

다층회귀신경예측 모델 및 HMM 를 이용한 임베디드 음성인식 시스템 개발에 관한 연구

A Study on Development of Embedded System for Speech Recognition using Multi-layer Recurrent Neural Prediction Models & HMM

*김 정훈, *장 원일, *김 영탁, *이 상배

*Jung hoon Kim, *Won il Jang, *Young tak Kim, *Sang bae Lee

*한국해양대학교 전자통신공학과

요 약

본 논문은 주인식기로 흔히 사용되는 HMM 인식 알고리즘을 보완하기 위한 방법으로 회귀신경회로망(Recurrent neural networks : RNN)을 적용하였다. 이 회귀신경회로망 중에서 실 시간적으로 동작이 가능하게 한 방법인 다층회귀신경예측 모델(Multi-layer Recurrent Neural Prediction Model : MRNPM)을 사용하여 학습 및 인식기로 구현하였으며, HMM과 MRNPM 을 이용하여 Hybrid형태의 주 인식기로 설계하였다. 설계된 음성 인식 알고리즘을 잘 구별되지 않는 한국어 숫자음(13개 단어)에 대해 화자 독립형으로 인식률 테스트 한 결과 기존의 HMM인식기 보다 5%정도의 인식률 향상이 나타났다. 이 결과를 이용하여 실제 DSP(TMS320C6711) 환경 내에서 최적(인식) 코드만을 추출하여 임베디드 음성 인식 시스템을 구현하였다. 마찬가지로 임베디드 시스템의 구현 결과도 기존 단독 HMM 인식시스템보다 향상된 인식시스템을 구현할 수 있게 되었다.

Abstract

In this paper, the recurrent neural networks (RNN) is applied to compensate for HMM recognition algorithm, which is commonly used as main recognizer. Among these recurrent neural networks, the multi-layer recurrent neural prediction model (MRNPM), which allows operating in real-time, is used to implement learning and recognition, and HMM and MRNPM are used to design a hybrid-type main recognizer. After testing the designed speech recognition algorithm with Korean number pronunciations (13 words), which are hardly distinct, for its speech-independent recognition ratio, about 5% improvement was obtained comparing with existing HMM recognizers. Based on this result, only optimal (recognition) codes were extracted in the actual DSP (TMS320C6711) environment, and the embedded speech recognition system was implemented. Similarly, the implementation result of the embedded system showed more improved recognition system implementation than existing solid HMM recognition systems.

Key words : MRNPM, Neural Network, HMM, VQ, Speech Recognition, DSP

1. 서 론

음성인식은 기계와 인간사이의 통신에 필수적인 기술이다. 현재까지 음성 인식 기술은 상당한 발전된 상태이며, 여러 가지 응용 제품들이 출시되고 있는 실정이다. 이러한 기술로 DTW(Dynamic Time Warping), NN(Neural Network), HMM(Hidden Markov Model) 등이 음성인식을 위해서 주로 연구되어 왔다.

DTW는 화자의 발성시간의 차이를 효과적으로 흡수하나 각 화자간의 음성학적 차이를 모델링 하는데 한계가 있어 주로 화자 종속형에 사용되어지고 있다.

NN으로 음성 인식 접근 방식은 90년 초반 큰 발전이 있었으며, BP(Back propagation), TDNN (Time Delay Neural Network), RNN(Recurrent Neural Network) 등이 있다. 이러한 방식으로 테스트 결과 음소별 음성인식에서는 HMM보다 더

높은 인식률 결과가 나타났다. 그러나 고립단어와 연속음성인식 시스템 에서는 HMM과 비교하여 현저히 낮은 인식률 나타냈으며, 이로 인해 실제 적용 사례는 적다.

HMM은 확률적인 모델로서 음성 인식 분야에서는 가장 많이 사용되고 있으나 학습 시 요구되는 수학적 가정들이 많아 임베디드 상에는 학습 단계를 실행하기 에는 어려움 점이 있지만, 인식 알고리즘은 실제 임베디드 환경에서 사용되고 있다. 이러한 문제점을 해결하는 방법으로 학습 단계는 범용 PC상에서 학습을 시키고, 학습된 정보만을 실제 임베디드 시스템에 적용하여 인식 부분만을 구현하는 방법이 있다. 현재까지 개발된 제품들은 보통 이 방식을 이용하여 구현되었으며, 인식률은 보통 80~90% 이다. 이러한 제품들을 사용함에 있어 실제 인식률은 더욱 크게 떨어지며, 그 이유로 잡음 환경이 가장 중요한 요인으로 작용하지만, 그에 못지않게 인식 알고리즘도 상당한 요인으로 작용한다. 이에 본 논문에서는 HMM와 NN의 혼합 형태인 인식 알고리즘을 사용하여 인식기를 구현하였다. HMM의 경우 기존의 알고리즘을 사용하였으며, 신경망의 경우 RNN 방식을 더욱 더 향상시킨 MRNPM을 사용하였다.

접수일자 : 2004년 12월 6일

완료일자 : 2004년 5월 18일.

본 논문의 구성은 다음과 같다. 2장에서는 본 시스템에서 설계된 음성인식 알고리즘에 대한 내용들을 기술하며, 3장에서는 제안된 음성인식 알고리즘을 통해 PC 및 TMS320C6711에서의 구현된 내용을 서술하며, 4장에서는 여러 가지 실험을 통해서 제안된 방식의 효율성을 검증하며, 마지막에는 결론으로 본 논문을 요약한다.

2. 복합형 음성 인식 알고리즘

본 논문에서 제안된 Hybrid 형태의 음성 인식 알고리즘은 크게 HMM과 MRNPM을 사용하였다. 주로 다층회귀신경 예측 모델 중심으로 서술할 것이다.

2.1 HMM 모델의 학습 및 인식

HMM의 학습에는 Baum-Welch 알고리즘을 사용하였으며, 인식부에서는 ML(Maximum likelihood) 알고리즘을 사용하였다[1].

HMM 학습 알고리즘(Baum-Welch)을 이용하여 13개의 단어별 λ 값들을 생성하였다.

$$\lambda_i = (\pi, A, B) \quad i: \text{단어 종류식} \quad (1)$$

π : Initial stat probability

A : State transition probability

B : State observation probability

그림 1은 Baum Welch 추정 알고리즘을 이용하여 HMM 학습 알고리즘에 대한 전체 구성도이며, 학습을 통해서 단어별 λ 값들이 최종 나오게 된다. 이 λ 값들을 이용하여 새롭게 입력되는 음성 벡터와 조건 확률 계산을 처리함으로써, 최종 확률 값을 구할 수 있다. 이 결과 가장 큰 확률 값을 인식 단어로 결정한다.

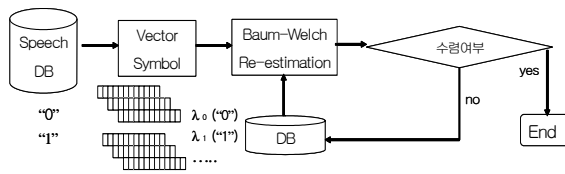


그림 1. HMM 학습 알고리즘

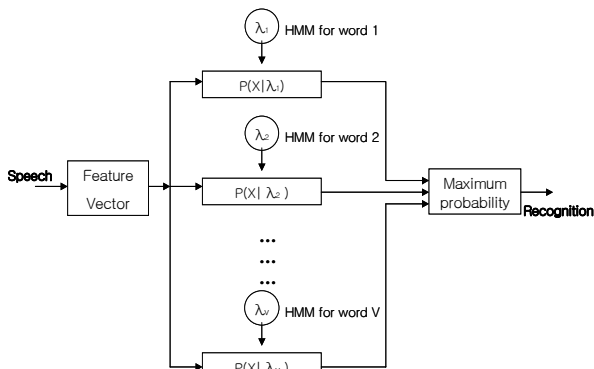


그림 2. HMM 인식 알고리즘

그림 2는 인식 알고리즘으로 Viterbi algorithm을 사용하였다.

2.2. 다층회귀신경예측 모델과 학습 및 인식 알고리즘

2.2.1 다층회귀신경예측 모델의 구조

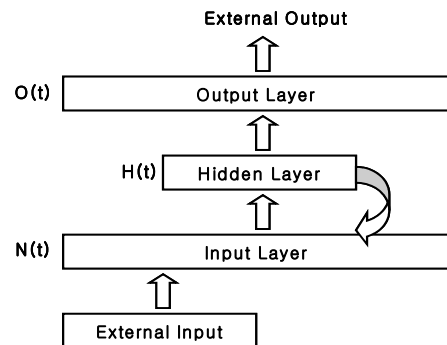


그림 3. 다층회귀신경예측 모델의 구조

다층신경회로망에는 여러 가지 회귀구조가 있는데, 본 논문에서는 Elman 유형의 구조인 예측기를 학습 과정에 사용하였고, 이 구조는 그림 3과 같다[2].

입력층과 출력층 사이에 은닉층을 첨가하고, 은닉층의 모든 뉴런에 입력층으로의 회귀 연결을 두었다. 출력층에는 선형출력함수, 은닉층은 시그노이드 함수를 사용하였다. 학습신호 $s^T = \{s(1), s(2), \dots, s(T)\}$ 를 길이 T의 학습 신호라 할 때, $s(t) = [s_1(t), s_2(t), \dots, s_N(t)]$ 는 시간 t에서의 N차원 음성 특징 벡터가 된다.

본 논문에서는 프레임당 MFCC 특징 추출 결과 1개의 벡터의 수(N)를 12개로 정했으며, 음성의 전체 프레임의 길이(T)는 일정하지는 않지만, 제안된 신경망 구조에서는 프레임의 길이와 관계없이 학습이 가능한 구조이다.

시간 t에서의 표적 벡터를 S(t)라 할 때, 예측 차수가 τ 인 경우, 외부 입력 벡터인 $x(t) = [x_1(t), x_2(t), \dots, x_N(t)]$ 은 $s(t-1), s(t-2), \dots, s(t-\tau)$ 의 τ 개의 벡터를 연결한 벡터이다. 은닉층에 H개의 은닉 뉴런이 존재하면, 시간 t에서의 은닉층의 전체입력(net input) 값들로 이루어진 벡터를 $n(t)$, 은닉층의 출력벡터를 $h(t)$, 출력층의 외부출력벡터를 $o(t)$ 라 하면, 표적벡터 s(t)에 대한 예측벡터는 $\hat{s}(t)$ 는 $o(t)$ 와 같다. $n_j(t), h_j(t), o_k(t)$ 를 각각 $n(t), h(t), o(t)$ 의 j, k번째 성분이라 하면, 전체 시스템은 다음의 식들로 기술된다.

$$f(y_j(t)) = \begin{cases} \frac{1}{1 + \exp(-y_j(t))} & j = 1, \dots, M \\ y_j(t) & j = M + 1, \dots, M + N \end{cases} \quad \text{식 (2)}$$

$$n_i(t) = \sum_{i=1}^{H+N\tau} v_{ij} x_i(t) \quad \text{식 (3)}$$

$$(i = 1, \dots, H+N\tau)$$

$$h_j(t) = f(n_j(t)) \quad \text{식 (4)}$$

$$(j = 1, \dots, H)$$

$$o_k(t) = \sum_{j=1}^M w_{jk} h_j(t) \quad \text{식 (5)}$$

$$(k = 1, \dots, N)$$

v_{ij} : i번째 입력층 뉴런에서 j번째 은닉층 뉴런
 w_{jk} : j번째 은닉층 뉴런에서 k번째 출력층 뉴런

식 (2)은 시그노이드 함수를 나타낸 식이며, 식 (3)은 위에서 언급된 외부 입력벡터 $x_i(t)$ 와 v_{ij} 의 가중치 계산을 나타낸 식이다. $x_i(t)$ 는 입력층 수(H)와 t-1에서의 은닉출력벡터 $h(t-1)$ 를 연결한 $H+N\tau$ 차원 벡터이다.

식 (4), (5)은 각각 은닉층과 출력층의 계산에 관한 식이며, 식 (4)의 f는 식 (2)을 사용하였다.

예측오차벡터 $e(t) = s(t) - s^{\wedge}(t)$ 로부터 정의되는 시간 t에서의 전체오차 J(t)는 다음과 같다.

$$J(t) = \frac{1}{2} \sum_{k=1}^N [e_k(t)]^2 \quad \text{식 (6)}$$

이로부터 v_{ij} , w_{jk} 에 대한 $\nabla J(t)$ 를 계산하여 각 가중치 집합에 대한 학습식을 유도할 수 있다. 학습계수를 α 라 하면 v_{ij} 에 대한 학습식은 다음과 같다[3].

$$\Delta v_{ij}(t) = -\alpha \frac{\partial J(t)}{\partial v_{ij}} \quad \text{식 (7)}$$

w_{jk} 에 대한 학습식은 $-\alpha (\partial J(t) / \partial w_{jk})$ 를 계산하여 유도할 수 있으며, 식 (8)과 같다. [3]

$$\Delta w_{jk}(t) = \alpha e_k(t) o_k(t) h_j(t) \quad \text{식 (8)}$$

2.2.2 MRNPM의 학습 및 인식 구조

앞 절의 유도된 수식을 이용하여 MRNPM의 학습 및 인식 구조는 그림 4, 5와 같다.

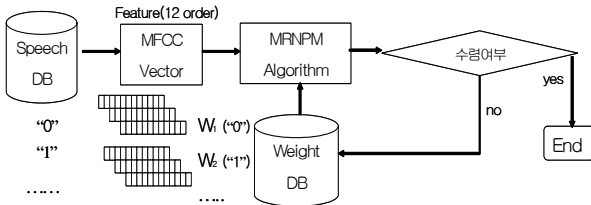


그림 4. MRNPM의 학습 구조

그림 4는 학습 단계로서 HMM에서 사용되었던 음성 DB를 그대로 사용하였으며, 단어 당 75개씩 각각 학습을 시켰다. 학습된 결과는 단어별로 웨이트 정보가 생성됨으로, 최종 13개의 웨이트 정보가 출력된다. 이러한 단어별 웨이트 정보를 이용하여, 인식 과정을 나타낸 것이 그림5이다. 새로운 음성이 입력되면, 각 신경망의 Forward Algorithm을 수행한 후 Finally Logic에서 각 단어별 출력층의 에러 값이 가장 낮은 것을 최종 인식 단어로 출력하게 된다.

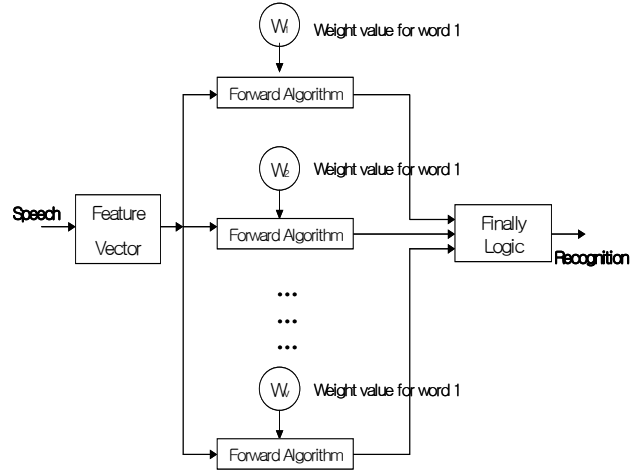


그림 5. MRNPM의 인식 구조

3. 음성 인식 시스템 구현

본 논문에서는 음성수집 및 학습/인식단계를 PC환경에서 구현하여 테스트를 하였으며, 최적화된 인식부 코드만을 가지고, TMS320C6711에 실제 구현하였다.

3.1 음성 데이터베이스 생성 및 전처리 처리

제안된 모델에 대하여 13개의 한국어 숫자음에 대한 인식 실험을 위해 총 50명(남자25명, 여자 25명)의 목소리를 수집하였다. 각 음성자들은 단어별로 3번씩 발성하여, 총 1950개의 단어를 수집하였으며, 이 중 50%은 학습데이터로 사용하였으며, 나머지 50%는 인식 테스트에 사용하였다. 본 논문의 음성 저장 방식은 PC 사운드 파일에서 보편적으로 쓰는 11.025kHz의 샘플링 주파수(Sampling Frequency)와 16비트의 분해능(Resolution)으로 수집하였다.

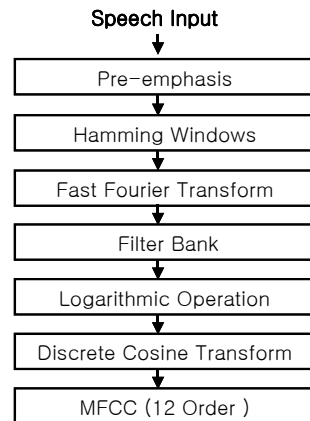


그림6. MFCC 전체 흐름도

수집된 음성 패턴을 MFCC(Mel-Cepstrum)[4]으로 처리하면 각 프레임당 12개의 특징 값들이 추출된다.

본 논문에서 설정된 프레임의 크기는 330, 프레임의 겹침(overlap)은 220으로 하였다.

그림 6은 MFCC의 처리 절차를 나타낸 것이다.

3.2 벡터 양자화

각 단어의 특징 추출된 값(MFCC)들을 이진 트리 알고리즘[5]을 이용하여 코드북(Codebook)을 생성시켰다.

코드북의 크기는 512, 에러비율은 0.00001, e (splitting parameter)값은 0.001로 각각 설정하였다. 이렇게 생성된 양자화 테이블과 특징 벡터의 MFCC값을 넣게 되면, 거리가 가장 가까운 양자화 테이블의 심볼값을 얻을 수 있게 된다[5]. 그림 7는 벡터양자화의 전체 처리 절차를 나타낸 것이다[6].

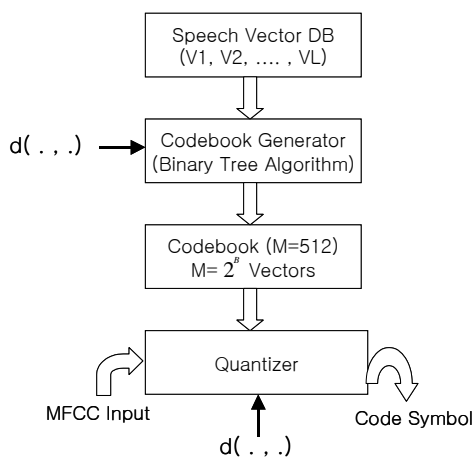


그림 7. 벡터양자화의 전체 흐름도

3.3 제안된 복합형 인식 알고리즘

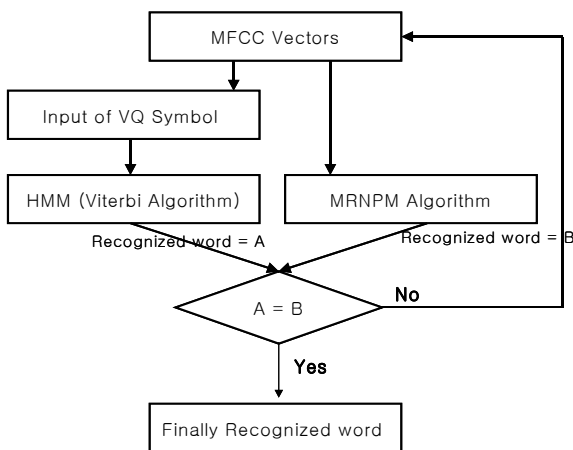


그림 8. 전체 인식기 처리부

본 절에서는 앞 장에서 다루었던 알고리즘들을 이용하여 복합형 인식 알고리즘을 설계에 관한 내용을 서술한다. 그림8에서 나타난 것처럼 각 인식부를 병렬로 처리하여 인식된 결과가 같으면 그 결과를 최종 인식 단어로 결정하고, 만약 두개의 인식부에서 처리된 단어가 상이할 경우,

인식 거부를 실행하여, 초기로 되돌아 음성 재입력을 시도하도록 설정하였다. 아울러 HMM의 입력은 벡터 양자화된 심볼 값으로 입력을 받도록 하였으며, MRNPM의 입력은 특징 벡터(MFCC)를 그대로 사용하였다. 이는 벡터 양자화 되면서 특징 요소들이 왜곡된다는 점을 착안하여 MRNPM에서는 벡터 양자화 처리를 생략하였다. 그리고, HMM은 벡터 양자화 된 벡터들을 입력을 넣은 이유는 HMM의 계산량을 줄이기 위해서 이다. 이렇게 설계된 내용은 3.4절에서는 각각 PC 및 DSP 환경에 실제 구현한 내용들을 서술한다.

3.4 실시간 음성 인식기 구현

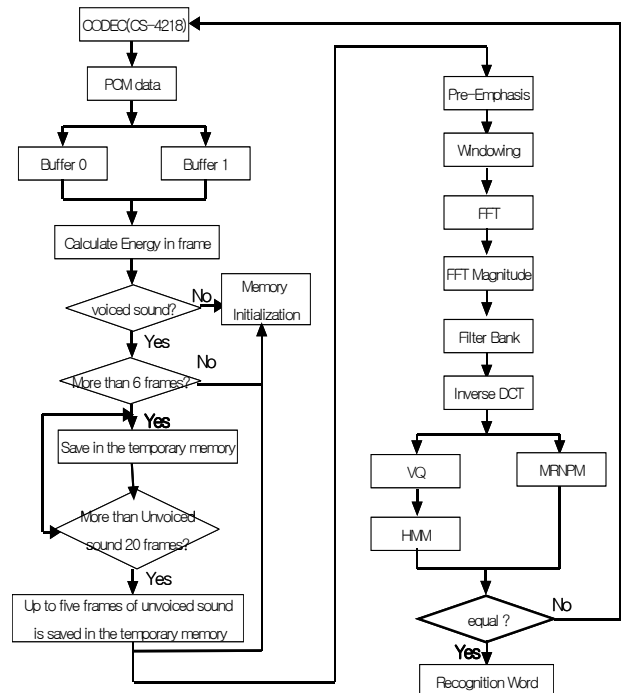


그림 9. 실시간 음성 인식 순서도

그림 9는 앞에서 제시한 내용들을 근거로 PC 및 DSP 상에서 구현된 실제 프로그램의 순서도이다. 전반부에는 유성음/무성음을 판별하는 부분으로 감지된 유성음 부분만을 전처리를 수행하게 된다.



그림 10. C7611 DSP에서의 음성인식 테스트 사진

이후 앞 절에서 학습된 데이터는 각각 λ 및 웨이트 정보들을 가지고, 인식부만을 설계하여 PC 및 TMS320C6711에서 인식시스템을 각각 구현하였다. TMS320C6711하드웨어 플랫폼은 TI사의 C6711 DSK[7]를 사용하였으며, 본 논문을 구현하기 위해 플래쉬 메모리를 2Mbyte로 확장 시켰다. 그 외 TIS TLC320AD535 16-bit Data Converter 및 JTAG용 CCS(Code Composer Studio)를 이용하여 구현하였다. 그림 10에서는 실제 DSP보드를 나타낸 사진이다.

4. 음성 인식 실험 및 논의

앞장에서 제시된 내용을 근거로 하여 다양한 음성인식 테스트를 시도하였다. 표 1은 HMM을 단독으로 인식 실험을 했을 때, 결과를 나타낸 것이다. 각 단어별로 75번씩 훈련 데이터 및 테스트 데이터를 오프라인 상태(PC환경)에서 인식을 테스트한 결과이다. 실험 결과, 훈련 데이터를 그대로 인식 테스트한 경우는 거의 99%의 높은 인식률이 나왔지만, 새로운 테스트 데이터를 실험한 경우는 평균 87%으로 나왔다.

표 1. HMM 인식 테스트 (단위 : 백분율)

Word	Training data	Test data
0 (영)	100%	94%
1 (일)	98%	86%
2 (이)	98%	89%
3 (삼)	98%	81%
4 (사)	98%	73%
5 (오)	100%	92%
6 (육)	100%	100%
7 (칠)	100%	84%
8 (팔)	100%	96%
9 (구)	100%	92%
10(십)	93%	58%
100(백)	100%	96%
1000(천)	100%	88%

표 2는 MRNPM을 단독으로 인식 실험을 했을 경우에 대한 실험 결과이며, 표1의 실험 방법과 동일하게 시도하였다. 학습 계수(α)는 실험 결과 0.00025가 최상의 인식률이 나왔다. 실험 결과, 훈련 데이터를 그대로 인식 테스트한 경우는 HMM보다는 다소 낮은 97%의 인식률이 나왔고, 새로운 테스트 데이터를 실험한 경우는 평균 89%으로 나와 HMM보다는 조금 높은 것으로 나왔다. 표 3는 본 논문에서 제안된 하이브리드 형태의 인식 알고리즘을 통해 실험한 결과이다. 실험 결과 표1과 표2와 비교하면 학습 데이터는 비슷한 인식률이 나왔지만, 테스트 데이터를 통한 실험에서는 평균 인식률이 92%으로 나타났으며, 표1과 비교해서 5%의 인식률 향상이 되었다. 표 3에 대한 실험 결과에서 인식거부가 발생한 내용은 제외시켰다.

표 2. MRNPM 인식 테스트

Word	Training data	Test data
0 (영)	96%	84%
1 (일)	92%	82%
2 (이)	100%	92%
3 (삼)	100%	94%
4 (사)	100%	94%
5 (오)	94%	82%
6 (육)	100%	96%
7 (칠)	92%	82%
8 (팔)	100%	96%
9 (구)	94%	86%
10(십)	100%	92%
100(백)	98%	90%
1000(천)	100%	94%

표 3. HMM / MRNPM 인식 테스트

Word	Training data	Test data
0 (영)	100%	96%
1 (일)	100%	92%
2 (이)	100%	94%
3 (삼)	100%	94%
4 (사)	100%	88%
5 (오)	100%	96%
6 (육)	100%	96%
7 (칠)	98%	90%
8 (팔)	100%	96%
9 (구)	100%	96%
10(십)	98%	86%
100(백)	100%	90%
1000(천)	100%	94%

표 4. 임베디드 시스템에 인식 테스트

	잡음이 없는 환경	잡음이 있는 환경
인식률	89%	74%
인식거부	25번	47번

임베디드 시스템에서 테스트한 실험 결과는 표4와 같으며, 총 100번의 발성에 대한 인식 결과를 나타낸 것이다. 이때 잡음이 없는 환경에서는 표3와 비슷한 인식률이 나왔으며, 인식 거부 횟수는 25번이었다. 이에 반해, 잡음 환경(라디오를 약하게 방송되는 환경)에서 테스트한 결과 약 15%의 인식률 저하가 나타났으며, 인식 거부도 많이 발생하였다. 마지막으로 표 5에서는 각 음성 인식 프로그램의 부분별 인식 속도를 나타낸 것이다. PC환경은 펜티엄 4 2.0GHz에서 테스트하였으며, 총 975번을 테스트하여 평균값이며, DSP환경에서는 2가지로 속도 테스트 하였다. 결과에서와 같이 본 논문에서 제안된 Hybrid 형태는 HMM 단독으로 인식 테스트한 결과와 비교할 때, 인식 시간이 크게 차이가 나지 않았다.

표 5. 인식 속도 측정 테스트

	PC환경	DSP(HMM)	DSP(Hybrid)
MFCC	0.1158초	0.65초	0.73초
VQ	0.016초		
HMM	0.015초		
MRNPM	0.028초		

5. 결 론

본 논문에서는 다양한 인식 알고리즘 중 가장 성능이 좋은 HMM과 신경망의 일종인 MRNPM을 병렬로 처리하게 하여 인식률을 향상시켰다. 오프라인 상태에서 실험한 결과 약 5%의 인식률 향상을 보여 한 개의 인식 알고리즘을 사용하는 것보다 더 효능이 좋다는 것을 증명하였다. 아울러 제안된 MRNPM는 기존의 예측신경회로망과 변별기로 학습되어지는 회귀신경망의 단점들을 효과적으로 극복하며 여러 장점을 보유하고 있다. 즉, 이 모델은 예측신경 회로망에 비해 시간축 정렬 알고리즘이 불필요하여 인식시간이 단축되고, 작은 크기의 기억장소를 요구하기 때문에 임베디드 시스템에 적용하는데 매우 적합하다는 것이다. 실제 DSP환경에서 테스트한 결과, 병렬 구조의 인식 시스템이지만, 인식하는데 걸리는 시간이 기존 HMM만을 사용했을 때와 큰 차이가 나지 않았다.

하지만 여러 가지 잡음 환경 내에서 인식률을 테스트 한 결과 인식률은 현저히 떨어지며, 최상의 인식률 올리기 위해서는 잡음 처리가 필요하다.

이에 여러 잡음 처리 기술이 있지만, 이 중 ICA (Independent Component Analysis)을 이용하면 각 요소별로 분석하기 때문에 음성과 잡음 영역을 분류할 수 있다. 분류된 음성만을 처리하게 됨으로써 획기적인 전처리 단계로 주목을 받고 있으며, 인식률 크게 향상 될 것이라 생각된다[8].

참 고 문 헌

[1] LAWRENCE R. RABINER, "A Tutorial on Hidden Markov Models and Selected Application in Speech Recognition", IEEE, FEBRUARY, 1989.
 [2] Je Kwan Ryu, Kyung Min NA, "Speech Recognition Using Recurrent Neural Prediction Models" IEEK, November, 1995.
 [3] S. J. Lee, K.C. Kim, H. S. Yoon and J. W. Cho, "Application of fully recurrent neural networks for speech recognition", Proc. ICASSP'91, pp. 270-280, 1989
 [4] L.Rabiner, B.H.Juang, "Fundamentals of Speech Recognition", Prentice-Hall Inc. 1993.
 [5] L. Rabiner, "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.

[6] Ryu Kang Joo, "A Study on the Data Fusion of Korean digit recognition Using DHMM", Korea Maritime University, 1998.
 [7] www.ti.com, "C6711 DSP Starter Kit (DSK)"
 [8] Aapo Hyvarinen, Juha Karhunen, Erkki Oja, "Independent Component Analysis", Wiley Interscience, 2001.

저 자 소 개



김정훈(Jung hoon Kim)
 2001년 : 동명정보대학교
 정보통신공학과 (학사).
 2003년 : 한국해양대학교
 전자통신공학과 (석사).
 2003년~현재 : 한국해양대학교
 전자통신공학과 박사과정.

2003년~ 현재 : 동명정보대학교
 메카트로닉스공학과 겸임교수
 관심분야 : 음성인식, DSP, 인공지능, 임베디드.



장원일(Won il Jang)
 2002년 : 동명정보대학교
 정보통신공학과(학사).
 2003년~현재 한국해양대학교
 전자통신공학과 석사과정.
 관심분야 : 패턴인식, 임베디드



김영탁(Young tak Kim)
 2001년 : 진주산업대학교
 전자공학과 (학사).
 2003년 : 한국해양대학교
 전자통신공학과 (석사).
 2003년 ~ 현재 : 한국해양대학교
 전자통신공학과 박사과정.
 2003년 ~ 현재 : 동명정보대학교
 메카트로닉스공학과 겸임교수
 관심분야 : 패턴인식, 임베디드.



이상배(Sang bae Lee)
 1989년 : 고려대학교(공학박사).
 현재 한국해양대학교
 전자통신과 정교수.