비선형 집단화와 완화기법을 이용한 VQ/HMM에 관한 연구

A Study on VQ/HMM using Nonlinear Clustering and Smoothing Method

정 희 석*, 강 철 호*
(Heui Suck Chung*, Chul Ho Kang*)

※이 연구는 1998년도 광운대학교 교내 학술연구비 지원으로 수행된 것임

요 약

본 논문에서는 이산적인 HMM(Hidden Markov Model)을 이용한 고립단어 인식 시스템에서 입력특징벡터의 변별력을 향상시키기 위해 수정된 집단화 알고리듬을 제안하므로써 K-means나 LBG 알고리듬을 이용한 기존의 HMM에 비해 2.16%의 인식율을 향상시켰다. 또한 HMM학습과정에서 불충분한 학습데이타로 인해 발생되는 인식율저하의 문제를 해소하기 위해 확률적으로 개선된 smoothing 기법을 제안하므로써 화자독립 실험에서 3.07%의 인식율을 향상시켰다.

본 논문에서 제안한 두가지 알고리듬을 모두 적용하여 최종적으로 실험한 VQ/HMM에서는 기존의 방식에 비해 화자독립 인식실험 결과 평균 인식율이 4.66% 개선되었다.

ABSTRACT

In this paper, a modified clustering algorithm is proposed to improve the discrimination of discrete HMM(Hidden Markov Model), so that it has increased recognition rate of 2.16% in comparison with the original HMM using the K-means or LBG algorithm. And, for preventing the decrease of recognition rate because of insufficient training data at the training scheme of HMM, a modified probabilistic smoothing method is proposed, which has increased recognition rate of 3.07% for the speaker-independent case.

In the experiment applied the two proposed algorithms, the average rate of recognition has increased 4.66% for the speaker-independent case in comparison with that of original VQ/HMM.

I. 서 론

음성은 인간의 가장 자연스러운 의사소통 수단이다. 따라서 음성인식은 인간과 기계간의 자연스런 의사소통을 이룰 인터페이스를 제공한다.

현재 연구되어지고 있는 음성인식의 방법으로는 벡터양자화(Vector Quantization), 시간적인 정합을 이용한 DTW(Dynamic Time Warping) 알고리듬, 확률적인 방법으로 잘 알려진 Hidden Markov Model(HMM), 그리고 신경망에 의한 인식 등이 있다. 그 중에서도 HMM은 음성의 시간적인 변이성을 통계적인 확률모델로 분석하므로써 높은 인식율을 보여 1980년대 이후 활발한 연구가 진행되어져 오고 있으며 최근에는 신경망과 함께 결합하여 다양하게 연구되고 있다.

그러나 HMM은 관측 시퀀스가 사실이라는 가정하에 이에 대한 출력 확률을 최대로 하기 위한 학습 알고리듬을 적용하는 것이므로 실제 학습하고자하는 데이터가 불충분하면 인식율을 저하시키게 된다[1].

또한 HMM 입력패턴의 벡터양자화를 위한 집단화 알고리듬으로써 흔히 이용되는 K-means나 LBG 등의 알고리듬은 모든 입력 패턴에 대해 벡터축상에서 최소 유클리드 거리를 갖도록 임의의 수의 클러스터로 집단화하므로 발생빈도가 높고 유사도가 큰 입력벡터에 대해 변별력을 상대적으로 저하시켜 결국 HMM에서의 인식율을 저하시킨다.

따라서 본 논문에서는 VQ/HMM을 기초한 음성인식시스템에서 확률적으로 개선된 smoothing 기법을 적용하여 학습데이터의 불충분으로 인한 인식을 저하문제를 해결하고, 비선형적으로 수정된 집단화 알고리듬을 통해 많은 입력 패턴벡터가 좁은 영역내에 밀집될 경우에 대한 패턴벡터간의 변별력을 향상시켜 HMM에서의 인식율을 개선하기 위한 집단화 알고리듬에 대한 연구 및 실험을 하였다.

II. VQ/HMM의 기본이론

1. Vector Quantization

* 광운대학교 전자통신공학과
접수일자 : 1998년 9월 30일

벡터 양자화는 무한한 수의 특징 벡터를 유한한 수의 이산 벡터 공간으로 사상시키는 부호화 방법으로서 1980년대 이후 음성인식분야에 적용되기 시작했다. 이는 입력된 음성신호의 많은 정보량을 상대적으로 매우 적은 수의 코드벡터들로 사상시킴으로써 정보량을 줄이고 이에 따른 연산량을 감소시키는 효과를 갖는다[2]. 그러나 이러한 과정에서 정보의 손실이 발생하며 이를 양자화 오차라 한다. 따라서 이를 최소화할 수 있는 최적의 코드벡터를 생성하는 집단화(clustering) 알고리듬이 중요시되고 있다.

자율학습에 의한 집단화 알고리듬으로는 동적 집단화(Dynamic clustering)와 계층적 집단화(Hierarchical cluster-ing)의 두 가지의 기본 형태로 나누어 볼 수 있다. 동적 집단화 방법은 정해진 클러스터의 수에 따라 입력된 모든 특징 벡터들이 안정된 분할을 이룰 때까지 반복적으로 클러스터 멤버와 클러스터 중심값을 갱신하는 것으로써 특히 음성 신호처리 분야에서의 특징 벡터 집단화 알고리즘으로 널리 이용되고 있다[2][3].

본 논문에서는 이산적인 HMM에서 일반적으로 많이 이용되는 동적 집단화 방법으로써 K-means 알고리듬을 이용하여 제안한 알고리듬과의 인식율을 비교하였다.

2. Hidden Markov Model

HMM은 관측할 수 없는 "hidden" process와 음성 신호로부터 이러한 hidden process의 상태로 유도되는 음향학적 벡터를 연결하는 관측 과정(observation process)으로 구성된다. 따라서 HMM에서는 관측할 수 없는 음성의 통계적인 특성을 관측 가능한 벡터열을 통해 추정하므로써 음성의 통계적인 변이성을 잘 반영하고 있다[4][5].

임의의 음성 특징벡터의 관측열 $O=(o_1 o_2 \ldots o_T)$ 이 사실임을 가정할 때 주어진 N-states HMM 모델에서의 상태열이 $q=(q_1 q_2 \ldots q_T)$ 라면 결국 관측열의 확률은 다음 식 (1)와 같이 주어진다.

$$P(O \mid q, \lambda) = \sum_{q_1 q_2 \ldots q_T} \pi_{q_1} b_{q_1}(o_1) a_{q_1 q_2} b_{q_2}(o_2) \cdots a_{q_{T-1} q_T} b_{q_T}(o_T) = \sum_{q_1 q_2 \ldots q_T} \prod_{t=1}^{T} a_{q_{t-1} q_t} b_{q_t}(o_t) \quad (1)$$

위의 식 (1)에서의 관측벡터에 대한 전향변수를 $\alpha_t(i) = p(o_1 o_2 \ldots o_t,\ q_t = i \mid \lambda)$ 로 정의하면 다음의 과정을 통해 식 (2)와 같이 관측열의 확률을 구할 수 있다.

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

여기서, $1 \le t \le T-1,\ 1 \le j \le N$

$$p(O \mid \lambda) = \sum_{i=1}^{N} \alpha_T(i) \quad (2)$$

또한, 후향변수를 $\beta_t(i) = p(o_{t+1} o_{t+2} \ldots o_T \mid q_t = i, \lambda)$로 정의할 때 다음의 과정을 통해 식 (3)과 같이 관측열의 확률을 얻을 수 있다.

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(o_{t+1}) \beta_{t+1}(j) \quad (3)$$

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

임의의 관측열의 확률을 최대로 하기위한 HMM 학습 알고리듬으로는 일반적으로 Maximum Likelihood Estimation (MLE)을 이용한다. 여기서는 Baum에의해 확률적으로 증명된 Baum-Welch 재추정 알고리듬을 적용하여 주어진 모델의 확률을 최대로하여 학습한다[6]. 모델 파라미터의 재추정식은 다음 식 (4)(5)(6)와 같이 표현된다.

$$\overline{\pi}_i = \frac{\alpha_0(i)\beta_0(i)}{\sum_{j=1}^{N} \alpha_T(j)} \quad (4)$$

$$\overline{a}_{ij} = \frac{\sum_{t=1}^{T} \alpha_{t-1}(i) a_{ij} b_j(o_t) \beta_t(j)}{\sum_{t=1}^{T} \alpha_{t-1}(i) \beta_{t-1}(i)} \quad (5)$$

$$\overline{b}_j(k) = \frac{\sum_{t=1}^{T} \alpha_t(i) \beta_t(i) \delta(o_t, v_k)}{\sum_{t=1}^{T} \alpha_t(i) \beta_t(i)} \quad (6)$$

식 (6)에서 표현한 관측심볼의 출력확률에서 delta함수는 다음 식 (7)로 정리되며 이는 관측열에서 관측심볼이 존재할 경우에 한해 확률값을 추정하는 결과를 가져온다.

$$\delta(o_t, v_k) = \begin{cases} 1, & o_t = v_k \\ 0, & o_t \neq v_k \end{cases} \quad (7)$$

결국, HMM모델의 음성인식 과정은 주어진 관측열에 대한 최대 확률분포를 갖는 모델을 결정하는 것으로 이루어진다. 여기서 전향/후향확률에 의한 연산을 이용하여 상태경로를 추적하는 경우 인식율이 다소 우수한 반면 주어진 모든 상태에서의 출력 심볼의 확률을 전부 추정하므로 계산량과 복잡도가 증가한다. 따라서 실시간을 요구하는 인식과정에서는 일반적으로 동적 프로그램 기술로서 잘 알려진 Viterbi decoding 방법을 이용하여 상태경로의 변이와 최적의 모델을 추정함으로써 인식할 수 있다[6].

III. 제안한 화자독립 VQ/HMM 알고리듬

1. Discriminative Pattern Clustering 알고리듬

일반적으로 이산적인 HMM의 Vector Quantizer에서 codebook을 생성하기 위한 알고리듬으로는 Lloyd 알고리듬으로 잘 알려진 K-means 알고리듬을 널리 이용한다. K-means 알고리듬의 기본이론은 무한히 많은 수의 입력벡터를 미리 정해진 K개의 대표값(중심값)으로 사상시키는 집단화 알고리듬으로써 임의의 입력벡터에 대한 Vector Quantizer의 양자화에러를 최소로하기 위해 반복적으로 클러스터의 중심값을 갱신하는 알고리듬이다[1][4]. 여기서의 Distortion 측정방법으로는 주로 벡터간

의 유클리드 거리에 의해 연산되며 적절한 연산량과 그다지 성능면에서 나쁘지 않다는 이유로 일반적인 HMM의 입력단에서 널리 이용되어왔다.

그러나 이러한 K-means 알고리듬의 경우 유일한 codeword로 집단화되지 못하고 초기치의 설정에따라 국부최소치를 갖기도 하는 등 단점을 갖는다[1]. 따라서 이를 개선하기 위한 수정된 집단화 알고리듬들이 발표되었으며 그 대표적인 방식으로는 LBG 알고리듬을 들 수 있다[1]. LBG 알고리듬은 반복적으로 클러스터를 분할하는 알고리듬으로써 K-means 알고리듬에서의 초기치 설정 문제를 해결하기 위해 모든 입력벡터를 하나의 클러스터로 집단화하여 초기 중심값을 설정한 후 매 반복횟수마다 현재의 클러스터들의 중심값을 이동시켜 분할한다. 결국 분할과정이 m번 진행되면 2^m 개의 클러스터를 만들게되고 미리 정해진 수 K개의 안정된 분할을 이룰때까지 반복된다. 다음은 LBG알고리듬에 의한 집단화 과정을 단계별로 살펴본 것이다.

step 1> Initialization : 모든 입력벡터에 대한 하나의 중심값을 설정한다.

$$c_1^{(p)} = \frac{\sum_{all} x^{(p)}}{N_T} \quad where, \quad N_T : \text{모든 입력벡터의 수}$$

$c_1^{(p)}$: 초기 중심값

step 2> Spliting : 각 클러스터의 중심값을 이동시켜 둘로 분할한다.

$$c_k^{(p)+} = c_k^{(p)}(1+\epsilon) \qquad 0.01 \le \epsilon \le 0.05$$
$$c_k^{(p)-} = c_k^{(p)}(1-\epsilon) \qquad k = 1, 2, \cdots\cdots, 2^{iterations}$$

step 3> Clustering : 모든 입력벡터에 대한 분할된 각 클러스터와의 유클리드 거리를 측정하여 가장 작은 거리를 갖는 클러스터의 멤버벡터로 집단화한다.

$$k^* = \arg\min_k d(x^{(p)}, c_k^{(p)})$$

step 4> Centroid update : 각 클러스터의 멤버벡터를 통해 클러스터의 중심점을 갱신한다.

$$c_{k^*}^{(p)} = \frac{1}{N_{k^*}} \sum_{x \in S_{k^*}} x^{(p)}$$
$$where, \quad N_{k^*} : S_{k^*} \text{에 소속된 멤버의 수}$$

step 5> Termination 1 : 오차의 갱신값이 설정된 임계값 이하이면 step 6 을 수행하고 그렇지 않으면 step 3 으로 되돌아간다.

step 6> Termination 2 : 정해진 수의 클러스터로 분할되었으면 작업을 종료하고, 그렇지 않으면 step 2 로 되돌아가 반복하여 수행한다.

본 논문에서 제안하고 있는 비선형적 집단화 알고리듬은 LBG 알고리듬과 유사한 과정을 통해 집단화하는 분할(spliting) 알고리듬이라 할 수 있으나 분할 조건을 반복횟수마다 무조건적으로 클러스터의 수를 두배수로 증가시키는 분할기법과는 달리 각 클러스터에 대한 멤버벡터의 수에 의거하여 최대 멤버벡터를 갖는 클러스터만을 재분할하는 것으로써 분할이 종료되면 비교적 모든 클러스터들의 멤버벡터의 수가 균등한 분포를 갖게된다. 따라서 일반적인 집단화 알고리듬과 같이 유클리드 거리만을 최소화하기 위해 집단화하게 되면 전체 벡터영역에서 특정영역에 많은 입력벡터들이 존재할 경우 이에 대한 일정한 갯수의 클러스터 중심값은 발생빈도가 높은 입력벡터들에 대해 상대적으로 저하된 변별력을 갖게되나 제안한 방법과 같이 입력벡터가 집중된 영역에 대해 더 많은 클러스터 중심값을 할당하는 비선형적인 방법으로 집단화하면 정해진 수 K개의 클러스터는 모든 입력벡터에 대해 상대적으로 우수한 변별 특성을 갖게 된다. 즉, 발생빈도수가 많은 임의 입력벡터들간의 변별력을 상대적으로 높여주는 효과를 가져오므로 통계적인 방법을 이용한 고립단어 음성인식분야에서 유사도가 높은 특징 파라미터간의 인식율을 크게 개선하는 우수한 성능을 보이게 된다.

다음은 본 논문에서 제안한 비선형적 집단화 알고리듬의 수행과정을 살펴본 것이다.

step 1> Initialization : 모든 입력벡터에 대한 하나의 중심값을 설정한다.

$$c_1^1 = \frac{\sum_{all} x^{(p)}}{N_T} \quad where, \quad N_T : \text{모든 입력벡터의 수}$$

S_1 : 모든 입력벡터 $x^{(p)}$의 집합

step 2> Searching : 현재의 모든 클러스터 중 최대 멤버 수를 갖는 클러스터를 검출한다.

$$\overline{k} = \arg\max_k M(S_k) \qquad k = 1,2,\cdots\cdots,K$$

$M(S_k)$: k번째 클러스터의 멤버 수

step 3> Spliting : 최대 멤버의 수를 갖는 클러스터의 중심값을 이동시켜 둘로 분할한다.

$$c_{\overline{k}}^{(p)+} = c_{\overline{k}}^{(p)}(1+\epsilon) \qquad 0.01 \le \epsilon \le 0.05$$
$$c_{\overline{k}}^{(p)-} = c_{\overline{k}}^{(p)}(1-\epsilon)$$

step 4> Clustering : 모든 입력벡터에 대한 분할된 각 클러스터와의 유클리드 거리를 측정하여 가장 작은 거리를 갖는 클러스터의 멤버벡터로 집단화한다.

$$k^* = \arg\min_k d(x^{(p)}, c_k^{(p)})$$

step 5> Centroid update : 각 클러스터의 멤버벡터를 통해 클러스터의 중심점을 갱신한다.

$$c_{k'} = \frac{1}{N_{k'}} \sum_{x \in S_{k'}} x$$

where, $N_{k'}$: $S_{k'}$에 소속된 멤버의 수

step 6> Termination 1 : 오차의 갱신값이 설정된 임계값 이하이면 step 7 을 수행하고 그렇지 않으면 step 4로 되돌아간다.

step 7> Termination 2 : 정해진 수의 클러스터로 분할되었으면 작업을 종료하고, 그렇지 않으면 step 2 로 되돌아가 반복하여 수행한다.

2. 제안한 DHMM Smoothing 기법

기존의 이산적인 HMM은 주어진 관측열의 관측시퀀스가 사실이라는 가정하에 최대확률분포를 갖도록 Baum-Welch 재추정 알고리듬을 통해 학습한다. 그러므로 학습데이타가 불충분하여 실제 음성에서는 자주 발생하는 출력 심볼이 학습시 관측열에서는 나타나지 않게되면 학습시 관측심볼에 대한 재추정 출력확률분포가 식 (6)과 (7)에서 살펴본 바와 같이 관측된 심볼이 있을 경우에 한해 학습되므로 매우 낮은 확률값으로 학습하거나 재추정된 확률분포를 0으로 만들게 되어 HMM 인식과정에서 인식율 저하의 근본적인 원인이 된다[1][5]. 따라서 학습 관측열의 시퀀스를 증가시키거나 상대적으로 HMM 모델의 크기를 감소시켜 free 파라미터의 수를 작게 설정하는 parameter tying과 전 체 학습데이터에 대해 이를 분할한 또다른 파라미터에 가중치를 두어 삽입하는 deleted interpolation이 제시되고 있다[1][4].

그러나 실제적으로 한정된 학습데이타 입력에 대해 인위적으로 학습 데이터 수를 증가시키는데는 한계가 있으며 모델의 크기를 축소하게 되면 변별력있는 학습에 제한을 받게 된다. 또한, deleted interpolation에 의한 HMM 모델에서는 입력 학습데이타를 적절히 분류하여 개별적인 학습모델을 생성하고 각 모델에 가중치를 두어 HMM 모델을 smoothing하므로 화자독립 음성인식에서 우수한 인식결과를 가져오나 하나 이상의 또다른 모델을 생성할 충분히 많은 학습데이터가 요구되며 재추정되는 모델의 수가 많아지므로 많은 학습량과 학습시간이 요구되는 단점이 있다.

따라서 본 논문에서는 출력심볼의 발생확률을 재추정하는 과정에서 학습 관측열에서 각 심볼별 확률분포를 통해 심볼의 출력확률을 smoothing하는 기법을 통해 deleted interpolation에서의 모델의 중복성에 의한 연산량을 줄이고 불충분한 학습데이타로 인해 발생되는 인식율의 저하를 방지하므로써 재추정 파라미터의 신뢰성을 높였다.

다음 식 (8)과 (9)는 본 논문에서 제안하고 있는 출력확률분포의 재추정과정에서 심볼의 발생확률을 고려한 재추정식을 보여준다.

$$\overline{b_i}(k) = \frac{\sum_{t=1}^{T} \alpha_t(i)\beta_t(i) \cdot f_{O_t}(v_k)}{\sum_{t=1}^{T} \alpha_t(i)\beta_t(i)} \quad (8)$$

여기서,

$$f_{O_t}(v_k) = \frac{\text{관측열에서 심볼 } v_k\text{가 발생할 기대치}}{\text{학습 데이타 열에서의 심볼의 총 수}}$$

$$\text{if } \forall t,\ o_t \neq v_k, \quad \text{then } f_{o_t}(v_k) = \epsilon \quad (9)$$

where, $\epsilon = 0.000005$

식 (9)에서 표현한 학습 심볼의 발생확률 $f_{O_t}(v_k)$은 HMM 입력단의 벡터양자화 과정에서 얻을 수 있으며 이는 주어진 관측열에서의 각 심볼의 발생확률을 의미한다. 기존의 이산적인 HMM이 식 (7)에서와 같이 출력심볼의 관측확률을 delta함수에 의해 재추정하는 반면 제안한 알고리듬의 경우 관측벡터열에 대한 심볼의 발생확률분포를 인가하므로써 관측데이타가 불충분할 경우 화자의 특이성에 따른 인식율 저하와 학습과정에서 전혀 관측되지 않은 심볼에 대해 최소확률로 학습하고 확률계산시 scaling하므로써 관측열의 부족으로 인해 실제 발생할 수 있으나 관측되지 않았던 데이터에 대한 인식율을 향상시켰다.

다음 그림 1은 본 논문에서 이용하고 있는 VQ/HMM 시스템의 전체 구성도이다.

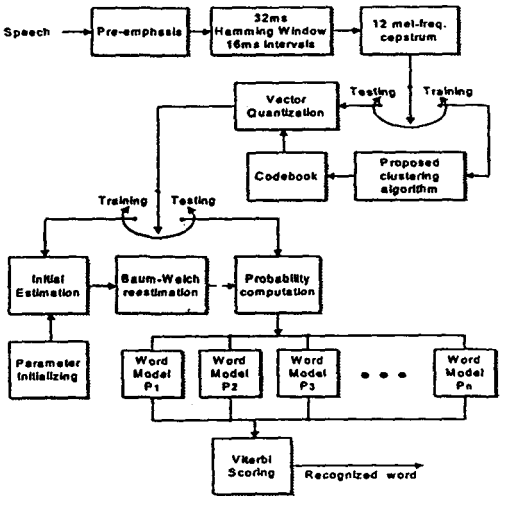

그림 1. 전체 시스템 구성도

Fig. 1. Block-diagram of total system.

IV. 모의 실험 및 결과

본 논문에서 사용한 음성데이타는 아래 표 1과 같이 10개의 윈도우 명령어로 구성하였으며 남성화자 30명으로부터 각 3회씩 발성한 900개의 고립단어를 학습데이타

로 사용하였고 남성화자 22명으로부터 각 4회씩 발성하여 얻은 880개의 고립단어를 이용하여 테스트하였다. 각 음성은 mono 8[KHz]로 표본화되었으며 각 샘플당 16[bit]의 resolution으로 디지털 변환되었다.

표 1. 모의 실험에서 사용된 고립단어
Table 1. Isolated words used in simulation.

구분	_0	_1	_2	_3	_4	_5	_6	_7	_8	_9
단어	시작	위로	아래로	오른쪽	왼쪽	실행	메뉴	탭	나가기	닫기

본 논문에서 사용한 음성특징벡터로는 음성의 스펙트럼에 기초한 선형예측계수를 liftering이나 weighting과정을 통해 쉽게 완화시켜 스펙트럼의 변이성을 제거하여 음성의 정적특성을 강조하므로써 높은 인식율을 보여주는 12차 Mel-frequency cepstrum 계수(MFCC)를 이용하였다.

본 논문의 모의실험 결과는 기존의 VQ/HMM과 제안한 알고리듬을 적용한 이산적인 HMM에서의 화자독립 고립단어 인식율의 비교를 위한 것으로 인식율 전반을 향상하기 위한 고차의 delta- cepstrum 이나 power spectrum 등의 특징 파라미터들을 생략하여 적은 연산량을 통해 알고리듬을 비교 분석하였다.

1. 기존의 VQ/HMM에 대한 인식실험 및 결과

기존의 VQ/HMM에 의한 모의실험에서는 30명의 화자로부터 추출한 900개의 고립단어에 대한 12차 mel-frequency cepstrum 계수를 음성 특징벡터로 변환하여 K-means 집단화 알고리듬을 이용해 128개의 코드북벡터를 생성하였다. 그리고, 이를 중심값 벡터로 하여 입력벡터에 대한 유클리드 거리를 최소로 하는 중심값 벡터의 인덱스를 출력하므로써 HMM의 입력 관측시퀀스를 만들게 된다. 따라서 이러한 과정으로 만들어진 각 단어별 30개의 관측 시퀀스는 이산적인 HMM (DHMM) 에 인가되어 학습되어진다.

HMM 학습과정에서는 Baum-Welch 재추정 알고리듬을 통해 학습하고자 하는 관측시퀀스의 확률을 최대로 한다. 이때 임계값을 줌으로써 갱신되는 모델 파라미터의 확률들의 총합이 임계값 이하이면 재추정 알고리듬을 종료하여 특정 고립단어에 대한 HMM 모델을 생성한다. 본 논문의 모의실험 과정에서는 재추정 임계값을 0.0005로 선정하여 각 단어별 HMM 모델을 생성하였고 22명의 화자에 대한 880개의 고립단어로 화자독립 인식실험을 하였다.

표 2는 기존의 일반적인 VQ/HMM을 이용한 화자독립 실험에서의 단어별, 화자별 인식율을 나타낸 것으로 평균 90.23%의 인식율이 나타났다.

표 6은 VQ/HMM 학습과정에서 각 단어별 모델을 생성하는데 소요되는 반복횟수를 나타낸 것으로 본 논문에서 설정하고 있는 임계값에 대해 기존의 VQ/HMM은 평균 417회의 반복횟수가 소요되었다.

2. 제안한 VQ/HMM에 대한 인식실험 및 결과

K-means 집단화 알고리듬을 이용한 기존의 이산적인 HMM에서의 화자독립 고립단어 인식율에 대해 본 논문에서 제안한 비선형적 집단화 알고리듬을 적용하였을 경우의 인식율을 비교하기 위해 HMM 학습시 전처리 과정에서 생성된 동일한 음성특징 파라미터를 제안한 비선형적 집단화 알고리듬을 이용하여 128개의 codebook을 형성한 후 학습과정을 통해 HMM의 각 단어별 모델을 생성하였다.

표 3에서는 이러한 비선형적 집단화 알고리듬을 적용하여 화자독립 인식실험에 대한 결과를 보여주고 있다. 비선형 집단화 알고리듬을 적용한 본 실험에서는 92.39%의 인식율을 보여 기존의 VQ/HMM 에 비해 2.16% 인식율이 향상되었다.

또한, 불충분한 학습 데이터로부터 한정된 수의 벡터 시퀀스를 학습할 경우 발생하는 인식율 저하를 극복하기 위해 HMM 학습과정에서의 출력 심볼의 관측확률 재추정시 전체 학습 시퀀스에서의 입력 벡터에 대한 확률분포를 고려하여 갱신하므로써 불충분한 데이터에 대한 HMM에서의 인식율을 보다 향상시켰다.

표 4는 기존의 VQ/HMM에 이러한 확률적인 smoothing 기법을 적용한 화자독립에서의 인식실험 결과를 보여준다. 여기서는 기존의 VQ/HMM에 비해 화자독립에서 93.30%의 인식율을 보여줌으로써 3.07%의 인식율이 향상되었다.

표 5에서는 제안한 두가지의 개선된 알고리듬을 적용하여 화자독립에서의 인식실험 결과를 나타내었다. 여기서는 비선형적 집단화 알고리듬을 적용하여 벡터양자화를 취하고 확률적으로 개선된 smoothing 기법을 적용하므로써 기존의 VQ/HMM에 비해 4.66% 향상된 94.89%의 인식율을 보여준다.

또한 표 6과 표 7에서는 기존의 VQ/HMM과 제안한 두가지의 알고리듬을 적용한 VQ/HMM에서의 학습시 각 단어별 모델 생성을 위한 재추정 반복횟수를 나타내었다. 제안한 알고리듬의 경우 인식율의 향상뿐만아니라 기존의 VQ/HMM에 비해 동일한 임계값에서 모델 파라미터가 갱신되는 동적 범위를 줄여줌으로써 오히려 평균 반복횟수도 217회 감소한 200회의 평균 반복횟수를 보여주고 있다.

그림 2에서는 기존의 VQ/HMM을 이용한 화자독립 인식실험에서의 평균인식율과 본 논문에서 제안한 두가지 알고리듬에 대한 각각의 평균 인식율 및 두가지 알고리듬을 모두 적용한 평균 인식율을 비교하고 있다.

그림 3에서는 기존의 VQ/HMM과 제안한 각 알고리듬을 적용한 VQ/HMM에서의 단어별 인식율을 비교하였다. 특히 여기서는 제안한 비선형 집단화 알고리듬이 발성구간에 따라 유사한 단어의 변별력을 향상하는데 우수한 성능을 가진다는 것을 잘 보여준다.

그림 4에서는 기존의 VQ/HMM과 제안한 알고리듬을 적용하여 확률적으로 smoothing된 VQ/HMM에서의 학습시 단어별 반복횟수를 비교하여 나타내었다.

표 2. 기존의 VQ/HMM에 의한 화자독립 고립단어 인식율
Table 2. The rate of speaker-independent isolated word recognition in original VQ/HMM.

단어 / 화자	시작 _0	위로 _1	아래로 _2	오른쪽 _3	왼쪽 _4	실행 _5	메뉴 _6	탭 _7	나가기 _8	닫기 _9	계	인식율 (%)
AE	4/4	3/4	4/4	4/4	4/4	4/4	4/4	4/4	3/4	4/4	38/40	95.00
AF	4/4	3/4	4/4	3/4	4/4	4/4	4/4	3/4	3/4	4/4	36/40	90.00
AG	4/4	3/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	39/40	97.50
AH	4/4	4/4	4/4	4/4	4/4	4/4	4/4	3/4	4/4	4/4	39/40	97.50
AI	4/4	4/4	4/4	1/4	4/4	4/4	3/4	4/4	4/4	4/4	36/40	90.00
AJ	4/4	3/4	3/4	1/4	3/4	4/4	3/4	4/4	4/4	1/4	30/40	75.00
AK	4/4	4/4	4/4	2/4	4/4	4/4	3/4	2/4	4/4	4/4	35/40	87.50
AL	4/4	3/4	2/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	37/40	92.50
AM	4/4	2/4	4/4	0/4	4/4	4/4	4/4	4/4	0/4	4/4	30/40	75.00
AN	4/4	3/4	4/4	1/4	4/4	4/4	4/4	4/4	1/4	4/4	33/40	82.50
AO	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	40/40	100
AP	4/4	0/4	4/4	4/4	2/4	4/4	4/4	4/4	3/4	4/4	33/40	82.50
AQ	4/4	1/4	4/4	3/4	4/4	4/4	2/4	3/4	4/4	4/4	33/40	82.50
AR	4/4	2/4	4/4	4/4	3/4	4/4	4/4	2/4	4/4	4/4	35/40	87.50
AS	4/4	4/4	4/4	4/4	4/4	4/4	4/4	3/4	4/4	4/4	39/40	97.50
AT	4/4	4/4	3/4	2/4	4/4	4/4	4/4	4/4	2/4	4/4	35/40	87.50
AU	4/4	4/4	4/4	3/4	4/4	4/4	4/4	4/4	4/4	4/4	39/40	97.50
AV	4/4	1/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	37/40	92.50
AW	4/4	4/4	4/4	4/4	2/4	4/4	4/4	3/4	4/4	3/4	36/40	90.00
AX	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	3/4	39/40	97.50
AY	3/4	2/4	4/4	4/4	4/4	4/4	3/4	4/4	4/4	4/4	36/40	90.00
AZ	4/4	4/4	4/4	3/4	4/4	4/4	4/4	4/4	4/4	4/4	39/40	97.50
계	87/88	66/88	84/88	67/88	82/88	88/88	82/88	79/88	76/88	83/88	794/880	90.23
인식율 (%)	98.86	75.00	95.45	76.14	93.18	100	93.18	89.77	86.36	94.32	90.23	

표 3. 비선형적 집단화 알고리듬을 적용한 VQ/ HMM에서의 화자독립 고립단어 인식율
Table 3. The rate of speaker-independent isolated word recognition in VQ/HMM applied nonlinear clustering algorithm.

단어 / 화자	시작 _0	위로 _1	아래로 _2	오른쪽 _3	왼쪽 _4	실행 _5	메뉴 _6	탭 _7	나가기 _8	닫기 _9	계	인식율 (%)
AE	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	3/4	4/4	39/40	97.50
AF	4/4	3/4	4/4	3/4	4/4	4/4	4/4	3/4	3/4	4/4	36/40	90.00
AG	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	40/40	100
AH	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	40/40	100
AI	4/4	4/4	4/4	4/4	4/4	4/4	4/4	3/4	3/4	4/4	38/40	95.00
AJ	4/4	3/4	4/4	2/4	3/4	4/4	4/4	4/4	2/4	3/4	33/40	82.50
AK	4/4	4/4	4/4	4/4	4/4	4/4	4/4	3/4	3/4	4/4	38/40	95.00
AL	4/4	3/4	4/4	2/4	4/4	4/4	4/4	4/4	4/4	4/4	37/40	92.50
AM	4/4	4/4	4/4	1/4	4/4	4/4	4/4	4/4	4/4	4/4	37/40	92.50
AN	4/4	2/4	4/4	2/4	4/4	4/4	4/4	4/4	1/4	4/4	33/40	82.50
AO	4/4	3/4	4/4	3/4	4/4	4/4	4/4	4/4	4/4	4/4	38/40	95.00
AP	4/4	1/4	4/4	3/4	3/4	4/4	4/4	4/4	3/4	4/4	34/40	85.00
AQ	4/4	0/4	4/4	2/4	4/4	4/4	4/4	4/4	4/4	4/4	34/40	85.00
AR	4/4	3/4	4/4	3/4	3/4	4/4	4/4	2/4	3/4	4/4	34/40	85.00
AS	4/4	3/4	4/4	3/4	4/4	4/4	4/4	4/4	4/4	4/4	38/40	95.00
AT	4/4	3/4	3/4	2/4	4/4	4/4	4/4	4/4	4/4	4/4	36/40	90.00
AU	4/4	4/4	4/4	1/4	4/4	4/4	4/4	4/4	4/4	4/4	37/40	92.50
AV	4/4	2/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	38/40	95.00
AW	4/4	3/4	4/4	4/4	3/4	4/4	4/4	4/4	4/4	4/4	38/40	95.00
AX	4/4	3/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	39/40	97.50
AY	4/4	3/4	3/4	4/4	4/4	4/4	3/4	4/4	4/4	4/4	37/40	92.50
AZ	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	3/4	4/4	39/40	97.50
계	88/88	67/88	86/88	67/88	84/88	88/88	87/88	83/88	76/88	87/88	813/880	92.39
인식율 (%)	100	76.14	97.73	76.14	95.45	100	98.86	94.32	86.36	98.86	92.39	

표 4. 확률적인 Smoothing 알고리듬을 적용한 VQ/HMM의 화자독립 고립단어 인식율
Table 4. The rate of speaker-independent isolated word recognition in VQ/HMM applied propabilistic smoothing algorithm.

단어 / 화자	시작 _0	위로 _1	아래로 _2	오른쪽 _3	왼쪽 _4	실행 _5	메뉴 _6	탭 _7	나가기 _8	닫기 _9	계	인식율 (%)
AE	4/4	3/4	4/4	4/4	4/4	4/4	4/4	4/4	3/4	4/4	38/40	95.00
AF	4/4	4/4	4/4	3/4	4/4	4/4	4/4	4/4	3/4	4/4	38/40	95.00
AG	4/4	3/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	39/40	97.50
AH	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	40/40	100
AI	4/4	4/4	4/4	4/4	1/4	4/4	4/4	3/4	4/4	4/4	36/40	90.00
AJ	4/4	3/4	3/4	4/4	4/4	4/4	4/4	4/4	3/4	4/4	37/40	92.50
AK	4/4	4/4	4/4	4/4	4/4	4/4	3/4	4/4	3/4	4/4	38/40	95.00
AL	4/4	2/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	38/40	95.00
AM	4/4	2/4	4/4	2/4	4/4	4/4	4/4	4/4	0/4	4/4	32/40	80.00
AN	4/4	4/4	4/4	3/4	3/4	4/4	3/4	4/4	0/4	4/4	31/40	77.50
AO	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	40/40	100
AP	4/4	3/4	4/4	4/4	2/4	4/4	2/4	4/4	3/4	4/4	34/40	85.00
AQ	4/4	1/4	4/4	4/4	4/4	4/4	2/4	4/4	3/4	4/4	34/40	85.00
AR	4/4	3/4	4/4	4/4	3/4	4/4	4/4	3/4	4/4	4/4	37/40	92.50
AS	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	40/40	100
AT	4/4	4/4	4/4	2/4	4/4	4/4	4/4	4/4	3/4	4/4	37/40	92.50
AU	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	40/40	100
AV	4/4	3/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	39/40	97.50
AW	4/4	4/4	4/4	4/4	1/4	4/4	4/4	4/4	4/4	3/4	36/40	90.00
AX	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	40/40	100
AY	4/4	4/4	3/4	4/4	4/4	4/4	2/4	4/4	4/4	4/4	37/40	92.50
AZ	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	40/40	100
계	88/88	75/88	86/88	82/88	76/88	88/88	80/88	86/88	73/88	87/88	821/880	93.30
인식율 (%)	100	85.23	97.73	93.18	86.36	100	90.91	97.73	82.95	98.86	93.30	

표 5. 비선형 집단화/확률적인 Smoothing 기법을 적용한 VQ/HMM 화자독립 고립단어 인식율
Table 5. The rate of speaker-independent isolated word recognition in VQ/HMM applied nonlinear clustering / probabilistic smoothing algorithm

단어 / 화자	시작 _0	위로 _1	아래로 _2	오른쪽 _3	왼쪽 _4	실행 _5	메뉴 _6	탭 _7	나가기 _8	닫기 _9	계	인식율 (%)
AE	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	3/4	4/4	39/40	97.50
AF	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	40/40	100
AG	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	40/40	100
AH	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	40/40	100
AI	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	3/4	4/4	39/40	97.50
AJ	4/4	4/4	4/4	4/4	3/4	4/4	4/4	4/4	3/4	4/4	38/40	95.00
AK	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	3/4	4/4	39/40	97.50
AL	4/4	3/4	4/4	3/4	4/4	4/4	4/4	4/4	4/4	4/4	38/40	95.00
AM	4/4	3/4	4/4	1/4	4/4	4/4	4/4	4/4	1/4	4/4	33/40	82.50
AN	4/4	4/4	4/4	3/4	3/4	4/4	4/4	4/4	0/4	4/4	34/40	85.00
AO	4/4	4/4	4/4	4/4	3/4	4/4	4/4	4/4	4/4	4/4	39/40	97.50
AP	4/4	2/4	4/4	4/4	4/4	4/4	2/4	4/4	4/4	4/4	36/40	90.00
AQ	4/4	1/4	4/4	3/4	4/4	4/4	3/4	4/4	4/4	4/4	35/40	87.50
AR	4/4	3/4	4/4	4/4	4/4	4/4	4/4	3/4	4/4	4/4	38/40	95.00
AS	4/4	4/4	4/4	3/4	4/4	4/4	4/4	4/4	4/4	4/4	39/40	97.50
AT	4/4	3/4	4/4	2/4	4/4	4/4	4/4	4/4	2/4	4/4	35/40	87.50
AU	4/4	4/4	4/4	3/4	4/4	4/4	4/4	4/4	4/4	4/4	39/40	97.50
AV	4/4	3/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	39/40	97.50
AW	4/4	4/4	4/4	4/4	2/4	4/4	4/4	4/4	4/4	4/4	38/40	95.00
AX	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	40/40	100
AY	4/4	3/4	3/4	4/4	4/4	4/4	3/4	4/4	4/4	4/4	37/40	92.50
AZ	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	4/4	40/40	100
계	88/88	77/88	87/88	78/88	83/88	88/88	84/88	87/88	75/88	88/88	835/880	94.89
인식율 (%)	100	87.50	98.86	88.64	94.32	100	95.45	98.86	85.23	100	94.89	

표 6. 기존의 VQ/HMM 학습시 재추정 반복횟수
Table 6. The number of iterations for learning of original VQ/HMM.

단어 모델	시작 _0	위로 _1	아래로 _2	오른쪽 _3	왼쪽 _4	실행 _5	메뉴 _6	탭 _7	나가기 _8	닫기 _9	평균
반복 횟수	357	740	843	314	509	464	332	283	197	133	417

표 7. 제안한 알고리듬에 의한 학습시 재추정 반복횟수
Table 7. The number of iterations for learning of VQ/HMM with proposed algorithm.

단어 모델	시작 _0	위로 _1	아래로 _2	오른쪽 _3	왼쪽 _4	실행 _5	메뉴 _6	탭 _7	나가기 _8	닫기 _9	평균
반복 횟수	88	228	451	307	225	57	47	191	151	257	200

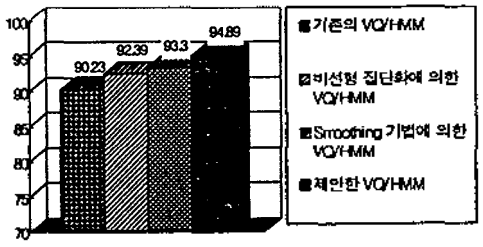

그림 2. 전체 인식율 비교
Fig. 2. Comparison of total recognition rate.

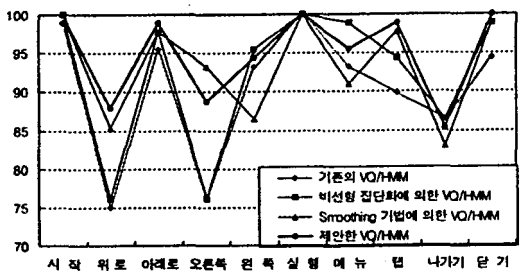

그림 3. 화자독립 실험에서의 단어별 인식율 비교
Fig. 3. Comparison of recognition rates on speaker-independent experimentation.

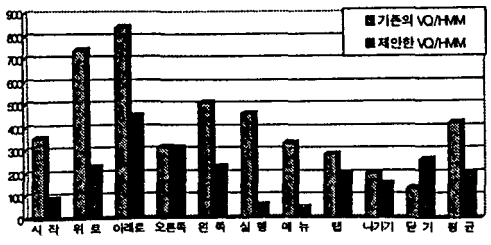

그림 4. 학습시 반복횟수 비교
Fig. 4. Comparison of numbers of iterations for learning.

V. 결 론

본 논문의 연구에서는 고립단어 인식을 위해 기존의 VQ/HMM에서의 집단화 알고리듬을 수정하여 유사한 음성구간의 변별력을 향상시켜주기 위한 비선형적 집단화 알고리듬을 제안하였으며, HMM 학습과정에서 불충분한 학습데이터로 인한 인식율 저하를 막기 위한 확률적인 smoothing 기법을 제안하였다. 그 결과 화자독립시 4.66%의 평균인식율의 향상을 보였다.

이는 HMM 학습과정에서 불충분한 학습데이터로 인한 인식율저하문제를 극복하고 있음을 잘 보여주고 있다. 특히 단어별 인식율에서 보여준 것과 같이 비교적 장시간 지속되는 모음구간의 유사성이나 동일모음의 발성음에서 초성, 종성자음의 변별력 문제로 인해 많은 오류를 발생시키는 "왼쪽", "메뉴", "닫기"의 인식율을 기존의 방식에 비해 각각 2.27%, 5.68%, 4.54% 향상시켰고 단어구간이 짧아 특징벡터의 변별력이 저하된 "탭"의 인식율에서도 4.55%의 향상을 보였다. 이는 실험과정에서 "왼쪽"의 발성음이 주로 "오른쪽"으로 오인식되고, "메뉴"의 발성음이 "위로"로 오인식되었던 점, 그리고 "닫기"라는 발성음이 "나가기"로 오인식되었던 점을 감안할 때 음성특징 파라미터의 추출과정에서 유사한 단어구간에서의 변별력을 상대적으로 향상시킨 결과를 잘 말해준다.

HMM 모델의 학습과정에서는 상태의 수, 상태별 출력심볼의 수, 재추정 알고리듬에서의 임계값 등에 의해 학습 반복횟수와 학습시간을 결정하게 되는데 동일한 조건하에서 제안한 알고리듬의 경우 HMM 학습과정에서 소요되는 재추정 알고리듬의 평균 반복횟수를 반감시켜 학습시 더욱 빠른 속도로 수렴하는 결과를 가져왔다.

이는 Baum-Welch 재추정 과정에서 갱신되는 모델의 확률에 대한 동적범위가 제안한 알고리듬의 경우 더욱 작아지므로 기존의 방식에 비해 오히려 빠른속도로 수렴된 결과이다.

참 고 문 헌

1. X. D. Huang, Y. Ariki, M. A. *Jack, Hidden Markov Models for Speech Recognition*, Edinburgh information tech., 1990.
2. Allen Gersho, Robert M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic Publishers, 1992.
3. John R. Deller, Joho G. Proakis, Jphn H. L. Hansen, *Discrete-Time Processing of Speech Signals*, Macmillan Publishing Company, 1993.
4. L. Rabiner, Biing-Hwang Juang, *Fundamentals of Speech Recognition*, Prentice-Hall International, Inc., 1993.
5. L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proc. IEEE*, Vol. 77, No. 2, pp. 257-286, 1989.
6. Valtcho Valtchev, *Discriminative Methods in HMM- based Speech Recognition, University* of Cambridge, 1995.

▲정 희 석(Heui Suck Chung) 1968년 9월 30일일생

1996년 8월 : 광운대학교 전자통신공학과 공학사
1998년 8월 : 광운대학교 전자통신공학과 공학석사
1999년 3월~현재 : 광운대학교 대학원 전자통신공학과 박사과정
※주관심분야 : 음성인식, 통신신호처리, 신경망응용

▲강철호(Chul Ho Kang)
한국음향학회지 제17권 8호(77쪽) 참조