

MLHF 모델을 적용한 어휘 인식 탐색 최적화 시스템

안 찬 식*, 오 상 엽**

Vocabulary Recognition Retrieval Optimized System using MLHF Model

Chan-Shik Ahn *, Sang-Yeob Oh **

요 약

모바일 단말기의 어휘 인식 시스템에서는 통계적 방법에 의한 어휘인식을 수행하고 N-gram을 이용한 통계적 문법 인식 시스템을 사용한다. 인식 대상이 되는 어휘의 수가 증가하면 어휘 인식 알고리즘이 복잡해지고 대규모의 탐색공간을 필요로 하게 되며 처리시간이 길어지므로 제한된 연산처리 능력과 메모리로는 처리하기가 불가능하다. 따라서 본 논문에서는 이러한 단점을 개선하고 어휘 인식을 최적화하기 위하여 MLHF 시스템을 제안한다. MLHF는 FLaVoR의 구조를 이용하여 음향학적 탐색과 언어적 탐색을 분리하여 음향학적 탐색에서는 HMM을 사용하고 언어적 탐색 단계에서는 Levenshtein distance 알고리즘을 사용한다. 시스템 성능 평가 결과 어휘 종속 인식률은 98.63%, 어휘 독립 인식률은 97.91%의 인식률을 나타냈으며 인식속도는 1.61초로 나타내었다.

Abstract

Vocabulary recognition system of Mobile terminal is executed statistical method for vocabulary recognition and used statistical grammar recognition system using N-gram. If limit arithmetic processing capacity in memory of vocabulary to grow then vocabulary recognition algorithm complicated and need a large scale search space and many processing time on account of impossible to process. This study suggest vocabulary recognition optimize using MLHF System. MLHF separate acoustic search and lexical search system using FLaVoR. Acoustic search feature vector of speech signal extract using HMM, lexical search recognition execution using Levenshtein distance algorithm. System performance as a result of represent vocabulary dependence recognition rate of 98.63%, vocabulary independence recognition rate of 97.91%, represent recognition speed of 1.61 second.

▶ Keyword : 음성인식(speech recognition), 어휘 인식(vocabulary recognition), 음향학적 탐색(acoustic search), 언어적 탐색(lexical search)

• 제1저자 : 안찬식 교신저자 : 오상엽

• 투고일 : 2009. 09. 28, 심사일 : 2009. 10. 13, 게재확정일 : 2009. 10. 26.

* 광운대학교 컴퓨터공학과 박사과정 **경원대학교 IT대학 컴퓨터소프트웨어

※ 본 연구는 2009년도 경원대학교 지원에 의한 결과임.

1. 서론

음성인식은 모델과 비교를 통해 인식 대상을 판단하는 것이다. 음성인식에서는 인식 대상 어휘를 대상으로 모델을 생성하며 모델의 형태는 HMM(Hidden Markov Model)이다. [1] HMM은 확률적으로 처리하는 기법으로 샘플들을 기반으로 필요한 확률 파라미터를 추정하게 된다. 이러한 샘플들은 특징 파라미터로써 신호 처리 기법을 이용하여 추출하며 순수하게 이 특징 파라미터들로만 이루어진 모델을 음향모델이라 한다. [2] 통계적 음성인식에서는 음향모델을 단어인식 단계에서 Viterbi 알고리즘 또는 DTW(Dynamic Time Warping) 알고리즘, Levenshtein distance 알고리즘 등의 패턴 정합 알고리즘을 이용하여 미리 학습된 패턴 중에서 유사도가 가장 높은 것을 찾아서 인식 결과를 출력한다. [2,6,11,12]

Viterbi 알고리즘은 실시간을 요구하는 인식과정에서 동적 프로그래밍 기술을 이용하여 상태경로의 변이와 최적의 모델을 추정한다. 연산에서의 부하를 월등히 감소시킨다는 장점이 있으나 전, 후향 확률추정을 이용한 디코딩에 비해 인식율이 다소 저하되는 단점이 있다. [15]

DTW 알고리즘은 비선형 시간 정규화를 갖는 패턴정합 알고리즘으로 공통적이고 균일하게 샘플간격을 갖게 음성패턴을 시간적으로 샘플화하여 인식 시간이 빠른 장점이 있으나 정합함수가 너무 많은 관계로 다른 분류의 단어 간에 분별이 어려워지게 되므로 강제로 정합함수 기울기를 제한하는 방법을 도입해야 하는 단점이 있다. [11]

Levenshtein distance 알고리즘은 변환이 이루어지는 음소들이 우열을 갖지 않는 경우에 효과적이나 추출되는 음소열의 경우 음향학적인 정보가 포함되어 있어 음소들 사이의 변화에 대한 가중치를 다르게 적용해야 하는 단점이 있다. [16]

최근 휴대용 단말기의 사용이 빈번하면서도 휴대용 단말기에서는 대용량의 인식이 어렵다는 인식을 가지고 있어 사용이 쉽지 않다. 인식 대상이 되는 어휘의 수가 증가하게 되면 어휘 인식 알고리즘이 복잡해지고 대규모의 탐색공간을 필요로 하게 되며 처리시간이 길어지므로 인식 시스템에서 소요되는 메모리의 용량이 커지는 단점이 있다. 이러한 단점을 개선하고 어휘 인식을 최적화하기 위하여 본 논문에서는 MLHF(Modified Levenshtein distance and HMM using FLaVoR)를 제안한다.

MLHF는 FLaVoR의 구조를 이용하여 음향학적 탐색과 언어적 탐색을 분리하여 음향학적 탐색에서는 HMM을 사용하고 언어적 탐색 단계에서는 Levenshtein distance 알고

리즘을 개선하여 어휘의 수가 증가 하더라도 음소열을 이용하여 처리하므로 탐색 공간을 줄일 수 있고 처리 시간을 단축할 수 있는 장점을 가진다.

음소 인식기에서 어휘 인식기로 전달하는 특징벡터를 각 음소 구간에서의 신뢰도 벡터와 음소별 HMM을 사용한 음소 인식기를 사용하고 단어 인식단계는 인식된 음소열로부터 Levenshtein distance 알고리즘을 사용한 시스템을 실험한 결과 빠르고 좋은 인식성능을 확인하였다. 제 2장에서는 어휘 인식기의 기본구조를 설명하고, 제 3장에서는 본 논문에서 제안한 MLHF 시스템에 대하여 설명한다. 제 4장에서는 제안한 시스템의 실험결과에 대하여 설명하고 제 5장에서 결론을 기술한다.

II. 기존 연구

2.1 일체형 어휘인식

음성인식에는 음성음향학적 지식을 이용하는 방법, 통계적 방법, 인공지능을 이용한 방법, 신경회로망을 이용한 방법 등이 널리 연구되었다. 통계적 음성인식에서는 아래 그림 1에서와 같이 학습단계에서 추출된 특징벡터를 이용하여 기준이 되는 음향모델 또는 기준패턴을 구한 다음, 어휘 인식단계에서 비터비 알고리즘 또는 DTW와 같은 패턴 정합 알고리즘을 이용하여 미리 학습된 패턴 중에서 가장 유사한 것을 찾아서 인식결과로 출력한다. 일체형 인식 구조는 탐색 과정에서 모든 가능한 지식 정보들을 가져오므로 일체형 탐색 방법은 복잡한 언어 모델을 사용하여 음향모델과 언어모델을 사용한다. [2]

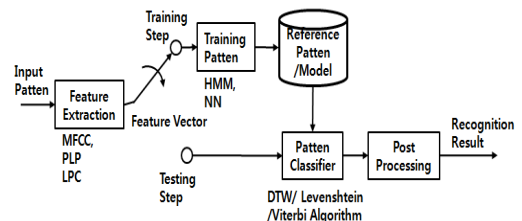


그림 1. 어휘인식 시스템
Fig. 1. Speech recognition system

일체형 어휘 인식의 장점은 검색의 효율성에 있다. 어휘에서 음향적 혼란이 많아 어휘 및 언어 모델에 의해 제공되는 정보를 일찍 포함하는 것은 탐색공간으로부터 가장 가능성이 낮은 부분들을 삭제하기 위해 효과적인 것으로 입증되었다.

2.2 FLAVoR 어휘인식

그림 2는 FLAVoR 어휘 인식 시스템의 구조이다. 첫 번째 단계인 음소인식기에서 음향-음소 모델에 대한 탐색 알고리즘은 주어진 입력신호로부터의 음향 특징을 이용하여 가장 유력한 음소열의 네트워크를 결정한다. 음소 네트워크는 시작과 종료 시점을 가지는 최적의 음소 집합을 포함하고 단어 탐색 단계에서 더 복잡한 모델링 구조가 가능하도록 돕는다. 음소 인식기는 특정 언어를 위한 어떠한 지식 도메인에서도 동작할 수 있다. 또한 음소 정보 자체는 언어 학습과 같은 특정한 응용 분야에 사용되거나 적절한 이름 인식처럼 특별한 문제를 해결하기 위해서도 사용할 수 있다.[3,4]

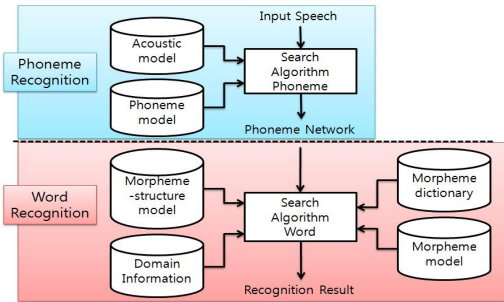


그림 2. FLAVoR 어휘인식 시스템
Fig. 2. FLAVoR speech recognition system

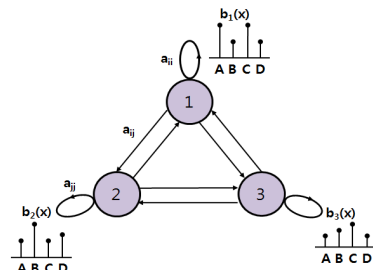
음소 네트워크와 관련된 메타 데이터는 실제 단어 인식을 수행할 두 번째 단계의 입력으로 주어지고 두 번째 단계인 단어 탐색단계에서 탐색 알고리즘은 임의로 음운과 구문의 두 지식 정보를 가진다. 음운 요소는 형태소(morphemes) 열과 단어 경계들의 가설로 음소 네트워크를 바꾸고 구문 지식은 형태소 사전, 형태소 구문 모델, 발음 규칙, 교체 행렬로 구성된다. 모든 지식 정보들은 탐색 네트워크나 유한 상태 변환기들로 결합되어지며 이러한 변환기는 탐색을 위한 아주 간결하고 효과적인 해결책이다.[4] 형태소 사전은 각 표제어(lemma)를 위한 음소 사본을 가지는 두 접사(affixes)와 어근(roots)의 집합으로 구성된다. 형태-구문언어 모델은 단어의 형태와 구문 정보와 그 문맥을 기초로 하는 각각의 가설된 단어를 위한 확률 값을 제공한다. 계층적이고, 확률적인 형태소 분석이 각 입력단어의 분석을 위해 제공되며 형태소 정보는 구문기반의 언어 모델로 병합된다. 이러한 구조가 가지는 장점들을 요약하면 동적 어휘의 사용보다 나은 일반적인 언어 모델들의 병합과 더 높은 수준의 모듈화와 개선되어진 풍부한 추력을 들 수 있다.[8]

2.3 음향모델과 어휘인식

음향학적 모델과 언어적 탐색을 분리하여 처리하는 방법이며 음향학적 모델 추출에는 HMM을 사용하여 기준이 되는 음향모델 또는 기준 패턴을 구한다. 음향모델은 음소 열로 구성되어 지며 언어적 탐색을 하기 위한 입력 값으로 주어진다. 언어적 탐색은 Levenshtein distance 알고리즘을 이용하여 어휘 인식을 수행한다.

2.3.1 HMM(Hidden Markov Model)

HMM은 시간에 따라서 변화하는 신호를 모델링하는 방법으로서 숨겨진 상태전이 확률 과정과 각 상태에서 관측신호를 발생하는 관측 확률과정의 두 개의 랜덤프로세스로 이루어진다. 그림 3은 3개의 상태를 갖는 HMM을 나타낸다. 각 상태에서는 A, B, C, D의 관측 심볼을 출력한다. HMM의 상태 전이 확률 및 관측확률은 시간에 독립이며 현재 상태에만 의존한다고 가정한다. HMM은 초기 ($t=0$)에 상태 i 의 확률 $\pi_i = \Pr(s_0 = i)$, 상태 i 에서 j 로의 전이 확률 $a_{ij} = \Pr(s_t = i, s_{t+1} = j)$, 상태 j 에서 심볼 k 를 관측할 확률 $b_j(k) = \Pr(x_t = k | s_t = j)$ 로 표현한다.[17]



State seq : 1 3 1 2 2 3
Observed seq. (X) : A C D B A C
그림 3. 3상태를 가지는 HMM
Fig. 3. 3state on HMM

2.3.2 Levenshtein distance Algorithm

두 패턴의 정합에 DP(Dynamic Programming)정합 매트릭스를 적용한 것으로 단순히 두 패턴의 최적 정합을 위해 필요한 변환의 횟수만을 패턴간의 거리로 나타낸다. 즉, 음소 열 X와 음소열 Y사이의 Levenshtein distance는 음소열 X와 음소열 Y가 같아지기 위해서 필요한 최소한의 삽입, 삭제, 대체 변환의 수를 의미한다. 두 개의 음소열이 $X = x_1, x_2, \dots, x_m$ 과 $Y = y_1, y_2, \dots, y_n$ 이고, x_i, y_j 는 각각 음소열 X, Y의 i 번째와 j 번째의 음소이고 음소열의 음소 길이는 각각

$|X| = m, |Y| = n$ 이다. Levenshtein distance $LD_{m,n}$ 의 의미는 하나의 음소열 x_1, x_2, \dots, x_m 을 다른 음소열 y_1, y_2, \dots, y_n 로 정합하는데 필요한 최소 연산수라고 한다. 두 음소열 사이의 Levenshtein distance는 아래 식1과 같은 알고리즘으로 계산된다.

$$LD_{0,0} = 0 \quad \dots\dots (1)$$

$$LD_{m,0} = m, LD_{0,n} = n$$

$$LD_{m,n} = \min \left\{ \begin{array}{l} LD_{m-1,n} + 1 \\ LD_{m-1,n-1} + 1 \\ LD_{m-1,n-1} + t_{m,n} \end{array} \right\}$$

여기서 $t_{m,n} = \left\{ \begin{array}{l} s, \text{ if } x_m = y_n (\text{일치}) \\ r, \text{ if } x_m \neq y_n (\text{불일치}) \end{array} \right\}$

Levenshtein weight set : $W = \{s, r\} = \{0, 1\}$

알고리즘에서 삽입, 삭제, 대체에 대한 가중치는 별점으로 작용하며 삽입이 일어나면 $LD_{m,n} = LD_{m,n-1} + 1$ 로 계산되고, 삭제가 일어나면 $LD_{m,n} = LD_{m-1,n} + 1$ 이 계산된다. 대체가 일어나면 $x_m \neq y_n$ 인 경우 $LD_{m,n} = LD_{m-1,n-1} + 1$ 로 계산되고, 일치할 때는 $x_m = y_n$ 인 경우 $LD_{m,n} = LD_{m-1,n-1}$ 으로 별점 없이 계산된다.[18,19,20]

III. MLHF 시스템의 설계 및 구현

3.1 시스템 모델

MLHF 시스템을 그림 4와 같이 구성하였다. FLaVoR 구조를 이용하여 음소 인식단과 단어 인식단으로 구분을 하였고 음소 인식단은 음성 신호가 입력될 경우 끝점검출(End-point detection)과 특징 추출이 이루어지는 음향학적인 탐색을 HMM을 이용하여 음소 인식을 수행한다. 음소 인식의 결과 음소열을 출력되고 출력되어진 음소열은 단어 인식단의 입력으로 입력되어 패턴 모델을 구성하고 구성 되어진 패턴 모델로부터 Levenshtein distance 알고리즘 어휘 인식이 수행되어 매칭된 어휘의 결과로 출력된다.

단 사이의 음소열을 음소 인식단에서 음향-음소 모델에 대한 탐색 알고리즘은 주어진 입력신호로부터의 음향 특징을 이용하여 가장 유력한 음소열의 네트워크를 결정하고 음소 네트워크와 관련된 메타-데이터는 실제 단어 인식을 수행할 두 번째 단계의 입력으로 주어진다. 탐색단계에서 탐색 알고리즘은 임의로 음운과 구문의 두 지식 정보를 가진다. 음운 요소는 형태소(morphemes) 열과 단어 경계들의 가설로 음소 네트워크를 바꾸고 구문 지식은 형태소 사전, 형태소 구문 모델, 발음규칙, 교체 행렬로 구성된다.

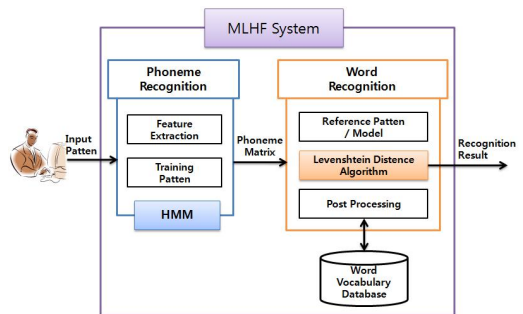


그림 4. MLHF 시스템 구조도
Figure 4. MLHF system architecture

3가지의 상태를 가지는 Left-to-Right의 상태를 갖는 유사음소(PLU)단위의 문맥중속적인 Triphone을 바탕으로 한 인식시스템으로써 특징과미터로 12차의 MFCC와 1차의 에너지를 사용하여 총 13차의 특징을 추출하여 인식실험에 사용하였다.

가변어휘 인식 실험용 음소모델 훈련을 위하여 총 3종류(ETRI의 PBW445 DB, POW3848 DB, 국어공학연구소의 PBW452 DB)의 음성 DB를 사용하였고, 음성 DB는 8k로 Sampling하고 16bit 양자화한 선형 PCM의 포맷을 갖는다.

3.2 MLHF를 이용한 어휘 인식

음소 인식은 3상태 혹은 5상태의 HMM 훈련에 의해 구성된 모델이 가지는 음소 연속 네트워크에 의해 입력된 음성의 최적 음소열을 발생시킨다. 이후, 가장 유사한 대표 단어(N-nearest word)를 선택한 후 보다 세밀한 text-dependent model(tri-phone model 또는 bi-phone model)을 이용한 인식 네트워크를 구성하는데 사용된다.

모델을 바탕으로 인식 단에서 입력음성으로부터 특징 과미터를 추출하고 HMM network는 발음사전에 근거하여 유사음소모델을 연결해서 만든 단어모델을 병렬로 나열하여 인식이 수행된다. 인식수행과정은 문맥중속적인 음향모델을 바탕으로 기본적인 인식을 수행할 수 있는 어휘를 발음사전에 등록시킨다. 등록어휘에 대한 인식을 수행하고 인식어휘가 변경되거나 추가되었을 때 인식 어휘변경 및 추가 과정을 거쳐 변경되거나 추가된 인식어휘에 대해서 음운학적인 어휘형태를 음성학적인 표기로 변화하기 위한 자동 발음열 생성과정을 수행하고 음소열로 변환된다. 음소열은 PLU(Phone Likely Unit)로 Tagging되어져 발음사전에 등록함으로써 인식을 할 수 있는 어휘로 등록 된다.

두 음소열 “강남”과 “강당”을 기존의 Levenshtein distance 를 사용하여 거리를 계산할 경우 그림 5와 같이 최소거리가 계산된다.

비교가 되는 두 개의 음소열을 X는 “ㄱ, ㅏ, ㅓ, ㅓ, ㅓ”이라 하고, Y는 “ㄱ, ㅏ, ㅓ, ㅓ, ㅓ”이라 할 때, 그림 5의 (1)은 두 음소열을 통하여 행렬을 초기화했을 때를 나타낸다. (2)는 음소열 X의 i=1번째 음소와 음소열 Y의 모든 음소를 대상으로 거리를 계산한 결과이다. 완성된 그림5의 (4)에서 오른쪽 하단의 숫자 ‘2’가 두 음소열에 대한 Levenshtein distance가 된다. 음소 ‘ㄱ’이 음소 ‘ㅓ’으로, 음소 ‘ㅓ’이 음소 ‘ㅓ’으로 변환된다.

		ㄱ	ㅏ	ㅓ	ㅓ	ㅓ	ㅓ
0	0	1	2	3	4	5	6
ㄱ	1	0					
ㅏ	2	1					
ㅓ	3	2					
ㅓ	4	3					
ㅓ	5	4					
ㅓ	6	5					

(1)

		ㄱ	ㅏ	ㅓ	ㅓ	ㅓ	ㅓ
0	0	1	2	3	4	5	6
ㄱ	1						
ㅏ	2						
ㅓ	3						
ㅓ	4						
ㅓ	5						
ㅓ	6						

(2)

		ㄱ	ㅏ	ㅓ	ㅓ	ㅓ	ㅓ
0	0	1	2	3	4	5	6
ㄱ	1	0	1	2			
ㅏ	2	1	0	1			
ㅓ	3	2	1	0			
ㅓ	4	3	2	1			
ㅓ	5	4	3	2			
ㅓ	6	5	4	3			

(3)

		ㄱ	ㅏ	ㅓ	ㅓ	ㅓ	ㅓ
0	0	1	2	3	4	5	6
ㄱ	1	0	1	2	3	4	5
ㅏ	2	1	0	1	2	3	4
ㅓ	3	2	1	0	1	2	3
ㅓ	4	3	2	1	1	2	3
ㅓ	5	4	3	2	2	1	2
ㅓ	6	5	4	3	3	2	2

(4)

그림 5. 두 음소열에 대한 레빈슈타인 거리 계산
Figure 5. Levenshtein distance calculation for two phoneme strings

제안한 방법을 사용하였을 경우 어휘수가 증가하여도 HMM에서 출력인 음소열을 사용하여 탐색하므로 Viterbi 알고리즘과 DTW 알고리즘에서 단점으로 나타나는 인식율을 보다 효율적으로 향상시킨다.

개선된 Levenshtein distance을 위한 기본 알고리즘은 다음과 같다.

```

Begin
  Initialize the LevenshteinDistanceSearch
Job1 : Get Input Vector Xt
  If Vector Xt is exist then next
  LD(m,n)=LD(m,n-1)-1
  else goto end
  Clear NL
  Process the likelihood(ALL n∈CL),

Job2 : Get Delete Vector Xt
  If Vector Xt is exist then next
  LD(m,n)=LD(m-1,n-1)+1
  else goto end
  Clear NL
  Process the likelihood(ALL n∈CL),

Job3 : Get alternate Vector Xt
  If Vector X≠Vector Y then
  LD(m,n)=LD(m-1,n)+1
  else Vector X=Vector Y
  LD(m,n)=LD(m-1,n)
  goto end
  Clear NL
  Process the likelihood(ALL n∈CL),

end
  
```

기존 Levenshtein distance Algorithm에 나타나 는 단점을 음소들 사이의 변화에 대한 가중치를 다르게 적용하여 개선하여 어휘수가 증가하여도 탐색공간의 효율성과 처리시간을 단축하였다.

IV. 실험결과 및 분석

본 논문에서 제안한 MLHF의 성능 검증을 위하여 인식 실험을 수행하였으며 시스템 환경은 표 1과 같다. 훈련과정과 실험환경과의 불일치 문제를 해결하기 위해 잡음처리는 위너(wiener) 필터를 사용하였다. 어휘 목록은 회사명, 지역명, 지하철역명으로 구성하였다. 인식 실험에서는 실험에 참가한 화자가 어휘 중에서 임의로 50단어씩 발음하여 총 1500단어를 대상으로 실험을 수행하였다. 어휘는 실내 환경과 잡음 환경에서 이동기기 등에 내장되어 있는 내장형 마이크로폰을 사용하여 16kHz Mono로 녹음 하였고, 16bit PCM 양자화를 사용하였다. 실험 어휘는 실내 10명, 실외 5명 등 총 15명의 성인 남성이 참가하였다. 실내 환경은 50~55dB이고, 실외 환경은 70~75dB의 소음환경 하에서 실험하였다.

표 1. 시스템 환경
Table 1. System Environment

항목	Client	Server
모델명	IPAQ 112	Pentium 4
CPU	624MHz	2.4GHz
RAM	128M	512M
ROM	256M	
OS	Windows Mobile 6 Classic	Windows XP

표 2는 기존의 방식인 viterbi, DTW, Levenshtein distance를 이용한 인식구조와 본 논문에서 제안한 MLHF를 이용한 인식구조 인식 속도에 관한 실험 결과이다. 실험은 어휘입력이 끝난 상태에서 인식 결과가 나오기까지의 시간차를 이용해 측정하였으며 결과에 의하면 어휘수에 따라 기존의 방식과 제안한 MLHF 모두 인식 속도가 증가하였지만 기존 방식 보다 현저히 빠른 속도를 나타내었다.

표 2. 인식 속도
Table 2. Recognition Speed

어휘수	인식 속도(Sec)			
	Viterbi	DTW	Levenshteini	MLHF
5만	1.23	1.18	1.17	1.17
15만	1.68	1.33	1.50	1.34
30만	2.01	1.57	1.73	1.61

표 3과 4는 기존의 방식인 viterbi, DTW, Levenshtein distance를 이용한 인식구조와 본 논문에서 제안한 MLHF를 이용한 인식 구조를 실내 환경에서의 실험과 실외 환경에서의 실험을 나타낸다. 실내 환경은 50~55dB에서 실험 하였으며, 실외 환경은 70~75dB의 소음환경 하에서 실험하였다.

표 3. 실내 환경 인식률
Table 3. Indoor Environment Recognition Rate

어휘	인식률(%)			
	Viterbi	DTW	Levenshteini	MLHF
어휘 종속	97.51	93.31	98.31	98.63
어휘 독립	96.97	91.91	96.91	97.91

표 4. 실외 환경 인식률
Table 4. Outdoor Environment Recognition Rate

어휘	인식률(%)			
	Viterbi	DTW	Levenshteini	MLHF
어휘 종속	91.01	89.21	91.11	91.21
어휘 독립	90.07	87.11	90.01	90.11

표 3과 4의 결과에서 보는 것과 같이 시스템 성능 평가 결과 어휘 종속 인식률은 98.63%, 어휘 독립 인식률은 97.91%의 인식률을 나타냈으며 인식속도는 1.61초로 나타내었다. viterbi의 결과 값과 비교하였을 때 속도 면에서 0.4초 향상되었으며 인식률에서 1.01% 향상된 결과를 나타냈다.

V. 결론

본 논문은 FLaVoR의 구조를 이용하여 음소 인식단에 음향모델인 HMM을 사용하고 단어 인식단에 Levenshtein distance 알고리즘을 사용하여 MLHF 시스템을 구성하여 제안한 것으로 음소열로부터 패턴 매칭을 수행하는 실험을 수행하였다.

제안한 MLHF 시스템 구조로 인하여 휴대용 단말기에서 음향학적 탐색과 언어적 탐색을 분리하여 어휘의 수가 증가 하더라도 음소열을 이용하여 처리하므로 탐색 공간을 줄일 수 있고 처리 시간을 단축 할 수 있는 장점을 확인하였으며 인식 속도와 인식률에서 기존 시스템보다 나은 결과를 얻을 수 있었다. 시스템 성능 평가 결과 어휘 종속 인식률은 98.63%, 어휘 독립 인식률은 97.91%의 인식률을 나타냈으며 인식속도는 1.61초로 나타내었다. viterbi의 결과 값과 비교하였을 때 속도 면에서 0.4초 향상되었으며 인식률에서 1.01% 향상된 결과를 나타냈다.

따라서 MLHF 시스템은 휴대용 단말기를 이용하므로 휴대성과 장소의 구애를 받지 않고, 기존 인식 시스템의 높은 인식률을 그대로 적용할 수 있다.

인식 시스템의 인식률을 높이기 위한 방법으로는 검색 알고리즘의 개발과 실 환경에서의 주변 잡음문제에 대한 연구가 필요하다.

참고문헌

- [1] S. Young, D. Kershaw, J. Odell, D. Ollason, Valtcher, P. Woodland, "The HTK Book," Cambridge University Engineering Department, 2002.
- [2] L. R. Rabiner, B. H. Juang, "Fundamentals of speech recognition", Prentice Hall, 1993.
- [3] Kris Demuynck, Tom Laureys, Dirk van Compernelle, and Hugo van Hamme, "FLaVor: a flexible architecture for LVCSR," In EUROSPEECH-2003, pp. 1973-1976, 2003.
- [4] K. Demuynck, J. Duchateau, and D. Van Compernelle, "A static lexicon network representation

- for cross-word context dependent phones," In Proc. EUROASPEECH, Vol.1, pp. 143-146, 1997.
- [5] 윤경섭, "휴대용 단말기를 위한 실시간 무선 영상 음성 전송 기술," 한국컴퓨터정보학회논문지, 제 14권, 제 4호, 111-117쪽, 2009년 4월.
- [6] D. Jurafsky and J. H. Martin, "Speech and Language Processing" Prentice-Hall, 2000.
- [7] M. F. Gales, "Model-based techniques for noise robust speech recognition" Ph. D. dissertation, University of Cambridge, Sept, 1995.
- [8] A. S. Manos and V. W. Zue, "A study on out-of-vocabulary word modeling for a segment-based keyword spotting system," Master Thesis, MIT, 1996.
- [9] 조영수, 이기정, 김광태, 홍재근, "HMM을 이용한 한국어 음소인식 (Korean Phoneme Recognition using HMM)," 대한전자공학회 학술발표회 논문집, 제 16권, 제 1호, 81-84쪽, 1994년.
- [10] 김동주, 김한우, "문맥가중치가 반영된 문장 유사도 척도," 전자공학회 논문지, 제 43권, 제 6호, 496-504쪽, 2006년.
- [11] Justin Zobel and Philip Dart "Phonetic String Matching: Lessons from Information Retrieval," SIGIR'96, pp. 166-173, 1996.
- [12] T. Jitsuhiro, S. Takatoshi, and K. Aikawa, "Rejection of out-of-vocabulary words using phoneme confidence likelihood," ICASSP, pp. 217-220, 1998.
- [13] 지진구, 윤성일, "음성을 이용한 화자 검증기 설계 및 구현," 한국컴퓨터정보학회논문지, 제 5권, 제 3호, 91-98쪽, 2000년 9월.
- [14] 김기백, 최중호, "음성인식 기반 콘텐츠 네비게이션 시스템," 한국컴퓨터정보학회지, 제 15권, 제 1호, 99-102쪽, 2007년.
- [15] L. Rabiner and B. H. Juang, "Fundamentals of Speech Recognition" Prentice-Hall, 1993.
- [16] Eiichi Tanaka and Tamotsu Kasai, "Synchronization and Substitution Error-correcting codes for the Levenshtein Metric," IEEE Trans. Information Theory, Vol.IT-22, No.2, pp. 156-176, 1976.
- [17] L. R. Bahl, P. V. deSouza, P. S. Gopalakrishnan, D. Nahamoo, and M. Picheny, "A Fast Match for Continuous Speech Recognition Using Allophonic Models," InProc. IEEE ICASSP-92, Vol.1, pp. 17-21, 1992.
- [18] S. Ortman, A. Eiden, H. Ney, and N. Coenen, "Look-ahead Techniques for Fast Beam Search," InProc. IEEE ICASSP-1997, pp. 1783-1786, 1997.
- [19] W. Daelemans, S. Buchholz, and J. Veenstra, "Memorybased shallow parsing," in Proc. CoNLL, pp. 53-60, 1999.
- [20] Justin Zobel and Philip Dart "Phonetic String Matching: Lessons from Information Retrieval," SIGIR'96, pp. 166-173, 1996.

저 자 소개



안 찬 식

2002 : 광운대학교 공학석사.
2002 - 현재 : 광운대학교 박사과정
관심분야 : 음성인식, 분산처리, 음성
/음향 신호처리



오 상 엽

1999 : 광운대학교 이학박사.
1998 - 현재 : 경원대학교 IT대학
컴퓨터소프트웨어
교수
관심분야 : 소프트웨어공학, 비전관리,
소프트웨어재사용, 형상관
리, 객체지향, 음성인식,
분산처리, 음성/음향 신호
처리