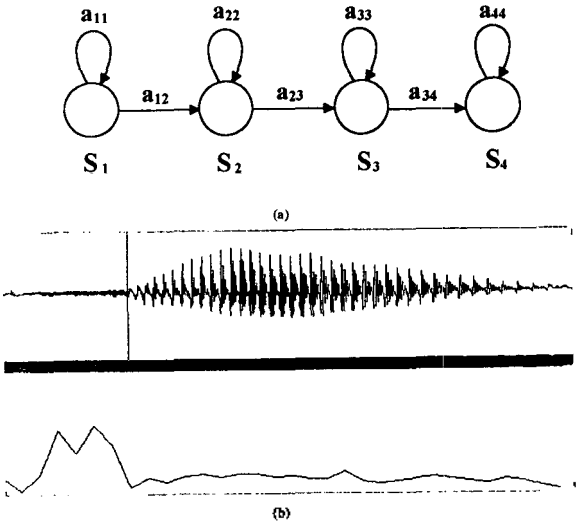

그림 2.1 (a)Left-to-right 상태 전이 모델 (b)음성 데이타 /팔/의 상태 전이

Fig. 2.1 (a) A left-to-right state transition model (b) the state transition of /phal/

HMM 음성 인식 시스템은 재추정 알고리즘을 이용한 모델 파라메터 학습 과정과 각 HMM에 음성 신호열을 입력하여 가장 큰 forward 값이나 Viterbi 값을 찾아내는 인식 과정으로 이루어진다. HMM을 이용한 기본적인 학습 및 인식 과정은 다음과 같다.

(1) 인식 대상어휘의 특징 벡터 생성

(2) 코드 북(code book) 생성 : K-means 집단화 알고리즘[6]

(3) HMM 생성 (학습) :

① 벡터 엔코딩 : 표준 패턴과 가장 가까운 코드 워드의 번호 출력

⇒ 관측열 생성 ($O_1, O_2, \ldots, O_T$)

② 모델 파라메터 초기화의 확률적 제한

$$\sum_i \pi_i = 1, \quad \sum_j a_{ij} = 1, \quad \sum_{k=1}^{M} b_j = 1$$

③ Baum-Welch reestimation 알고리즘 반복수행

⇒ 대상 어휘에 대한 HMM (A, B, π) 생성 ⇒ 모든 대상 어휘에 대해 반복

(4) 인식

미지의 시험패턴 입력을 전처리 및 벡터 엔코딩하여 시험패턴에 대한 관측열을 생성하여 각 모델에 적용한 후 Viterbi 값 계산

⇒ 가장 큰 값을 나타내는 모델의 인덱스 출력

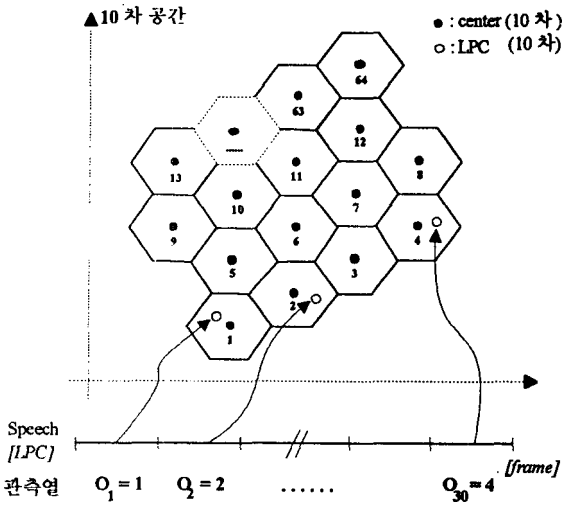

그림 2.2 엔코딩 과정

Fig. 2.2 Encoding

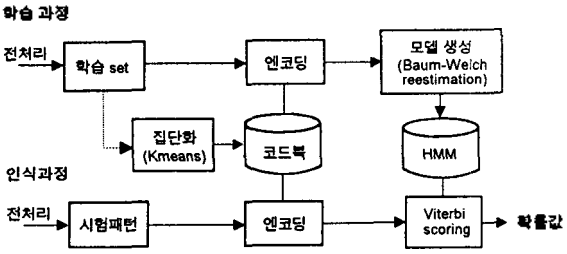

그림 2.3 HMM을 이용한 음성인식 블록도

Fig. 2.3 Block diagram of a speech recognizer based on a HMM

2.2 연결어 인식 알고리즘

2.2.1 Level Building 알고리즘 [1]

Level building 알고리즘은 연결어 인식에 사용되는 대표적인 알고리즘으로 단독어 표준 패턴들을 미지의 숫자음열과 비교하여 일치하는 최적 단어열을 결정한다. 즉, 단독어 표준 패턴들의 최적열을 결정한다. DTW에서는 단독어 표준 패턴으로 템플리트(template)를 사용하고, 이와 유사하게 HMM에서는 단독어 표준 패턴으로 확률 모델을 사용하고 시험 패턴과 표준 패턴과의 정합은 Viterbi 값을 이용한다. HMM을 이용한 Level building 알고리즘의 흐름은 다음과 같다.

① $P^v_1(t)$, $1 \le t \le T$: 레벨 1에서 표준 모델 v에 대한 시

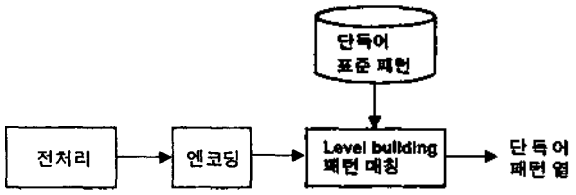

그림 2.4 Level building을 이용한 연결어 인식 블록도 [3]
Fig. 2.4 Block diagram of a connected word recognizer based on a level building

험 패턴의 프레임 t까지의 누적 확률 (2-5)

②$F^v_l(t)$, $1 \le t \le T$: 레벨 l의 시작 부분에서 경로가 출발하는 곳을 나타내는 backpointer (2-6)

각 level l의 끝에서 최적 모델을 구하기 위해 모델 v에 대한 최대화를 다음과 같은 알고리즘을 수행한다.

③$$P^B_l(t) = \max_{1 \le v \le V} P^v_l(t),\ 1 \le t \le T \quad (2\text{-}7)$$

$$W^B_l(t) = \arg\max_{1 \le v \le V} P^v_l(t),\ 1 \le t \le T$$: 최대 값을 나타내는 단어 모델의 번호 (2-8)

$F^B_l(t) = F_l^{W^B_l(t)}(t)$, $1 \le t \le T$: 최적 단어 모델의 backpointer (2-9)

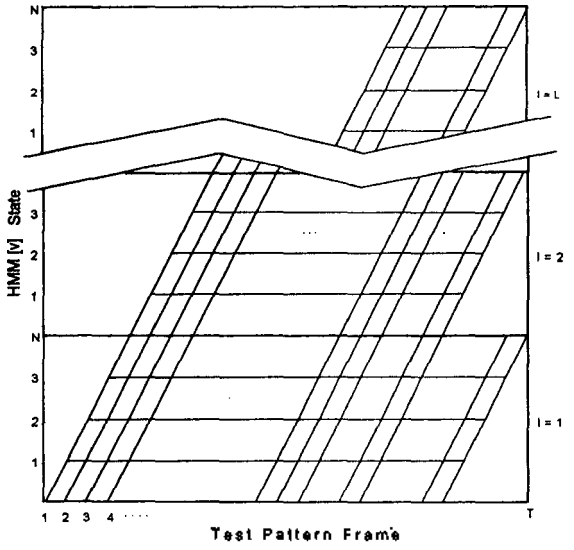

그림 2.5 HMM이 적용된 level building 과정 설명 [7]
Fig. 2.5 Illustration on the level building with HMM

각각의 새로운 레벨은 이전의 레벨에 속하는 선행 프레임에서 초기 최적 확률값으로 시작되고, 새로운 초기 프레임에서 시작되는 단어 모델과의 매칭에 의해 Viterbi 값을 증가시킨다. 이러한 과정을 레벨수가 기대되는 숫자수의 최대값에 일치할 때까지 반복한다. 각 레벨의 끝에서 확률 $P^B_l(T)$를 나타내면서 크기 l인 최적 단어열은 backpointer array $F^B_l(t)$에 의해 얻어지고, 전체적인 최적열은 식 (2-10)과 같이 가능한 모든 레벨에 대한 최대 $P^B_l(T)$ 이다.

$$P^* = \max_{1 \le v \le V} [P^B_l(T)] \quad (2\text{-}10)$$

2.2.2 Segmental K-means 알고리즘 [3]

초기 연결어 인식은 고립단어를 학습시켜 수행되어 왔다. 이것은 느린 발성음과 명확하게 발음된 정상 속도의 발성음의 경우 비교적 잘 동작하였으나 조음 현상이 나타나는 일반적인 발성음에는 적합하지 못한 방법이었다. 이러한 문제점을 해결하기 위해 고립단어의 표준 패턴을 학습시키는 것이 아니라 고립단어 표준 패턴을 이용하여 단어열로 부터 추출해낸 단어를 학습시키는 방법이 고안되었다. 이 학습 방법에서는 표준 패턴을 개선 하기 위해 추출된 패턴과 고립단어의 표준 패턴을 조합시킨다. 이 알고리즘의 단점은 같은 숫자음이라도 연결되었을 경우 소리가 달라진다는 것인데, 특히 한국어 숫자음의 경우 연음의 영향이 매우 크기 때문에 이러한 단점은 더욱 큰 문제를 일으킨다. 본 논문에서는 HMM을 작성할 때, 숫자음 10개에 대한 모델에 연속된 발음시에 발생할 수 있는 소리에 대한 모델도 몇 가지 추가하여 이러한 문제를 보완하였다.

①분절 (segmentation)

Level Building 알고리즘을 이용하여 학습될 단어열을 각각의 단어로 분절(segment) 하고 각각에 대한 단어 화일을 생성한다. 예를 들면 /940-5123/ 이라는 단어열을 /9/, /4/, ~, /3/으로 분절하여 각각을 화일로 저장한다.

②단어 패턴 building

템플리트 형태의 패턴에서는 K-means 집단화(clustering) 알고리즘을 사용하고, HMM 패턴에서는 상태 단위 분절 학습 과정(state-by-state segmental training procedure)을 이용하여 모델링될 단어 token들의 발생 확률을 최대화하는 파라메터를 선택한다.

③수렴성 조사

이론적으로는 현재의 표준 패턴 set을 이전의 것과 비교하여 차이가 근소하면 종료한다. 실제적으로는 시험 화일에 저장된 독립적인 연결 단어열들을 대상으로 인식 실험을 하여 갱신된 표준 패턴의 인식률이 개선되었으면 알고리즘을 반복하고 그렇지 않으면 종료한다.

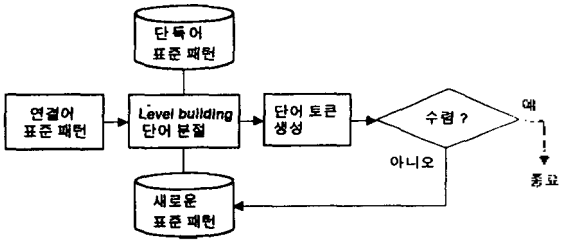

그림 2.6 Segmental K-means 알고리즘 블록도 [3]
Fig. 2.6 Block diagram of the Segmental-K-means procedure

학습은 목적 함수(objective function)를 개선하는 순환 과정으로 템플리트를 기본으로 하는 인식 알고리즘에서는 평균 거리를 줄이는 방향으로, 모델을 기본으로 하는 인식 알고리즘에서는 likelihood를 증가시키는 방향으로 이루어진다.

Ⅲ. 후처리

3.1 연음 현상

연음 현상이란 앞 음절의 받침에 모음으로 시작되는 허사가 이어질 때, 앞의 받침이 뒤 음절의 첫소리에 이어져 소리나는 현상이다. 즉 음절이 연속으로 발음될 때, 선행 음절의 종성 받침이 'ㅇ'으로 시작되는 후행 음절에 늘어붙어 발음되는 것이다. 이러한 연음 현상은 조음 현상과 더불어 오인식을 유발하는 대표적인 문제점으로 그림 3.1에서 처럼 단음절 발성때 나타나지 않는 변형음을 발생시킨다.

한국어 숫자음은 연속되었을 때, "35" (/사모/) 등과 같이 발음 특성상 많은 연음 현상이 발생하여 자동 음성 다이얼링 시스템과 같은 연결 숫자음 인식에 큰 문제로 작용한다. 영어 숫자음과 달리 한국어 숫자음은 모두 단음절이고, 10개의 숫자음 중에서 6개의 숫자음(/공/, /일/, /삼/, /육/, /칠/, /팔/)이 종성 받침을 갖고 있고, 4개의 숫자음(/일/, /이/, /오/, /육/)의 초성이 'ㅇ'으로 선행 음절 종성 받침의 영향을 받는다.

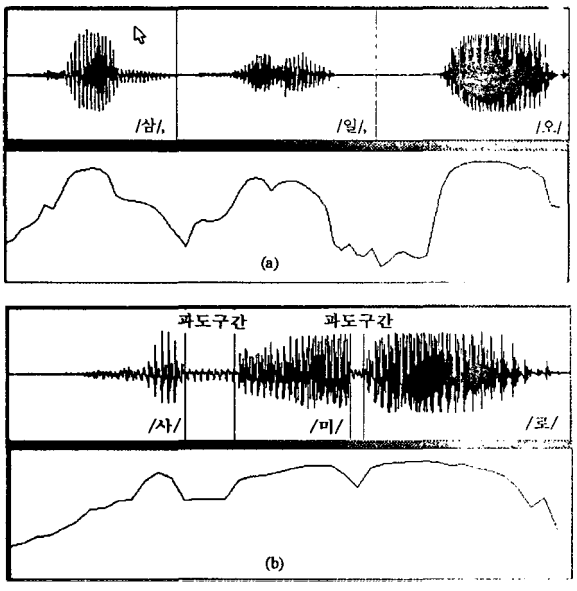

그림 3.1 (a) 단독 숫자음 /삼/, /일/, /오/의 음성 파형
(b) 연음 현상이 나타나는 연결 숫자음 /삼일오/의 음성 파형

Fig. 3.1 (a) The waveforms of isolated digits /sam/, /il/, /o:/
(b) A waveform of a connected digit /samilo:/ presenting the prolongation

본 논문의 목표가 음성 다이얼링에 있으므로 10개의 숫자음 외에도 전화 번호 발성시 사용되는 /에/의 발음을 추가하여 연음 현상을 고려하였다.

아래의 표 3.1은 연결 숫자음 발성시 개인의 발성 습관에 따라 발생할 수 있는 발음을 나타낸 것이다.

표 3.1 연결 숫자음에서 나타날 수 있는 연음 현상
Table 3.1 The prolongation in the connected digit speech

선행숫자	후행 숫자				
	1	2	5	6	예
0 /공/	공일 고-ㅇ-일	공이, 공니 고-ㅇ-이	공오 고-ㅇ-오	공육 공뉵	공에 고-ㅇ-에
1	일일, 일닐 일릴 일-ㄹ-일	일이, 일니 일리 일-ㄹ-이	일오 이로 일-ㄹ-오	일육 일뉵 일륙	일에, 일레 이레 일-ㄹ-에
3	삼일 사릴 사-ㅁ-일	삼이, 삼니 사미 사-ㅁ-이	삼오 사모 사-ㅁ-오	삼육 삼뉵 사뮤	삼에 사메 사-ㅁ-에
6	육일 유길 유-ㄱ-일	육이 유기 유-ㄱ-이	육오 유고 유-ㄱ-오	융육	육에 유게 유-ㄱ-에
7	칠일 치릴 치-ㄹ-일	칠이, 칠니 치리 치-ㄹ-이	칠오 치로, 치-ㄹ-오	칠육, 칠륙 치륙, 칠뉵	칠에 치레 치-ㄹ-에
8	팔일 파릴 파-ㄹ-일	팔이, 팔니 파리 파-ㄹ-이	팔오 파로 파-ㄹ-오	팔육 파륙	팔에 파레 파-ㄹ-에

3.2 연음 처리

위와 같은 연음 현상을 고려 하기 위하여 추가되어야 할 새로운 단어 모델은 다음과 같다. /일-ㄹ/, /유-ㄱ/, /치-ㄹ/, /파-ㄹ/, /고-ㅇ/, /모/, /미/, /밀/, /메/, /융/, /뉵/ 이고, 여기서, /모/, /미/, /밀/, /메/, /융/, /뉵/ 등은 그대로 발성된 것을 이용하고, /일-ㄹ/, /유-ㄱ/과 같은 발음은 /12/, /62/ 등의 자연스런 발음에서 수동 분절한 것을 이용한다.

추가된 모델들에 대해 일반적인 교정학습과 Segmental K-means 학습을 그대로 적용하여 연결어 표준 패턴을 작성하고, 인식 과정에서 이들 모델에 대한 몇 가지 규칙을 적용한다. 예를 들면, 모델 /모/에 대한 최종 인식 결과로 /5/, 모델 /미/에 대한 최종 인식 결과로 /2/를 출력하고, 모델 /모/, /미/, /밀/의 선행 숫자음이 /4/일 경우 /4/를 /3/으로 출력한다.

Ⅳ. 실험 및 고찰

4.1 표준 패턴 작성

전처리 과정은 다음과 같다.

- 아날로그 필터 : 70 Hz~4.5 KHz
- A/D : 10 KHz 샘플링, 16 bit 양자화
- 끝점 검출 : 자기상관(autocorrelation)의 0차 계수로 나타나는 에너지값 이용
- Pre-emphasis : $H(z) = 1 - 0.95z^{-1}$

· Window : Hamming window $W(n) = 0.54 - 0.46 \cos(\frac{2\pi n}{N-1})$

· LPC cepstrum 계수 : 10 차

단독 숫자음 표준 패턴은 연속 음성으로부터 연결어 인식을 위한 표준 패턴을 추출하기 위한 것이다. 연음 현상을 고려하여 실제 인식하고자 하는 숫자음 10개와 연음으로 나타날 수 있는 몇 가지 단어를 추가하여 표준 패턴으로 이용하였다. 표준 패턴은 서울 지역 20대 남성 화자 3명이 각각 3번씩 발음한 것으로 실험실 잡음 환경 하에서 실험 주체자의 감독 하에 작성되었다. 표준 패턴 작성시 연음 현상으로 추가된 단어 모델은 앞장에서 언급된 것에서 계산량을 고려하여 선정된 것이다.

본 논문에서는 막대한 계산량 때문에 제외되었으나 추가로 고려되어야 할 발음은 /공-ㅇ/, /응/, /늉/, /꽁/, /꾸/, /쌈/, /싸/ 등이다. 연음 현상을 처리하기 위해 추가한 모델들은 /12/, /62/ 등의 발음으로 부터 수동 분절(hand segment)하여 작성하였다.

본 실험에서 사용된 연속 숫자음 표준 패턴은 지난 90년도에 본 연구실에서 선정된 것으로서 연속해서 일어날 수 있는 모든 경우의 수를 조합하여 구성한 21 개의 연속 숫자음이다.

4.2 후처리를 이용한 인식 실험

4.2.1 교정 학습 방법 [4]

관측 심볼의 출력 확률값을 조종하기 위해서는 먼저 각 심볼인 코드 워드(code word)의 상대적인 발생 횟수를 구해야 한다. 이것을 빈도수[5]라 하며 식 (3-1)로 나타낼 수 있고, 실제로는 Baum-Welch reestimation 알고리즘을 이용하여 구한다. 이것은 결과적으로 한 음성 데이타에서 사용된 전체 코드 워드의 발생 횟수에 대한 각 코드 워드 발생 횟수의 상대적 비율이 된다.

b^j_{lm}의 빈도수(count) :

표 4.1 연음 현상을 고려하여 선정된 단어 모델

Table 4.1 Selected word models with prolongation

모 델	1	2	3	4	5	6
발 음	일 [il]	이 [i]	삼 [sam]	사 [sa]	오 [o :]	육 [yuk$^{\lceil}$]
모 델	7	8	9	10	11	12
발 음	칠 [c^{h}il]	팔 [p^{h}al]	구 [ku]	공 [kon]	일-ㄹ[il-r]	유-ㄱ[yu-k]
모 델	13	14	15	16	17	18
발 음	치-ㄹ[c^{h}i-r]	파-ㄹ[p^{h}a-r]	모 [mo]	미 [mi]	밀 [mil]	에 [e]

총 단독 숫자음 표준 패턴 = 3 화자 * 3 회 * 18 모델 = 162 개

표 4.2 연결 숫자음 표준 패턴

Table 4.2 Reference patterns for the connected digit

	전화번호	발 음	선	정	된	모	델	
1	512-0257	오일-ㄹ이에공이오칠	5	11	2	18	10	2	5	7
2	630-1349	육삼공에일삼사구	6	3	10	18	1	3	4	9
3	745-6780	칠사오에육칠팔공	7	4	5	18	6	7	8	10
4	826-9318	팔-ㄹ이유-ㄱ에구사빌팔	14	2	12	18	9	4	17	8
5	904-0371	구공사에공삼치-ㄹ일	9	10	4	18	10	3	13	1
6	910-2388	구일공에이삼팔팔	9	1	10	18	2	3	8	8
7	843-4616	팔사삼에사유-ㄱ일육	8	4	3	18	4	12	1	6
8	729-5522	치-ㄹ이구에오오이이	13	2	9	18	5	5	2	2
9	607-7641	육공치-ㄹ에칠육사일	6	10	13	18	7	6	4	1
10	358-8736	사모파-ㄹ에팔칠삼육	4	15	14	18	8	7	3	6
11	153-0599	이-ㄹ오삼에공오구구	11	5	3	18	10	5	9	9
12	270-9483	이칠공에구사팔삼	2	7	10	18	9	4	8	3
13	396-0011	삼구유-ㄱ에공공일일	3	9	12	18	10	10	1	1
14	408-6281	사공파-ㄹ에유-ㄱ이파-ㄱ일	4	10	14	18	12	2	14	1
15	689-6542	육팔구에유-ㄱ오사이	6	8	9	18	12	5	4	2
16	209-1921	이공구에일구이일	2	10	9	18	1	9	2	1
17	147-3324	일사치-ㄹ에삼사미사	1	4	13	18	3	4	16	4
18	986-5066	구팔유-ㄱ에오공육육	9	8	12	18	5	10	6	6
19	569-1775	오유-ㄱ구에일칠치-ㄹ오	5	12	9	18	1	7	13	5
20	795-9785	칠구오에구칠파ㄹ오	7	9	5	18	9	7	14	5
21	448-1234	사사파-ㄹ에일-ㄹ이삼사	4	4	14	18	11	2	3	4

$$\eta(b_{lm}|y,\lambda_j)=b_{lm}^{j}\frac{\partial L(y|\lambda_j)}{\partial b_{lm}}\Big|_{b=b_f}$$

$$=\frac{\sum_{I=1}^{S}P(y,I|\lambda_j)*\eta_{lm}(I)}{\sum_{I=1}^{S}P(y,I|\lambda_j)}\Big|_{b=b_f} \quad (3\text{-}1)$$

한 관측열(observation sequence)을 자신의 category에 해당하는 모델에 적용한 값과 다른 category의 모델에 적용한 값의 차이를 가중치로 설정하는데, 서로 독립적인 확률 분포를 결합할 때 어떤 데이타가 정확히 어떤 영향을 미치는지 알 수 없으므로 두 개의 상수값 β_W(within-class learning rate), β_B(between-class learning rate)를 설정하여 가중치의 크기를 실험적으로 조절한다. β_W는 학습하고자 하는 category의 데이타중에서 다른 catagory로 틀리게 인식되는 데이타들을 얼만큼 고려할 것인가를 결정하는 상수이고, β_B는 다른 category의 데이타중에서 학습하고자 하는 category로 틀리게 인식되는 데이타들을 얼만큼 고려할 것인가를 결정하는 상수이다.

인식이 정확히 되었더라도 근소한 확률값의 차이로 인식에 성공했을 경우, 이와 같은 category의 어떤 데이타가 들어왔을 경우 틀리게 인식될 확률이 높다. 이러한 문제점을 고려하여 확률값의 차이가 어느 상수값(near-miss criterion) 이하를 나타내면 이 모델도 다시 학습시킨다. [4][5]

$$R=\log\left[\frac{P(y|\lambda_i)}{P(y|\lambda_j)}\right]=\log P(y|\lambda_j)-\log P(y|\lambda_j),$$

$$0\le R\le\delta \quad (3\text{-}2)$$

모든 모델에 대하여 다음과 같은 방법으로 심볼의 출력 확률값을 갱신한다.

그리고 최종 알고리즘은 다음과 같다.

$$b_{lm}^{*i}=\frac{\eta_{lm}^{*}}{\sum_{n}\eta_{ln}^{*}} \quad (3\text{-}3)$$

⇒ HMM i의 상태 l의 코드 워드 m의 새로운 출력 확률값

$$\eta_{lm}^{*}=\sum_{y\ni C_i}\eta(b_{lm}|y,\lambda_i+\beta_w\sum_{j=1,j\neq i}^{p}\sum_{y\in C_i}\eta^{*s}(b_{lm}|y,\lambda_i/\lambda_j)$$

$$-\beta_b\sum_{j=1,j\neq i}^{p}\sum_{y\in C_j}\eta^{*o}(b_{lm}|y,\lambda_i/\lambda_j) \quad (3\text{-}4)$$

① 항 : 학습 모델이 λ_i 일 때 λ_i의 category C_i에 속하는 음성 데이터 관측열 y에 대한 빈도수

② 항 : $\eta^{*s}(b_{lm}|y,\lambda_i/\lambda_j)=\gamma^{s}(y,\lambda_i/\lambda_j)\times\eta(b_{lm}|y,\lambda_i)$ (3-5)

$$\gamma^{s}(y,\lambda_i/\lambda_j)=\begin{cases}0 & , R>\delta_n\\ 1-R/\delta_n & , 0<R\le\delta_n\\ 1 & , R<0\end{cases} \quad (3\text{-}6)$$

β_W : within class constant

δ_n : near-miss constant, score 차이가 작은 데이타의 영향고려

$$R=R(y|\lambda_i/\lambda_j)=\log P(y|\lambda_i)-\log P(y|\lambda_j) \quad (3\text{-}7)$$

: 음성 데이터 관측열 y를 같은 category의 모델 i와 다른 category 모델 j에 적용 했을 때 두 확률값의 차이

③ 항 : $\eta^{*o}(b_{lm}|y,\lambda_i/\lambda_j)=\gamma^{o}(y,\lambda_i/\lambda_j)\times\eta(b_{lm}|y,\lambda_i)$ (3-8)

$$\gamma^{o}(y,\lambda_i/\lambda_j)=\begin{cases}0 & , R>\delta_n\\ 1-R/\delta_n & , 0<R\le\delta_n\\ 1 & , R<0\end{cases} \quad (3\text{-}9)$$

β_B : between class constant

$$R=R(y|\lambda_i/\lambda_i)=\log P(y|\lambda_j)-\log P(y|\lambda_i) \quad (3\text{-}10)$$

4.2.2 교정 학습 실험 결과

초기 표준 패턴으로 사용된 단독 숫자음의 인식률과 Segmental K-means 학습 과정에서 발생하는 단어 토큰들을 새로 생성된 모델에 적용했을 때의 인식률이다. 교정 학습[4]은 생성된 토큰중의 일부를 사용하였고 Segmental K-means 학습의 최종 단계에서 적용하였다. 단독 숫자음 인식에 사용된 데이타는 총 126개의 18종 음성이고, 연결어 인식에 사용된 데이타는 총 256개의 21종 전화 번호 음성으로 연결어 학습 과정에서 단어 토큰 확일을 발생시킨다.

표 4.3 단독 숫자음 인식 실험 결과
Table 4.3 Isolated digits recognition result

β_W	β_B	인식률(%)
1.0	1.0	81.0
0.0	0.01	91.8
0.02	0.01	93.8
0.001	0.001	90.8

표 4.4 단어 토큰 인식 실험 결과
Table 4.4 Word token recognition result

β_W	β_B	인식률(%)
1.0	1.0	75.9
0.01	0.01	91.0
0.001	0.001	87.9

그림 4.1은 후처리 하지 않았을 때 level building 분절과 후처리했을 때 level building 분절, 즉, 연음 모델을 적용하지 않았을 때와 적용했을 때의 연결 숫자음 /358-8736/이 level building에 의해 자동 분절된 것을 나타낸다. 그림 (a)는 연음 현상을 고려하지 않은 인식 알고리즘에 의해 /35/가 /45/로 인식되고, 그 과정에서 나타난 분절 결과이다. 그림 (b)는 연음 현상을 고려한 인식 알고리즘에 의해 모델 /4/, /15/, /8/, /18/, /8/, /7/, /3/, /6/ 이 인식 결과로 출력되고 후처리 과정에서 /15/에 의해 /4/가 /3/으로 바뀌어 최종적으로 /358-8736/을 출력했을 때의 분절된 결과이다.

그리고 표 4.5은 후처리를 하지 않았을 때의 인식 결과를 나타내고, 표 4.6는 후처리했을 때의 인식 결과를 나타낸 인식 실험 결과의 일부분이다. 표 4.7는 표준 패턴의 작성에 참가한 화자 종속 실험 결과이고, 표 4.8은 표준 패턴의 작성에 참가하지 않은 화자 독립 실험 결과에 대한 인식률이다.

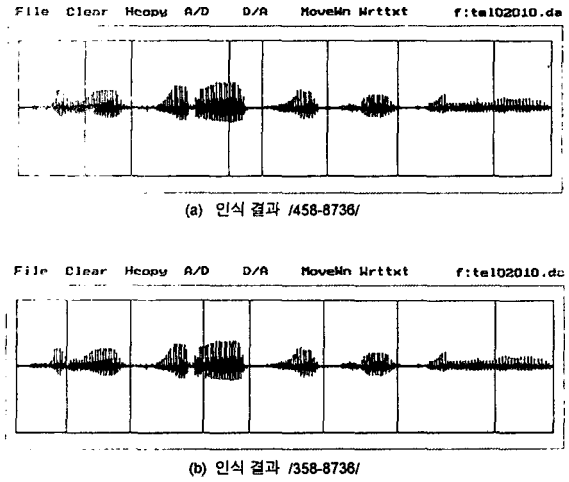

그림 4.1 (a) 후처리 하지 않았을 때 level building 분절
(b) 후처리했을 때 level building 분절

Fig 4.1 (a) Segmentation based on a level building without postprocessing
(b) Segmentation based on a level building with postprocessing

표 4.5 후처리를 하지 않았을 때의 인식 결과

Table 4.5 Recognition result without postprocessing

	전화번호	인식결과		전화번호	인식결과
1	512-0257	511-0257	12	270-9483	270-5483
2	630-1349	630-1349	13	396-0011	356-0011
3	745-6780	745-2780	14	408-6281	408-6287
4	826-9318	816-9318	15	689-6542	689-6542
5	904-0371	904-0377	16	209-1921	209-1922
6	910-2388	910-2388	17	147-3324	847-3324
7	843-4616	843-4616	18	986-5066	086-9066
8	729-5522	729-5222	19	569-1775	569-2775
9	607-7641	607-7641	20	795-9785	795-9785
10	358-8736	458-8736	21	448-1234	448-1134
11	153-0599	153-0595			

표 4.6 후처리했을 때 인식 결과

Table 4.6 Recognition result with postprocessing

	전화번호	인식결과		전화번호	인식결과
1	512-0257	511-0257	12	270-9483	270-9483
2	630-1349	630-1349	13	396-0011	396-0011
3	745-6780	745-6780	14	408-6281	408-6281
4	826-9318	826-3148	15	689-6542	689-6542
5	904-0371	904-0371	16	209-1921	209-2791
6	910-2388	910-2388	17	147-3324	147-3324
7	843-4616	843-4616	18	986-5066	586-5016
8	729-5522	729-1152	19	569-1775	569-1775
9	607-7641	607-7641	20	795-9785	795-2785
10	358-8736	358-8786	21	448-1234	448-1234
11	153-0599	153-0599			

표 4.7 화자 종속 실험 결과

Table 4.7 The recognition result in speaker dependent

화 자	화 자 1	화 자 2	화 자 3	종 합
인식률	93.9%	93.1%	87.8%	91.6%

표 4.8 화자 독립 실험 결과

Table 4.8 The recognition result in speaker independent

화 자	화 자 1	화 자 2	화 자 3	종 합
인식률	83%	79.4%	79%	80.5%

4.3 고 찰

실험 결과 교정 학습은 약간의 인식률을 개선 시켰다. 그러나 모델의 수, 상태 수, 코드 워드 수가 증가함에 따라 계산량의 증가가 극심하고, 파라메터를 일일이 실험을 통해 구해야 하는 문제점이 있다. 따라서, 이 방법은 숫자음 인식이나, 음소 단위 인식과 같은 소규모 어휘에 적용 하는 것이 바람직 하며, 단어 단위 대어휘 인식 시스템 등에 적용은 다소 어려우리라 생각된다.

또한, 적은 수의 화자를 대상으로 한 실험에서는 표준 패턴 작성에 참가하는 화자수를 늘이는 것이 교정 학습보다 인식률 및 계산 시간 면에서 더욱 효과적이었다. 그러므로, 표준 패턴 작성에 참가한 화자 수의 증가에 따른 인식률 증가가 임계치에 달했을 때, 이 방법을 적용하는 것이 바람직하다.

후처리에 대한 실험에서 인식률의 개선이 예상 보다 저조하였다. 그 원인은 첫째, 개개인의 발성 습관을 무시하고 일률적으로 연음 처리된 모델을 적용하여 학습시켰다는 점이다. 표준 패턴 작성에 참가한 화자들의 연결 숫자음 패턴을 조사해 보면, 개개인의 차이는 물론 같은 화자라도 발성 횟수에 따라 연음의 정도 차이가 존재 했다. 둘째, 계산량을 이유로 몇 가지 모델을 제외시켰다는 점이다. 특히, /공에/, /공일/ 등의 연결 숫자음에서 /공/이 /오/로 인식된 경우가 종종 있었는데, 이것은 /공/의 종

성 'ㅇ'이 후행 숫자음 /에/, /일/, /이/, /오/ 등의 초성에 연음되기 때문이다. 셋째, 비교 실험을 위해, 후처리 때문에 증가된 18단어들에 대해서도 후처리 과정이 없는 10단어와 동일한 상태수와 코드 워드수의 모델을 사용했다는 점이다. 즉, 모델이 10개일 때, 4개의 상태와 64개의 코드 워드는 인식에 충분한 수라고 할 수 있지만 후처리 때문에 18개로 증가된 모델에 대해서 다소 불충분한 수였다고 생각된다.

결과적으로 인식률이 증가되는 최대 화자수 설정, 연음 정도에 따른 정확한 모델의 선택, 발생 가능한 모든 연음 모델의 추가, 모델 수에 따른 최적 상태수와 코드 워드수의 설정 등이 이루어졌을 때 최대 인식률이 얻어지리라 생각된다.

그리고 TMS320C30 시스템 보드는 1982년 TMS320 계열의 DSP 프로세서가 출현한 이래 본격적인 32 bit 부동 소숫점 연산 능력을 가진 DSP 프로세서로 개발된 것이 TMS320C30 DSP 프로세서이다. TMS320C30 프로세서는 병렬 곱셈을 비롯, 논리연산을 한 실행 주기에 실행할 수 있도록 하는 구조를 갖고 있다. 또한 고성능의 사용하기 편리한 구조를 유지하기 위해서 병렬화를 이용하였고 특별한 DSP 명령들을 수행할 수 있다. 이러한 TMS320C30의 주요한 특징은 다음과 같다.

1) 60ns의 한 실행 주기는 33.3 MFLOPS(million floating point operations per second)와 16.7 MIPS (million instructions per second)로 구성된다.
2) 4K＊32 비트 단일 주기 이중 접근이 가능한 ROM을 내장하고 두개의 1K＊32 비트 단일 주기 이중 접근이 가능한 RAM을 내장한다. 그리고 64＊32 비트 명령어 캐쉬(cache)를 내장한다.
3) 32 비트의 명령어 및 데이타와 24 비트의 번지로 구성되고, 입력 32 비트, 출력 40 비트의 부동 소숫점/정수 곱셈 및 논리 연산을 한다.
4) 32 비트의 배럴 천이기(barrel shift)와 논리연산 및 곱셈의 병렬처리 기능을 한다.
5) DMA(Direct Memory Access) 기능의 내장하고, 여덟개의 extended-precision 레지스터의 내장하고 있다.

V. 결 론

HMM의 교정 학습은 각 모델들이 출력하는 확률값의 차이를 더욱 크게 하기도 하지만, 학습 대상의 선정에 따라서는 오히려 인식률이 저하되는 것이 관찰되었다. 이것은 한 카테고리의 데이타들 중에서 몇 개의 데이타만을 선정하여 학습 대상으로 하기 때문이다. 이 학습법이 모델을 만들기 위한 과정이라는 점에서, 학습 시간이 다소 길더라도 가능한 한 많은 수의 학습 데이타를 신중하게 선정하는 것이 바람직하다. 또한, 파라메터를 일일이 실험으로 구해야 하는 불편함을 해결하기 위해 이들 파라메터가 각 모델에 미치는 영향에 대한 연구와 그 영향에 따라 자동적으로 파라메터를 구해 내는 연구가 이루어져야 할 것이다.

후처리 실험 결과, 특히, /35/ 등의 발음에서 좋은 결과를 나타내어 연음 처리한 Level building의 타당성을 보여 주었다. 그러나, 개인적인 발음 습관에 따라서 연음 및 조음 현상의 정도 차이가 있으므로, 이 방법을 일률적으로 적용하는 것보다는 표준 패턴으로 선정된 데이터들에 대해 적응적으로 적용하는 것이 바람직하다. 그러므로써, 정확한 기준이 마련된다면 인식률의 향상을 기대할 수 있을 것이다.

여러 연구에 따르면, 한국어 연결 숫자음의 인식률은 다른 인식 대상에 비하여 상대적으로 저조한 인식률을 보인다. 이것은 한국어 숫자음이 모두 단음절이라는 점 외에도 숫자음 조합에는 규칙성이 존재하지 않는다는 점 때문이다. 따라서, 인식률 개선을 위한 연구와 아울러 개개인이 필요한 전화 번호가 한정되어 있다는 점을 이용한 실용화 연구가 병행되어야 할 것이다.

참 고 문 헌

1. C.S. Myers and L.R. Rabiner, "Connected digit recognition using a level building DTW algorithm," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-29, pp. 351-363, Jun 1981.
2. L.R. Rabiner and B.H. Juang, "An Introduction to Hidden Markov Models," IEEE ASSP magazine, pp4-17, January 1986.
3. L.R. Rabiner, J.G. Wilpon, and B. H. Juang, "A segmental k-means training procedure for connected word recognition based on whole word reference patterns," AT&T Tech, J., vol. 65, no.3 pp.21-31, May/June 1986.
4. T.H. Applebaum, B.A.Hanson, "Enhancing The Discrimination of Speaker Independent Hidden Markov Model with Corrective Training," ICASSP, p.S6.13, 1989.
5. Bahl.L.R., P.F.Brown, P.V.deSouza and R.L.Mercer, "A New Algorithm for the Estimation of Hidden Markov Model Parameters," ICASSP, p.S11.2, 1988.
6. J.G. Wilpon, L.R. Rabiner, "A modified K-Means Clustering Algorithms for use in Isolated Word Recognition," IEEE Trans. on Acoust. Speech and Signal Proc., Vol. ASSP-33, No.3, June 1985.
7. X.D. Huang, Y.Ariki, M.A. Jack, *Hidden Markov Models for Speech Recognition*, Edinburgh University Press, 1990.

▲양진우(Jin-Woo Yang)
한국음향학회지 제14권 3호 참조

▲김순협(Soon-Hyob Kim)
한국음향학회지 제14권 3호 참조