

# Incremental Learning을 이용한 화자 인식

## The Speaker Identification Using Incremental Learning

심귀보\* · 허광승\* · 박창현\* · 이동욱\*\*

Kwee-Bo Sim, Kwang-Seung Heo, Chang-Hyun Park, and Dong-Wook Lee

\* 중앙대학교 전자전기공학부

\*\* 중앙대학교 정보통신연구원

### 요 약

음성 속에는 화자의 특징이 포함되어 있다. 본 논문에서는 신경망에 기초한 Incremental Learning을 이용하여 화자 수에 제한 받지 않는 화자 인식 시스템을 제안한다. 컴퓨터를 통하여 녹음된 음성 신호는 End Detection과정을 통하여 유성음과 무성음을 분류하고 LPC를 이용해 12차수의 Cepstral Coefficients를 추출한다. 이 계수는 화자 식별을 위한 학습 입력 값으로 사용된다. Incremental Learning은 이미 학습한 Weight들을 기억하고 새로운 data에 대해서만 학습을 하는 학습 방법으로 Neural Network 구조가 화자 수에 따라 늘어나므로 화자 수에 제한을 받지 않고 학습이 가능하다.

### Abstract

Speech signal has the features of speakers. In this paper, we propose the speaker identification system which use the incremental learning based on neural network. Recorded speech signal through the Mic is passed the end detection and is divided voiced signal and unvoiced signal. The extracted 12 order cpestrum are used the input data for neural network. Incremental learning is the learning algorithm that the learned weights are remembered and only the new weights, that is created as adding new speaker, are trained. The architecture of neural network is extended with the number of speakers. So, this system can learn without the restricted number of speakers.

**Key Words** : Speaker identification, LPC, Ceptrum coefficients, Neural Network, Incremental learning

### 1. 서 론

음성을 이용하여 화자를 식별하는 분야는 크게 화자 인식과 화자 검증으로 나누어진다. 화자 검증은 검증된 화자와 검증되지 않은 화자를 구분하는 것이 목적이고 화자 인식은 화자를 구분하는 것이 기본 목적이다. 화자 인식은 다시 문장 독립과 문장 종속으로 나누어진다. 문장 독립은 특정한 문장이 아닌 정해지지 않은 문장을 이용하여 화자를 인식하고 문장 종속은 미리 정하여진 문장 안에서만 화자를 인식하는 것을 말한다.

화자 식별을 수행하는 시스템은 두 부분으로 나누어진다. 하나는 화자의 음성에서 특징점을 추출하는 부분이고 나머지는 화자를 구별하기 위한 학습부분이다.

음성에서 특징점을 추출하는 방법은 두 부분으로 나눌 수 있다. Time domain과 Frequency domain의 두 영역이다. 첫 번째 Time domain은 시간축을 기본으로 해서 음성을 분석하고 특징점을 추출하는 방법으로써 parameter로는 Linear Prediction Coefficients, Reflections Coefficients, Log area

ratio Coefficients, Cepstrum Coefficients 등이 쓰이고 있다. Frequency domain은 주파수를 기본으로 해서 분석하는 방법으로 parameter로는 Inverse Filter Spectral Coefficients, Speech Spectrum Parameters 등이 쓰이고 있다. 하나의 Parameter를 쓰는 것보다 두 가지 이상의 Parameter를 쓰는 것이 더욱 정확한 음성 특징점을 추출할 수 있다[1].

화자를 식별하는 학습방법은 크게 3가지로 분류할 수 있다. 첫 째는 Euclidean distance를 사용하는 방법으로 reference data와 test data 사이의 Euclidean distance를 구하여 가장 가까운 거리에 있는 data를 구별하는 학습법을 사용하여 화자를 식별한다[2]. 그러나 이 식별 방법은 test와 reference data를 모두 기억하고 있어야 하므로 memory와 computation면에서 많은 양을 요구한다. 두 번째는 VQ(Vector Quantization)을 이용하는 방법이다. 화자로부터 추출된 음성 특징점들을 가지고 codebook을 만들고 codebook의 centroid를 구해서 가장 가까운 codebook에 distance값을 줌으로써 가장 적은 거리를 가지고 있는 화자 특징점을 찾는 방법이다[3]. 세 번째는 Neural Network을 가지고 화자 식별을 수행하는 방법이다. Neural Network은 적은 parameter로 학습을 하며 간단한 연산과 각 화자에 따른 인식기를 구성하지 않으므로 memory 측면에서도 상당히 효율적이다. 그러나, 화자가 바뀔 때마다 새롭게 학습해야 하는 불편한 점도 가지고 있다. 화자 식별에 쓰이는 Neural Network을 살펴보면 MLP(Multi Layer Perceptron), TDNN(Time Delay Neural Network), RBF(Radial Basis Function), LVQ(Learning Vector Quantization) 등이 있다[4].

접수일자 : 2002년 12월 7일

완료일자 : 2003년 9월 1일

감사의 글 : 본 연구는 산업자원부 차세대 신기술 개발 사업(과제 번호: N09-A08-4301-09)에 의해 지원 받았습니다.

위의 Neural Network외에도 Decision Tree Neural Network을 이용하여 화자를 구별 하기도 한다[4].

Neural Network이 가지고 있는 입력 Data가 새로 들어 올 때 마다 새롭게 학습해야 하는 단점을 해결하기 위해서 제시되고 있는 학습 방법이 Neural Network구조에 Incremental Learning 학습 방법을 적용한 모델이다[5].

Incremental Learning은 기존의 학습된 Neural Network에 새로운 입력이 들어오면 입력에 대한 Node가 추가되는 학습으로써 새로운 입력이 들어 올 때 마다 학습을 다시 하는 기존의 Neural Network와는 다르게 새로운 입력에 대해서만 학습하는 방식이다[6]. 이 학습방법은 많은 화자를 효율적으로 구분할 수 있다.

본 논문에서는 화자 식별을 하기 위해서 크게 두 단계로 나누어진 모델을 제시한다. 첫 번째는 음성 특징점 추출 단계로써 컴퓨터를 통하여 녹음된 speech signal을 이용하여 전체 speech signal을 Frame별로 나누어 분석하고 LPC과정을 거쳐 얻은 Cepstrum Coefficients를 학습 Data로 사용한다. 두 번째 단계는 학습 단계로써 Backpropagation을 이용한 초기 학습과 새로운 화자에 대한 Incremental learning을 수행하는 학습으로 진행된다. 본론에서는 음성 특징점 추출 과 학습 과정을 소개하고 실험을 통해서 Incremental Learning을 이용한 화자 인식 모델을 제시한다.

## 2. 화자 인식 모델

화자 인식을 하기 위해 화자의 음성에서 각 특징 점들을 이끌어 내는 것이 중요하다.

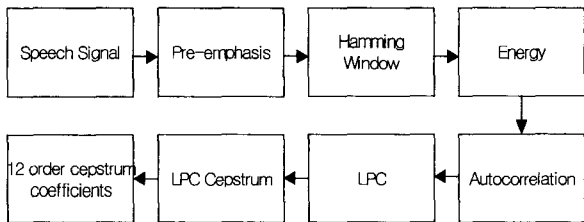


그림 1. 음성 신호 처리  
Fig. 1. Speech Signal Processing

그림 1은 음성 신호 처리에 대한 전체 적인 과정을 보여 주고 있다. 아날로그 음성 신호는 A/D converter에 의해서 Digital 신호로 변환된다. 이렇게 변환된 신호를 처리하는 방법은 크게 두 가지가 있는데, 하나의 디지털 샘플만을 대상으로 처리하는 방법과 일정한 양의 샘플을 저장한 후 처리하는 프레임 처리하는 방법이 있다[7]. 프레임 처리 방식에서는 음성 전처리 과정으로써 Window의 역할이 중요하다.

### 2.1 Hamming Window

Window는 긴 신호가 있을 때, 신호의 일부분만을 볼 수 있게 해 주는 역할을 한다. Hamming Window는 관심 있는 영역에 일정한 pattern의 대칭적인 Weighting을 취한다[1].

$$W(n) = \begin{cases} 0.54 - 0.46\cos(2\pi n/M) & 0 \leq n \leq M \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Hamming Window는 식(1)과 같이 표현된다. 원 신호에

대해서 다른 Window 보다 좋은 smoothing과 frequency leakage가 없는 신호를 얻을 수 있는 장점이 있다.

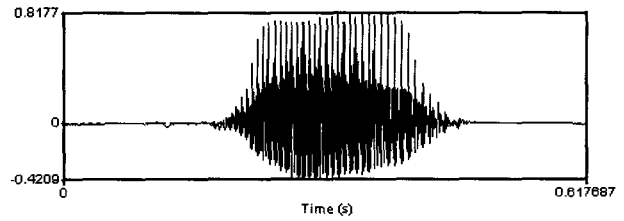


그림 2. Speech Signal  
Fig. 2. Speech Signal

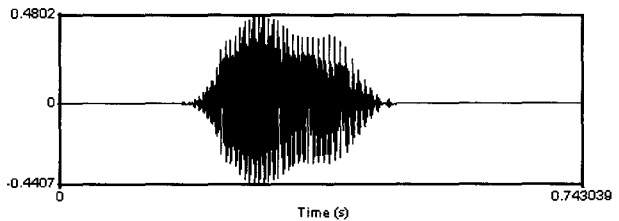


그림 3. Hamming Window된 Speech Signal  
Fig. 3. Hamming Windowed Speech Signal

그림 2와 3은 “아”에 대해서 Speech Singal과 Hamming Window처리된 speech signal을 보여 주고 있다. 위 두 그림을 통해 Hamming Window가 원래의 speech signal을 좀 더 smoothing하게 하는 특징을 볼 수 있다.

### 2.2 Energy

음성 신호의 short-time energy를 이용하여 음성 신호의 진폭의 변화를 나타낼 수 있다. 무성음 구간에서의 진폭의 변화는 작고 유성음 구간에서는 그 반대이다[8]. 이런 특성을 가지고 있는 energy를 이용하여 유성음과 무성음을 분류할 수 있다.

$$E(m) = \sum_{n=m-N+1}^m [S(n)W(m-n)]^2 \quad (2)$$

식(2)에서  $E(m)$ 은 energy값을 나타내고  $S(n)$ 은 Frame 내의 각 샘플값을 의미한다. short-time energy는 무성음에서는 작게 나타나고 유성음 구간에서는 크게 나타난다.

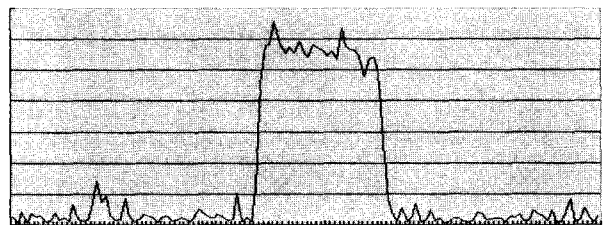


그림 4. Energy  
Fig. 4. Energy

그림 4는 “아”에 대한 Energy를 나타내고 있다. 유성음과 무성음이 분류되는 것을 알 수 있다. Magnitude가 큰 부분이 유성음 부분이고 작은 부분이 무성음과 묵음 부분이다.

Energy만으로는 무성음과 목음을 분류할 수 없다. 이것을 해결하기 위해서는 Zerocrossing rate를 추가 하면 무성음과 목음을 분류 할 수 있다.

### 2.3 Autocorrelation

Correlation은 두 신호에 대한 유사성을 측정하는 매우 정교한 측정방법 이다. 또한, Correlation Function은 Fourier Transform에 의한 Power spectrum과 연관 되어 있고 음성 신호에 포함된 주파수의 특징을 예측하기 위한 가장 좋은 방법 중 하나이다. Correlation은 간단한 벡터 내적으로 표현할 수 있다[9].

$$R[d] = \sum_{i=0}^{N-1} x[i]y[i+d] \quad (3)$$

식(3)에서  $i$ 는 음성신호 데이터의 순서이고  $d$ 는  $x$ 와  $y$ 사이의 이동된 샘플 수이다. Correlation Length  $N$ 은 Correlation 결과로써 얼마나 많은 데이터를 사용할 지를 결정한다.  $R[d]$ 는 Correlation Function으로써 두 신호의 차이 값을 가지고 연속적으로 계산한다.

신호  $x$ 와  $y$ 가 같은 신호일 때,  $R[d]$ 를 Autocorrelation Function이라고 한다[10].

### 2.4 LPC(Liner Predictive Coding)

LPC의 기본 모델은 주어진 시점에서  $n$ 에서의 음성신호는 지나간  $p$ 개의 음성신호의 linear combination으로 근사할 수 있고 인간의 성도 모델은 LPC모델을 통해서 잘 표현되고 LPC Coefficients를 통해서 성도모델을 나타낸다[7].

LPC Coefficients는 Durbin's recursive solution에 의해서 구해진다[1].

$$\sum_{k=1}^p \alpha_k R_n(|i-k|) = R_n(i) \quad (4)$$

Durbin's recursive solution은 다음과 같은 식을 이용하여 구한다[9].

#### Step 1. Initialize

$$E^0 = R[0]$$

#### Step 2. Iteration

$$k_i = (R[i] - \sum_{j=1}^{i-1} \alpha_j^{i-1} R[j-1]) / E^{i-1} \quad (5)$$

$$\alpha_i^i = k_i \quad (6)$$

$$\alpha_j^i = \alpha_j^{i-1} - k_i \alpha_j^{i-1} \quad (7)$$

$$E^i = (1 - k_i^2) E^{i-1} \quad (8)$$

#### Step 3. Final Solution

$$\alpha_j = \alpha_j^p \quad 1 \leq j \leq p \quad (9)$$

Coefficient  $k_i$ 는 반사 계수이고 시스템의 안정을 위해서 -1 과 1 사이에 있어야 한다. 또한 Coefficients  $E^i$ 는 에러 율

을 나타낸다. 예측 계수 값은 차수( $p$ )보다 작은 모든 차수에 서 구한다[11].

### 2.5 LPC Cepstral Coefficients

Cepstrum  $c(n)$ 은  $C(z)$ 의 inverse  $z$ -transform으로 정의 된다[12].

$$C(z) = \sum_n c(n)z^{-n} \quad (10)$$

All poles  $z=z_i$ 가 unit circle안에 있고 gain 값을 1로 주면 LPC Cepstral Coefficients( $c_{lp}(n)$ )는 아래와 같이 정의 된다[12].

$$c_{lp}(n) = \begin{cases} \frac{1}{n} \sum_{i=1}^p z_i^n & n > 0 \\ 0 & n < 0 \end{cases} \quad (11)$$

LPC Cepstral Coefficients는 recursive에 의해 예측 계수 값으로부터 구한다[13].

$$c_1 = \alpha_1 c_n = \sum_{k=1}^{n-1} (1 - \frac{k}{n}) \alpha_k c_{n-k} + \alpha_n, \quad 1 < n < p \quad (12)$$

식 (12)에서  $c$ 는 Cepstrum Coefficients값을 나타내고  $\alpha$  값은 예측 계수 값을 나타낸다. 또한  $p$ 는 차수 값을 의미하며 차수의 설정은 너무 작은 값은 화자의 특징을 추출하기가 상당히 어렵고 너무 높은 차수의 값은 speech signal의 noise가 섞여 있는 부분까지도 나타내므로 특징점으로 추출하기에는 적당하지 않다. 본 논문에서 제안하는 모델에서는 12차수를 사용한다.

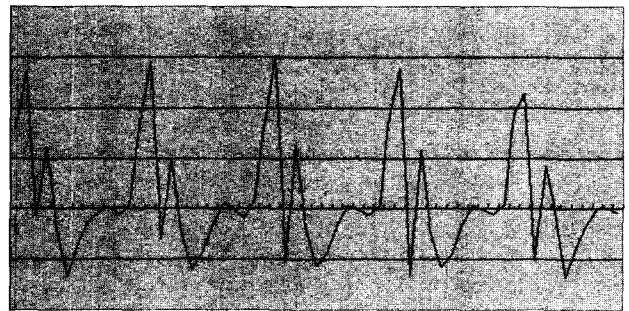


그림 5. LPC Cepstrum  
Fig. 5. LPC Cepstrum

그림 5는 "아"에 대한 Cepstrum Coefficients를 보여 주고 있다. 그림에서 가로축은 Frame을 의미하여 세로축은 Cepstrum Coefficients의 Magnitude를 나타낸다. Cepstrum은 화자만이 가지고 있는 성도를 모델링 하는데 많이 사용 되고 있다. 이런 특징은 또한 화자를 구별하기 위한 음성 특징점으로 이용 되고 논문에서 제시하는 화자 식별 모델에서도 12차수의 LPC Cepstrum Coefficient를 학습 Data로 사용한다.

### 2.6 Incremental Learning Neural Network

Neural Network은 화자들의 음성 특징점들을 가지고 화자를 판별하는 역할을 한다. 여기에서는 Backpropagation과 Incremental Learning 학습 알고리즘을 사용하는 MLP(Multi Layer Perceptron) Neural Network을 제시 한다. 이 Neural

Network의 구조는 12차수의 Cepstrum Data를 받기 위한 입력층과 화자 수에 따라 결정되는 중간층과 출력층으로 되어 있다.

### 2.6.1 Backpropagation

초기 학습을 위해서 Backpropagation 학습 알고리즘을 사용한다. Backpropagation은 gradient descent을 사용하여 Weight 공간에서의 최소 error값을 찾는 학습 알고리즘이다 [14]. 각 Node는 sigmoid function으로 되어 있다. 초기 MLP 구조는 4명의 화자를 구분하기 위해서 12개의 Input Node와 8개의 Hidden Node 그리고 4개의 Output Node로 구성 되어 있다. 초기 Weight들은 random하게 설정 되고 Weight는 기준 Error값 보다 작을 때까지 즉 학습이 완료 될 때 까지 계속 Update가 된다.

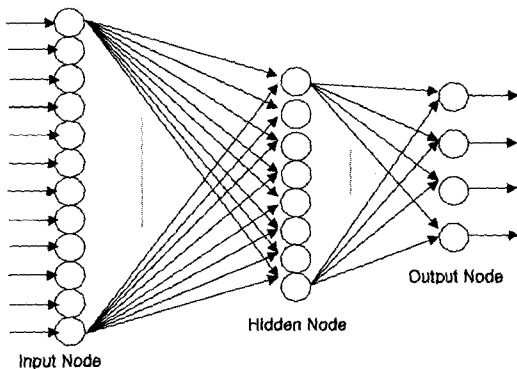


그림 6. MLP  
Fig. 6. MLP

그림 6은 Backpropagation 학습 알고리즘을 적용한 MLP의 구조를 보여 주고 있다. 초기 학습은 8개의 중간층과 4개의 출력층을 가지고 진행 된다. 4개의 Output Node는 식별되는 화자수를 의미한다. 입력 Data는 4명의 화자에 대해서 Frame별로 입력 되고 중간층과 출력층을 통해서 나온 값과 목표값을 이용하여 Error 값을 구하게 된다. 목표값은 4개의 Output Node에서 출력값이 "1"인 경우 활성화 되므로 첫 번째 화자에 대해서는 {1,0,0,0}으로 설정이 되고 두 번째는 {0,1,0,0}으로 설정된다. 세 번째와 네 번째 화자의 경우도 동일하다.

### 2.6.2 Incremental Learning

Incremental Learning은 기존의 학습된 Neural Network에 새로운 입력이 들어오면 입력에 대한 Node가 추가되는 학습으로써 새로운 입력이 들어 올 때 마다 학습을 다시 하는 기존의 Neural Network와는 다르게 새로운 입력에 대해서만 학습하는 방식이다[12].

Backpropagation 학습을 통해 완성된 초기 MLP는 4명의 화자를 구분한다. 새로운 화자의 음성 특징 Data가 들어오면 학습된 MLP로는 새로운 화자를 구분하지 못한다. 즉, Output Node에서 활성화 된 값이 나오지 않는다. 기존의 Neural Network은 학습을 전체적으로 다시 하게 된다. 들어오는 화자수에 대해서 더욱 많은 학습 시간이 걸린다. 효과적인 학습을 위해서 새로운 화자에 대한 Data가 들어오면 중간층 노드는 2개, 출력층 노드는 1개가 기본 학습된 MLP위에 추가 된다. Incremental Learning은 이런 학습 과정을 통해서

진행된다.

그림 7은 Incremental Learning을 위한 MLP NN의 구조를 보여 주고 있다. 새로운 화자에 대한 음성 특징 Data가 들어 올 때 마다 중간층 노드와 출력층 노드는 증가되고 증가된 중간층 노드와 출력층 노드는 서로 연결 되어 있다. 증가된 출력층이 활성화 되기 위해서는 나머지 출력층들이 활성화 되지 않아야 한다. 목표값은 추가된 화자에 대해서 {\*, \*, \*, \*, 1}으로 설정된다. '\*'은 don't care를 의미한다. '\*'부분은 학습을 하지 않고 연결된 weight들은 '0'이 된다. 학습이 끝나면 MLP를 통해서 학습된 weight들을 메모리에서 가지고 오게 됨으로써 추가된 node만이 학습한다.

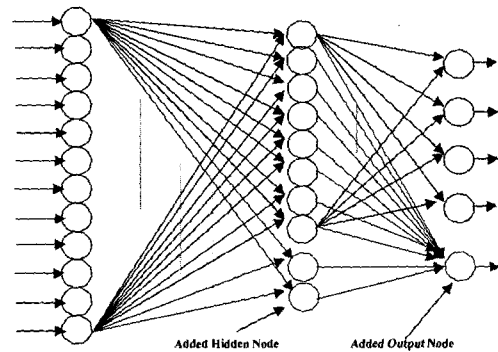


그림 7. Incremental Learning NN구조  
Fig. 7. Incremental Learning NN Structure

## 3. 실험 결과

### 3.1 Speech Signal Processing

화자 인식을 실험하기 위해서 실험 환경을 설정했다. 가족 구성원 즉 4명의 화자를 기본 구성원으로 하고 새로운 화자를 방문객으로 하여 진행 하였다.

실험 음성database는 6명의 화자에 대해서 7개의 문장을 평상시의 습관대로 말하게 한 후 녹음하였다. 각 음성은 11.025Khz, 16bit, stereo 방식으로 녹음하였다. 각 녹음된 음성들은 말하는 길이가 다르다. 문장의 길이가 다르고, 말하는 사람들의 평소 말하는 습관으로 인해서 같은 말이라도 음성 길이가 달라진다. 이런 문제를 해결하기 위해서 녹음된 음성들의 각 frame을 조사해서 가장 작은 frame(20)을 기준으로 해서 특징점을 추출하였다. 여기서 말하는 특징은 LPC Cepstrum Coefficients(LPCC)를 말한다. 20Frame에서 각각의 frame에서 12차수의 LPCC를 추출하므로 전체 특징점의 수는 문장마다 240개의 특징점들을 추출한다.

### 3.2 화자 인식

Incremental Learning을 이용한 화자 인식은 4 가지 단계를 거쳐 실험이 진행 된다. 첫 번째는 기본 4명에 대한 학습 단계로써 MLP구조를 가지고 있다. 학습 data는 4명의 화자가 말한 짧은 문장에서 추출한 LPCC이다. Error값이 10%이하가 되면 학습을 멈추게 된다. 각 weight값들은 저장된다. 두 번째는 기본 학습된 MLP에서 4명의 화자에 대한 화자 인식 단계이다. 녹음된 7개의 문장을 화자 식별 data로 사용한다. 4개의 output node중에서 가장 큰 output값을 출력하는 node가 활성화 된다. Output node들은 각각의 화자를 의미한

다. 다음 단계는 incremental learning을 이용한 학습 단계이다. 학습된 화자 외에 새로운 화자가 들어오면 4명에 대해서 학습한 MLP로는 새로운 화자를 인식하지 못한다. 즉, 4개의 output node들의 출력값이 한계값을 넘지 못하므로 node들은 활성화 되지 않는다. 이 때, incremental learning을 시작하게 된다. 새로운 화자에 대해 hidden node가 2개, output node가 1개 추가 된다. 추가된 hidden node와 output node의 weight들은 완전하게 연결되어 있지만, 추가된 hidden node가 전에 학습된 MLP의 output node와는 weight들이 연결되어 있지 않다. incremental learning의 장점은 전에 학습한 weight들을 기억함으로써 새 data에 대해서도 MLP전체가 학습을 하는 불편함을 제거하게 된다. 마지막 단계는 incremental learning이 추가된 MLP에 대한 화자 인식이다. 기본 4명의 화자 외에 추가되는 화자에 대한 식별이 이루어진다. 또 다른 새로운 화자가 들어오면 incremental learning을 이용하여 hidden node와 output node가 추가 되고 학습하게 된다.

증가 되는 hidden node의 수는 실험을 통해 설정하였다. 2개 일 때 가장 적은 학습 시간으로 학습을 했고 2개 이상이 되어도 2개와 크게 학습 시간이 차이가 나지 않았다. 학습률은 0.1로 하였고 incremental learning에서는 추가된 node만으로 학습을 하므로 학습 종료 조건인 에러율을 12.5%로 높였다.

표 1. 화자 인식률.

Table 1. Speaker Identification rate.

Speaker	Identification rate(%)
Speaker 1	100
Speaker 2	100
Speaker 3	100
Speaker 4	100
New speaker 1	95
New speaker 2	90

표1은 6명의 화자에 대한 화자 인식률을 보여 주고 있다. speaker는 MLP를 이용하여 화자 식별을 한 결과를 보여 주고 있다. 4명에 대해서는 화자 식별이 정확하게 수행되는 것을 확인 할 수 있다. New speaker는 incremental learning을 이용하여 화자 식별을 한 결과를 보여 주고 있는데 인식률이 화자가 증가 될수록 떨어지고 있다. 추가된 node만으로 학습이 이루어지는 incremental learning은 node들이 더 추가 될수록 추가된 node간에 학습이 서로 영향을 주지 않으므로 인식률을 떨어뜨리는 결과가 발생하였다. 전체 화자 인식 시스템의 인식률은 97.5%이다.

#### 4. 결 론

제한된 화자 인식 시스템은 화자 식별에서 좋은 성능을 보여주고 있는 LPCC 음성 특징점을 사용하고 있다. 화자 식별을 위한 학습부에서는 4명의 가족 구성원 화자에 대해서 MLP를 이용하여 학습과 식별을 수행하여 4명의 화자에 대해서 화자 식별이 되는 것을 확인한 후 incremental learning

을 적용하여 새로운 화자를 식별하고 새로운 화자를 식별 할 때 Neural Network전체가 학습하는 단점을 보완 하였다. 시스템은 NN이 미리 정의 되어 있지 않고 화자수가 증가함에 따라 NN이 증가함으로써 화자 수에 제한을 받지 않는 incremental learning의 효율성과 장점을 실험을 통해 증명하였다.

#### 참 고 문 헌

- [1] N. Mohankrishnan, M. Shridhar, M.A. Sid-Ahmed "A Composite Scheme for Text-Independent Speaker Recognition", Acoustic, Speech and Signal Processing, IEEE International Conference on '82, Vol: 7, pp. 1653-1656, 1982
- [2] S. Pruzansky, "Pattern-matching procedure for automatic talker recognition", J. Acoustic. Soc. Amer, Vol: 35, pp. 354-358, Apr, 1971
- [3] F.K. Soong, A.E. Rosenberg, L.R. Rabiner, B.H. Juang, "A vector quantization approach to speaker recognition", in Proc. ICASSP, pp. 387-390, 1985
- [4] Kevin R.Farrell, Richard J.Mammone, Khaled T.Assaleh, "Speaker Recognition Using Neural Networks and Conventional Classifiers", IEEE Transaction on speech and audio processing, Vol: 2, No.1, pp. 194-205, January 1994
- [5] K.Farrell, R.J.Mammone, A.L.Gorin, "Adaptive Language Acquisition Using Incremental Learning", Acoustics, Speech, and Signal Processing, 1993, ICASSP-93, 1993, IEEE International conference on, Vol: 1, pp. 501-504, Apr 1993
- [6] R.Poliker, L.Udpa, S.S.Udpa, V.Honavar, "Learn++: An Incremental Learning algorithm for Multilayer perceptron networks", Acoustic, Speech and Signal Processing, 2000, ICASSP'00, Proceedings, 2000, IEEE International Conference on, Vol: 6, pp. 3414-3417, 2000
- [7] 한진수, 음성신호처리, 오성미디어, pp. 20-23, 2000
- [8] John R.Deller, Jr., John H.L. Hansen, John G. Proakis, Discrete-Time Processing of Speech Signals, IEEE Press, pp. 246-251, 1993
- [9] A.M.Kondoz, Digital Speech-Coding for Low Bit Rate Communications Systems, John Wiley & Sons Ltd, pp. 44-46, 1994
- [10] Paul M.Embree, Bruce Kimble, C language Algorithms for Digital Signal Processing, Prentice-Hall International Editions, pp. 31-32, 1991
- [11] Xuedong Huang, Alejandro, Hsiao-Wuen Hon, Spoken Language Processing A Guide to Theory, Algorithms, and System Development, pp. 294-295, 2000
- [12] Ehab F.M.F.Badan, Hany Selim, "Speaker Recognition Using Artificial Neural Networks Based on Vowel phonemes", Signal Processing Proceedings 2000, WCCC-ICSP 2000 5th International Conference on, vol.2, pp. 796-802, 2000

[13] Xician Yue, Datian Ye, Chongxum Zheng, Xiaoyu Wu, "Neural Networks for Improved Text-Independent Speaker Identification", IEEE Engineering in Medicine and Biology Magazine, vol. 21, issue. 2, pp. 53-58, March-April, 2000

[14] Raul Rojas "Neural Networks A Systematic Introduction" Springer, pp. 149-182, 1996



**허광승(Kwang-Seung Heo)**

2002년 : 중앙대학교 전자전기공학부 학사  
2002년~현재 : 동 대학원 전자전기공학부 석사과정

관심분야 : 화자인식, 신경회로망 등  
E-mail : ball600@wm.cau.ac.kr

**저 자 소개**



**심귀보(Kwee-Bo Sim)**

1984년 : 중앙대학교 전자공학과 공학사  
1986년 : 동대학원 전자공학과 공학석사  
1990년 : The University of Tokyo 전자공학과 공학박사  
2003년~현재 : 한국퍼지 및 지능시스템학회 부회장

2001년~2002 : 대한전기학회 제어및시스템부문의 편집위원 및 학술이사  
2000년~현재 : 제어자동화시스템공학회 이사  
2003년~현재 : 일본계측자동제어학회(SICE) 이사  
1991년~현재 : 중앙대학교 전자전기공학부 교수

관심분야 : 인공생명, 진화연산, 지능로봇시스템, 뉴로-퍼지 및 소프트 컴퓨팅, 자율분산시스템, 로봇비전, 진화하드웨어, 인공면역계 등

Phone : +82-2-820-5319  
Fax : +82-2-817-0553  
E-mail : kbsim@cau.ac.kr



**박창현 (Chang-Hyun Park)**

2001년 : 중앙대학교 전자전기공학부 학사  
2003년 : 동 대학원 전자전기공학부 공학석사  
2003년~현재 : 동 대학원 전자전기공학부 박사과정

관심분야 : 진화연산, 신경회로망 등  
E-mail : 3rr0r@wm.cau.ac.kr



**이동욱(Dong-Wook Lee)**

1996년 : 중앙대학교 제어계측공학과 공학사  
1998년 : 동 대학원 제어계측학과 공학석사  
2000년 : 동 대학원 제어계측학과 공학박사  
2002년~현재 : 중앙대학교 정보통신연구원 연구전담교수

관심분야 : 인공생명, 진화연산, 인공면역계, 인공두뇌 등  
E-mail : dwlee@wm.cau.ac.kr