'93 한국자동제어학술회의논문집 (1993.10.20~22)

연속음성중 키워드(Keywork) 인식을 위한 Binary Clustring Network

°최관선, 한민홍
고려대학교 공과대학 산업공학과

Binary Clustering Network for Recognition of Keywords in Continuous Speech

°Kwan-Seon Choi and Min-Hong Han
Department of Industrial Engineering, Korea University

Abstract

This paper presents a binary clustering network (BCN) and a heuristic algorithm to detect pitch for recognition of keywords in continuous speech. In order to classify nonlinear patterns, BCN separates patterns into binary clusters hierarchically and links same patterns at root level by using the supervised learning and the unsupervised learning. BCN has many desirable properties such as flexibility of dynamic structure, high classification accuracy, short learning time, and short recall time.

Pitch Detection algorithm is a heuristic model that can solve the difficulties such as scaling invariance, time warping, time-shift invariance, and redundance. This recognition algorithm has shown recognition rates as high as 95% for speaker-dependent as well as multi-speaker-dependent tests.

1. 서론

음성인식이란 것은 인간이 한 말을 이해할 수 있는 기계(컴퓨터)를 만들려고 하는 기술이다. 음성인식 기술이 완성되면 현재와 같은 키보드, 마우스등 사람의 손을 직접 사용해야 하는 수단을 사용하지 않고도 컴퓨터에 명령을 줄 수 있게 되고, 사용자(user) 인터페이스 측면에서도 다방면의 혜택을 누릴 수 있다.

음성인식을 구현하기 위해서는 음성신호로부터 특징되는 패턴을 안정적으로 추출할 수 있는 능력이 있어야 한다. 그러기 위해서 음성인식시스템은 다음과 같은 몇가지 불변성(invariant)을 갖고 있어야 한다. 먼저, 음성신호의 시간축상 왜곡현상(time warping)을 처리할 수 있도록 시간왜곡불변(time warping invariant)해야 한다. 다음에는 단어의 시작과 끝을 미리 알 수가 없으므로, 인식단위구간중 어느 부분에 단어가 위치하더라도 찾아낼 수 있도록 시간불변(time invariant)해야 한다. 그리고 특징추출의 능력이 음성신호의 절대적 크기에 관계해야 하므로 크기불변(scale invariant)해야 한다. 또한 특징추출 능력이 화자에 따라 변하지 않도록 화자불변(speaker invariant)해야 한다. 이런 불변(invariant) 특성과 패턴인식 능력을 이용할 수 있다면 화자독립 연속음성인식을 실현할 수 있을 것이다[3].

기존 연구되어온 음성인식알고리즘을 살펴보면 크게 DTW (Dynamic Time Warping), HMM(Hidden Markov Model), 지식기반시스템(Knowledge-Based Approach), 신경망(Neural Network)으로 구분할 수 있다. DTW는 Dynamic Programming을 이용하고, HMM은 확률추정(Stochastic Estimation)을 이용한다. 지식기반시스템은 인공지능의 추론방법을 이용하고, 신경망은 학습을 통한 패턴분류(pattern classification)의 기능을 이용한다. 현재 가장 앞선 시스템이 HMM이라 할 수 있고 신경망 시스템은 아직 연구단계에 있다[3].

그동안 연구되어온 음성인식연구에서는 음성신호에서 특성치(parameter)를 추출할 경우 단지 시간축상 일정한 길이의 프레임단위(frame unit)로 나누어 특성치를 추출하였다. 이 방법으로는 어느 시점에서 프레임을 끊느냐에 따라 그 프레임의 파형이 틀려지며, 당연히 그 프레임의 특성치가 달라지는 시간축상의 왜곡현상이 발생한다. 그러므로 이러한 비일관적인 기존 프레임 분할법을 사용하는 경우, 음성인식의 정확도를 높이기 위해서는 많은 양의 음성자료을 이용한 학습이 필요하게 된다.

본 연구에서는 음성신호의 시간축상 왜곡현상 및 동일파형의 중복성 문제를 해결하기 위해서 먼저 음성신호중 피치(pitch)를 발견하고 이 피치들을 기준으로 일정 길이를 갖는 프레임으로 분할하는 피치기준 프레임분할방법에 대해서 연구한다. 그리고 비선형인 패턴클래스를 계층적으로 선형적인 분류기준에 의해 분류하는 Binary Clustering Network(BCN)에 대해서 연구한다. 또한 대용량 어휘를 인

식하기 위해서 음성인식단위를 음소(phoneme) 단위로 하고, BCN을 이용하여 음소특성치를 학습하고 인식하는 방법에 대해서 연구한다. 그리고 음절인식, 단어인식을 통해 연속음성중 키워드(Key Word)을 인식하는 방법에 대해서 연구한다.

2. 연속음성중 키워드(Key Word) 인식절차

연속 음성인식을 위해서는 그림 1과 같은 단계적인 기능이 필요하다. 이러한 기능들을 살펴보면,

1) 특성치 분석(feature analysis): 음성파형은 용장성(redundancy) 및 시간적 변동성등의 특성때문에 안정된 인식을 위해서 특성치 분석이 필요하다. 특성치 분석방법으로는 스펙트럴 분석(spectral analysis)과 시간적 분석(temporal analysis)이 사용된다.

2) 음성인식단위(speech recognition unit) 인식 : 음성신호의 인식단위는 단어(word), 음절(syllable), 음소(phoneme)가 될 수 있다. 입력된 음성신호는 음성인식단위 순서(speech recognition unit sequence)로 인식된다. 이 부분이 패턴매칭(pattern matching)에 해당된다.

3) 어휘적 분석(lexical analysis) : 인식된 음성인식단위순서와 단어사전에서 일치하는 음성인식단위순서가 있는가를 조사하는 단계이다. 이것은 인식하고자 하는 모든 단어들이 사전에 정의되어야 한다는 것을 의미한다.

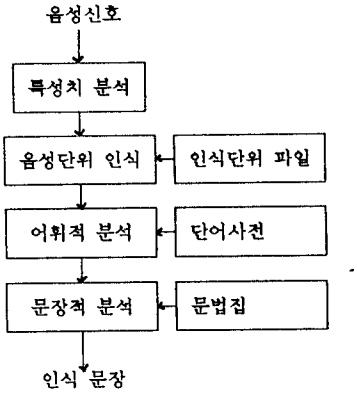


그림 1. 연속음성인식 처리의 흐름

4) 문장적 분석(syntactic analysis) : 어휘적 분석에서 인식된 단어순서(word sequence)들을 사전에 정의된 문법(grammer)에 의해 일치하는 문장을 찾는 단계이다.

3. 연속음성중 키워드(Key Word) 인식방법

3.1 특성치 분석(feature analysis)

특성치 분석은 음성신호중 용장성을 없애는 일종의 정보 압축이라고 할 수 있다. 일반적으로 음성신호의 특성치는 프레임단위로 추출되고, 특성치 분석은 스펙트럴 분석(spectral analysis)과 시간적 분석(temporal analysis)에 의해 이루어 진다.

3.1.1 피치기준 프레임분할법에 의한 프레임 분할

음성신호는 몇가지 단위파형이 반복되어 구성된다. 이러한 단위파형을 파형부호화 용어로 파형소편(波形素片)이라 하며, 한 피치의 파형을 말한다. 음성신호중 피치는 모음과 유성자음부분에서 명확하게 나타난다. 이러한 피치를 기준으로 프레임을 분할한다면 일관성있게 프레임의 특성치를 얻을 수 있고, 패턴매칭(pattern matching)시 시간축상 왜곡(time wraping) 문제를 해결할 수 있을 것이다. 또한 음성신호 특성치를 추출하는데 반복되는 파형소편을 제거함으로써 인식시간을 단축시키고 인식의 정확도를 향상시킬 수 있다. 음성신호에서 피치를 발견할 수 없는 무성음 부분은 기존 방법과 같이 임의 시점을 시작점으로 프레임을 분할하게 된다.

본 연구에서는 PCM(Pulse Code Modulation)방식으로 샘플링 주파수를 10 KHZ, 양자화 정밀도(resolution)를 8 bit로하여 음성신호를 녹음하였다. 녹음된 음성신호는 다시 3개의 샘플데이타의 평균(smoothing method)한 값을 갖고 피치, 즉 파형소편을 추출하였다.

다음은 파형소편 추출절차를 설명한다. 추출절차에 사용된 용어정의로 s(t)는 t시점의 음성신호 진폭(실질적으로는 3개 샘플의 평균값), block[i]는 음성신호중 i번째 단위구간, 파형소편이다. high[i]는 i번째 파형소편의 시작점 진폭(amplitude)이고, pitch[i]는 i 파형소편의 길이로 피치기간(pitch period)에 해당한다.

단계 1 : 유성음에 해당되는 파형소편 추출

단계 1-1 :

조건 (s(t) = 꼭지점(peak 또는 valley))

: 인접 신호들(s(t-2), s(t-1), s(t), s(t+1), s(t+2)) 의 진폭을 비교하여 꼭지점에 해당되는 신호를 찾는다.

행위 1) 단계 1-2로 간다.

그렇지 않으면

행위 1) t = t + 1, 단계 1-1로 돌아간다.

단계 1-2 : 전 peak(valley) 점과 현 valley(peak) 점으로 이루어지는 선분에서

조건 (전 peak(valley) 높이>상역치(high threshold)) 이고

(현 valley(peak) 높이<하역치(low threshold)) 이고

(선분경사도<경사도역치(incline threshold)) 이고

(선분크기>크기역치(size threshold)) 이고

(전 block 시작점과의 간격>간격역치(interval threshold) 이고

(현 선분크기 > 전 선분크기) 이면
: 파형소편의 시작점으로 판단한다.
행위 1) block[i] = t
2) hign[i] = s(t)
3) pitch[i] = t - block[i-1]
4) i = i + 1
: hign[i]와 pitch[i]는 파형소편을 이용한 음절 구분시 사용된다.
그렇지 않으면
행위 1) t = t + 1, 단계 1-1로 돌아간다.

단계 1-3 :
조건 (t = 음성신호 끝)
: 음성신호 끝인가 판단한다.
행위 1) 단계 2로 간다.
그렇지 않으면
행위 2) 단계 1-1로 돌아간다.

단계 2 : 음성시작점 추출

단계 2-1 : 단계 1에서 추출된 첫번째 파형소편(block[0])부터 역방향으로 일정구간 (N)의 음성신호의 진폭 분산(var_high[t])을 계산한다.
여기서 t = 0,...,block[0], N : 프레임 크기(본 연구에서는 20를 사용)

단계 2-2 :
조건 (var_high[t]>분산역치(variance threshold))
: 음성신호(speech)인지 침묵(silence)인지 판단한다.
행위 1) speech = t
2) silence = 0
그렇지 않으면
행위 1) silence = silence + 1

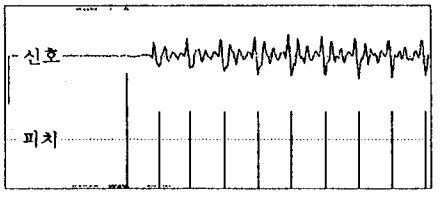
단계 2-3 :
조건 (silence>침묵기간역치(silence threshold))
: 침묵기간이 역치이상이면 중지한다.
행위 1) 음성시작점 = speech
그렇지 않으면
행위 1) 단계 2-1로 돌아간다.

본 연구에서 파형소편 추출를 위해 사용한 역치(threshold)값은 다음과 같다.

- 상역치(high threshold) : 130
- 하역치(low threshold) : 117
- 경사도역치(incline threshold) : -1
- 크기역치(size threshold) : 17
- 간격역치(interval threshold) : 20
- 분산역치(variance threshold) : 2.0
- 침묵기간역치(silence threshold) : 100

이러한 역치값은 반복실험에 의해서 얻어진 값으로 샘플링 레이트(sample rate) 및 양자화 정밀도(resolution)에 따라 적당히 조절되야 한다.

그림 2의 (a),(b)는 파형소편 추출알고리즘을 사용하여 피치기준 프레임분할을 위한 피치를 추출한 결과를 보여준다. 그림 (a)는 초성이 무성음인 음절파형의 전반부이고, 그림 (b)는 초성이 유성음인 음절파형의 전반부이다. 각 그림에서 신호축의 파형이 음성신호를 나타내며, 피치축의 수직선들이 피치를 나타낸다. 피치들이 정확하게 추출되었음 확인할 수 있다.



(a)

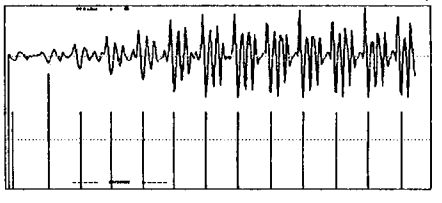

(b)

그림 2. 파형소편 추출알고리즘에 의한 피치추출 결과

3.1.2 프레임 특성치 추출

음성신호에서 피치를 기준으로 분할된 프레임들은 해밍창(hamming window)에 의해 창화(windowing)된다.

창화(windowing)된 음성신호의 프레임들은 선형예측법(LPC : Linear Predictive Coding)/Cepstral 분석으로 LPC 계수(coefficients) 및 cepstral 계수를 추출하였다. 한 프레임의 특성치는 20차수(order)의 cepstral 계수벡터(vector)로 이루어진다.

3.2 음성인식단위(speech recognition unit) 인식

음성은 문장(sentense), 구(phrase), 단어(word), 음절(syllable), 음소(phoneme)라는 원소로 유기적으로 구성되어 있다. 음성인식은 주로 단어단위, 음절단위, 음소단위로 이루어진다. 단어인식시스템은 인식어휘수가 적은 경우에 대단히 유리하다. 아울러 시스템 구조가 대단히 간단해 실용적이다. 그러나 일반적인 음성신호를 처리하기에는 단어 사전의 크기가 너무 커 구현하기 힘들다. 음소인식시스템은 대용량 어휘 화자독립 연속단어음성인식을 위한 궁극적인 해결책이 될 것이다. 그러나 음성중 음소와 음절을 정확히 뽑아내기 위해 복잡한 시스템이 요구된다. 음절인

식시스템은 이 두 방식의 중간적인 특성을 띠고 있다. 현재 상품화 되어 있는 시스템은 거의 전부 단어인식시스템을 채택하고 있다고 볼 수 있다[3].

본 연구에서는 음소단위로 음성을 인식하는 방법을 연구한다. 음소특성치를 학습하고 인식할 수 있는 BCN 알고리즘과 음소구분이 불명확한 모음이 연음되는 음절에서 음소를 구분할 수 있는 파형소편에 의한 음절구분방법에 대해서 연구한다.

3.2.1 Binary Clustering Network(BCN) Algorithm

패턴인식(pattern recognition)에서 대부분의 패턴 클래스(pattern class)들은 비선형(nonlinear)으로 이루어져 있다. 비선형인 패턴 클래스들을 비선형분리기(nonlinar classifier)로 분리한다면 상당히 복잡한 비선형판별함수(nonlinear discriminant function)가 필요하게 된다[7][8]. 모든 패턴 클래스들를 정확하게 판별할 수 있는 비선형판별함수를 구현하는 것은 거의 불가능할 것이다. 본 연구에서는 단순한 선형분리기(linear classifier)로 모든 비선형 클래스들을 분류하고, 패턴인식을 위한 탐색시간을 상당히 단축시킬 수 있는 방법을 제시한다.

비선형인 패턴클래스을 단일수준(one level)에서 모든 패턴 종류별 분류기준을 정형화하는 것은 매우 어렵다. 그러나 Top-Down 방식으로 분류한다면 각 수준(level)에서는 분류기준수를 작게 하고 수준수(level number)를 크게 함으로써 쉽게 분류기준을 정형화할 수 있다. 하위수준(root level)은 단일 패턴클래스가 되고, 상위수준(top level)부터 하위수준(root level)의 분류기준의 연결이 그 패턴종류에 대한 분류기준이 되는 것이다.

Binary Clustering Network(BCN) Algorithm은 모든 패턴들을 계층적으로 이진분할하여 트리(tree)구조를 형성시킨다. 그리고 탐색방법은 이진탐색나무(Binary Search Tree)방법을 사용한다. 이 방법의 기대탐색시간은 패턴클래스 수가 N일 경우 $\Theta(\log N)$이 된다[6]. 그림 3는 BCN의 구조를 보여준다. 여기서 원의 크기는 클러스터의 수를 의미한다.

Binary Clustering Network(BCN) Algorithm은 한 클러스터에서 가장 상이한 두개의 서브클러스터(subcluster)로 분류한다. BCN의 척도는 K-means algorithm과 같이 클러스터 중심점에서 클러스터 영역 모든 점들까지 자승거리를 사용한다. 분할 방법은 한 클러스터에서 가장 거리가 먼 두점을 구하고 이 두점을 초기 서브클러스터의 분류기준으로 삼아 두개의 서브클러스터를 결정한다. 가장 거리가 먼 두점을 초기 분류기준으로 결정하는 것은 가장 상이한 두개의 서브클러스터로 분류한다는 점에서 중요하다. 초기 분류기준을 임의의 두점으로 할 경우 한 클러스터를 가장 상이한 두개의 서브클러스터로 분류한다고 보장할 수 없다. 결정된 두개의 서브클러스터는 평균값을 취하여 새로운 두개의 중심점을 구하고 이 새로운 중심점을 분류기준으로 대치하여 재분류한다. 이 과정은 재분류시 두개의 서브클러스터간에 이동되는 점들의 수(잘못 분류된 패턴수)가 없을 때까지, 즉, 분류기준의 변화가 더이상 발생되지 않을 때까지 계속한다. 분류기준의 변화가 더이상 발생되지 않는 경우, 한 클러스터를 두개의 서브클러스터로 분류하는 최적의 분류수준이 되며, 이때의 두개의 중심점이 두 서브클러스터의 분류기준이 된다. 이 과정이 최적 분류기준 결정단계이다.

서브클러스터들은 다시 상기의 분류기준 결정단계를 반복하여 양분된다. 한 클러스터에서 분류을 중단하는 방법으로는 클러스터의 거리편차가 일정한 역치값이내가 되는 경우 중단하는 자율학습방법과 사전정보에 의한 동일패턴으로 서브클러스터가 구성될 때 분류를 중단하는 감독학습방법이 있을 수 있다.

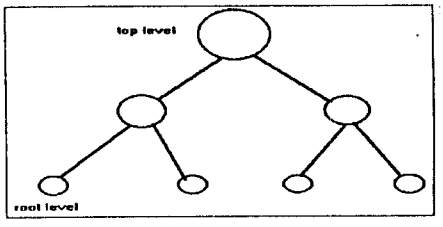


그림 3. BCN의 구조

패턴 X는 p차 벡터값이며 패턴샘플들의 세트를 $\{X_1,\ldots,X_N\}$이라고 하고, k번째 분할기준 수정시 i 클러스터를 $C_i(k)$, 분할기준이 수정완료된 i 클러스터를 C_i이라고 한다. i 클러스터의 k번째 분류기준은 $c_i^1(k)$와 $c_i^2(k)$이라고 한다. 이때 초기 클러스터는 전체 샘플세트가 된다. 즉,

$C_1 = \{ X_1,\ldots,X_N \}$

두 패턴간의 거리는 유클리디안 거리(euclidean distance)을 사용한다. 즉,

$$d^2(X_I, X_J) = (X_I - X_J)^T(X_I - X_J) = | X_I - X_J |^2 \quad (2)$$

BCN 알고리즘의 세부 절차는 다음과 같다.

단계 1. i 클러스터, C_i에서 가장 거리가 먼 두점(패턴)을 구하여 초기 분류기준 $c_i^1(1)$, $c_i^2(1)$으로 한다.

$c_i^1(1) = \max | X_m - X_n |$의 두점중 한점

$c_i^2(1) = \max | X_m - X_n |$의 두점중 $c_i^1(1)$을 제외한 한점

여기서 $X_m, X_n \in C_i$

단계 2. i 클러스터, C_i를 초기 분류기준 $c_i^1(1)$, $c_i^2(1)$를 적용한 다음 규칙으로 두개의 서브클러스터 $C_{i+1}(1)$, $C_{i+2}(1)$로 분류한다.

$X_m \in C_{i+1}(1)$, 만약 $|X_m-c_i^1(1)|<|X_n-c_i^2(1)_n|$

$X_n \in C_{i+2}(1)$, 만약 $|X_n-c_i^2(1)|<|X_m-c_i^1(1)_n|$

여기서 $X_m, X_n \in C_i$

단계 3. 클러스터 $C_{i+1}(k)$, $C_{i+2}(k)$의 분류기준 $c_i^1(k)$, $c_i^2(k)$를 수정한다.

$$c_i^1(k+1) = \frac{1}{N_1} \sum_{X \in C_{i+1}(k)} X \qquad (3)$$

여기서 N_1는 $C_{i+1}(k)$의 샘플수

$$c_i^2(k+1) = \frac{1}{N_2} \sum_{X \in C_{i+2}(k)} X$$

여기서 N_2는 $C_{i+2}(k)$의 샘플수

단계 4. i 클러스터, C_i를 수정된 분류기준 $c_i^1(k+1)$, $c_i^2(k+1)$를 적용한 다음 규칙으로 두 개의 서브클러스터 $C_{i+1}(k+1)$, $C_{i+2}(k+1)$로 다시 분류한다.

$X_m \in C_{i+1}(k+1)$, 만약 $|X_m-c_i^1(k+1)|<|X_m-c_i^2(k+1)_n|$

$X_n \in C_{i+2}(k+1)$, 만약 $|X_n-c_i^2(k+1)|<|X_n-c_i^1(k+1)_n|$

여기서 $X_m, X_n \in C_i$

단계 5. 분류기준 수정은 $c_i^1(k+1) = c_i^1(k)$ 또는, $c_i^2(k+1) = c_i^2(k)$인 경우에 종료한다. 그렇지 않은 경우 단계 3으로 돌아간다.

단계 5까지의 절차는 K-means algorithm에서 분류할 패턴종류가 2인 경우와 동일하다.

단계 6. 분류기준 수정이 완료된 C_i는 두개의 분류기준과, 서브클러스터인 C_{i+1}, C_{i+2}의 주소값을 기억시킨다. 이는 최종적으로 클러스팅의 구조가 Binary Tree 구조를 갖게 한다.

단계 7. 생성된 서브클러스터 C_{i+1}, C_{i+2}..... 들은 다음 조건을 만족하는 경우 분류를 중지한다.

1) 자율학습인 경우
   클러스터의 표준편차 $< \theta$
   여기서 θ는 클러스터 분류의 한계이다.
2) 감독학습인 경우
   클러스터의 패턴들의 종류가 동일한 경우
3) 자율학습 및 감독학습을 병행하는 경우
   클러스터의 표준편차 $< \theta$ 이고
   클러스터의 패턴들의 종류가 동일한 경우

분류중지 조건에 만족되지 않는 서브클러스트들은 단계 2로 돌아간다.

3.2.2 파형소편에 의한 음절구분 절차

음절이 명확히 구분되는 뒤 음절에 자음이 있는 단어에서는 음소를 정확하게 인식할 수 있다. 그러나 음절이 모음으로 연음되는 경우의 모음과 이중모음은 구분하기 어렵다. 모음이 이중모음인 경우, 예로 '위'에서는 모음의 앞부분에는 'ㅜ'음소가 나타나고, 모음의 뒷부분에는 'ㅣ' 음소가 나타난다. '우이동'에서 '우'와 '이'음절은 모음이 연음된다. 연음된 '우이'의 파형에서는'위' 음절에서와 같이 'ㅜ'음소와 'ㅣ'음소가 나타난다.

이중모음을 갖는 음절과 두 모음이 연음되는 두 음절의 차이는 파형의 길이가 된다. 그리고 이중모음과 달리 두 모음이 연음되는 경우는 연음되는 중간부분에 진폭이 줄어드는 경향이 있다. 이러한 경우는 음성신호파형에서 순서적인 파형소편들의 진폭과 피치를 이용하여 음절을 구분하면 음소를 구분하면 할 수 있다.

그림 4는 파형소편에 의한 음절구분 결과를 보여준다. * 표식의 수직선이 한 음절의 끝을 나타낸다.

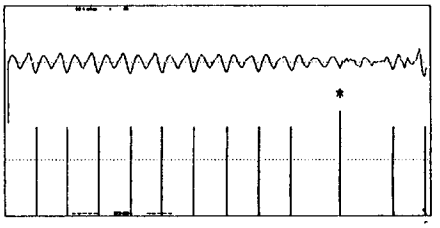


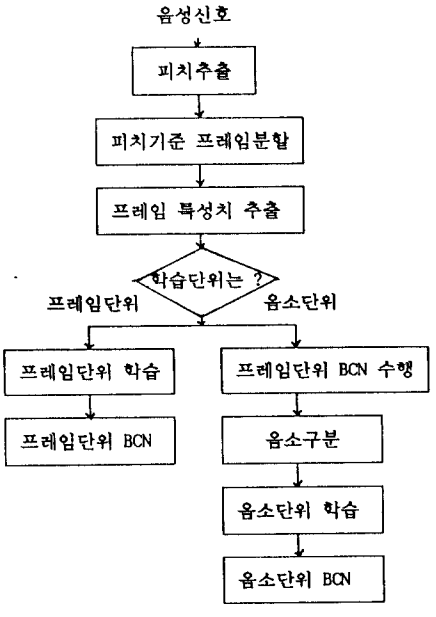
그림 4. 파형소편에 의한 음절구분 결과

3.2.3 음소인식 절차

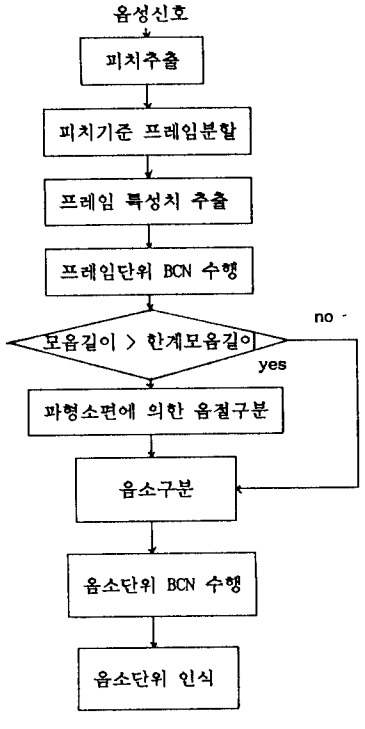
본문에서는 BCN 알고리즘을 이용하여 음성신호중 음소를 인식하는 방법에 대해서 소개한다.

먼저 피치기준 프레임분할법에 의해 음성신호를 프레임 단위로 분할하여 특성치를 산출하였다. 분할된 프레임의 특성치들은 시각적으로 확인한 음소명을 부여하였다. 모든 음소들를 포함하는 충분한 학습용 음소특성치자료를 수집하여, BCN으로 클러스팅하였다. 분할중단기준은 자율학습 및 감독학습을 병행하는 방법을 사용하였다. 이 프레임 특성치로 학습된 BCN을 프레임단위 BCN이라고 칭한다.

프레임단위 BCN은 일차적으로 자음부분과 모음부분을 식별하고 각 부분에서 균등한 비율분할에 의한 임의 수의 프레임을 선택하여 한 음소의 특성치 벡터로 하였다. 프레임 수가 적은 경우는 이전 프레임을 반복시켰다. 임의 수의 프레임특성치는 음소파형들의 시간적인 흐름에 대한 상태(state)의 특성을 나타낸다. HMM 알고리즘에서는 관측된 프레임들의 순서(sequence)를 훈련(training)을 통해 상태를 결정하고 상태전이확률분포(state transition probability distribution)를 구한다[4]. 그러나 자음과 모음부분이 구분된 상태에서 상태변화는 특성치의 시간적인 변화만 의미가 있다고 본다. 그리고 동일한 음소의 자음 또는 모음부분의 길이는 유사한 프레임의 수에 대략적으로 비례한다. 즉, 파형이 길면 비슷한 파형의 수도 많음을 알 수 있다. 이 점은 균등한 비율분할에 의한 상태결정방법의 정당성을 보여 주는 것이다. 본 연구에서는 상태(state) 수를 5개로 하였다. 음소에 대한 5개 프레임특성치들은



(a) 학습절차



(b) 인식절차

그림 5. 음소학습 및 음소인식 처리의 흐름

다시 BCN으로 학습시켰다. 이 BCN을 음소 단위 BCN이라고 칭한다.

음소인식은 학습으로 형성된 2단계의 BCN을 통하여 이루어 진다. 즉, 음성파형을 피치기준 프레임분할법에 의해 프레임단위로 분할하여 특성치를 구하고, 이 프레임특성치 순서(sequence)를 프레임단위 BCN을 통해서 자음부분과 모음부분으로 구분한다. 이때 두개의 모음이 연음되는 경우가 발생할 수 있는데, 이러한 경우는 한계모음길이(반복실험에 의해 산출한 한음절에서 발생할 수 있는 최대 모음길이)보다 길 경우, 파형소편을 이용한 음절을 구분함으로써 한 음소에 해당하는 모음부분을 추출한다. 그리고 학습에서와 동일한 방법으로 각 부분에서 시간적 상태를 나타내는 5개의 프레임특성치를 선택한다. 음소인식은 선택된 5개의 프레임특성치로 음소단위 BCN을 통해서 이루어진다. 그림 5는 음소학습 및 음소인식절차를 보여준다.

3.3 어휘적 분석(lexical analysis)

음절인식은 음소인식에 의해 이루어 진다. 음절이 인식되면 다음 단계로 단어를 인식하게 된다. 인식된 음절의 순서(sequence)는 단어사전에서 단어를 구성하는 음절의 순서(sequence)와 비교되고, 일치하는 단어를 인식단어로 한다.

음성의 연음현상은 음소인식에서와 같이 단어인식에도 어려움을 준다. 자음과 유성모음이 연결되는 단어는 받침이 없는 음절에 받침이 생성된다. 예로 '동대문'의 단어에서 '대'의 파형은 '문'의 'ㅁ'의 영향으로 받침이 'ㅁ'인 '댐'으로 인식된다. 이러한 단어의 인식은 '동대문'을 인식된 음절이 '동'+'대'+'문' 또는 '동'+'댐'+'문'으로 구성되었을 때 인식되는 것으로 하였다. '안암동'의 단어를 보면 '안'의 'ㄴ'이 '암'의 음절에 연음되어 '남'음절로 인식된다. 마찬가지로 '안암동'을 '안'+'암'+'동' 또는 '안'+'남'+'동'으로 구성되었을 때 인식되는 것으로 하였다. 실질적으로 단어를 구성하는 음절의 발음은 연음에 의해서 다르게 발음된다. 그러므로 음절을 인식하고 단어를 인식하는 경우 인식단어의 음절은 발음에 의한 음절로 구성해야 한다.

3.4 문장분석(syntactic analysis)

이 단계는 연속음성중 키워드(Key Word)를 인식하는단계이다.

연속음성중에는 연속음성의 의미를 찾는데 중요하지 않은 부분이 있다. 사람은 음성을 인식할 때 모든 단어를 다 기억하지 않는다. 음성중 중요한 단어만 기억하고 중요한 단어간의 관계만으로 그 의미를 찾는다. 그러므로 연속음성중에서 중요한 키워드를 인식하는 것은 연속음성인식을 위해 매우 효과적인 방법이 될 수 있다.

연속음성중 키워드인식을 위한 문법은 키워드의 관련성

으로 이루어진다. 예로, "홍길동씨의 전화번호를 안내해 주십시오."라는 문장에서 가장 중요한 키워드는 이름과 전화번호가 될 것이다. 예의 문장이 음성으로 들어왔을 경우, "홍길동"과 "전화번호"가 단어사전에서 찾아지게 된다. 그리고 문법집에서 "전화번호" 키워드를 확인하고 전화번호파일중 "홍길동"을 찾아 전화번호를 알려주게 된다.

연속음성신호중에서 키워드가 아닌 음절은 인식시간과 조음현상에 의한 인식정확도에는 영향을 주지만 실질적 키워드 인식에는 영향을 주지 않는다.

4. 음성인식 사례연구및 결과

개발된 음성인식시스템은 자동 전화번호 및 주소안내시스템, 운전보조시스템, 주행안내시스템으로 응용할 수 있다.

자동 전화번호 및 주소안내시스템은 사용자가 "홍길동 전화번호는 무엇", "전화번호 홍길동", "홍길동 전화번호","홍길동 주소는 무엇" 등으로 음성명령을 하였을 경우 "홍길동씨의 전화번호는 000-0000입니다", "홍길동씨의 주소는 안암동입니다"로 전화번호 및 주소안내를 해준다.

운전보조시스템은 음성을 이용한 운전보조시스템으로 "핸들 좌로 30도", "속도 60", "속도 후진 10", "정지", "경적"등 운전에 필요한 몇가지 명령어에 대해서 응답할 수 있다. 이 시스템은 현재 컴퓨터 화면상으로 모의되고 있다.

주행자동안내시스템은 GPS(Global Positioning System) 장비를 이용한 주행중 운전자에게 도로안내를 해주는 시스템이다. 운전자가 "시청 주행", "시청 지도 안내", "주유소 위치"등 운전중 알고 싶은 사항을 음성으로 알려주면 주행자동안내시스템은 적절한 정보를 제공해 준다.

본 음성인식시스템의 사례연구에서는 3 명의 화자들로부터 음성자료를 수집하여 음성인식 실험을 하였다. 실험결과 화자종속인 경우 95% 정도의 인식결과를 보였다. 그러나 화자독립의 경우는 60% 정도의 낮은 인식률을 보였다. 그러한 결과는 학습용 음성자료를 많은 사람들로부터 수집하지 않고 소수의 사람들로부터 수집하여 불충분한 자료를 학습에 사용하였기때문이다. 그러므로 화자독립시 음성인식률을 높이기 위해서는 많은 사람들로부터 음성자료를 수집하여 학습에 사용해야 한다. Lawrence R. Rabiner[4]는 고립단어(isolated word) 화자독립 인식실험에서 100명의 화자들로부터 각 단어를 100번 반복하여 수집한 음성자료를 사용하였다.

5. 결론

본 연구에서는 연속음성중에서 키워드(Key Word)을 인식하는 방법에 대해서 연구하였다.

음성파형중 프레임 분할은 먼저 피치를 발견하고 피치를 기준으로 일정한 길이로 분할하는 피치기준 프레임분할법을 사용하였다. 이러한 피치기준 프레임분할법은 음성파형을 일관성있게 프레임단위로 분할할 수 있었으며, 음성인식시 시간축상의 왜곡문제를 해결할 수 있었다.

새로운 패턴분류 및 인식방법으로 BCN(Binerary Clusting Network) 알고리즘을 개발하였다. BCN은 클러스터를 계층적으로 이분화하여 트리(tree)을 구성한다. 패턴분류를 위한 학습방법은 자율적인 학습 및 감독적인 학습을 병행할 수 있으며, 패턴인식을 위한 기대탐색시간은 패턴종류 수가 N일 경우 $\Theta(\log N)$이 된다.

음성인식단위는 음소단위로 하였다. 음소인식시스템은 대용량 어휘 화자독립 연속단어음성인식을 위한 궁극적인 해결책이 될 것이다. 음소인식절차는 먼저 피치기준 프레임분할법에 의해 음성신호를 프레임단위로 분할하고 특성치를 산출하였다. 음소는 2단계의 BCN을 통해 학습 및 인식이 이루어 졌다. 연속음성중 키워드 인식은 음소 및 음절인식을 통해 인식된 단어들의 순서(sequence)를 미리 정의된 문법집의 단어들의 관련성과 비교하여 이루어 졌다.

화자독립 연속음성인식을 위해서는 아직 많은 추가연구가 필요하다. 특히 많은 사람들로부터 학습에 필요한 충분한 음성자료를 수집하는 일이 큰 과제인 것 같다.

참고문헌

[1] 동역메카트로닉스 연구소, "음성합성과 음성인식시스템", 영진출판사, 1990

[2] Panos E. Papamichalis," Practical Approaches To Speech Coding", 1987

[3] 정 훈,"신경망을 이용한 음성인식", 전기공학회 논문지, 10권 2호, pp 49~59, 1992년 4월

[4] Lawrence R. Rabiner," A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition", Proc. IEEE, vol.77, NO.2, pp.257-286, Feb.1989

[5] Jay G. Wilpon, Lawrence R. Rabiner, Chin-Hui Lee, E. R. Goldman, "Automatic Recognition of Keywords in Unconstrained Speech Using Hidden Markov Models", IEEE Trans. Acoust., Speech, Signal Processing, vol.38, no.11, pp.1870-1878, Nov.1990

[6] Christopher J. Van Wyk, "Data Structures and C Programs", 1990

[7] Sing-Tze Bow, "Pattern Recognition and Image Preprocessing", 1992

[8] Stephen P. Banks "Signal Processing, Image Processing and Pattern Recognition", 1990

[9] James A. Freeman, David M. Skapura, "Neural Networks Algorithms, Applications, and Programming Techniques", 1991

[10] 김대수, "신경망 이론과 응용(I)", 하이테크정보, 1992