화자 종속 알고리즘을 이용한 음성 인식 보안 시스템 구현

*김영현, 문철홍
광주 대학교
전화 : (062) 670-2652 / 팩스 : (062) 670-2191 / 핸드폰 : 011 - 628 - 0542

Implementation of Speech Recognition Security System Using Speaker Defendent Algorithm

*Young Hyoun Kim, Cheol-Hong Moon
Gwangju University
cosura@hanmail.net

Abstract

In this paper, a speech recognition system using a speaker defendant algorithm is implemented on the PC. Results are loaded on a LDM display system that employs Intel StrongArm SA-1110. This research has completed so that this speech recognition system may correct its shortcomings. Sometimes a former system is operated by similar speech, not a same one. To input a vocalization is processed two times to solve mentioned defects. When references are creating, variable start-point and end-point are given to make efficient references. This references and new references are changed into feature parameter, LPC and MFCC. DTW is excuted using feature parameter. This security system will give user permission under fore execution have same result.

1. 서 론

첨단 산업 분야에서 기술개발에 대한 경쟁이 치열해지고 있다. 따라서 핵심적인 기술과 전략 정보가 외부로 유출되는 것을 방지하기 위하여 여러 종류의 보안 시스템들이 사용되고 있다. 보안 기술의 발전과 더불어 복제 기술 또한 발전하고 있기 때문에 보안 시스템이 침해당하는 일이 자주 발생하고 있다. 이러한 보안 시스템의 침해를 막기 위한 많은 연구들이 진행되어왔고 이러한 연구 중 하나가 인간의 특징을 이용한 생체 보안이다.

생체 보안이란 인간의 신체 부위를 인식하는 보안 기술이다. 생체 보안이 가능한 부분 중 대체로 지문, 손, 얼굴, 음성, 홍채 이렇게 다섯 가지가 많은 연구에 사용된다. 이 다섯 가지 중 음성은 인간이 가지고 있는 기본적인 능력 중 가장 보편적이고 편리한 정보 전달의 수단이다. 음성에 의해 표현되는 말은 인간과 인간 사이의 의사소통 수단뿐 만 아니라 인간의 음성을 이용하여 인간과 기계와의 통신으로 장치들을 동작 시키는 수단으로서도 중요한 역할을 한다.

그러나 인간의 음성을 음성 인식 보안 시스템에 적용하여 음성의 파형만을 보고 분석하기에는 불규칙한 부분이 많고, 매번 발음할 때마다 동일하지 않기 때문에 단순한 파형만을 비교해서는 각각의 음성들을 구별해 내기 쉽지 않다. 그래서 음성의 변화에 대한 강인하고 명확한 특징을 제공하기 위해 음성이 가지고 있는 고유의 성질만을 추출해서 이용한다. 추출된 고유의 성질을 특징 파라미터라 하고, 그 종류로는 선형 예측 코딩(Linear Prediction Coding), 켑스트럼 계수(Cepstrum Coefficient), 인지 선형 예측(Preceptual Linear Prediction) MFCC(Mel-Frequency Cepstrum Coefficient)등이 있다.

본 논문에서는 선형 예측 코딩과 MFCC을 이용하여 동적 시간 정합(DTW)을 적용하였으며, 외부로 연결된 마이크를 통해서 PC 상에서 음성인식을 구현하였고, 음성 인식 보안 시스템의 단점인 비슷한 발성에 대해서는 시스템이 동작되는 것을 방지하기 위한 방법으로 동일한 발성을 연속적으로 두 번 발성하여 동작하는 시스템을 구현하였다. 또한 RISC 프로세서인 ARM SA-1110을 이용하여 PC 상에서 구현된 음성 인식 시스템의 결과를 USB 또는 Serial 통해 전송하고, 전송된 데이터는 LDM에 시각적으로 표현되게 구성해 보

았다.

본 논문의 구성은 다음과 같다. 본론에는 ARM SA-1110을 이용한 하드웨어 구성과 하드웨어 구현 소프트웨어 및 화자 종속 알고리즘을 이용한 음성 인식 보안 시스템에 관한 실험 및 결과에 대해서 논하고, 마지막으로 결론을 도출했다.

2. 음성 인식

음성 인식은 발음에 따라 분절음 인식과 연속음 인식으로 나뉘고 분절음 인식은 고립단어 인식과 연결 단어 인식으로 나누어지고 인식 대상에 따라 화자 종속과 화자 독립으로 나뉜다. 본 논문에서는 화자 종속 고립단어 음성 인식을 이용하였다. 고립단어에 유용하게 이용되는 알고리즘이 DTW이다.

DTW는 warping 함수의 기울기에 대한 제한을 두고 있다. 음성은 같은 발음에 대해서 지나치게 많은 차이가 나지 않기 때문에 기울기의 제한을 두어 인식률을 높인다. DTW의 제한조건은 끝점 제한, 단조 제한, 국부 경로 제한, 전역 경로 제한의 조건을 가진다. 끝점 제한 조건은 입력 음성 패턴의 시작점과 참조 패턴의 시작점과 일치하고 입력 음성 패턴의 끝점은 참조 패턴의 끝점과 일치해야 하는 조건이며, 단조 제한 조건은 최적 경로는 항상 단조 증가해야 한다. 국부 경로 제한 조건은 격자상의 한 노드에 도달하기 위한 경로에 제한을 두어 시간 상 지나치게 수축되거나 팽창되는 것을 방지한다. 마지막으로 전역 경로 제한 조건은 동일 음성이 발성될 때 지속 시간의 차이가 1/2에서 2배를 넘지 않는 가정 하에 입력 음성 패턴과 참조 패턴간의 전 구간에 걸친 허용 가능한 영역을 제한하여 탐색시간을 줄이는 조건이다.

DTW의 초기화 조건 : $g_1(c(1)) = d(c(1)w(1))$ (1)

동적 프로그래밍 식 :

$$g_k(c(k)) = \min_{c(k-1)} [g_{k-1}c(k-1) + d(c(k))w(k)] \quad (2)$$

시간 축으로 정규화된 거리 : $D(A,B) = \frac{1}{N} g_k(c(k))$ (3)

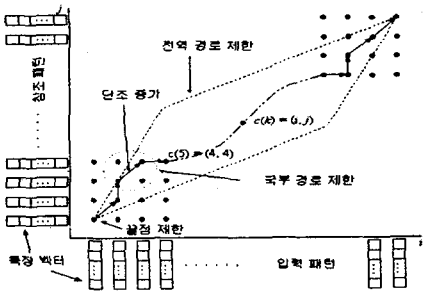


그림 1. DTW 제한 조건

그림 1은 DTW를 하기 위한 제한 조건들을 보여준다. 제한 조건 중 국부 경로 제한의 일부분을 표 1에서 보여준다.

초기화 조건 : $g(1,1) = 2d(1,1)$ (4)

동적프로그래밍 :

$$g(i,j) = \min \begin{cases} g(i,j-1) + d(i,j) \\ g(i-1,j-1) + 2d(i,j) \\ g(i-1,j) + d(i,j) \end{cases} \quad (5)$$

시간축 정규 화된 거리 : $D(A,B) = \frac{1}{N} g(I,J)$ $where \quad N = I + J$ (6)

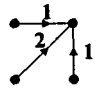
	국부경로제한	동적 프로그램 식
Type I		$g(i,j) = \min \begin{cases} g(i-1,j) + d(i,j) \\ g(i-1,j-1) + 2d(i,j) \\ g(i,j-1) + d(i,j) \end{cases}$
Type II		$g(i,j) = \min \begin{cases} g(i-2,j-1) + 3d(i,j) \\ g(i-1,j-1) + 2d(i,j) \\ g(i-1,j-2) + 3d(i,j) \end{cases}$

표 1 국부 경로 조건에 따른 warping 함수 및 동적 프로그램 식

3. 시스템 설계

3.1. 하드웨어 설계

3.1.1 ARM SA-1110

그림 2는 Intel Strong ARM SA-1110 마이크로프로세서의 내부 블록도를 보여준다.

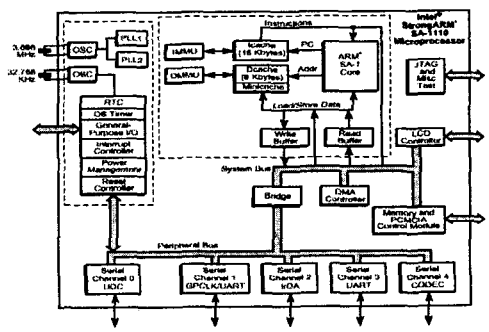


그림 2. SA-1110의 내부 블럭도

3.1.2 GPIO

SA-1110을 이용하여 LDM을 구동을 시키기 위해서 SA-1110의 GPIO에 연결을 시킨다. GPIO Control block내에는 8개의 레지스터가 있다. 한 가지는 핀의 상태를 moniter 한 것이고(GPLR), 두 가지는 pin의 상태를 제어하기 위한 것(GPSR)이며 하나는 pin의 방향

(input 또는 output)을 제어하기 위한 것(GPDR)이고 두 가지는 detected 되어야 될 pin의 edge type을 규정하는 것(GRER, GFER)이다. 다른 하나는 pin에서 정해놓은 edge type이 검출 되었을 때 flag하는 것(GEDR)이다. 마지막 하나는 pin이 일반적인 GPIO로서 사용될지 alternate function으로 사용될지 알려주는 것(GAFR)이다. GAFR과 GPDR을 제외한 레지스터는 Reset후에 어떤 값인지 알 수 없으며 프로그램에서 초기화해야 한다.

3.1.2.1 USB Device Controller (UDC)

SA-1110 UDC는 대부분 USB 1.1을 따르고, Host가 내보낸 Standard device requests를 지원한다.

3.1.2.2 USB Operation

SA-1110이 Reset되거나 USB케이블이 SA-1110에 연결되면, SA-1110은 자동으로 Endpoints들을 구성한다. 그리고 Host는 그때 SA-1110에 유일한 Address를 할당하고, SA-1110 UDC는 Host의 Control상에 놓이게 된다. SA-1110 UDC는 Host에 의해서 SA-1110 UDC의 Endpoint 0으로 보내지면 commands(control transactions)에 응답한다. Host는 "Bulk OUT" data Frame들을 Endpoint 1에 전송한다. Host는 "Bulk IN" Data Frame들을 Endpoint 2로부터 받는다.

3.2 소프트웨어 설계

Windows 2000 환경에서 VC++ 6.0을 이용하여 DTW 및 특징 파라미터 추출에 관한 알고리즘을 구현하는 프로그램을 작성 및 디버깅하였다.

종래의 음성 인식 시스템은 유사한 음성에 대해 구동되는 경우가 종종 발생하였다. 이를 방지하기 위해 연속적인 동일한 음성 레퍼런스를 받아 DTW를 수행하게 하였다. 또한 음성 파라미터의 추출 시간을 줄이기 위해 특징 파라미터 추출을 병렬 처리하였다.

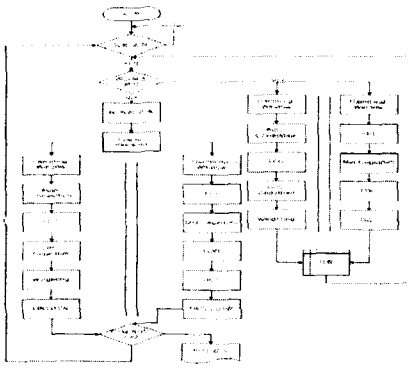


그림 3. PC상 음성 인식 구현

그림 3는 음성 인식 구현을 위한 순서도를 보여준다.

4. 실험 및 결과

본 논문에서는 구현한 음성인식 알고리즘이 하드웨어와의 연동성을 가지게 하기 위해서 아래 그림 4와 같은 실험환경을 설정하였다.

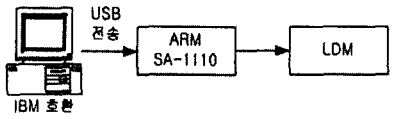


그림 4. 실험 환경

Windows 2000 운영 체제를 기반으로 한 PC환경에서 입력받은 음성데이터의 레퍼런스에 의해 인식된 음성 데이터를 USB 인터페이스를 통하여 SA-1110 보드로 전송하여 LDM에 디스플레이 하여 음성인식의 결과를 나타내도록 하였다. 훈련되어지지 않은 사람에게서는 연속적인 같은 단어에 대해 발성 할 경우 똑같은 발성을 하기 힘들다. 따라서 연속적으로 발성을 하게 하여 특징 파라미터를 추출하였으며, 음성인식의 신뢰성을 위하여 목을 모델링한 LPC와 귀를 모델링한 MFCC를 이용하여 각각의 특징 파라미터를 추출하였다.

식 (4), (5), (6)를 이용하여 음성인식 알고리즘을 구현을 하였으며, Time warping으로 DTW의 누적거리를 측정 판별할 수 있도록 프로그램 하였다. 그림 5에서는 누적거리 측정에 대한 결과를 나타내었다.

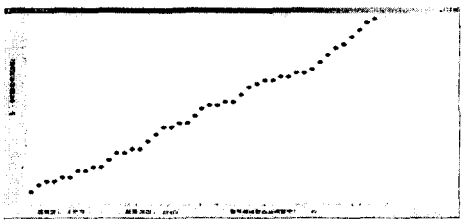


그림 5. DTW Warping

Warping에 의해서 출력된 결과들은 그림 6의 보안 시스템 프로그램에 적용된다. 그림 6은 보안 시스템 구동 순서도를 보여준다.

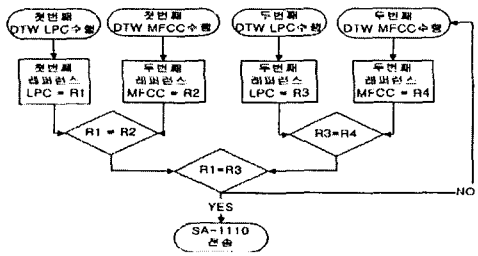


그림 6. 보안 시스템 구동 순서도

또한, 음성인식결과는 PC상태에서도 확인할 수 있게 하여 하드웨어와 PC상태의 음성인식 결과를 비교해 보았다. 그림 7은 PC상태에서 음성인식 결과를 보여주고, 그림 8은 PC상에서 음성인식 결과를 SA-1110으로 전송되어 표현된 대한 실물사진을 나타내었다.

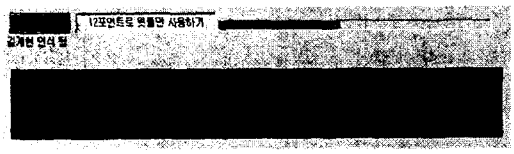


그림 7. PC상의 음성 인식 결과

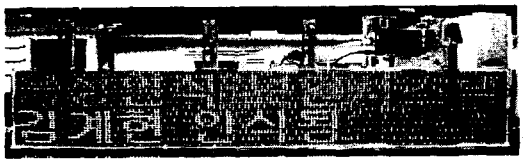


그림 8 LDM에 인식된 레퍼런스 데이터 출력

SA-1110을 통해서 LDM을 구동하기 위해서는 LDM을 초기화를 해야 한다. 초기화를 위하여 GPIO와 관련 레지스터들을 설정을 해주어야 하며 USB의 관련 레지스터도 초기화를 해야 한다.

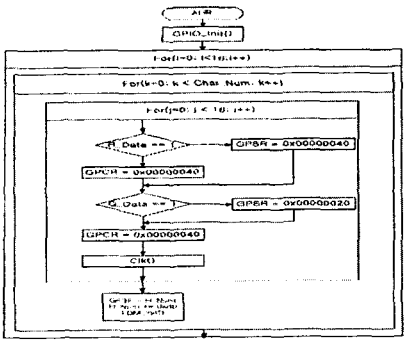


그림 9. SA-1110 LDM 구동 순서도

그림 9는 SA-1110에서 LDM을 구동하기 위한 순서도를 보여준다. 그림 10은 본 논문에서 이용된 실물의 전체 모습을 보여준다.

그림 10. 실제 구현된 시스템

5. 결 론

본 논문에서는 마이크를 통하여 음성데이터를 PC로 입력하고, PC에서 화자 종속 알고리즘을 이용하여 음성의 특징을 추출하여 인식 데이터를 검출하였으며, PC에서 구현된 보안 시스템의 데이터는 USB 또는 Serial을 통해 LDM으로 전송되어 시각적으로 표현할 수 있는 음성 인식 보안 시스템을 연구하였다.

PC상에서의 인식되어진 음성에 대한 보안 시스템이 약 97%이상의 인식률을 보였다. 보다 높은 인식률을 위해서는 지향성 마이크를 사용하고, 특징 파라미터를 추출할 때 환경잡음을 차단하여 음성의 질을 높이고, 발성자가 가변적으로 설정된 시작점 계수와 끝점 계수를 이용해 최적의 레퍼런스를 생성하므로써 각각의 레퍼런스들에 대한 특징 파라미터를 더 정확히 추출한다면 인식률이 개선될 것이다.

본 논문에서는 PC상에서 USB를 이용하여 SA-1110으로 데이터를 전송하여 LDM에 표현하고 음성 인식 보안 시스템으로써의 가능여부를 알아보았다. 하드웨어를 Display인 LDM으로 구성하였지만, 보안에 이용되는 하드웨어와 연동을 시킨다면 음성 인식 보안 시스템으로 활용할 수 있을 것이다.

참고문헌

[1] Claudio Becchetti and Lucio prina Ricotti, "Speech Recognition Theroy and C++ implementation", JOHN WILEY & SONS, 1999.

[2] L.R.Rabiner/R.W.Schafer, "Digital Processing of Speech Signal", PRENTICE HALL, 1978.

[3] Lawrence Rabinter and Biing-Hwang, "Fundamentals Of Speech Recognation", Prentice Hall, 1993

[4] 오영환, "음성 언어 정보 처리", 홍릉과학 출판사. 1998

[5] 박경범, "음성의 분석 및 합성과 그 응용", 그린, 1997

[6] 안현순, "터버 C로 구현한 과학기술 계산 프로그래밍", 가남사, 1993

[7] 김형훈, "USB GUIDE", OHM사, 2002

[8] Intel, "Intel StrongARM SA-1110 Microprocessor Develper's manual", 2001

[9] Compaq, HP, "Universal Serial Bus Specification 2.0", 2000

[10] Coulter, "digital AUDIO Processing", R&D, 2000

[11] 임창환, "DSP 음성인식과 초음파 센서를 이용한 자율주행로봇 구현", 석사학위 논문, 광주대학교 2000, 2

[12] 김봉춘, "DSP 음성 분석을 이용한 연주 시스템 구현", 석사학위 논문, 광주대학교 2002, 2

[13] 김성수, "임베디드 시스템을 이용한 공장제어 시스템 구현",광주대학교 2003, 2