한국어 음성인식 플랫폼의 설계*

권오욱(충북대학교), 김회린(ICU), 유창동(KAIST),
김봉완(원광대학교), 이용주(원광대학교)

<차 례>

1. 서 론
2. 음성인식 플랫폼 사례
3. 한국어 음성인식 플랫폼
3.1. 설계 방향
3.2. 요구 기능
3.2.1. 초보자용 기능
3.2.2. 전문가용 기능
3.3. 구 조
3.3.1. 신호처리 및 특징추출
3.3.2. 음향모델
3.3.3. 발음 사전
3.3.4. 언어모델
3.3.5. 탐 색
3.3.6. 후처리
3.4. 규 격
3.4.1. 성능 규격
3.4.2. 기능 규격
3.4.3. 입출력 규격
3.4.4. API 규격
4. 플랫폼 설계
4.1. 소프트웨어 구조
4.2. 소프트웨어 동작
4.3. 향후 계획
5. 결 론

<Abstract>

Design of a Korean Speech Recognition Platform

**Oh-Wook Kwon, Hoi-Rin Kim, Changdong Yoo,
Bong-Wan Kim, Yong-Ju Lee**

For educational and research purposes, a Korean speech recognition platform is designed. It is based on an object-oriented architecture and can be easily modified so that researchers can readily evaluate the performance of a recognition algorithm of interest. This platform will save development time for many who are interested in speech recognition. The platform includes the following modules: Noise reduction, end-point detection, mel-frequency cepstral coefficient (MFCC) and perceptually linear prediction (PLP)-based feature extraction, hidden Markov model (HMM)-based acoustic modeling, n-gram language modeling, n-best search, and Korean language processing. The decoder of the platform can handle both lexical search trees for large vocabulary speech recognition and finite-state networks for small-to-medium vocabulary speech recognition. It performs word-dependent n-best search algorithm with a bigram language model in the first forward search stage and then extracts a word lattice and rescores each lattice path with a trigram language model in the second stage.

* Keywords : Korean speech recognition platform

* 이 논문은 음성정보기술산업지원센터의 연구비 지원으로 휴먼인터페이스연구조합을 통하여 "한국어 음성인식 플랫폼 개발" 과제에서 수행한 내용입니다.

1. 서 론

최근에 음성인식기의 상용화 조류에 따라서 새로운 아이디어 및 관심을 가진 학생, 연구자 및 응용프로그램 개발자들이 늘어나고 있다. 기존에 공개된 음성인식기가 다수 있으나, 아직 음성인식에 대한 전문성을 갖추지 못한 사람들에게는 음성인식기의 내부구조를 파악하고 수정하여 자기 자신의 아이디어를 실현하기에는 공개된 음성인식기에서 제공하는 문서 및 관련 정보가 매우 부족한 실정이다. 이는 특히 새로운 아이디어를 가진 연구자의 음성인식 분야 진입을 어렵게 만드는 요인이 되기도 한다. 또한 최근에 한국내의 대학 및 연구소에서도 공통의 음성 데이터베이스를 이용한 한국어 음성인식 실험결과를 발표하고 있으나, 서로 다른 음성인식 플랫폼을 사용함으로써 새로운 알고리듬의 검증 및 각자의 시스템에 올바르게 활용하는데 어려움이 있다.

이러한 문제점을 해결하고자 교육 및 연구를 위하여 쉽게 이해할 수 있고 문서화가 잘되어 있는 한국어 음성인식 플랫폼을 설계하고자 한다. 내부구조에 대한 문서화가 갖추어진 공통 개발 도구는 연구역량을 핵심기술에만 집중할 수 있도록 하며, 대학 및 연구소에서 발표하는 음성인식 연구결과의 검증에도 활용될 수 있고, 공통 플랫폼을 사용함으로써 새로운 알고리듬의 구현이 용이하다는 이점이 있다. 음성인식에 친숙하지 않은 사용자도 쉽게 응용할 수 있게 함으로써 음성인식 관련 응용프로그램 개발을 자극할 수 있으며, 새로운 아이디어를 가진 신규 그룹의 진입 장벽을 낮출 수 있다.

본 논문에서 개발하고자 하는 음성인식 플랫폼은 ECHOS (Easy Compact Hangeul Object-oriented Speech recognizer)로서, 쉽고 작으면서 한글 처리가 가능한 객체기반의 구조를 갖는다. 이 플랫폼은 교육 및 연구를 위한 한국어 연속음성인식 플랫폼 개발을 통해 초보자, 숙련자 및 개발자가 공통된 플랫폼의 토대에서 인식성능 향상을 위한 알고리듬을 개발하고 비교할 수 있어서, 음성인식 기술의 발전에 기여할 것으로 기대한다. 프로그램은 객체 단위의 모듈로 설계를 하여 재사용이 용이하도록 하고 각 모듈이 독립적으로 사용되는 샘플 프로그램을 제공한다. 모듈에 대한 일관성 있는 프로그램 인터페이스 구조를 가지고 고수준 표준언어를 사용하여 프로그램을 작성한다. 그리고 산업적인 측면을 고려하여 고속 구현, 효율적인 구현에 대한 주석 및 방안을 상세히 기술한다. 기존의 소프트웨어와의 차별성을 위해 신호처리 부분을 강화하고, 한국어 처리가 가능하도록 하며, 성능이 입증되지 않는 부분은 삭제한다.

제2장에서는 공개된 음성인식기의 사례를 조사하고, 제3장에서는 한국어 음성인식 플랫폼의 설계 방향을 정하고 기존의 사례를 검토하여 시스템 규격을 정하고 시스템 설계과정을 기술하였다. 제4장에서는 향후 계획에 대하여 간략히 언급하고 제5장에서 결론을 맺는다.

2. 음성인식 플랫폼 사례

음성인식 분야의 진입장벽 감소 및 저변 확대를 위하여 외국의 대학을 중심으로 음성인식 소프트웨어를 공개하는 추세이다. 공통의 음성 데이터베이스와 공개 소프트웨어를 사용함으로써 새로운 아이디어를 누구나 쉽게 적용하여 검증할 수 있도록 함으로써, 최근의 음성인식률 포화를 돌파하여 음성인식의 새로운 전기를 마련하려는 의도를 품고 있다.

HTK (Hidden Markov Toolkit) [1]은 영국 켐브리지 대학에서 개발되어 한동안 상용 제품으로 판매하다가 2000년 9월부터 무료로 사용할 수 있는 소프트웨어로 전환되었다. HTK는 음성인식기의 훈련 및 테스트에 사용되는 사실상의 표준 공개 플랫폼으로서, C언어로 구현되어 있으며, 성능면에서도 NIST에서의 벤치마크 테스트에서도 좋은 성적을 나타낸 바 있다. 최근에는 휴대폰 환경에서의 분산음성인식을 위한 특징추출 알고리듬을 표준화하기 위한 ETSI의 Aurora-2 및 Aurora-3 프로젝트에서의 기준 인식기로 사용되었다.

Sphinx [2]는 CMU의 자원관리(Resource management) 태스크에 사용된 것으로서 오픈 소스 운동의 일환으로서 공개되어 음성인식 개발에 도움을 주고 있다. 최초로 성공적인 연속음성인식 기술을 선보였다는 점에서 널리 알려진 음성인식 플랫폼이다. C언어로 구현되어 있으며, 음성인식기의 훈련 및 인식에 사용되고 있다.

Mississippi 대학에서 만든 음성인식기[3]는 foundation class의 형태로 개발되어 일반적인 음성처리에 응용될 수 있도록 되어 있다. C++언어로 구현되었으며, 객체기반 연속음성 인식기의 훈련 및 인식을 위한 도구이다. ETSI의 Aurora-4 (5000단어 WSJ) 프로젝트의 기준 인식기로 사용되고 있다.

일본에서는 1996년부터 쿄토대를 중심으로 Julius [4]라는 2만 단어급의 대어휘 연속음성인식을 목표로 C언어로 구현된 음성인식기를 개발하였으며 최근에는 컨소시엄을 구성하여 인식기의 개발을 주도하고 있다. 이 컨소시엄은 일본 대학들에서의 대어휘 음성인식 연구개발을 지원하고 있다.

MGR은 HTK를 객체지향 프로그램으로 바꾼 버전으로서 인식부분의 핵심기능만을 구현한 것이다. 음향모델 및 언어모델의 기능에서도 아주 간단한 정도의 기능만을 지원하고 있다.

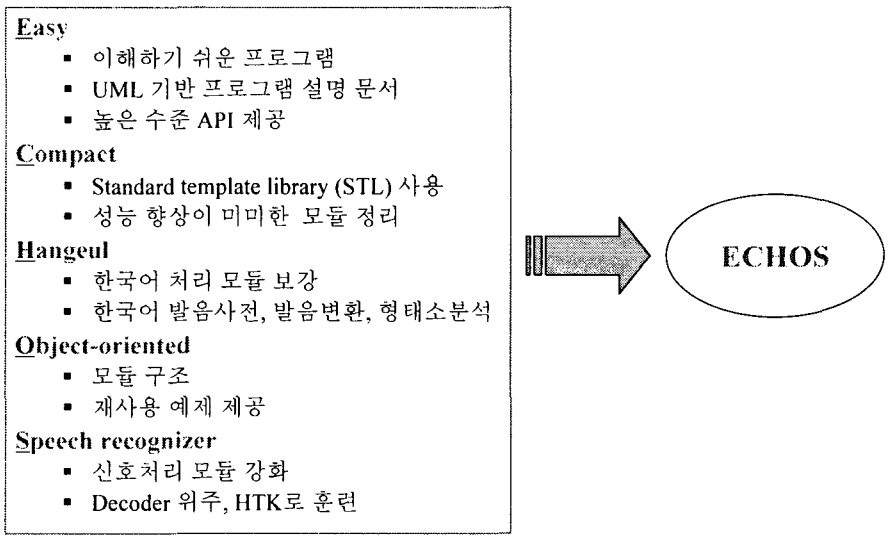
국내에서는 충북대학교에서 공개한 객체지향 인식기인 ezCSR [5]이 있다. C++언어로 구현되었으며, 탐색트리(search tree) 및 유한상태망(finite state network)으로 주어진 탐색 알고리듬을 모두 지원하고 있다. 한글 발음사전 생성 및 잡음제거 기능을 가지고 있다.

3. 한국어 음성인식 플랫폼

3.1. 설계 방향

한국어 음성인식 플랫폼은 쉽고 문서화가 잘되어 초보자도 쉽게 접근할 수 있도록 설계된다. 또한 최근의 연구동향을 파악하여 성능향상에 필수적인 모듈을 기본으로 제공하도록 한다. 사용자가 쉽게 자신의 알고리듬을 치환하여 검증할 수 있도록 객체 지향의 프로그램 구조를 갖도록 한다. 플랫폼의 일부 모듈을 음성인식이외의 다른 용도를 위하여 쉽게 가져다 쓸 수 있도록 각 모듈마다 독립적으로 사용할 수 있는 응용예제 프로그램을 제공한다. 고수준의 표준 라이브러리인 standard template library (STL) [7]를 사용함으로써 프로그램의 가독성을 높이고 알고리듬의 본질을 구현하는데 주력할 수 있도록 한다. 변수 및 함수 이름의 작명법, 프로그램 스타일 등을 통일함으로써 사용자들이 프로그램을 이해하는데 도움이 되고 혼동을 일으키지 않도록 한다.

문서화를 위하여 C++언어로 이루어진 소프트웨어의 문서화에 특히 적합한 unified modeling language (UML) [6]을 채택하였다. 플랫폼은 전체적인 구성도에 의하여 설명되며, 각 모듈은 다시 기능, 입출력 및 외부 모듈과의 인터페이스, 내부 구조의 순으로 작성된다. UML의 표준을 약간 수정하여 본 플랫폼의 매뉴얼 작성 기준으로 사용한다. <그림 1>은 이러한 설계방향에 따른 ECHOS의 특징을 나타낸다.



<그림 1> ECHOS의 특징

3.2. 요구 기능

ECHOS는 고립단어 인식, 연속음성인식, 음성 분할 기능을 수행할 수 있다. 동작모드는 사운드 카드로부터 직접 입력되는 음성을 인식하는 온라인 인식과 파일에 저장된 음성을 인식하는 오프라인 인식을 지원한다. 응용 프로그램의 개발을 위한 라이브러리, 음성인식 실험을 위한 도구, 음성 파일의 음소단위 분할기로서 사용 가능하다. 음성인식 기능은 사용자의 수준에 따라서 초보자용 기능과 전문가용 기능으로 나누어진다.

3.2.1. 초보자용 기능

- 인식기 제어: 시스템을 시작 및 종료시킨다. 음성 입력을 파일로 받을 것인지, 마이크를 통해 입력을 받을 것인지, 그리고, 입력 음성 데이터의 샘플링 주파수, 포맷 등 입력 조건을 제어한다.
- 인식 단어 제공: ECHOS가 인식해야 될 단어를 사용자가 직접 제공할 수 있다. 물론 제공되는 단어는 미리 주어진 사전에 있어야 한다.
- 인식결과: 입력 음성에 대해 하나의 인식결과를 나타낸다.

3.2.2. 전문가용 기능

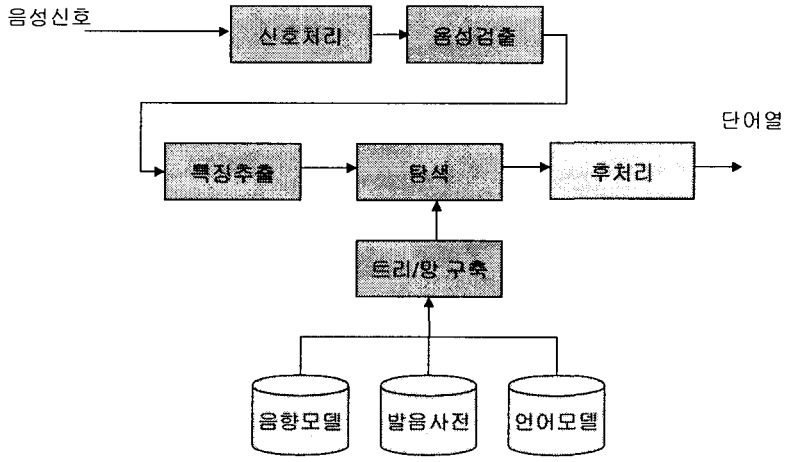
초보자가 사용할 수 있는 모든 기능뿐만 아니라 추가적인 작업 및 인식기의 정밀한 제어 또는 좀 더 자세한 인식결과를 얻을 수 있다.

- 인식기 제어: 음성입력 조건에 대한 정밀한 설정을 할 수 있다.
- 인식 단어, 문법: ECHOS가 인식해야 단어나 문법만을 제공할 뿐만 아니라, 사전에 단어를 추가하거나 변경할 수 있다.
- 음향모델, 언어모델 제공: 음향모델을 추가하거나, 새로 제공할 수 있으며, 언어 모델을 바꿀 수 있는 기능을 제공한다.
- 탐색 모듈 선택: 요구되는 인식기의 조건에 따라 인식의 핵심 모듈이 탐색 모듈을 선택할 수 있는 기능을 제공한다. 제공되는 탐색 모듈의 알고리듬에는 FSN 탐색과 tree 탐색 방식이 있다.
- 상세 인식 결과: 1-best 결과 외에 lattice 형태의 인식결과를 받아서 인식 결과에 대한 자세한 분석 및 1-best forced alignment 기능을 통해 단어 및 state 단위의 분할 정보를 얻을 수 있다.

3.3. 구 조

한국어 음성인식 플랫폼은 고립단어 인식, 연속음성인식, 음성 분할 기능을 수행할 수 있다. 동작모드는 사운드카드로부터 직접 입력되는 음성을 인식하는 온라인 인식과 파일에 저장된 음성을 인식하는 오프라인 인식을 지원한다. 응용 프로그램을 개발을 위한 라이브러리, 음성인식 실험을 위한 도구, 음성 파일의 음소단위 분할기로서 사용 가능하다.

플랫폼은 <그림 2>와 같이 신호처리 및 특징추출로 이루어진 전처리부, 음향모델, 발음사전, 언어모델, 탐색 모듈, 후처리 모듈을 가진다.



<그림 2> 한국어 음성인식 플랫폼의 구조

3.3.1. 신호처리 및 특징추출

입력된 음성신호로부터 특징을 추출하는 과정은 다음과 같다. 음성입력 모듈에서 음성입력 장치나 파일을 통해 들어온 음성신호를 신호처리 모듈로 넘겨준다. 신호처리 모듈은 넘겨받은 음성신호에서 잡음을 제거하거나 음성인식성능을 높이기 위한 신호처리를 한다. 음성검출은 신호처리 과정에 사용되기도 하고, 음성구간에서만 인식하기 위한 음성 구간 정보를 추출하기 위해 사용된다. 마지막으로 신호처리된 음성구간의 음성신호를 가지고 특징벡터를 추출하여 음성인식을 하게 된다.

3.3.2. 음향모델

음향모델은 시간적으로 변화하는 음성신호의 특징을 모델링한다. 음향모델링

방법은 HMM [8], 신경회로망(NN) 등이 사용되었으나, 최근에는 성능의 우수성, 유연성, 확장성 측면에서 유리한 HMM 기반의 음향모델이 대세를 이루고 있다. 외국에서 발표된 대부분의 음성인식 플랫폼에서도 HMM을 기반으로 하고 있다.

3.3.3. 발음 사전

다중 발음을 지원한다. 한글 단어에 대한 발음을 자동으로 생성할 수 있다. 발음사전 및 언어모델에서는 한글 표제어를 사용할 수 있도록 한다.

3.3.4. 언어모델

언어모델은 단어간의 문법을 고려하여 인식 후보에 가중치를 줌으로써 문법에 맞는 문장이 더 높은 점수를 얻도록 함으로써 인식률을 향상시킨다. 최적의 인식 단어열을 찾기 위한 탐색에서는 비교하여야할 후보의 개수를 줄이는 역할도 하게 된다. 인식되는 대상 어휘의 수와 인식 속도, 인식 성능을 고려하여 언어모델을 선택할 수 있다. 제공되는 언어모델에는 FSN, bigram과 trigram [9]이 있다. 음향모델, 언어모델과 발음사전을 이용하여 음성인식에 필요한 탐색공간을 형성한다.

3.3.5. 탐 색

명령어 인식 및 숫자음 인식과 같은 적은 어휘의 인식을 위한 FSN 형태의 탐색공간과 대어휘 인식과 빠른 인식을 위한 tree 형태의 탐색공간을 음향모델, 언어모델과 발음사전을 통해 형성한다. 형성된 탐색공간과 입력된 음성으로부터 구해진 특징벡터를 사용하여 인식을 한다. 또한 two-pass 탐색도 가능하다. 인식된 결과는 1-best 인식결과와 lattice형태의 인식결과를 얻을 수 있으며, lattice형태의 인식결과로부터 N-best의 인식결과를 얻을 수 있다.

3.3.6. 후처리

인식결과를 처리하여 인식 성능을 더욱 향상하거나 인식결과의 신뢰도를 계산하는 모듈이다.

3.4. 규 격

3.4.1. 성능 규격

음성인식 플랫폼은 30,000 단어까지 인식할 수 있다. 인식해야 하는 단어의 수, 인식속도와 인식성능에 따라 알맞은 음향모델 및 언어모델을 선택하여 사용한다. 플랫폼의 인식속도를 적절히 유지하기 위하여 태스크에 따라서 적당한 알고리듬을 선택하여 구현한다. 플랫폼의 메모리도 인식성능과 서로 타협관계에 있기 때문에 요구되는 메모리의 양에 따라 적당한 모델과 알고리듬을 선택하여 사용한다.

3.4.2. 기능 규격

음성인식 플랫폼의 기능은 신호처리 및 특징추출, 음향모델, 발음사전, 언어모델, 탐색 알고리듬, 한글 처리 등으로 나누어진다. <표 1>은 다른 공개 음성인식 소프트웨어와 비교한 ECHOS의 기능을 나타낸다. 표에서 '*'로 표시한 부분은 1차적으로 구현되며, 표시가 없는 기능은 차후에 우선 고려된다.

■ 신호처리 및 특징추출

- 잡음제거: ECHOS의 사용 환경에서 발생할 수 있는 주변 배경잡음과 입력장치에서 발생하는 채널잡음을 제거하여 환경에 강인한 음성데이터를 만든다. ECHOS에서 잡음을 제거하기 사용되는 잡음제거 알고리듬에는 spectral subtraction, Wiener filtering, ETSI 알고리듬이 있다.
- 음성검출: 입력신호로부터 음성만을 검출하는 기능으로써, 잡음제거를 위한 신호처리에서 사용되기도 하고, 또한 신호처리된 입력신호로부터 실제 음성구간의 정보를 얻기 위해 사용된다. ECHOS는 에너지 기반 음성 검출 알고리듬과 ETSI 기반 음성 검출 알고리듬을 제공한다.
- 특징추출: 플랫폼은 MFCC [10], ETSI [11], PLP[12] 특징을 추출할 수 있다.

■ 음향모델

- Continuous HMM: ECHOS에서는 continuous HMM을 사용하고, diagonal 또는 full covariance matrix를 사용할 수가 있다. HTK와 호환이 될 수 있는 포맷을 가진 음향모델을 사용할 수 있다. HTK에서 제공되는 모든 옵션을 지원하지는 않는다.
- State-tying: 형성된 음향모델에 대해 훈련된 음향모델이 부족하거나 유사한 확률분포를 가지는 state끼리 묶는 알고리듬을 제공한다.
- Decision tree: Decision tree를 사용하여 음향모델을 새롭게 갱신하거나 또는 탐색하는 과정에서 decision tree를 사용하여 음향모델을 선택할 수 있다.

<표 1> 다른 음성인식 소프트웨어와 비교한 ECHOS의 기능

Module	ECHOS-1.0	HTK-3.2	Julius-3.4.1	Sphinx-3	ISIP-r00_n11	EzCSR-1.03	MGR
Signal processing & Feature extraction	Spectral subtraction*, Wiener filtering*, ETSI* MFCC*, ETSI*, PLP* EPD_Energy*, ETSI* Channel comp : CMS, ETSI*	MFCC, PLP, VTLN, cepstral mean & variance normalization, variance scaling Energy-based speech/silence detection	MFCC_E_D_N_Z AIFF, AU, NIST, SND and WAV(ADPCM) Spectral subtraction Remove DC offset	MFCC, PLP, CMN	SoF (signal object file) LP, MFCC	MFCC, PLP CMS Speech detection Wiener filtering, Spectral subtraction	MFCC
Acoustic model	Continuous HMM* Covariance: Diag*, Full* State sharing* HTK compatible* Decision tree* Semi-continuous HMM Online speaker adaptation	Continuous/Semi-continuous/Discrete Diagonal/Full covariance Gaussian mixture HMMs Decision tree state-clustering Cross-word modeling Offline supervised SA using MLLR and MAP Online unsupervised SA using MLLR Two-model reestimation Global feature transform	HMM context dependent phoneme models (tri-phone) tied mixture and phonetic tied-mixture model Support model skip transition Support inter-word short pause handling Support binary HMM	Continuous and semi-continuous HMM Flexible feature vector Single or 4 streams Flexible HMM topology State tying with senones CART-based decision tree	Word, phone, word internal triphone, cross-word triphone Decision tree-based state-clustering Forced alignment	Continuous density HMM Diagonal/Full covariance Gaussian mixture HMMs Shared-state tri-phones Decision tree-based state sharing	Continuous HMM HTK format
Dictionary	Multiple pron.* Hangeul dictionary* Hangeul text-to-pron.* Hangeul morphology analysis	Multiple pron.	Multiple pron.	Multiple pron.		Multiple pron.	
Language model	FSN*, Bigram*, Trigram* Class N-gram, Keyword	Lattice-based grammar format Word-pair grammar Back-off bigram N-gram tool set, class n-gram	2-gram and reverse 3-gram(standard ARPA) Binary format Class N-gram	N-gram Statistical Language Modeling Toolkit	Network N-gram Can read SRI toolkit	Network Bigram Hangeul dictionary	FSN "compnet" Grammar generator
Search	FSN Search* Search tree* Lattice*→N-best list* Two-pass search* first pass : 2-gram and tree network search* second pass : 3-gram stack decoding* 1-best forced alignment* Utterance verification Confidence measure Dynamic vocabulary	Token passing algorithm Bigram or FSN Cross-word triphone models Lattice & N-best output Forced alignment Lattice post-processing, Lattice pruning, Finding 1-best, LM expansion	Two-pass strategy first pass : 2-gram and tree network search second pass : reverse 3-gram decoding stack decoding Gaussian Pruning Confidence measure	Flat decoder (slow): Pseudo-trigram Lextree decoder (fast): Any N-gram Subvector quant. based on Gaussian selection	Time-synchronous, Viterbi beam search N-best output Word-Graph (lattice) generation, word-graph rescoring Not support Phone, state-graph	One-pass dynamic programming Lexical tree search Network search Forced alignment	Viterbi token passing algorithm FSN Frame-synchronous beam search
Prog. Lang.	C++/STL	C	C	C++	C++	C++	C++
Systems	Linux, Windows	Unix/Linux, Windows, Cygwin	Linux, Solaris, Digital UNIX	Linux, Unix, Windows	Solaris, Linux, Cygwin	Linux, Windows	Linux, Windows
Comments	Decoder Object-oriented	De facto standard for training	Decoder, Control from client process via Network	First speech recognizer	Object-oriented	Decoder Object-oriented	Object-oriented Modularized
Limitations	Cross-word, Trigram forward search, Look-ahead, Gaussian selection	Dynamic vocabulary Multi-channel, multi-thread Phoneme look-ahead Gaussian selection			SVM, RVM, MLLR in next version Signal flow diagram	Cross-word, Speaker adaptation Dynamic vocabulary	

■ 발음사전

- 한글 발음사전: 인식대상 어휘에 대해 음향모델과 연결하기 위한 한글 발음사전이 제공된다. 대량의 어휘에 대한 발음사전을 구축하기 힘들기 때문에 발음사전 생성기를 통해 한글 발음사전을 생성하고, 한 어휘에 대한 다중 발음을 허용한다.
- 발음사전 생성기: 한글 어휘에 대한 자동적으로 발음기호로 표현해 주는 모듈이다. 한글 어휘에는 예외처리가 많고 다중 발음도 존재하기 때문에 기본적으로 발음사전 생성기에 의해 생성된 발음 기호에 대한 후처리가 필요하다.

■ 언어모델

- FSN: FSN(Finite State Network)는 인식하고자 하는 단어의 연결관계를 네트웍으로 표현하는 것으로서 자유도가 낮기 때문에 인식단어 수가 적은 인식기에 주로 사용된다.
- Bigram: 통계학적인 문법으로서 단어간의 연결관계를 확률로 표현한 문법이다. Bigram은 과거의 한 단어로부터 다음에 나타날 단어의 확률을 정의한 문법이다.
- Trigram: Trigram은 과거의 두 단어로부터 다음에 나타날 단어의 확률을 정의한 문법으로서 bigram보다 정교하고 인식성능을 높일 수 있으나 연산량이 많고 복잡하다.

■ 탐 색

- FSN 탐색: 인식대상 어휘가 작거나 인식속도가 느리더라도 인식성능에 중점을 두었을 때 사용되는 탐색 방식이다.
- Tree 탐색: 인식대상 어휘가 많거나 인식성능은 떨어지더라고 인식속도를 고려한 탐색방식이다.
- Two-pass 탐색: FSN 또는 search tree를 사용하든 먼저 bigram으로 일차적인 탐색을 하고 재차 trigram으로 stack decoding을 하여 보다 정확한 인식결과를 얻을 수 있다.
- 1-best forced alignment: 인식된 결과를 가지고 다시 강제로 인식된 결과를 다시 인식하여 단어, 음소, state단위의 분할 정보와 likelihood 값을 얻는다.

3.4.3. 입출력 규격

■ 음성입력

- Windows에서는 유/무선 마이크를 통해 직접 입력을 받거나 파일을 통해 입

력 받을 수 있으며, 운영체제(OS)가 Linux 또는 Unix인 경우에는 파일을 통해서만 입력 받을 수 있다.

- 8/16 kHz 샘플링 주파수, 샘플당 16 비트의 RAW 음성 입력 또는 8비트의 PCM 포맷을 지원한다. 압축된 음성입력은 지원하지 않는다.

■ 출 력

- 인식결과: 1-best 인식결과와 lattice 형태로 인식결과를 제공한다. lattice 형태의 인식결과로부터 backward tracking을 통해 N-best 인식결과를 얻을 수 있다.
- 단어 단위 likelihood 및 경계정보: 탐색과정에서 얻어지는 단어 단위의 경계(segmentation) 정보와 likelihood에 대한 정보를 얻을 수 있다.
- 1-best forced alignment: 인식된 결과 중 1-best 인식결과를 가지고 강제적으로 다시 인식하여 음소 또는 state 단위의 분할 정보와 likelihood 값을 얻을 수 있다.

3.4.4. API 규격

응용프로그램 작성을 위하여 독자적인 인터페이스 규격을 제공한다. 마이크로소프트의 SAPI는 지원하지 않는다. API 규격은 사용자의 수준에 따라서 두가지 단계로 제공된다. 상세한 API 규격은 향후 논문에서 기술한다.

4. 플랫폼 설계

4.1. 소프트웨어 구조

ECHOS 소프트웨어는 <그림 3>과 같이 다음과 같은 패키지로 구성된다.

■ SpeechRecog: 사용자 프로그램과의 인터페이스를 담당한다.

■ SearchBase: 탐색모듈. 인식에 필요한 모든 모듈을 관리하고, 탐색 알고리듬에 따라서 해당하는 탐색객체를 호출한다. 인식결과는 1-best 및 lattice이다.

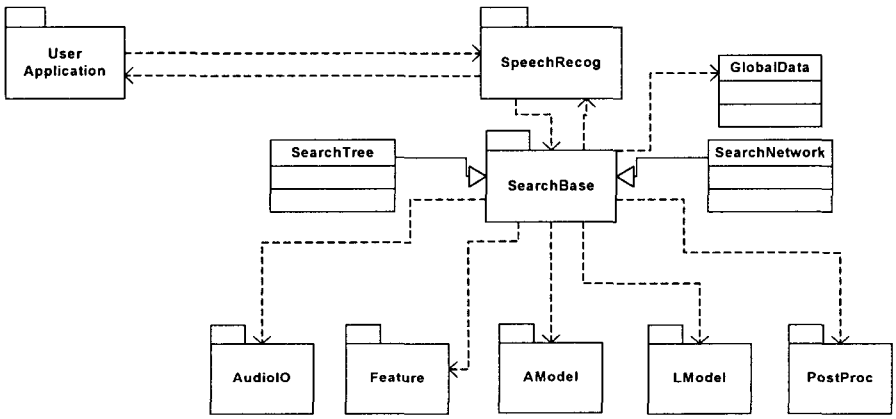
■ SearchTree: 대어휘를 위한 lexical 트리를 구성하여 탐색한다.

■ SearchNetwork: 소규모 또는 중규모 어휘를 갖는 음성인식을 위하여 FSN를 구성하여 탐색한다.

■ AudioIO: 사운드카드 또는 파일로부터 음성을 읽어 들인다. 대략적인 끝점검

출 기능도 동시에 수행된다.

- Feature: 입력신호로부터 잡음을 제거하고 특징을 추출한다. 정교한 끝점검출이 사용되어 탐색모듈에게 음성입력 완료를 알려준다.
- AModel: 음향모델을 읽어들이고, 입력된 특징에 대한 로그확률을 계산한다.
- LModel: 언어모델을 읽어들이고, 이전의 단어열이 주어질 때 현재단어의 로그확률을 계산한다.
- PostProc: Lattice입력에 대하여 다른 지식원을 사용하여 향상된 인식결과를 제공한다.

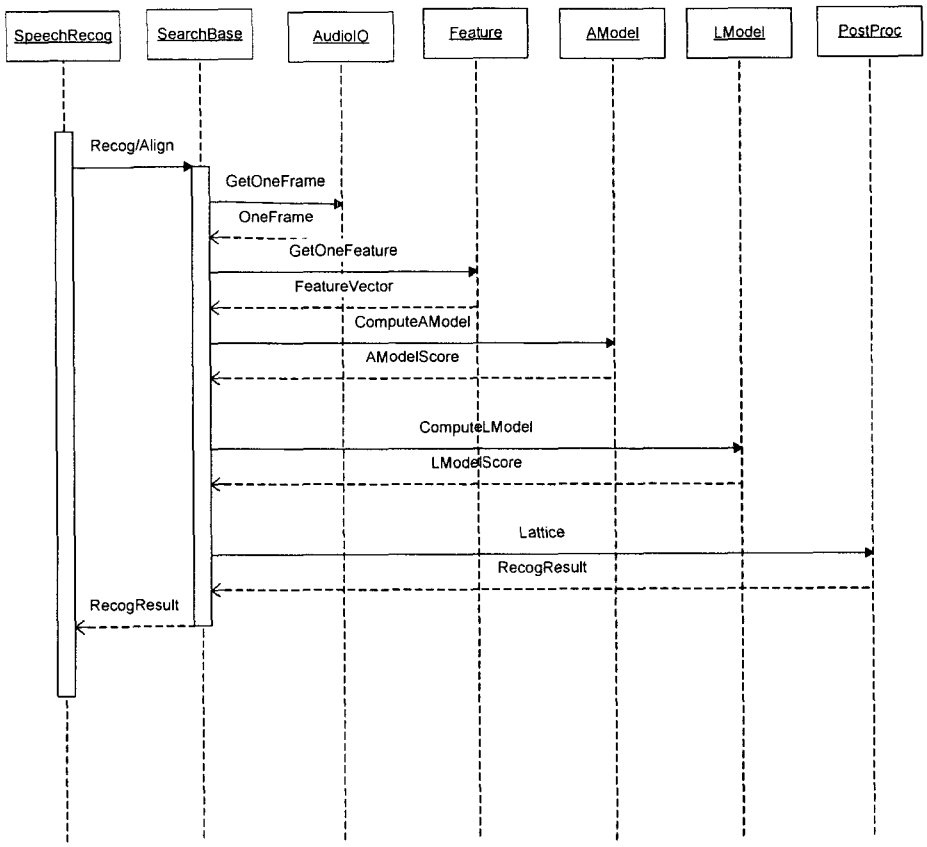


<그림 3> ECHOS의 소프트웨어 구조

4.2. 소프트웨어 동작

ECHOS는 <그림 4>와 같이 동작한다.

- SpeechRecog 모듈은 사용자 응용 프로그램으로부터의 요구사항에 따라서 적절히 탐색모듈을 호출한다.
- 탐색모듈은 프레임 단위로 관련 모듈을 순차적으로 호출한다.
- 탐색모듈은 오디오입출력 모듈로부터 한 프레임씩 읽어들여 특징추출 모듈로 전달한다. 특징추출 모듈에 특징추출로부터 한 프레임씩 읽어들여 탐색트리 또는 FSN에서 한 프레임씩 전진한다. 끝점이 검출되면 backtracking을 수행하여 lattice 또는 1-best 출력을 얻는다. Search 모듈은 lattice를 후처리 모듈에 전달하여 최종결과를 받는다.



<그림 4> ECHOS 시스템의 동작

4.3. 향후 계획

현재 한국어 음성인식 플랫폼의 기능규격 결정, 시스템 설계를 완료하였으며, 2005년 초까지 ECHOS의 기본적인 연속음성인식 모듈의 설계 개발을 완료할 예정이다. 각 모듈에 대한 검증 및 사용자에 대한 교육적 측면을 고려하여 독립적인 샘플 프로그램을 제공한다. 개발된 음성인식 플랫폼은 대학 및 연구소 등에 공개될 예정이며, 테스트 결과에 대한 피드백을 반영하여 플랫폼을 향상시킬 것이다.

국내의 음성인식 관련 전문가 회의를 통하여 요구기능, API규격, 개발방향, 검증방안, 활용방안, 활성화 방안에 대한 자문을 받아 플랫폼 개발에 반영 예정이다.

5. 결 론

본 논문에서는 교육 및 연구를 위한 한국어 음성인식 플랫폼의 개발 필요성을 언급하고, 그 목적에 맞는 플랫폼의 설계 방향, 시스템 규격, 소프트웨어 구조 등을 제안하였다. 개발되고 있는 플랫폼은 쉽고 작으면서 한글 처리가 가능한 객체 기반의 구조를 가지며, 30,000단어 정도의 연속음성인식 기능을 갖는다. 이 플랫폼은 국내 음성인식기술의 저변을 확대하고, 기술 수준을 향상시키고, 연구자들이 자신의 알고리듬 연구에 전념할 수 있는 토대를 마련하며, 음성인식기술의 비교 기준의 역할을 할 것으로 기대된다.

참 고 문 헌

[1] HTK Home page. http://hrk.eng.cam.ac.uk

[2] CMU Sphinx: Open Source Speech Recognition. http://www.speech.cs.cmu.edu/sphinx/Sphinx.html

[3] Automatic Speech Recogntion:Software. http://www.isip.msstate.edu/projects/speech/software/

[4] Multipurpose Large Vocabulary Continuous Speech Recognition Engine Julius. http://www.ar.media.kyoto-u.ac.jp/members/ian/doc

[5] ezCSR. http://speech.chungbuk.ac.kr/~owkwon/srhome/softwares/index.html

[6] Practical UML: A Hands-On Introduction for Developers- by Randy Miller. http://bdn.borland.com/article/0,1410,31863,00.html

[7] Standard Template Library Programmer's Guide. http://www.sgi.com/tech/stl/

[8] L. Rabiner and B.-H. Juang, Fundamentals of Speech Recognition, *Prentice-Hall*, 1993.

[9] F. Jelinek, Statistical Methods for Speech Recognition (Language, Speech, and Communication), MIT Press, 1999.

[10] S.B. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Trans.* ASSP, vol. 28, pp. 357-366, Aug. 1980.

[11] Aurora, Distributed Speech Recognition. http://portal.etsi.org/stq/kta/DSR/dsr.asp

[12] H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech," *Journal of the Acoustical Society of America*, vol. 87, no. 4, pp. 1738-1752, 1990.

접수일자: 2004년 8월 17일
게재결정: 2004년 9월 9일

▶ 권오욱(Oh-Wook Kwon)
주소: 361-763 충북 청주시 흥덕구 개신동 12번지
소속: 충북대학교 전기전자컴퓨터공학부
전화: 043) 261-3374
E-mail: owkwon@chungbuk.ac.kr

▶ 김회린(Hoi-Rin Kim)
주소: 305-714 대전광역시 유성구 문지동 103-6번지
소속: 한국정보통신대학원대학교(ICU)
전화: 042) 866-6139
E-mail: hrkim@icu.ac.kr

▶ 유창동(Changdong Yoo)
주소: 305-701 대전광역시 유성구 구성동 373-1번지
소속: 한국과학기술원(KAIST) 전자전산학과
전화: 042) 869-3470
E-mail: cdyoo@ee.kaist.ac.kr

▶ 김봉완(Bong-Wan Kim)
주소: 570-749 전북 익산시 신용동 344-2번지
소속: 원광대학교 음성정보기술산업지원센터(SiTEC)
전화: 063) 850-7452
E-mail: bwkim@sitec.or.kr

▶ 이용주(Yong-Ju Lee)
주소: 570-749 전북 익산시 신용동 344-2번지
소속: 원광대학교 음성정보기술산업지원센터(SiTEC)
전화: 063) 850-7451
E-mail: yjlee@wonkwang.ac.kr