論文 90-27-9-20

MSVQ를 이용한 HMM에 의한 단독어 인식

(Isolated Word Recognition By HMM using Multisection MSVQ)

安 泰 玉*, 邊 龍 圭*, 金 淳 協*

(Tae Ock Ann, Yong Kyu Byun, and Sun Hyub Kim)

要 約

본 연구는 단독어를 multisection VQ (vector quantization) 와 HMM (hidden markov model) 을 이용해 인식하였다. 인식 어휘로는 3 명의 화자에 의해 5 번 발음한 20개의 지역명을 선정하였다. multisection codebook은 인식어휘의 단어에 대해 같은 길이, 즉 section으로 나누고, 각각의 section에 대해 표준 VQ codebook을 만든 다음 section별 관측치를 구한 후 HMM 학습 후 최종 인식하였다.

Multisection VQ codebook은 시간정보를 가지는 잇점과 전체의 codeword에 의해 관측열을 구하는 것보다 각 section별로 관측열을 구하므로 계산량을 훨씬 줄일 수 있고 또한 인식율도 높일 수 있었다.

실험에서, multisection VQ codebook을 이용한 HMM의 경우가 인식률이 향상됨을 증명하였다.

Abstract

In this paper, isolated words are recognized using multisection VQ and HMM. As recognition vocabuaries, 20 area-name which is uttered 5 times by 3 speakers is selected.

In generating codebook, we devide recognition vocabulary into equal length, section, and make standard VQ codebook to each section and calculate observation by section and than recognize isolated words by HMM training.

Multisection VQ codebook has time information and as observation is calculated by each section, computation is lesser and recognition rate is higher than by whole codword.

As a result, it is proved that recognition rate is higher in case of HMM using multisection VQ codebook.

I. 서 론

기존의 음성 인식 방법은 크게 패턴 정합 방법과 확률적인 모델링 방법으로 나눌 수 있다. 패턴 정합 방법은 입력 되는 음성의 특징인자(parameters)들을 추출해서 추출한 인자와 이미 저장되어 있는 특징 인자들과 비교해 가장 근사한 것으로 인식하는 방법이다.[1] 이러한 방법은 인식률은 높으나 시간과 특징 인자의 저장 용량이 커야한다는 큰 단점이 있다.

음성의 특징은 개인간의 차가 클 뿐만 아니라 같은 사람이 같은 내용을 말해도 그 특징이 불안정하게 나타난다. 확률적인 모델링은 패턴 정합을 특수한 경우로써 포함하는 좀 더 일반화된 방법으로서 실제 음성의 불안정성을 보다 정확하게 반영하며, 확

*正會員, 光云大學校 電子計算氣工學科
(Dept. of Computer Eng., Kwangwon Univ.)

接受日字 : 1990年 3月 20日

률적인 모델은 표준 음성들에 대한 모델링을 한 후, 이들 모델로부터 입력패턴의 관측 확률을 구하여 가장 높은 확률을 가지는 모델로 인식한다.

확률적인 모델의 일종인 HMM은 최근 인식률이나 여러가지면에서 우수한 결과를 보였다.[13,14,15,16] 그런데, 기존의 VQ codebook에 의한 HMM 음성 인식에서는 시간 정보를 포함하지 못함으로 시간 정보를 이용할 수 있는 방법에 비해 인식률이 떨어진다. 따라서, 본 논문에서는 MSVQ(multisection VQ) codebook을 이용한 HMM을 제안하는데 그 이유는, MSVQ에서는 시간 정보를 이용할 수 있기 때문이다.[10,11] MSVQ에서는 단어를 동일한 몇 개의 section으로 나누어서 각 section별로 codebook을 작성하는데 각 section별 codebook은 시간 정보를 가지게 된다.

HMM에 의한 인식 단계에서는 MSVQ의 각 section별로 구한 codebook으로 부터 관측열을 구해 인식시킴으로써 높은 인식률을 얻었고, 또한 MSVQ를 8 section으로 하여 codebook을 작성했을 경우 관측열을 구하는데 있어서 계산량을 1/4까지 줄일 수 있었다.

본 실험에서 시작점과 끝점을 검출하는데 peak-to-peak에 의한 끝점 검출 방법을 사용하였으며, Shikano 등의 실험결과[3] LPC(linear prediction coefficient) 파라메타[4,5]에 의한 음성 인식보다 LPC cepstrum 파라메타가 더 우수하다는 고찰에 따라 LPC cepstrum 파라메타를 특징 파라메타로 사용하였다.

본 연구의 전체 인식 시스템의 구성도 그림1과 같다.

전체 구성은 V절로 되어 있다. Ⅱ절에서는 VQ 이론과 MSVQ codebook작성에 대해서 기술하였고, Ⅲ절에서는 HMM의 이론과 MSVQ이 어떻게 HMM에서 이용되는지에 대해서 기술하였고, Ⅳ절에서는 실험조건 및 대상어에 대해서 기술하였고, 실험방법 및 실험결과에 대해 기술하였다. 끝으로 V절에서는 결론을 내렸다.

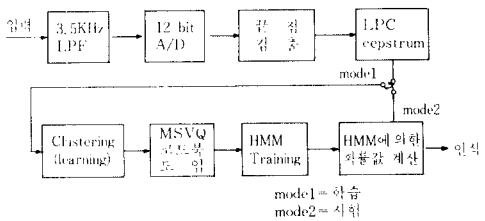

그림 1. 전체적인 시스템의 구성도
Fig. 1. Overall block diagram of system.

Ⅱ. 벡터 양자화 이론

1. 개요

VQ란 벡터의 열들을 통신이나 디지탈 채널에 저장하기에 적당한 수치값의 열과 매핑하기 위한 시스템이다.[6][7] VQ의 가장 큰 목적은 데이타 압축으로 데이타의 신뢰성을 잃지 않으며, 최대 한도로 bit rate를 줄이는데 있다. 데이타 압축에 기여한 Shannon의 rate distortion 이론에 의하면 스칼라 대신에 벡터를 코딩함으로써 더 좋은 성능을 얻을 수 있다는 것이다.[8] 따라서, 음성인식에 있어서 데이타 압축이라는 측면에서 표준 패턴을 생성하는데 VQ를 이용한다. 즉, 음성 인식에서의 VQ는 입력된 음성의 특징 벡터를 미리 저장해 둔 특징 벡터 중에서 가장 잘 정합되는 하나의 벡터와 매핑시켜 주는 것이다.

시험 벡터들에 의해 codebook이 만들어지며, 입력 벡터는 codebook의 벡터들 중에서 최소의 왜곡률을 갖는 벡터로 양자화 된다.[9]

2. 학습 데이타

VQ를 이용한 음성 인식 시스템에서 학습 데이타를 구성하는 방법은 두 가지가 있다. 하나는 단일 section codebook을 만들기 위한 구성법이고 또 다른 하나는 multisection codebook을 만들기 위한 구성법이다.

단일 section codebook이란 한 단어당 표준 단어 템플리트로서 하나의 codebook을 취하는 방법이다. 그러므로, 이 방법을 채택하면 학습 데이타는 고전적인 학습 데이타와 같이 단어를 M번 발음한 음성 데이타로 구성된다.

Multisection codebook이란 한 단어당 표준 단어 템플리트로서 2개 이상의 codebook을 취하는 방법이다. 그러므로 매 단어의 표준 패턴을 만들기 위해서는 두개 이상의 학습 데이타가 필요하다. 학습 데이타를 만들기 위하여 처음에 한 단어를 한번 발음한 것을 원하는 section N만큼 시간축으로 분할하여 N×M개의 초기 학습 데이타를 만든다. 다음에 이 초기 학습 데이타를 매 발음 할 때마다 각 section별로 모아서 최종 학습 데이타를 만든다.

3. 일반적인 VQ codebook 작성

VQ를 이용한 음성 인식 시스템에서 codebook은 곧 표준 단어 템플리트가 되므로 학습 데이타의 특성이 잘 나타나도록 codebook을 만들어야 한다.

본 연구에서는 K-Means 알고리듬을 이용하여 codebook을 작성하는 방법을 사용하였다.[12] Codebook을 작성하는 K-Means 알고리듬은 다음과 같다.

K-Means 처리 방법은 집단을 분류하고 집단의 중심을 계산하여 수렴 여부를 결정하는 등 세가지 기본 단계로 반복 수행된다. 임의로 집단의 수를 M으로 고려할 때 집단을 대표할 수 있는 중심점 M개의 토큰으로 결정되며, 이 M개의 집단에 대하여 맨 처음 집단의 중심점은 임의의 M개의 토큰을 선택할 수 있다. 간단히 식으로 표현하면 다음과 같다.

$$x_p^{(i)} = x_i, \quad 1 \le i \le M \tag{1}$$

그리고, 집단의 분류는 SNN(shared nearest neighbors)[12]의 기본 법칙, 즉

$$x_j \in w_i \delta \text{ iff } (x_j, x_p^{(i)}) < \delta(x_j, x_p^{(k)})$$
$$1 \le k \le M, \ 1 \le j \le N \tag{2}$$

에 준한다. 여기서 N는 전체 토큰수이다.

집단의 중심점을 구하고, 수렴여부는 앞의 반복 실행에서와 같이 집단의 중심점으로 동일 토큰이 지정되었는지를 조사하는 것을 말하며, 만일 동일 토큰이 아닐 경우에는 또 다시 반복 수행한다.

집단의 중심점이 여러 집단의 중심점 구하는 방법에 의하여 계산된다고 하더라도 두 토큰 사이의 진동은 있을 수 있다. 그러나, 이러한 경우가 종종 있는 것은 아니고 일반적으로 수렴한다. 본 연구에서는 codebook size를 128로 하여 실험하였으며, 중심점 계산 방법으로는 MINIMAX방법을 사용하였다.

4. Codebook 작성시 왜곡률 계산[15]

LPC cepstrum 벡터의 학습 데이타 집합을 $C_i = i = 1, \cdots, I$라 하자. 이 벡터들은 어휘에서의 단어들이 다양한 화자에 의해서 발음될때 일어나는 LPC cepstrum이다. VQ에 숨은 주요 개념은 주어진 M에 대하여 가장 가까이에 있는 codebook entry C_m에 의해 학습 데이타 집합 벡터 C_i의 각각에 대해 평균 왜곡률이 최소가 되도록 LPC cepstrum 벡터로 최적의 codebook의 집합 $C_m (m=1, 2, \cdots, M)$을 결정하는 것이다.

좀 더 형식적으로 말해서, 두 LPC cepstrum 벡터 C_r간 C_t간의 거리로서 $d(C_r, C_t)$라고 정의한다면 그때 VQ의 목표는

$$||D_M|| = \min_{c_m} \left[\frac{1}{I} \sum_{i=1}^{I} \min_{1<m<M} [d(C_m, C_i)] \right] \tag{3}$$

을 만족하도록 집합 C_m을 찾는 것이다.

위 식에서 M값 $||D_M||$은 VQ의 평균 왜곡률(거리)이다.

본 시스템에서 사용하는 국부적인 왜곡률은 12차 LPC cepstrum을 특징 파라메타로 사용하였을 때 다음과 같다.

$$d(C_r, C_t) = \sum_{i=0}^{12} (C_{ri} - C_{ti})^2 \tag{4}$$

여기서, C_{ri}, C_{ti}은 LPC cepstrum 계수이다.

5. Multisection VQ

1) MSVQ 이론

단어 음성 인식에서는 발성 속도에 따른 시간변동의 제거를 위해 DP(dynamic programing) 정합이 많이 이용되어 왔다. 이 방법은 시간축의 선형변환에 따른 계산량이 증가한다. 그러므로 시간 정규화가 필요 없는 단어 별로 작성된 VQ codebook에 의해 단어들의 음향적인 특성만을 비교하는 방법을 쓴게 되었다.

그러나, 일반적인 VQ codebook에는 시간적 정보가 포함 되어 있지 않아서 음향적 특성이 유사한 단어들 사이에 부정확한 인식이 일어난다. 따라서 한 단어를 발성순서에 따라 몇개의 section 으로 나누고 section 별로 독립된 codebook을 작성하므로써 시간적 정보를 포함시키는 multi-section VQ가 Burton 등에 의해 제안되었다.[10]

Burton의 MSVQ에 따르면 MS codebook 작성에 따르는 모든 음성은 발성시간에 관계없이 일정 수의 정해진 길이를 갖는 프레임으로 정규화 되어야 한다. 그러므로 발성 시간이 짧은 단어는 정규화 길이에 일치 시키기 위해서 인접 프레임을 중첩시켜야 한다. 이것은 분석과 거리계산에 있어 불필요한 요인이 될 수 있다. 본 연구에서는 전체 프레임을 구한 후, 일정한 section으로 나누어 codebook을 작성하였다.

2) MSVQ Codebook 작성

VQ codebook을 작성하는데 있어서 시간 정보를 포함시키는 방법 중 하나를 MSVQ codebook 이라고 한다. 어떤 단어의 MSVQ codebook은 그 단어를 동일 길이의 section으로 나누고 각 section 마다 K-Means 알고리듬을 써서 작성한다. 본 논문에서는 음성 데이타의 전체 프레임을 구한후 일정한 section으로 나누어 codebook을 작성 하였다. 따라서, 본 연구는 이것에 바탕을 두고 4 multisection VQ codebook과 8 multisection VQ codebook을 작성하여 실험하였다.

(1) 4 MSVQ Codebook

그림 2에 4 MSVQ codebook을 작성하는 과정이 나타나 있다. 한 단어 W를 1회 발성한 음성을 학습열로 사용한다. 한 프레임을 LPC 분석하여 얻은

cepstrum 벡터를 v 라 하면 1회 발성한 음성은

$$W = \{v_1, v_2, v_3, \cdots, v_k\} \quad (5)$$

와 같이 나타낼 수 있다. 인식 대상 어휘가 모두 L 개의 단어로 되어 있을때 각 단어 마다 I 회 발성된 음성으로 MSVQ codebook을 구성하기 위해서는 이들 J개의 section으로 나누었다.

$$W_l(i) = \{V_1(i)\ V_2(i) \cdots V_J(i)\}^t = (l = 1, 2, \cdots, L) \quad (6)$$

여기서, 만약 각 단어마다 I 회 발성된 음성에서의 i번째의 section이 N_j (j=1, 2, ···, J) 프레임으로 구성 되어 있다면

$$V_1(j) = \{v_1(i)\ v_2(i)\ v_3(i) \cdots\cdots v_{N1}(i)\}$$
$$V_J(i) = \{v_{N1+\cdots+N(J-1)+1}(i) \cdots v_{N1+\cdots+NJ}(i)\} \quad (7)$$

와 같이 각 section을 벡터열로 표시 할 수 있다. 그림 2 에서 보는 바와 같이 각 section의 프레임수는 다르나, 여기에서는 4 section으로 되어 있으므로 4개의 독립된 VQ codebook의 조합에 의해 MSVQ codebook이 구성 된다. section j에 해당하는 학습열의 집합을 S_j라 하면

$$S_j = \{V_j(1)\ V_j(2) \cdots\cdots V_j(I)\} = (j = 1, 2, 3, 4) \quad (8)$$

이 된다. 각 section에 대한 codebook C_j는 S_j를 학습열로 하여서 K-means 알고리듬에 의해 작성된다.

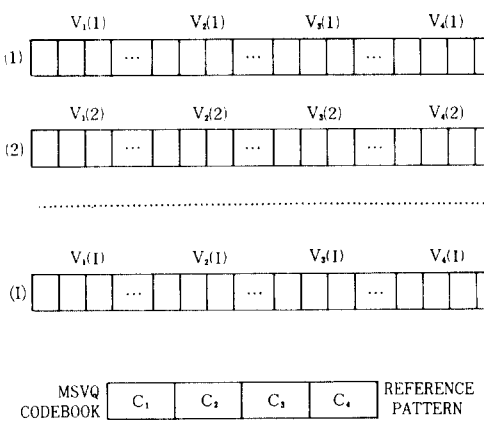

그림 2. 4 MSVQ codebook 작성
Fig. 2. 4 Multisection VQ codebook generation.

이 과정을 통해 작성된 codebook의 열

$$C = \{C_1, C_2, C_3, C_4\} \quad (9)$$

는 MSVQ codebook을 의미한다.

4 MSVQ codebook의 section별 codebook size를 32로 하여 총 128개의 codeword로 실험하였다.

(2) 8 MSVQ Codebook

8 MSVQ codebook을 만드는 과정은 4 MSVQ codebook을 만드는 과정과 동일하나 단지 section의 수를 4에서 8로 늘리는 것만이 다르다. 그래서 8개의 각 section codebook이 모여서 한 단어의 전체 codebook을 이루게 된다. 각 section codebook C_j는 2개의 codeword로 이루어진다.

8 multisection VQ codebook의 section별 codebook size은 16으로 하여 총 128개의 codeword로 실험하였다.

Ⅲ. Hidden Markov Model

1. HMM 이론[16]

HMM은 천이에 의해 연결된 상태들의 모임이다. HMM은 두 확률 즉, 천이확률과 출력 밀도함수(observation)을 가진다.

HMM은 다음과 같이 정의된다.

{S} 상태의 집합, S_1 초기 상태, S_F 최종 상태

$\{b_{ij}\}$ 천이의 집합, 여기서 a_{ij}은 i상태에서 j 상태로 천이할 확률

$\{b_{ij}(k)\}$ 출력 확률 메트릭스, i 상태에서 j상태로 천이시 k 심볼이 나올 확률.

a, b 확률은 다음과 같은 특성을 지녀야 한다.

$$a_{ij} \geq 0,\ b_{ij}(k) \geq 0,\ \forall i, j, k \quad (10)$$

$$\sum_j a_{ij} = 1\ \forall i \quad (11)$$

$$\sum_k b_{ij}(k) = 1\ \forall i, j \quad (12)$$

따라서 a, b는 다음과 같이 쓸 수 있다.

$$a_{ij} = P(X_{t+1} = j | X_t = i) \quad (13)$$

$$b_{ij}(k) = P(Y_t = k | X_t = i,\ X_{t+1} = j) \quad (14)$$

여기서, $X_t = i$은 Markov chain이 시간 t에서 상태 i에 있고, $Y_t = k$는 시간 t에서의 출력 심볼이 k라는 것을 의미한다.

HMM에서는 기본적인 두가지의 가정을 한다.

첫번째 가정은 식 (15)에서처럼 현재의 상태는 그 바로 이전 과거의 상태에만 의존한다는 것이다.

$$P(X_{t+1}=x_{t+1}|X_1^t=x_1^t)=P(X_{t+i}=x_{t+1}|X_t=x_t) \quad (15)$$

두번째 가정은 출력이 독립적이다는 것이다. 즉,

$$P(Y_t=y_t|Y_1^{t-1}=y_1^{t-1},\ X_1^{t+1}=x_1^{t+1}$$
$$=P(Y_t=y_t|X_t=x_t,\ X_{t+1}=x_{t+1}) \quad (16)$$

여기서, Y_i^j는 출력열 $Y_i, Y_{i+1}\cdots Y_j$을 나타낸다.

식 (16)에서 어떤 특정한 심볼이 시간 t에서 나타날 확률은 단지 그 시간에 취해진 천이에만 의존한다. 이러한 가정을 함으로서 파라메타의 수를 줄일 수 있으며, 실제 학습에서도 효과적인 알고리듬을 만들 수 있다. 여기에서 HMM을 정의하는데 세가지 문제점을 해결하여야 한다. 그 첫째는 평가(evaluation) 문제이고 두번째는 decoding 문제이며, 세번째는 학습(learning)문제이다. 그러면 첫번째로 평가 문제부터 살펴볼자.

모델과 관측열이 주어졌을 때 관측열의 확률을 계산하는 법이다. 이에 대한 방법으로는 forward 알고리듬을 이용한다.

즉, T길이 경로(length path)의 확률을 다음과 같이 정의하면

$$P(Y_1^T=y_1^T)=\sum_{X1^{T+1}} P(X_1^{T+1}=x_1^{T+1})P(Y_1^T=y_1^T|X_1^{T+1}=x_1^{T+1}) \quad (17)$$

위의 식 (17)의 첫번째 요소를 markov 가정에 의해 다시 쓰면,

$$P(X_1^{T+1}=x_1^{T+1})=\prod_{t=1}^{T} P(X_{t+1}=x_{t+1}|X_t=x_t) \quad (18)$$

위의 식 (17)의 두번째 요소를 출력 독립 과정에 의해 다시 쓰면,

$$P(Y_1^T=y_1^T|X_1^{T+1}=x_1^{T+1}$$
$$=\prod_{t=1}^{T} P(Y_t=y_t|X_t=x_t,\ X_{t+1}=x_{t+1}) \quad (19)$$

식 (18), (19)을 식 (17)에 대입하면 다음과 같다.

$$P(Y_1^T=y_1^T)$$
$$=\sum_{X1^{T+1}}\prod_{t=1}^{T}(X_{t+1}=x_{t+1}|X_t=x_t)P(Y_t=y_t|X_{t+1}=x_{t+1},X_{t+1}=x_{t+1}) \quad (20)$$

위의 $P(Y=y)$를 t에 대해 회귀적으로 표현하면 아래와 같다.

$$\alpha_i(t)=\begin{cases}0 & t=0\wedge i\neq S_I\\ 1 & t=0\wedge i=S_I\\ \sum_j \alpha_i(t-1)a_{ij}\ b_{ji}(y_t) & t>0\end{cases} \quad (21)$$

$\alpha_i(t)$는 markov 과정이 일반화된 y_1^t를 가지는 상태 i라는 가정이다.

두번째 문제는 decoding에 관한 것으로 관측열이 주어졌을 때 최적인 상태열을 선택하는 문제인데 이는 Viterbi 알고리듬을 이용한다.

각 상태에서 최적의 경로를 찾아가면

$$v(t)=\begin{cases}0 & t=0\wedge i\neq S_I\\ 1 & t=0\wedge i=S_I\\ MAX_j\ v_j(t-1)\ a_{ji}\ b_{ji}(y_t) & t>0\end{cases} \quad (22)$$

이다.

세번째 문제는 학습에 관한 문제로 최적의 모델링을 하기위해 각 파라메타들을 조정하는 문제인데 이는 Baum-Welch의 재평가(reestimation) 알고리듬으로 해결한다.

앞에서 언급한 $\alpha_i(t)$는 forward 알고리듬으로 상태 i에서 y_1^T를 생성할 확률이고, $\beta(t)$는 backward알고리듬으로 상태 i에서 y_{t+1}^T를 생성할 확률이다. 이를 t에 대해 회귀적으로 표현하면,

$$\beta_i(t)=\begin{cases}0 & i\neq S_F\wedge t=T\\ 1 & i=S_F\wedge t=T\\ \sum_j a_{ij}\ b_{ij}(y_{t+1})\beta_j(t+1) & 0\leq t<T\end{cases} \quad (23)$$

이고, $\gamma_{ij}(t)$는 i에서 j로 천이할 확률이다.

$$\gamma_{ij}(t)=P(X_t=i,\ X_{t+1}=j|y_1^T)$$
$$=\frac{\alpha_i(t-1)\ a_{ij}\ b_{ij}(y_t)\ \beta_j(t)}{\alpha_{SF}(T)} \quad (24)$$

이며, 이때 $\alpha_{SF}(T)$는 모델 M이 y_1^T에서 생성될 확률이다. 따라서,

$$\bar{a}_{ij}=\frac{\sum_{t=1}^{T}\gamma_{ij}(t)}{\sum_{t=1}^{T}\sum_k\gamma_{ik}(t)} \quad (25)$$

이고

$$\bar{b}_{ij}(k)=\frac{\sum_{t,y_t=k}\gamma_{ij}(t)}{\sum_{t=1}^{T}\gamma_{ij}(t)} \quad (26)$$

이다.

2. 제안된 MSVQ codebook을 이용한 HMM 모델 모델링

일반적인 VQ codebook에 의해 관측열 집합을 구할 때에는 전 codeword에서 가장 왜곡률이 적은 것을 선택하는데 반하여, 본 연구에서 제안하는 multisection VQ codebook을 이용한 실험에 있어서는 그

림 3에서 보는 바와 같이, MSVQ를 하기 위한 전체 단어 집합에서 각 section 단위로 clustering 작업을 해 codebook을 만든다. 본 실험에서는 총 codeword의 수를 VQ codebook에 의해서 HMM에 의한 인식을 하는 경우나 MSVQ codebook를 이용한 HMM의 경우나 동일하게 128개로 해서 HMM 인식 실험을하는 관계로 4 MSVQ codebook의 경우에는 각 section별 codebook의 크기를 32로 하여 4 section×32개 = 128개로 codeword의 수를 맞추었으며, 8 MSVQ codebook의 경우에는 각 section별 codebook의 크기를 16으로 하여 8 section×16개=128개로 codeword의 수를 맞추어 이에 의해 인식 실험을 하였다.

또한 그림 3에서 보는 바와 같이 Baum-Welch의 재평가(reestimation)알고리듬을 이용할 때 쓰는 학습 데이타의 집합의 경우나, 실제 인식시에 확률값을 구하기 위한 시험 데이타 집합을 사용할 때 모두 이 section별로 구한 codeword를 이용하여 관측열을 설정하므로 관측열 집합을 설정하는 방법도 일반적인 VQ codebook에 의한 방법과는 달리 입력 음성에 의한 패턴들도 N(N=4, 8)등분하여 section을 나눈 후 앞서 구한 multisection VQ의 각 section의 codeword들과 비교하여 가장 거리값이 적은 것을 관측치로 선택한다.[11]

따라서, MSVQ codebook을 이용한 HMM에 의한 음성 인식의 경우에는 관측열을 구하는데 시간 정보를 이용할 수 있다는 잇점이 있고 MSVQ codebook를 사용할 때에는 section별로 관측열을 구함으로 관측열을 구할시에 계산량을 1/4까지 줄일 수 있다.

Ⅳ. 인식 실험 및 결과

1. 실험조건 및 대상어

1) 실험조건

연구실의 보통의 연구 환경에서 마이크를 통하여 입력된 신호는 sampling 주파수를 8KHz로 하였으며 3.5KHz 저역 여파기를 통과한 후 12bit A/D 변환을 거쳐 음성신호를 구한 다음, 시작과 끝 구간을 검출한 후 LPC 계수를 구했다. 그런 다음 이것을 12차 LPC cepstrum 계수로 변환하여 codebook을 작성한다.

HMM에 의한 불특정 화자의 음성 인식을 하는데 있어서 대상 어휘로는 20개의 앞으로 바뀔 DDD 번호에 의한 지역명을 선정하였고, 3명의 남성 화자에 의해 각각 5번씩 발성된 것을 데이타로 선정하였다. MSVQ codebook 작성시에는 각 화자가 각각 단어에 대해 각각 1번씩 발음한 것으로 각 단어에 대한 MSVQ codebook을 작성했다. 따라서 이 codebook을 이용해서 HMM을 학습시키고 인식을 시키는데는 다음과 같은 방법으로 실험하였다.

1) 각 화자가 각각 1번씩 발음한 것으로 HMM 의 학습을 시켰으며, 각각 4번씩 발음한 것으로 인식실험을 하였다.

2) 각 화자가 각각 2번씩 발음한 것으로 HMM의 학습을 시켰으며, 학습시킨 데이타를 포함하여 각각 5번씩 발음한 것으로 인식 실험을 하였다.

본 연구에서는 기존의 음성 인식에서 HMM 모델링시에 사용한 것처럼 그림 4와 같이 Left-to-right 모델을 사용하였다.

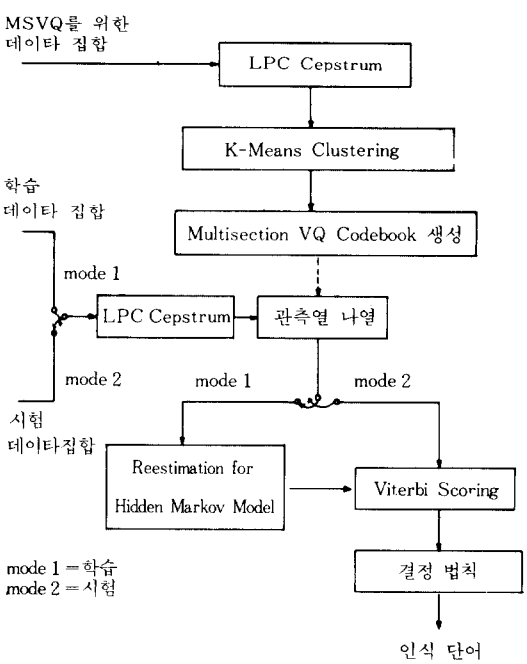

그림 3. 인식 시스템
Fig. 3. Recognition system.

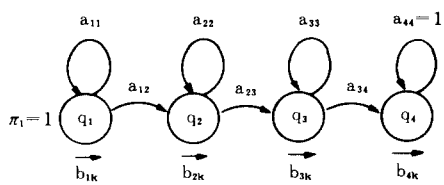

그림 4. 본 연구의 HMM 모델
Fig. 4. The HMM model of this paper.

2) 대상어

대상어휘는 앞으로 바뀔 것으로 알려진 20개의 DDD 지역명을 선택하였다. 선택된 지역명은 다음과 같다.

서울, 인천, 수원, 경기, 춘천, 강원, 대전, 청주, 충북, 충남, 부산, 마산, 대구, 경북, 경남, 전주, 광주, 전북, 전남, 제주.

2. 인식 실험 결과

본 실험은 20개의 지역명을 인식 대상으로 삼았으며 codebook size를 128로 하고 파라메타로는 12차 LPC cepstrum을 사용하였으며, state는 5인 경우와 8인 경우로 나누어 실험하였다.

1) 각 화자가 각각 1번씩 발음한 것으로 HMM의 학습을 시키고, 나머지 각각 4번씩 발음한 것으로 인식 실험을 한 경우

표 1. 인식율
Table 1. Recognition rate.

	single section VQ	4 Multisection VQ	8 Multisection VQ
5 state	94.1%	95.0%	95.4%
8 state	94.1%	94·1%	95.0%

2) 각 화자가 각각 2번씩 발음한 것으로 HMM 의 학습을 시키고, 학습시킨 데이타를 포함 각각 5번씩 발음한 것으로 인식 실험을 한 경우

표 2. 인식율
Table 2. Recognition rate.

	single section VQ	4 Multisection VQ	8 Multisection VQ
5 state	97.3%	97.6%	98.0%
8 state	97.3%	98.4%	98.4%

Ⅳ. 결 론

본 연구에서는 단일 section codebook과 multisection codebook을 작성하여 이것을 바탕으로 관측열을 만들어 이에 의해 HMM 학습을 시켰으며, 또한 이것을 바탕으로 인식 실험을 행하였다.

단일 section VQ codebook에 의해 관측열을 구해 인식시킨 경우보다, 본 연구에서 제안하는 multisection VQ codebook에 의해 관측열을 구해 HMM에 의해 학습시키고, 인식시켰을 경우가 시간정보를 포함하고, 또한 section별로 관측치를 구함으로 인식시간도 적게 걸린다. 따라서, Multisection VQ codebook을 이용한 HMM 인식의 경우가 더 인식률이 향상됨을 알 수 있다.

또한, multisection VQ codebook을 이용한 인식에서는 전반적으로 4 section의 경우보다 8 section의 경우가 더 인식율이 좋았다. 본 연구에서 특이한 사실은 3사람이 각 단어에 대해 한번씩 발음한 것으로 HMM 학습 및 이를 바탕으로 인식시 상태를 5로 한 경우가 상태를 8로 한 경우보다 multisection VQ codebook을 이용시에 인식율이 더 좋았다. 그리고, 8 multisection VQ codebook을 이용하여 HMM에 있어서 상태를 5로 하였을 때 인식율이 95.4%로 가장 좋았다.

반면에 5사람이 각 단어에 대해 두번씩 발음한 것으로 HMM 학습 및 이를 바탕으로 인식시 상태를 8로 한 경우가 상태를 5로 한 경우보다 multisection VQ codebook을 이용시에 인식율이 더 좋았다. 그리고, 이 실험에서도 상태 5의 경우 8 multisection VQ codebook을 이용하였을 때가 인식율이 가장 좋았고, 상태가 8인 경우는 4 MSVQ codebook을 이용한 경우나 8 MSVQ codebook을 이용한 경우나 똑같이 98.4%로 가장 좋은 인식률을 보였다.

이 실험으로 알 수 있는 것은 HMM의 학습시에 데이타량이 많으면 많을수록 좋다는 사실이며, 또한 학습 데이타가 충분할 경우에 상태가 5인 것보다 상태가 8인 경우가 인식률이 더 좋다는 것을 보여주었다. 이 때 오인식된 대부분의 단어들은 마이크를 이용하여 잡음이 있는 연구실에서 발음한 관계로 잡음이 들어간 것이었거나, 또한 너무 짧게 발음하여 애매하게 발음된 것들이었다.

끝으로, 우수한 잡음 제거 알고리듬과 음성 부분의 시작점과 끝점을 잘 검출해 낸다면 음성 인식하는데 많은 도움이 되리라 생각된다.

參 考 文 獻

[1] 김순협, "한국어 음성의 분석과 자동 인식에 관한 연구," 박사 논문, 연세대학교 대학원, 1982. 12.

[2] S.E. Levinson, L.R. Rabiner, A.E. Rosenberg and J.E. Wilpon, "Interactive clustering techniques for selecting speaker independent reference techniques, for isolated word recognition," *IEEE Trans. on ASSP* vol. 27, no. 2, pp. 134-141, APR. 1979.

[3] Shikano, K., Kohda, M. "On the LPC distance measures for vowel recognition in continuous utterance," *Institute of Electrical and Communication Engineers of Japan, Trans.* on D, J63-D, May 1980.
[4] Manfred R. Schroeder "Linear predictive coding of speech review and current directions" *IEEE Comm. Magazine,* vol. 23, no. 8, August 1985.
[5] J.D. Markel and A.H. Gray, *Linear Prediction of Speech* Spring Verlag Berline Heidelberg 1976.
[6] R.M. Gray "Vector quantization," *IEEE ASSP Magazine,* vol 1, pp. 4-29 Apr 1984.
[7] Y. Linde, A. Buzo, and R.M. Gray "An algorithm of vector quantizer design," *IEEE Trans. Comman,* vol. COM-28 pp. 84 95, Jan. 1980.
[8] C.E. Shannon, ,'A mathematical theory of communication," *Bell Sys. Tech. J.* 27, pp. 379 423, 623-656, 1948.
[9] KUK-CHIN PAN, Frank K. Soong and L.R Rainber, "A vector quantization-based preprocessor for speaker-independent isolated world recognition," *IEEE Trans. of Acoustics, Speech, Signal Processing,* vol. ASSP-33, no. 3, June 1985.
[10] D.K. Burton, J.E. Shore, J.T. Buck "Isolated word speech reocgnition using multisection vector qantization codebooks," *IEEE Trans. of Acoustics, Speech, Signal Processing, vol.* ASSP-33, no. 4, August 1985.
[11] 이성권, "VQ를 이용한 DDD 지역명 인식에 관한 연구," 석사학위 논문, 광운대학교 대학원, 1989. 12.
[12] J.T. Tou, R.C. Gonzalez, Pattern recognition Principles Addison-Wesley Publishing Company, Inc. 1974.
[13] L.R. Rabiner B.H. Juang, "An introduction to hidden markov models," *IEEE ASSP MAGAZINE JAN.* 1986.
[14] L.R. Rabiner, J.G. Wilpon, F.K. Soong, "High performance connected digit recognition using hidden markov models," *IEEE Trans. on Acoustics, Speech, and Signal Proceeding,* vol. 37, no. 8, August, 1989.
[15] L.R. Rabiner, S.E. Levinson, M.M. Sondhi, "On the application of vector quantization and hidden markov models to speaker-independent, isolated word recognition," *Bell System Technical Journal,* vol. 62 no. 4, April 1983.
[16] Kai Fu Lee, Automatic speech recognition: The development of the SPHINX system, Kluwer Academic Publishers

著 者 紹 介

安 泰 玉 (正會員) 第27卷 第7號 參照
현재 광운대학교 전자계산기 공학과 박사과정

金 淳 協 (正會員) 第27卷 第7號 參照
현재 광운대학교 전자계산기 공학과 교수

邊 龍 圭(正會員)
1933年 11月 10日生. 1958年 3月 연세대 전기과 졸업. 1979年 8月 단국대 대학원 전자과 졸업. 1986年～현재 광운대 대학원 박사과정. 1969年～현재 경기공업 개방대학 전산과 부교수