

工學碩士 學位論文

도어락에 適用한 話者 從屬形 音聲
認識 시스템 具現에 관한 研究

A Study on the Implementation of Speech
Recognition System for Speaker Dependent applied
to Doorlock

指導教授 李 尙 培

2006年 2月

韓國海洋大學校 大學院

電子通信工學科

鄭 聖 勳

本 論 文 을 鄭 聖 勳 의
工 學 碩 士 學 位 論 文 으 로 認 准 함

委 員 長 梁 圭 植 印

委 員 朴 東 國 印

委 員 李 尙 培 印

2006年 2月

韓 國 海 洋 大 學 校 大 學 院

電 子 通 信 工 學 科

목 차

Abstract	i
제 1 장 서 론	1
제 2 장 화자 인증 이론	3
2.1 잡음 처리 방법	3
2.2 끝점 검출 알고리즘	13
2.3 음성 특징정보 추출	16
2.4 화자 인식 알고리즘	19
제 3 장 음성 인식 시스템의 설계 및 구현	22
3.1 하드웨어 구성 및 특징	22
3.2 소프트웨어 구성	27
제 4 장 실험 및 고찰	31
4.1 잡음 처리된 화자 인증 실험 결과	33
4.2 FAR 및 FRR 화자 인증 테스트 결과	37
제 5 장 결론	40
참 고 문 헌	41
부 록	43

Abstract

Among four common methods (DTW, VQ, HMM and GMM) to implement the speech recognition system, although GMM is verified as the best method in aspect of performance, there are too many problems discovered to be applied to the real system.

In this research, therefore, we suggested to implement the embedded system using the DTW algorithm, which is simple and high performance, for real-time speech recognition. We used the RLS (Recursive Least Square) adaptive filter among various noise handling methods of the speech recognition system to have more accurate calculation as well as better convergence speech.

This algorithm runs a variety of calculations whenever parameters vary, reduces the amount of necessary calculations and converges to the best possible parameters. Also, we used the fact that the convergence speed is fast. In this research, TI's DSP (TMS320C32) was used for large amount of speech data handling, various noise handling as well as feature extraction.

First, speech data that is inputted from microphone extract features of speech through RLS algorithm, pre-emphasis, hamming window, MFCC (Mel-Frequency Cepstrum Coefficient) after speech area was detected by short time energy. The extractive features of speech area are stored to database and achieves recognition process by DTW (Dynamic Time Warping) algorithm

using speech features at speaker recognition process.

A total of three people pronounced isolated words twice and each person performed 100 tests while using the speech recognition system with the DTW. The result gave 97% at average.

There are two forms to estimate the recognition rate: the first one is FAR (False Acceptance Ratio) which is the error of accepting an impostor and the second one is FRR (False Rejection Ratio) which is the error of rejecting a legitimate user.

Therefore, in order to commercialize the system in reality, minimizing these ratios increases the security of the system and can get the reliance in aspect of stability.

In this research, each of three people performed the pronunciation test 30 times, and the result is that FRR is 5.56% at average while FAR is 1.1% at average.

In this research, we showed the possibility that the SITK-32C board in which the DTW recognition algorithm and the DSP (TMS320C32) are built can be applied to the real life through the test of the real persons' speech command recognition. As the result of the recognition test on PC using the DTW as the recognition algorithm for a speaker dependent system, we obtained almost 98% recognition rate.

제 1 장 서 론

오늘날 생체 보안 산업은 실생활에서 널리 적용되고 있으며, 적용 분야도 점점 확대 되고 있는 실정이다. 이러한 생체 산업은 유비쿼터스 사회와 연결되어 더욱 더 활발히 발전할 것으로 추정된다.

생체인식 기술이란 인간의 생체적 특성상 개인만의 독특한 특징을 본인확인을 위한 측정단위로 활용하며, 생체적 특징을 자동화된 장치를 거쳐 분석하여 본인 확인을 하는 기술이다. 생체인식 기술에 활용되는 생체 부분은 지문, 망막, 홍채, 얼굴, 손, 정맥, 목소리, 필체, 몸 냄새, DNA 등이 있다. 생체인식 기술은 남이 나를 대신 할 수 없는 가장 완벽한 본인확인 수단이다. 신분증이나 열쇠, 카드, 비밀번호 등과 같은 인식 수단은 복제, 분실, 망각, 노출, 도용, 변형 등이 될 수 있지만 생체인식은 한 차원 높은 인식 수단으로 그 어떤 것도 대신할 수 없는 완벽한 방법을 제공한다.

본 논문에서는 이러한 생체 인식 중 음성인식 분야(화자 인증)의 연구를 통해 보안 기능의 시스템(자동문)을 구현하는데 그 목적이 있다. 이는 사람의 목소리가 각기 다르다는 점을 이용한 방식으로써 사용자의 ID를 음성이라는 편리한 정보전달 수단을 기계에 인식하는 방법과 이를 인식하는 방법을 이용한 것이다.

여기서 우리는 화자 인증을 살펴보면, 화자 인증에는 크게 2가지로 분류할 수 있다. 첫 번째가 화자 검증(Speaker Verification)이며, 두 번째가 화자 식별(Speaker Identification)이 있다. 첫 번째 방식은 1:1 매칭 기법으로 ID와 함께 음성을 입력해서 데이터베이스에 ID에 저장되어 있는 음성 패턴을 찾아 매칭도를 구해 승인 또는 거

절을 행하는 과정이며, 두 번째 방식은 1:N 방식으로 등록된 화자들 중에서 가장 유사한 화자를 골라내는 것을 말한다.^{[11]~[31]}

이에 본 논문에서는 화자 식별을 구현하는데 초점을 맞추었으며, 화자 종속형(Speaker Dependent)으로 개발하였다.

하지만 화자 인증 시스템의 실용화 측면에서 볼 때, 가장 문제화 되는 점은 잡음 처리와 시스템의 소량화 문제이다. 이에 본 논문에서 제안된 점은 잡음 처리를 위해서 적응 필터^{[4],[5]}를 사용하여 잡음을 현저히 줄여 인식률을 올렸으며, 아울러 시스템의 소형화를 위해 DSP(TMS320C32) 프로세서^{[5]~[7]}를 사용하여 임베디드 시스템화^{[9]~[11]}시켰다.

본 논문의 구성은 다음과 같다.

2장에서는 화자 신호 처리의 일반적인 내용과 수많은 음성 데이터에서, 특징 추출 값을 뽑아내는 방법과 화자 인식 알고리즘을 설명한다. 3장에서는 앞장에서 설명되었던 이론을 배경으로 실제 화자 인식 시스템 설계 및 구현을 서술하며, 4장에서는 화자 인증 테스트 결과를 보여주고, 마지막으로 5장에서 결론을 짓는다.

제 2 장 화자 인증 이론

2.1 잡음처리 방법

잡음이 존재하는 환경에서는 음성 인식 시스템의 인식률은 현저히 차이가 난다. 이처럼 잡음이 음성 인식 시스템의 성능을 저하시키는 가장 큰 원인은 끝점 검출 오차에 의한 영향과 잡음에 의한 특징 벡터의 왜곡이다. 본 음성 모듈에서는 잡음환경에 강한 시스템을 구현을 하기 위해 첫 번째로 전제되어야 할 부분이 잡음 처리하여 깨끗한 음성을 사용하기 위함이다.

본 절에서는 트랜스버설 FIR 필터 및 점화적 최소자승 알고리즘^{[4],[5]}에 관해 살펴본다.

2.1.1 트랜스버설 FIR 필터

트랜스버설 FIR 필터를 이용한 적응필터를 그림 2.1과 같이 나타낼 수 있다. 필터 응답 $y(n)$ 은 입력신호를 $x(n)$, 필터의 계수를 $h(k)$ ($k = 0, 1, 2, \dots, M-1$)로 하면,

$$y(n) = \sum_{k=0}^{M-1} h(k)x(n-k) \quad (1)$$

식 (1)과 같이 주어진다. 여기서 필터 계수 $h(k)$ 는 시스템 특성을 규정하는 파라미터이고, 출력 $y(n)$ 은 이상 응답 $d(n)$ 에 접근하도록 과

라미터의 값을 조정한다.

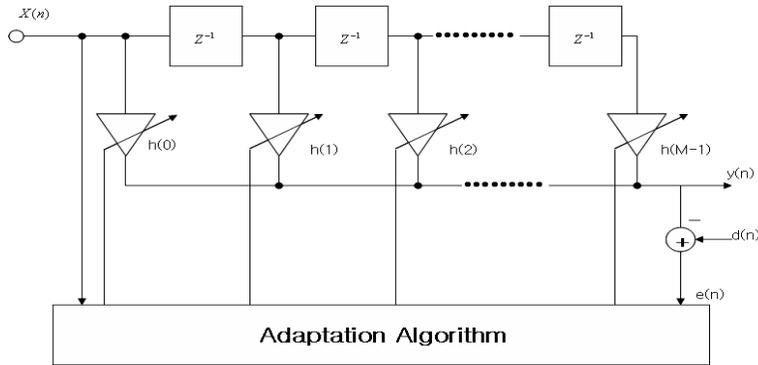


그림 2.1 트랜스버설 FIR 필터 개요도

Fig 2.1 The configuration of an transversal FIR filter

파라미터 조정을 위해서는 이상 응답 $d(n)$ 과 실제 응답 $y(n)$ 이 얼마나 근사한가를 평가할 척도가 필요하다. 여기서는 이상 응답과의 오차신호 $e(n)$ 는 식(2)와 같다.

$$e(n) = d(n) - y(n) \quad (2)$$

여기서, 오차 신호의 자승평균치(Mean Square Error)는

$$MSE = E[e^2(n)] \quad (3)$$

식(3)과 같이 나타나는데, MSE 최소가 되도록 필터 계수를 결정하는 문제를 생각한다. 여기서 $E(x)$ 는 x 의 기대치이다. 식(3)에 식(2)를 대입하여 정리하면,

$$\begin{aligned}
MSE &= E[e^2(n)] \\
&= E[d^2(n)] - 2E[d(n)y(n)] + E[y^2(n)] \\
&= E[d^2(n)] - 2E[d(n) \cdot \sum_{k=0}^{M-1} h(k) \cdot x(n-k)] \\
&\quad + E[(\sum_{k=0}^{M-1} h(k)E[d(n)x(n-k)])^2] \\
&= E[d^2(n)] - 2 \sum_{k=0}^{M-1} h(k) \cdot E[d(n) \cdot x(n-k)] \\
&\quad + E[(\sum_{k=0}^{M-1} h(k) \cdot x(n-k)) (\sum_{m=0}^{M-1} h(m) \cdot x(n-m))] \\
&= E[d^2(n)] - 2 \sum_{k=0}^{M-1} h(k) \cdot E[d(n) \cdot x(n-k)] \\
&\quad + \sum_{k=0}^{M-1} \sum_{m=0}^{M-1} h(k)h(m)E[x(n-k)x(n-m)]
\end{aligned} \tag{4}$$

식(4)가 된다. 식(4)의 제 1항은 이상응답의 자승평균치(전력)와 같다. 이것을 P_d 라 하자. 즉,

$$P_d = E[d^2(n)] \tag{5}$$

또한, 제 2항의 기대치는 이상 응답과 필터 입력신호의 상호 상관함수이다. 이것을 $R_{dx}(k)$ ($k = 0, 1, \dots, M-1$)로 표시하는데, 식(6)과 같다.

$$R_{dx}(n) = E[d(n)x(n-k)] \tag{6}$$

그리고 제 3항의 기대치는 입력신호 $x(n)$ 의 자기상관함수이다.

$$R_{xx}(m-k) = E[x(n-k)x(n-m)] \quad (7)$$

식(7)을 식(4)에 대입해서 정리하면, 식(8)과 같이 나오게 된다.

$$\begin{aligned} MSE = P_d - 2 \sum_{k=0}^{M-1} h(k)R_{dx}(k) \\ + \sum_{k=0}^{M-1} \sum_{m=0}^{M-1} h(k)h(m)R_{xx}(k-m) \end{aligned} \quad (8)$$

여기서 오차(MSE)는 필터 계수 $h(n)$ 의 2차 함수로 표현되어 2차 곡면으로 된다. 이 곡면을 오차 특성 곡면(Error Performance Surface)이라 한다.

오차 (MSE)를 최소로 하는 필터계수 $h(n)$ 에 대한 도함수를 0으로 하는 것이다. 그러므로 MSE의 최소치를 부여하는 최적 필터계수를 구할 수 있다.

식(8)을 $h(n)$ 로 미분하면,

$$\frac{\partial MSE}{\partial h(k)} = 2 \sum_{m=0}^{M-1} h(m)R_{xx}(m-k) - 2 \sum_{k=0}^{M-1} R_{xx}(k) \quad (9)$$

로 된다. 따라서 최적의 필터계수는 M원 연립방정식

$$R_{dx}(k) = \sum_{m=0}^{M-1} h(m)R_{xx}(m-k) \quad (10)$$

의 해로서 구할 수 있다. 이러한 방정식은 이미 알려져 있는 Wiener-Hope 방정식이 이산시간영역 표현에 대응하여 정규방정식 (normal equation)이라 부르고 있다. 여기서 필터계수 벡터를

$$H = (k(0), h(1), \dots, h(M-1))^T \text{로 두고,}$$

상관벡터를 $P = (R_{dx}(1), R_{dx}(0), \dots, R_{dx}(M-1))^T$, 상관행렬 R 을

$$R = \begin{bmatrix} R_{xx}(0) & R_{xx}(1) & \dots & R_{xx}(M-1) \\ R_{xx}(1) & R_{xx}(0) & \dots & R_{xx}(M-2) \\ \dots & \dots & \dots & \dots \\ R_{xx}(M-1) & R_{xx}(M-2) & \dots & R_{xx}(0) \end{bmatrix} \quad (11)$$

로 두면 식(9)에서 기울기는 다음과 같이 표현할 수 있는데,

$$\begin{aligned} \nabla &= \left[\frac{\partial MSE}{\partial h(0)}, \frac{\partial MSE}{\partial h(1)}, \dots, \frac{\partial MSE}{\partial h(M-1)} \right] \\ &= 2RH - 2P \end{aligned} \quad (12)$$

이는 식(12)와 같이 행렬로 표현할 수 있다. 그리고 정규방정식은 식(13)와 같고,

$$RH = P \quad (13)$$

식(13)으로부터 최적의 필터계수 벡터 H 는 식(14)와 같다.

$$H = R^{-1}P \quad (14)$$

한편 최적계수 $H = (h(0), h(1), \dots, h(M-1))^T$ 로 조절했을 때 최소 오차(최소평균 자승오차)는 식(15)와 같다.

$$\begin{aligned} (MSE)_{\min} &= P_d - 2 \sum_{k=0}^{M-1} h(k) R_{dx}(k) \\ &= P_d - P^T H \end{aligned} \quad (15)$$

2.1.2 점화적 최소자승(Recursive Least Square : RLS)알고리즘

오차 특성곡면은 2차 곡면이므로 계수 벡터를 반복적으로 조절하면서 최적의 계수벡터에 이르도록 하는 방법을 이용할 수 있다. RLS 알고리즘은 상관행렬과 기울기의 추정치를 반복계산에 이용하므로 계산의 정확성이 향상되고 수렴속도의 개선도 가능하다. 시점 k 에서 상관행렬의 추정치 \widehat{R}_k 와 기울기의 추정치 $\widehat{\nabla}_k$ 를 반복 계산하는 알고리즘은 식(16)과 같다.

$$H_{k+1} = H_k - \mu \widehat{R}_k^{-1} \widehat{\nabla}_k \quad (16)$$

그리고, 기울기 $\widehat{\nabla}_k$ 의 추정식은

$$\nabla_k = \frac{\partial MSE}{\partial H_k} \quad (17)$$

이다.

상관행렬의 추정치는 평활(smoothing)필터를 이용하여 식(18)과 같

이 주어진다.

$$\begin{aligned}\widehat{R}_k &= (1-a) \widehat{R}_{k-1} + a X_k X_k^T \\ &= a \sum_{i=0}^k (1-a)^i X_{k-i}^T\end{aligned}\quad (18)$$

그런데 실제로 필요한 것은 상관행렬의 역행렬이다. 추정치 \widehat{R}_k 의 역행렬 계산을 매 시점마다 행하는 것은 계산량의 증가를 초래하므로 적절하지 못하다. 따라서 처음부터 역행렬의 추정이 가능한 알고리즘으로는 점화적 최소자승(Recursive Least : RLS)알고리즘^{[4],[5]}이 이용된다.

식(18)에 행렬 \widehat{R}_k^{-1} 을 왼쪽에서, \widehat{R}_{k-1}^{-1} 을 오른쪽에서 곱하여 정리하면,

$$\widehat{R}_{k-1}^{-1} = (1-\alpha) \widehat{R}_k^{-1} + \alpha \widehat{R}_{k-1}^{-1} X_k^T \widehat{R}_k^{-1}\quad (19)$$

식(19)가 된다. 오른쪽으로부터 벡터 X_k 를 곱하면

$$\begin{aligned}\widehat{R}_{k-1}^{-1} X_k &= (1-\alpha) \widehat{R}_{k-1}^{-1} X_k X_k^T \widehat{R}_{k-1}^{-1} X_k \\ &= \widehat{R}_{k-1}^{-1} X_k (1-\alpha + \alpha X_k^T \widehat{R}_{k-1}^{-1} X_k)\end{aligned}\quad (20)$$

식(20)과 같이 정의 할 수 있다. 여기서 행렬 \widehat{S}_k 를

$$\widehat{S}_k = \widehat{R}_{k-1}^{-1} X_k\quad (21)$$

식(21)과 같이 두면 이 행렬의 전치행렬은

$$\hat{\mathbf{S}}_k^T = \mathbf{X}_k^T \cdot \hat{\mathbf{S}}_{k-1}^{-1} \quad (22)$$

식(22)와 같이 된다. 그런데 식(21)을 이용하면 식(20)은

$$\hat{\mathbf{S}}_k = \hat{\mathbf{R}}_{k-1}^{-1} \mathbf{X}_k (1 - \alpha + \alpha \mathbf{X}_k^T \hat{\mathbf{S}}_k) \quad (23)$$

식(23)이 된다. 우변의 괄호 안은 스칼라 양이므로 행렬 $\hat{\mathbf{R}}_k^T$ 를 오른 쪽으로부터 곱하여 정리하면,

$$\hat{\mathbf{R}}_{k-1}^{-1} \mathbf{X}_k \mathbf{X}_k^T \hat{\mathbf{R}}_{k-1}^{-1} = \frac{\hat{\mathbf{S}}_k \hat{\mathbf{S}}_k^T}{1 - \alpha + \alpha \mathbf{X}_k^T \hat{\mathbf{S}}_k} \quad (24)$$

식(24)가 된다. 이 관계식을 식(16)의 우변 제 2항에 대입하면 다음과 같은 역행렬의 반복 추정식 $\hat{\mathbf{R}}_k^{-1}$ 을 얻는다.

$$\hat{\mathbf{R}}_k^{-1} = \frac{1}{(1 - \alpha) \left[\hat{\mathbf{R}}_{k-1}^{-1} \frac{\alpha \hat{\mathbf{S}}_k \hat{\mathbf{S}}_k^T}{1 - \alpha + \alpha \mathbf{X}_k^T \hat{\mathbf{S}}_k} \right]} \quad (25)$$

초기 추정치로서 $\hat{\mathbf{R}}_0^{-1} = (1/\sigma^2)I$ 를 이용하여 계산을 할 수 있다. 여기서 I 는 단위행렬이다. 기울기의 추정치인 식(17)을 식(16)에 대입하면 식(26)과 같이 된다.

$$H_{k+1} = H_k - 2u \varepsilon_k \widehat{R}_k^{-1} X_k \quad (26)$$

여기서 \widehat{R}_k^{-1} 의 계산은 식(25)에서 구한다.

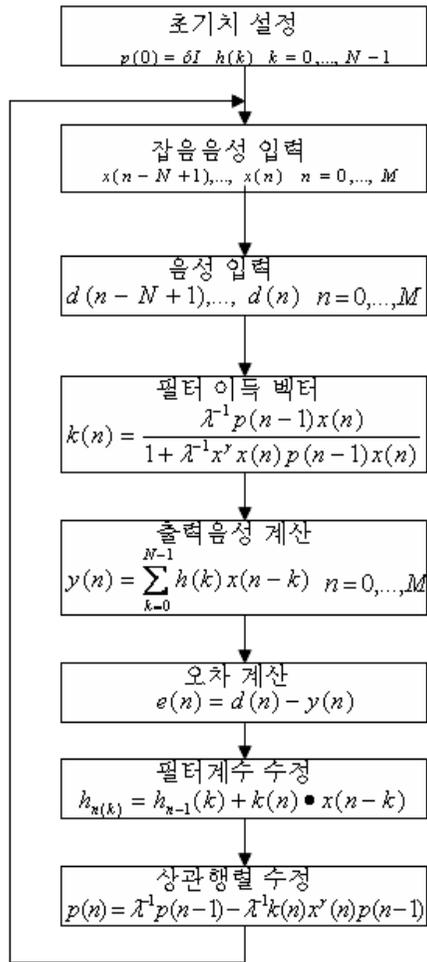


그림 2.2 RLS 적응 필터의 흐름도

Fig 2.2 The flowchart of the RLS adaptive filter

본 논문에서 소개한 알고리즘은 파라미터 값이 바뀔 때마다 각종 계산을 한다. 적응 알고리즘은 반복할 때마다 필요한 계산량이 줄어들고, 가능한 최적의 파라미터 값에 수렴해야 하며 또한 그 수렴속도가 빨라야 한다.

그림 2.2는 RLS 적응필터의 흐름도 이다. 필터의 계수는 32차로 하였다. 그리고 초기값 설정에 있어서 상관행렬의 역행렬의 초기값을 단위행렬로 주었다.

2.2 끝점 검출 알고리즘

끝점 검출 알고리즘은 유성음이나 무성음이나를 판단하는 단계이며, 실시간적으로 마이크에 입력되는 소리 중에 음성과 잡음을 구별하는 방법으로 영교차율 방식과 절대에너지 방식이 있다. 보통 인식 시스템에서는 이 두 방식을 모두 사용하지만, 본 논문에서는 절대에너지 방식^{[12]-[14]}만을 채택하였다.

2.2.1 영교차율(ZCR : Zero Crossing Rate)

영교차율은 분석구간 프레임 내에서 파형이 영점축과 교차하는 횟수를 말하며 이산적신호에서 연속 샘플링 값이 서로 다른 대수적 부호를 가질 때 일어나는데 이는 음성의 분할, 분석, 인식에 있어 매우 유용하게 사용되어 왔다.

영교차율은 유성음과 무성음을 판별하는데 유용한 파라미터가 되며 음성신호 중 영교차율이 높으면 무성음으로, 영교차율이 낮으면 유성음으로 판단한다. 영교차율에 대한 일반식은 식(27)과 같다.

$$Z_i = \sum_{n=0}^{N-1} |\text{sgn}[s_i(n)] - \text{sgn}[s_i(n-1)]| \quad (27)$$

여기서,

$$\text{sgn}[x] = \begin{cases} 1, & x \geq 0 \\ -1, & x < 0 \end{cases}$$

이다.

2.2.2 절대에너지 방식

음성 신호에서 절대 에너지는 무성음보다 유성음 부분이 크다. 절대에너지 방식은 크게 Short-Time Energy, Short-time Magnitude 2가지 방식이 있다. 에너지 E/M는 이산계(discrete system)에서 다음 식(28), 식(29)으로 주어진다.

$$E_i = \sum_{n=0}^{N-1} [s_i(n)]^2 \quad (28)$$

$$M_i = \sum_{n=0}^{N-1} |s_i(n)| \quad (29)$$

본 논문에서는 가장 간단한 방식인 절대에너지, 식(28)을 사용하여 음성 감지에 사용하였다. 여기서 말하는 음성의 감지는 유성음이나 무성음을 구별하는 방법이다.

2.2.3 프리엠퍼시스(Pre-emphasis)

프리엠퍼시스는 검출된 음성영역의 저주파 성분을 약화시키고 고주파 성분만을 강조시켜 마이크에서 발생된 음성신호의 DC성분을 제거하는 방식^[15]이다.

$$\bar{S}(n) = S(n) - aS(n-1), \quad 0.9 < a < 1.0 \quad (30)$$

$S(n)$ 은 검출된 음성데이터이고, 프리엠퍼시스 계수 a 는 0.95를 사용했을 때 최적의 결과가 나왔다.

2.2.4 해밍 윈도우(Hamming Window)

프리엠퍼시스 처리를 한 음성데이터는 각 프레임의 시작과 끝에서 신호의 불연속이 발생하는데 이것을 최소화시키기 위해 해밍윈도우^{[4],[15]}를 사용한다. $x(n)$ 은 프리엠퍼시스 처리된 음성신호이고 해밍윈도우 $w(n)$ 은 식 (32)와 같이 표현된다. 여기서 N 은 프레임의 길이이다.

$$\tilde{x}(n) = x(n) \cdot w(n) \quad (31)$$

$$w(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right), \quad (0 < n < N-1) \quad (32)$$

2.3 음성 특징 정보 추출

잡음에 강인한 특징 벡터로는 멜캡스트럼 계수(Mel-Frequency Cepstrum Coefficient)^{[4],[5],[15]}와 루트 캡스트럼 계수 등이 있다. 이 중 본 논문에서는 음성인식 시스템에서 가장 많이 적용되고 있는 멜캡스트럼에 대해서 설명하도록 하겠다.

2.3.1 MFCC(Mel-Frequency Cepstrum Coefficient)

캡스트럼 분석 방법 중 인간의 청각 특성을 이용한 것으로 멜캡스트럼이 있다. 멜은 톤 신호의 인지된 피치 또는 주파수 측정치를 나타내는 단위로서, 인간의 청각 시스템이 피치를 선형적으로 인지하지 않는 것처럼 톤 신호의 물리적인 주파수에 선형적으로 대응하지 않는다. Stevens와 Volkman은 임의로 1000Hz를 1000mel로 선정하고, 사람들에게 인지된 피치가 기준 주파수의 두 배가 되도록 변화시켜 이를 2000mel로 표기하였다. 이러한 방법으로, 실제 물리적인 주파수와 인지된 주파수 사이의 대응 관계를 구하였다. 이 대응 관계를 보면 1kHz 이하에서는 선형적으로 1kHz 이상에서는 대수적(log scale)으로 대응됨을 볼 수 있다. 멜과 주파수사이의 대응 관계를 근사적으로 나타낼 수 있는데 식 (33)과 같다.

$$F_{mel} = 2595 \log_{10} \left(1 + \frac{F_{Hz}}{700} \right) \quad (33)$$

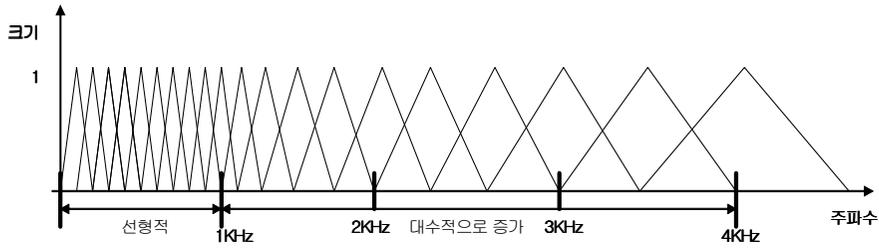


그림 2.3 필터 बैं크 대역

Fig. 2.3 Bandwidth of filter bank

여기서, F_{mel} 은 각각 근사식에 의해 구해진 인지된 주파수, F_{Hz} 는 실제의 주파수를 나타낸다.

멜 캡스트림은 DFT(Discrete Fourier Transform) 또는 FFT(Fast Fourier Transform) 크기를 멜과 주파수 사이의 대응 관계에 따라 주파수 축에서 와핑(warping)하여 이의 대수 값을 역 DFT/FFT하여 8에서 14차 정도의 계수를 구한다. 그림 2.3에 나타난 20개의 삼각 대역 통과 필터를 이용하여 임계 대역 필터를 통과한 로그 에너지 출력을 X_k 라 하면 M개의 캡스트림 계수는 식(33)과 같이 나타내어진다.

$$c_n = \frac{1}{20} \sum_{k=1}^{20} X_k \cos \left[n \left(k - \frac{1}{2} \right) \frac{\pi}{20} \right] \quad (34)$$

여기서, C_0 는 음성 프레임의 평균 에너지이며, 초기화 값이다.

2.3.2 MFCC 처리 절차

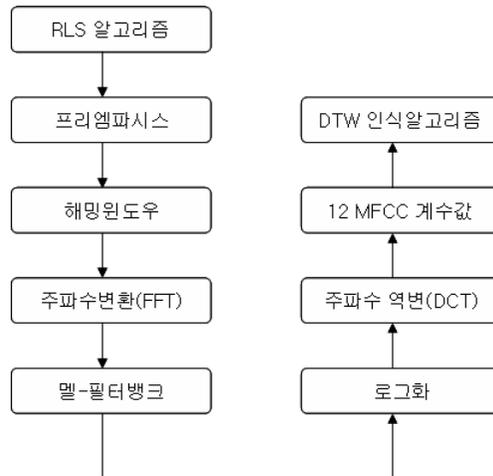


그림 2.4 MFCC 처리의 절차

Fig. 2.4 Procedure of MFCC process

그림 2.4에서는 MFCC의 처리 절차를 블록도로 표현했으며, RLS 알고리즘과 프리엠파시스를 통해서 고주파 성분이 강화된 음성은 해밍 윈도우를 씌워서 신호의 불연속 일치를 약화 시킨다.^{[4][5][15]} 그리고 FFT를 사용해서 주파수 영역으로 바꾼다. 시간영역에서 주파수 영역으로 바뀐 음성 데이터는 20개의 로그형 mel-filter banks를 거친다. 이 때 사용되어진 필터는 그림 2.3과 같은 필터를 사용한다. 필터를 거친 음성은 로그화를 통하고, DCT(Distance Cosine Transform)를 통해서 한 프레임 당 12개의 계수 값을 생성하게 된다. 이 때 DCT는 식 (34)에 나타나 있다.

2.4 화자 인식 알고리즘

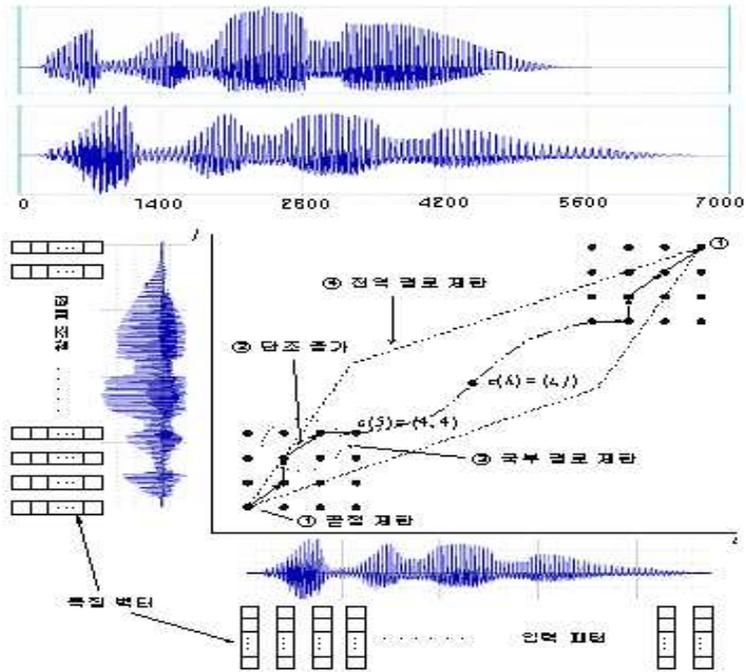


그림 2.5 DTW의 제약된 조건
Figure 2.5 Limited conditions of DTW

기존의 화자인식 알고리즘은 HMM(Hidden Markov Models), DTW, VQ(Vector Quantization), GMM(Gaussian Mixture Model) 등 여러 가지 알고리즘을 이용하여 개발되었으며, 성능면에서는 GMM이 가장 좋은 방법으로 검증되었다.

그러나 GMM은 훈련과정을 통해서만 인식함으로써 실제 시스템에 적용하기 위해서는 많은 문제점이 있다.

본 논문에서 제안하는 화자인식 알고리즘은 실시간 화자인식을 위해 간단하면서도 성능이 뛰어난 DTW(Dynamic Time Warping) 알고리즘^{[1]-[3]}을 적용하였으며, DTW는 입력패턴과 참조 패턴 사이의 거리를 측정해서 그 유사도를 측정하는 방법이다. 다시 말하면, 제한된 경로 내에서 단조 증가를 통해서 가장 가까운 거리를 판별, 유사도를 측정한다.

예를 들어, 길이가 M인 입력 음성 패턴을 $T=T(1),T(2),\dots,T(M)$ 길이가 N인 기준 패턴을 $R=R(1),R(2),\dots,R(N)$ 이라고 하면 두 패턴 간의 유사도 d는 다음 식 (35)과 같이 누적거리로 표현된다.

$$d = \sum_{n=1}^N D(R(n),T(w(n))) \quad (35)$$

이 때 $D(R(n),T(W(n)))$ 는 R의 n번째와 T의 W(n)번째의 국부적 유사도(Local Distance)이며, DTW는 두 패턴간의 누적 거리 최적화를 하는 (m,n) 평면의 최적 경로 $m=W(n)$ 를 찾는 방법이다. 이 방법은 음성신호의 특성을 고려해서 최적 경로 탐색에 다음과 같은 제약조건을 가한다. DTW 알고리즘의 내용은 그림 2.5에서 잘 나타나 있다.

- 끝점 제한 (Endpoint Constraints)
- 단조 조건 (Monotonic Condition)
- 국부 경로 제한 (Local Path Constraints)
- 전역 경로 제한 (Global Path Constraints)
- 기울기 가중치 (Slop Weighting)

시작점이 (1,1)이고 끝점이 (T_x,T_y)일 때 T_x,T_y를 통한 최적경로를 찾는 방법은,

1. Initialization : $D_A(1,1) = d(1,1)m(1).$
2. Recursion : $1 \leq i_x \leq T_x, 1 \leq i_y \leq T_y$ 이면,

$$D_A(i_x, i_y) = \min_{(i'_x, i'_y)} [D_A(i'_x, i'_y) + \zeta((i'_x, i'_y), (i_x, i_y))]$$
3. Termination : $d(x, y) = \frac{D_A(T_x, T_y)}{M_\phi}$

본 논문에서는 지역거리는 유클리드 거리법을 사용하고 있으며, 지역 제약방법은 Itakura방식¹¹⁾사용하고 있다.

DTW의 경우 기준 모델 집합의 작성은 간단하다. 인식하고자 하는 명령어들을 발음하고 분석한 후 연속된 프레임들의 특징 벡터들을 저장하고 있으면 된다. 인식 시에는 입력된 음성을 분석해 특징 벡터를 추출한 후 이들 기준 모델 집합의 구성원과 개별적으로 DTW를 수행하여 가장 적은 누적 거리를 주는 구성원을 찾으면 된다. 보통 음성인식에서는 DTW 알고리즘을 이용하여 고립단어 인식에 많이 사용하지만, 화자인식에 사용하기 위해서 본 논문에서는 패턴 매칭의 유사도(Distance)값만 이용하여 구현하였다.

제 3 장 음성 인식 시스템의 설계 및 구현

3.1 하드웨어 구성 및 특징

Main Board는 기본적으로 DSP Processor를 비롯하여 CODEC, Oscillators, PPI(Programmable Peripheral Interface), SRAM, Parallel Port, PAL(Programmable Array Logic)등으로 구성^{[7],[8]}이 되었다. 이 회로의 동작은 프로그램을 호스트 PC로부터 다운로드해서 동작하게 되는데 PC와의 호스트 인터페이스는 Board의 Parallel Port를 통해 이루어진다. 여기서는 DSP Processor와 CODEC, PPI에 대해서만 간단히 살펴보기로 하겠다.

3.1.1 DSP Processor

현재 사용되고 있는 프로세서의 종류는 크게 범용 프로세서, 마이크로 컨트롤러, 그리고 DSP 프로세서로 나눌 수 있다. 범용 프로세서는 보통의 컴퓨터가 수행하는 여러 가지 일들을 소화할 수 있도록 필요한 기능을 갖추고 있다. 예를 들면 Intel Pentium CPU가 여기에 속한다. 마이크로 컨트롤러는 주로 주변의 장치들을 관리하기에 적합하도록 만들어진 프로세서이다. 주변 장치에 여러 가지 명령을 내리면 되는 프로세서이므로 연산을 빨리 해야 하거나, 혹은 기억용량이 커야 할 필요는 없으므로 그에 맞추어 프로세서의 기능들을 줄이고 대신 가격을 싸게 하는 등의 장점을 갖도록 설계되어 있다. Intel 80196등이 이에 속한다. DSP 프로세서는 디지털 신호처리(영

상처리, 음성처리, Filtering)에 알맞게 설계된 프로세서이다.

표 3.1 TMS320C32의 특징

Table 3.1 Feature of TMS320C32

특징	C32
외부 버스	One bus: <input checked="" type="checkbox"/> 32비트 데이터 24비트 주소 0h-7FFFFFFh와 880000f-8FFFFFFh에서 <u>STRB 0</u> 동작 <input checked="" type="checkbox"/> 8, 16, 32비트 와이드메모리에 8, 16, 32비트데이터 900000h-FFFFFFFh에서 <u>STRB 1</u> 동작 <input checked="" type="checkbox"/> 8, 16, 32비트 와이드메모리에 8, 16, 32비트데이터 81000h-82FFFFFFh에서 <u>IOSTRB</u> 동작
ROM	없음
Boot loader	있음
On-chip RAM	512 주소: 810000h-82FFFFFFh
DMA	2채널 순위지정가능
시리얼 포트	1
타이머	2
인터럽트	레벨트리거 또는 에지, 레벨트리거 혼합
인터럽트 벡터 테이블	이동 지정가능 (Relocatable)
패키지	144 PQFP
전압	5V
온도	0° ~ 85°C (상업용) -40° ~ 125°C (확장판) -55° ~ 125°C (군사용)

본 연구에서는 TI(Texas Instrument)사의 DSP 프로세서인 TMS320C32를 사용하여 구현하였다. 이 프로세서는 실수 연산에 접

합하도록 설계된 산술연산장치(ALU : Arithmetic Logic Unit)뿐만 아니라, 32bit 데이터 버스와 24bit 어드레스 버스, 2개의 타이머, 2-channel, DMA, 그리고 serial port 등이 갖추어져 있다. 이 TMS320C32의 특징 중 중요한 것들을 정리하면 표 3.1과 같다.

3.1.2 CODEC

본 연구에 사용된 CODEC(CS4213)은 스테레오 오디오 CODEC으로서 단일 칩으로 구성되어 있고 A/D & D/A converting, Filtering, Level Setting 등을 수행하며 동작 방법에 따라 Serial Mode(SM) 3,4,5로 구분되는데 본 연구에서는 SM-4(Master Sub-Mode)를 사용하였다. 그리고 SM-4에서의 Sampling rate은 Processor Memory map의 814000H번지에 값을 적절히 넣어줌으로써 8KHz-48KHz 사이에서 8가지의 Sampling rate 중 하나를 선택할 수 있는데 본 연구에서는 8KHz의 Sampling rate를 사용하였다.

8255는 Intel사의 범용 Parallel I/O Interface이다. PPI는 프로그래밍을 통해 자신의 기능을 정하고 CPU와 주변장치 사이에서 그 규칙대로 신호들을 해석하여 전달해 주는 일을 하는 장치이며, 8255는 3개의 Port(Port A, B, C)의 8bit 입출력을 갖고 있는 범용 입출력 장치이다. 8255가 프로그래밍이 가능하다는 것은 필요에 따라 동작 규칙을 바꿀 수 있다는 것이다. 8255의 세 가지 모드는 Processor control register에 특정 값을 써 넣음으로써 각 Port가 입력인지 출력인지 그 규칙을 정하게 되는 것이다.

본 연구에서는 8255의 각 Port가 정해진 기능만을 수행하지 않고 상황에 따라 계속해서 변하며 사용자와 Processor 사이의 중계자 역

할을 수행한다. 그리고 사용자의 요구에 따른 외부장치들의 Control 신호도 8255를 통해 내 보내게 된다.

3.1.3 화자 인증용 시스템 보드 특성

본 논문은 실험에 사용된 화자인증용 시스템 보드 (SITK-C32)는 TMS320C32를 기반으로 설계한 실험/교육 장비로서 앞 절에서 설명되었던 모든 구성요소들이 내장되어 있으며, 그 외관 및 구성은 그림 3.1과 같다.^[6]

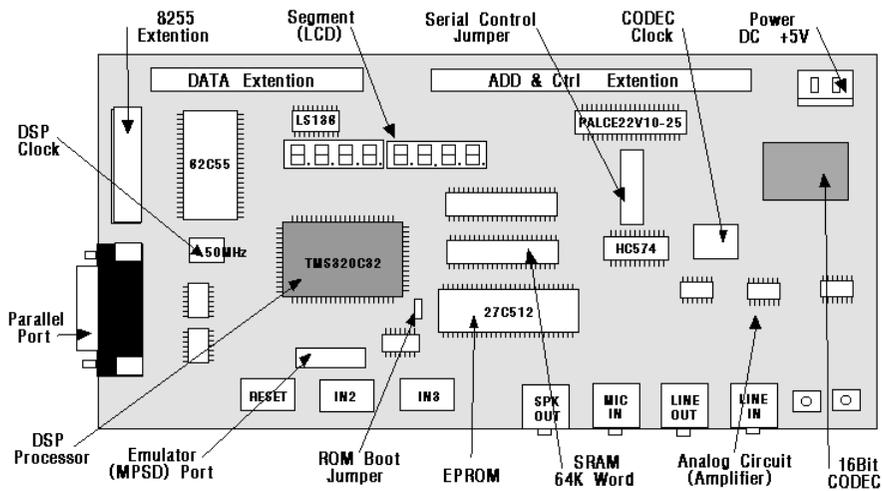


그림 3.1 SITK-C32 구성

Fig. 3.1 SITK-C32 composition

이 장비 안에는 호스트 컴퓨터와 통신하기 위한 병렬 포트가 8255A로 구성되어 있고, TMS320C32의 직렬 통신 포트와 CODEC chip CS-4218을 이용한 음성 신호 입출력 기능과 간단한 디지털 신

호를 입출력하기 위한 두 개의 스위치와 LED, 그리고 병렬 포트로 구동되는 8개의 7-Segment LED 등이 장착되어 있다.

TMS320C32의 내부를 살펴보면 실수 연산에 적합하도록 설계된 내부 산술연산 장치(Arithmetic Logic Unit : ALU) 뿐만 아니라, 32Bit 데이터 버스와 24Bit 어드레스 버스, 2개의 타이머, 2채널 DMA, 그리고 직렬통신 포트 등이 갖추어져 있음을 알 수 있다.

아울러 ezDSK라는 개발환경⁶⁾ 통해 사용자가 TMS320C32용 응용 프로그램을 쉽게 개발할 수 있도록 되어 있다.

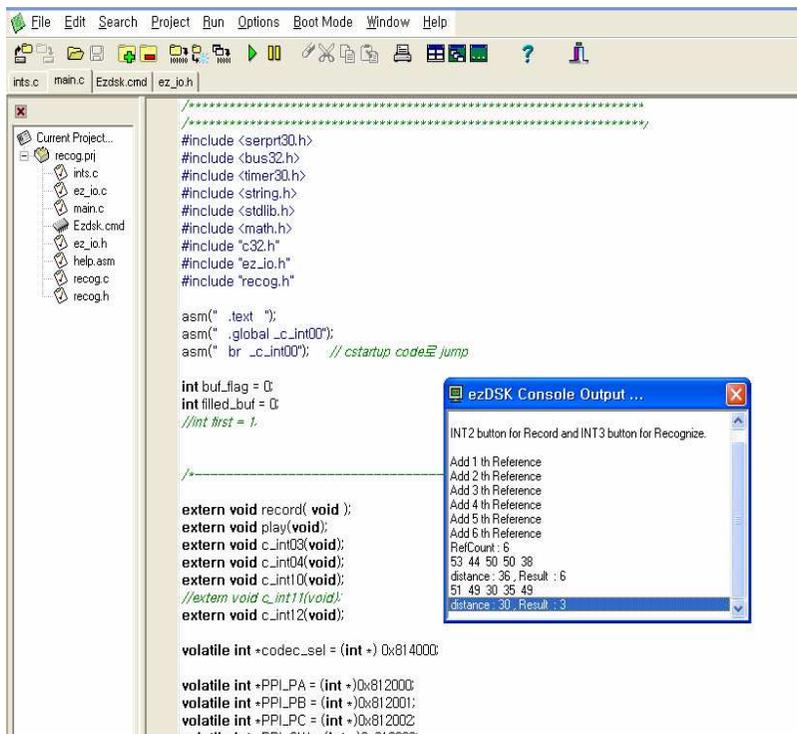


그림 3.2 ezDSK 개발 프로그램

Fig 3.2 ezDSK development program

3.2 소프트웨어 구성

그림 3.3에서는 화자 종속형 음성 인식 소프트웨어의 전체 흐름도를 나타낸다. 마이크를 통해서 들어온 음성데이터로부터 실제 음성 영역만 검출해야 할 것이다.

본 논문은 실시간으로 음성을 처리하기 때문에 실시간에서 비교적 성능이 떨어지는 영교차율을 적용하지 않았고, 절대에너지 방식을 사용하고 있다. 절대 에너지는 무성음보다 유성음부분이 크다는 이론을 바탕으로 하고 있다.

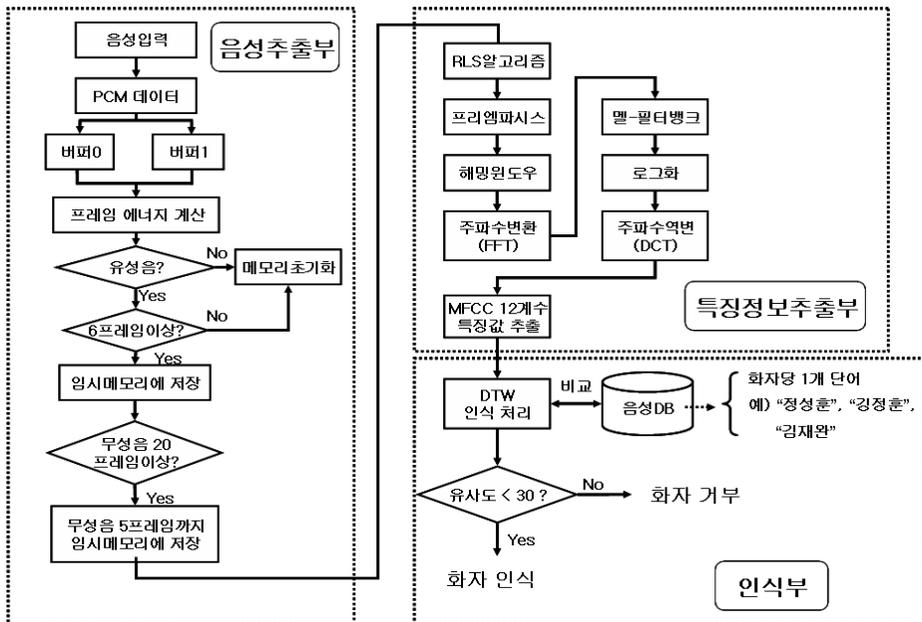


그림 3.3 화자 종속형 음성 인식 소프트웨어 전체 흐름도

Fig 3.3 Total flowchart of speech recognition software for speaker dependent

마이크에서 입력받은 음성데이터에서 각 프레임의 절대 에너지가 설정된 값보다 크면 이 구간을 실제 음성으로 간주한다.

본 논문에서는 음성프레임 길이를 100으로 설정하고, 데이터의 손실을 방지하기 위해서 프레임을 중복시키는 방식인 프레임 블록킹을 30으로 적용하고 있다. 음성구간추출은 절대 에너지가 설정된 값보다 큰 프레임이 6개 이상이 되면 그 부분은 유성음 구간으로 간주하여 메모리에 저장되고 이 후 절대 에너지가 설정된 값보다 작은 프레임이 20개 이상이 되는 무성음 구간까지 음성의 끝점을 검출하여 저장하였다. 아울러 끝점이 검출되면 20프레임의 무성음 구간 중 15프레임은 버리고, 나머지 5프레임만을 음성영역으로 사용하였다.

음성 데이터양은 인식 알고리즘을 실시간으로 처리하기에는 상당히 크므로 이 데이터를 효율적으로 줄여야 한다. 이러한 처리를 위해 특징정보 추출 과정이 필요하다. 본 논문에서는 MFCC(Mel-Frequency Cepstrum Coefficient) 방식을 사용하여 음성의 특징정보를 추출하였다. MFCC의 처리 절차는 그림 2.4의 특징정보추출부에 나타난 것과 같다. RLS알고리즘과 프리엠퍼시스(Pre-emphasis) 처리 후, 해밍윈도우(Hamming Window)를 씌우고 FFT(Fast Fourier Transform)을 수행하여 주파수 영역으로 변환한다. 변환된 값들을 그림 2.3와 같이 미리 설정된 MFCC(멜 캡스트럼) 대역필터뱅크를 통과시키고, 로그화, DCT(Distance Cosine Transform)을 거치면 프레임 당 12개의 계수 값들을 구할 수 있다. 이 값들이 음성의 특징정보들이다.

이러한 과정을 N번 수행하여 얻어진 참조 데이터 모두 저장한 후 실제 테스트 데이터가 입력이 되면 다시 끝점 검출과 MFCC과정을

수행하고 여기서 나오는 데이터와 참조 데이터를 DTW(Dynamic Time Wrapping)과정을 통해 각각을 비교하여 N개의 Distance 값을 최종결정 단계로 넘겨주게 된다. 최종결정 단계 과정에서는 DTW를 수행한 결과를 가지고 Test로 들어온 화자 데이터가 등록된 음성에 의한 데이터가 맞는지 판단하게 된다. 그리고 그 결과를 각각의 상황에 맞게 PPI 8255 등을 통해 외부의 장치들로 보내게 된다.

사용자와 인터페이스 간에 도어락을 열고 닫는 동작을 위한 제어를 위해 앞에서 설명했듯이 PPI 8255를 사용하였다. 8255의 A Port 8bit와 B Port의 하위 3bit은 LCD Display Control용으로, B Port의 상위 1bit은 도어락 모터 제어에 각각 사용하였다.

만일 화자가 등록된 음성이 아니라면 8255를 통해 LCD에 Error 메시지를 출력하게 되고 만일 화자가 등록된 음성이라면 8255를 통해 LCD에 인증 메시지를 출력하고 동시에 도어락 모터에 신호를 내어보내서 도어락을 여는 동작을 하게 하였다. 그리고 도어락이 다 열리면 Processor에서 타임 인터럽터가 동작을 해서 일정한 시간이 지나면 다시 8255에게 도어락을 닫는 신호를 내어보내서 도어락이 닫히게 된다. 도어락 작동은 그림 3.4와 같이 동작 된다.

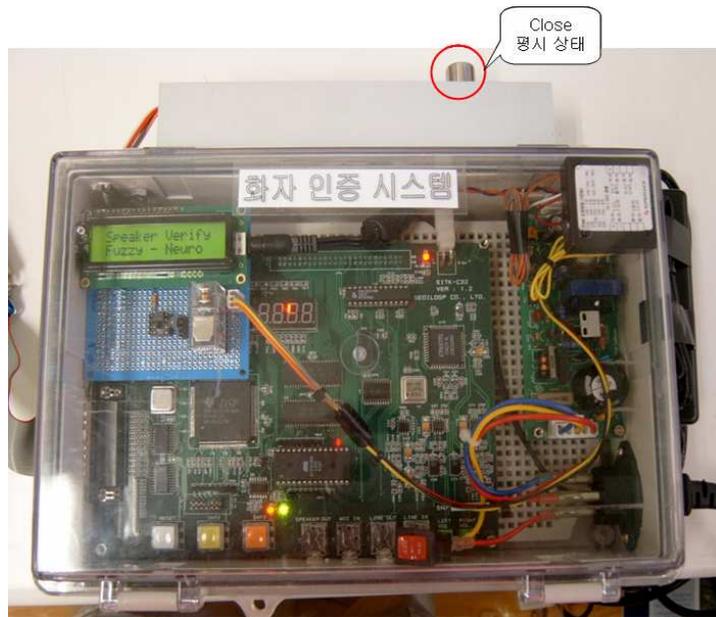


그림 3.4 도어락 시스템의 인증

Fig 3.4 Verification of the doorlock system

제 4 장 실험 및 고찰

본 실험에 이용된 음성 데이터의 분석은 총 3명의 화자를 2번씩 발성하여 총 6개의 화자 종속형 화자 인증을 위한 실험을 하였다.

표 4.1 화자(음성) 데이터 분석

Table 4.1 Analysis of speaker(voice) data

옵션	설정값
샘플링 주파수	8kHz
하드웨어 필터	LPF(Low Pass Filter), 8kHz
Frame Size(윈도우 크기)	256
Frame Blocking(윈도우 겹침)	100
절대 에너지 Threshold 값(ITU)	50000
A/D 변환	8kHz, 16bit(Resolution)

본 실험에서 사용된 샘플링 주파수는 8kHz로 설정하였으며, 코덱(CS4218, 8bit, 16bit)에 내장되어 있는 LPF로 기본적인 잡음은 제거된다. 윈도우 크기는 256, 윈도우 겹침은 100으로 설정하였으며, 음성 및 잡음의 분류 알고리즘은 절대 에너지 기법을 사용하였다. 절대 에너지의 Threshold 값은 50000으로 설정되었는데, 이는 수많은 화자 인증 테스트를 걸쳐 설정된 값이다.

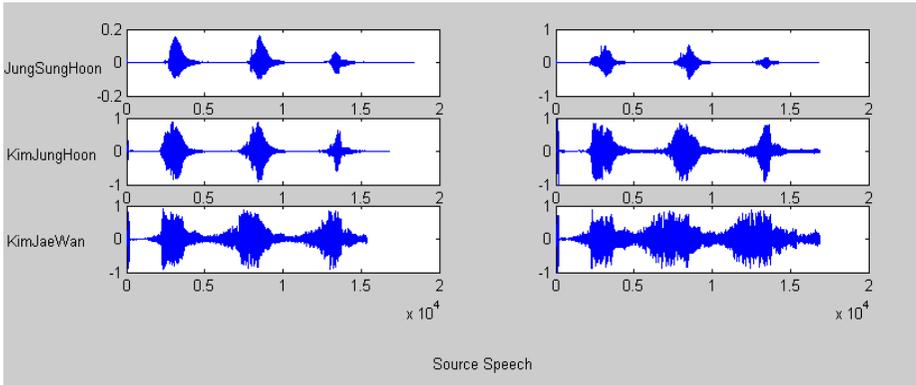


그림 4.1 잡음이 없는 깨끗한 상태의 음성 데이터

Fig 4.1 Noiseless voice data

표 4.1을 사용하여 총 3명의 음성(화자) 패턴을 아래의 그림에 나타내었다. 그림4.1은 잡음이 없는 깨끗한 상태의 음성 데이터를 도시화하였으며, 그림4.2는 화자인식에 사용된 잡음 음성 데이터베이스이다. 이때 원 음성에 잡음을 임의적으로 생성하기 위해 10dB의 라디오 쿼 상태에서 음성들을 저장시켰다.

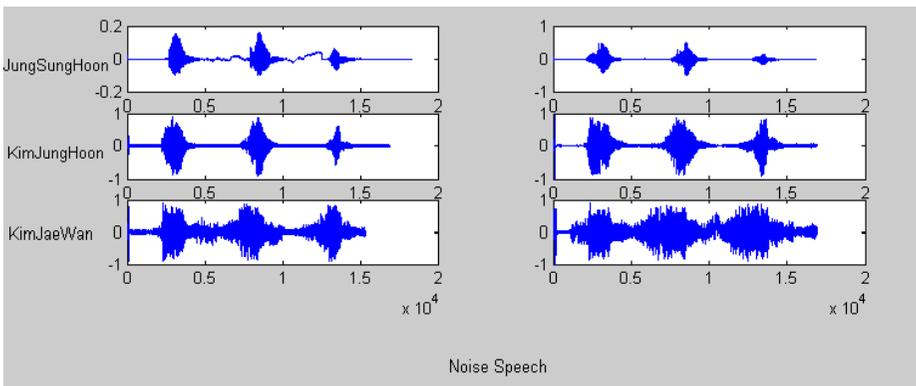


그림 4.2 잡음이 추가된 음성 데이터

Fig 4.2 Voice data added noise

4.1 잡음 처리된 화자 인증 실험 결과

본 실험에서는 여러 가지 잡음 환경에서 화자 인증 테스트를 수행하기 위하여 화자 인증 시스템의 전반에 걸쳐서 소음에 대한 대책을 강구하였다. 본 절에서는 이러한 방법들의 유효성을 평가하기 위하여 각각의 방법을 사용하는 화자 인증 시스템의 성능을 비교하였다. 소음 환경에서 화자 인증을 수행하기 위하여 본 실험에서 사용하고 있는 방법을 정리하면 다음과 같다.

- (1) 역 필터링과 고역 통과 필터 사용한 음성 구간 검출
- (2) 스펙트럼 차감법을 사용하여 잡음 제거
- (3) RLS 잡음 제거 알고리즘
- (4) 끝점 검출을 사용한 음성 구간 검출

위와 같은 방법들의 유효성을 평가하고 가장 우수한 성능을 나타내는 방법을 선정하기 위하여 표 4.2와 같은 5가지 형태의 화자 인증 시스템을 구현하였다.

Case 1은 필터링을 사용하지 않고 끝점 검출만 사용한 방법이고, Case 2는 끝점 검출을 하지 않는 방법으로, 필터와 스펙트럼 차감법을 사용하였다. Case 3은 Case 2에서 스펙트럼 차감법 대신에 필터와 RLS 잡음 제거 알고리즘 사용한 방법이며, Case 4는 필터와 스펙트럼 차감법, 끝점 검출을 사용한 방법이다. 마지막으로 Case 5는 Case 4에서 스펙트럼 차감법 대신에 필터와 RLS 잡음 제거 알고리즘 그리고 끝점 검출 알고리즘을 사용하였다.

표 4.2 여러 가지 형태의 화자 인증 시스템과 파라미터

Table 4.2 Various types of speech recognition systems and parameter

방법 \ 형태	Case 1	Case 2	Case 3	Case 4	Case 5
역필터링과 고역 통과 필터		☑	☑	☑	☑
스펙트럼 차감법		☑		☑	
RLS 잡음 제거 알고리즘			☑		☑
끝점 검출을 사용한 음성 구간 검출	☑			☑	☑

결론적으로 표 4.2에서 유추된 결과는 Case 5의 방법으로 잡음 제거하는 방법이 가장 효율적이라 생각하여 본 논문에서는 이 방법을 이용하였다. Case 5의 방법을 이용하여 잡음을 제거된 음성은 그림 4.3에 나타나 있으며, 종합적으로 정리하면 그림 4.4와 같이 된다.

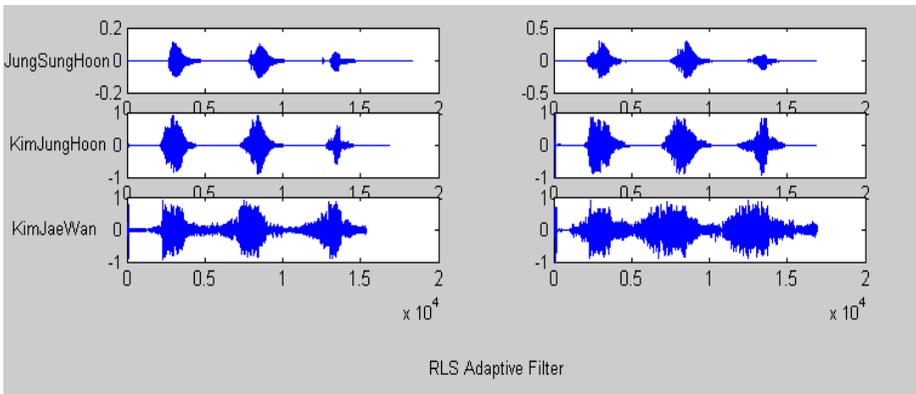


그림 4.3 RLS 적응필터 처리된 음성

Fig 4.3 Processed voice by RLS adaptive filter

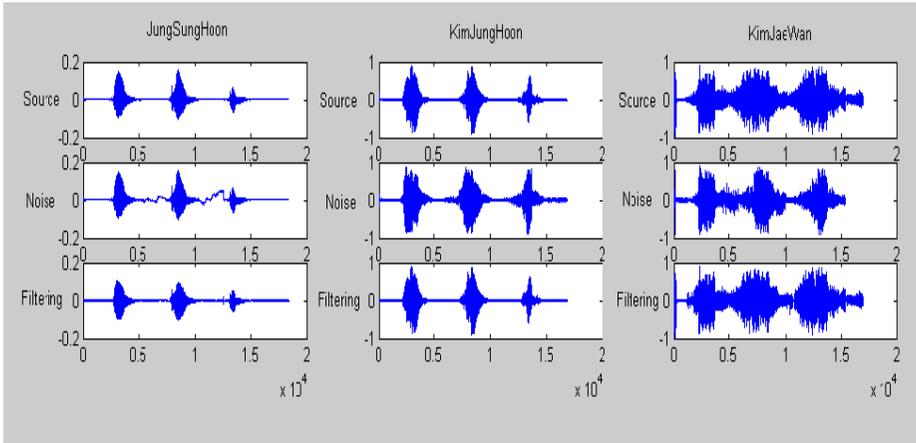


그림 4.4 음성 데이터베이스 처리 결과

Fig 4.4 Result of processing voice database

표 4.3은 5가지 경우에 대한 인식 실험 결과를 보여준다. 인식 실험은 총 3명을 대상으로 각 단어 당 하나의 기준 패턴을 사용하였다.

표 4.2에서 인식률 가장 좋은 결과는 Case 5로 Case 4와 유사하게 인식 결과가 나왔다. Case 2와 Case 3에서는 끝짐 검출을 하지 않을 경우에는 현저히 인식률 저조하게 나왔다.

표 4.3의 결과는 각 단어 당 기준 패턴의 수를 한 개로 하였을 때의 인식 결과이다. 또한 기준 패턴의 선택 과정도 각각의 단어를 여러 번 발음한 후 집단화하여 기준 패턴을 생성하지 않았다. 따라서 각 단어 당 하나의 기준 패턴을 생성하는 것보다는 두개 이상을 생성하여 음성의 변화에 보다 잘 대응할 수 있도록 하는 것이 필요하다. 본 절에서는 각 단어 당 기준 패턴의 수가 1개일 경우와 2개 및 3개 경우에 화자 인증 시스템의 성능을 비교하였다. 표 4.4는 표 4.3에서의 Case 5를 사용하여 테스트하였으며, 발성 테스트는 단어별로 총 250회를 실시하였다.

표 4.3 화자 인증 시스템의 형태에 따른 인식율

Table 4.3 Recognition rate per type of speech recognition system

대상\형태	Case 1	Case 2	Case 3	Case 4	Case 5
테스트 A	93.00	83.75	85.00	93.00	94.00
테스트 B	94.75	85.50	87.50	93.50	94.50
테스트 C	93.25	84.00	86.00	93.75	95.25
평균	93.67	84.42	86.17	93.42	94.58

※ Case 1 : 끝점 검출 사용.

Case 2 : 필터와 스펙트럼 차감법을 사용.

Case 3 : 필터와 RLS 잡음 제거 알고리즘 사용한 방법.

Case 4 : 필터와 스펙트럼 차감법, 끝점 검출을 사용.

Case 5 : 필터와 RLS 잡음 제거 그리고 끝점 검출 알고리즘을 사용.

표 4.4 단어 당 기준 패턴의 개수에 따른 인식률 변화

Table 4.4 Change of recognition rate depending on number of reference patterns per word

인식률\발성수	1번 발성	2번 발성	3번 발성
Recognition rate	90.2%	95.5%	95.8%

표 4.4에 나타난 것처럼 기준 패턴의 수는 1번보다는 2번이 낮지만, 3번이나 2번은 별 차이가 없는 걸로 나왔다. 그러므로 본 논문에서는 기준 패턴을 각 화자 당 2번으로 결정하였다.

표 4.5에서는 위의 실험을 토대로 Case5의 방식에 기준 단어수를 각각 5개씩 사용하여 인식 실험을 100회 실시하였다. 단, 실험환경

은 잡음은 없는 장소에서 실시하였다. 실험결과 평균 97.0%의 인식률이 나왔다.

표 4.5 화자 종속형 DTW를 이용한 화자 인증 최종 테스트

Table 4.5 Final speech recognition test using speech dependent DTW

명령어	테스트 횟수	인식 횟수	인식률
정성훈	100	99	99%
김정훈	100	96	96%
김재완	100	96	96%

4.2 FAR 및 FRR 화자 인증 테스트 결과

인식률을 평가하는 기준으로는 2가지가 있다. 그 첫째는 접근이 허락되지 않은 사람을 통과시키는 오류인 타인 수락율(FAR-False Acceptance Ratio)이다. 이 비율이 1%라면 100번의 인식이 수행되는 동안 1번의 오인식이 발생하였다는 것이다. 타인 수락율이 높을수록 시스템에서는 치명적이 된다. 일반적인 보안을 목적으로 하는 시스템에서 그 기술의 우위를 가늠하는 대표적인 척도로 대개 이 FAR이 꼽힌다. 두 번째는 출입이 허락된 사람을 통과시키지 않는 오류인 본인 거부율(FRR-False Rejection Ratio)이다. 이 오류율이 높다면 당연히 출입해야 할 사람인데도 출입불가가 되는 것이다. 따라서 현실에서 상용화되기 위해서는 이 비율을 최소화시키는 것이 곧 시스템의 보안성을 높여 안정성 면에서 신뢰를 얻을 수 있다.

본 절에서는 음성에 대해 각각 FRR과 FAR을 테스트하였다.

표 4.6 화자 인식 FRR 테스트

Tale 4.6 Speaker recognition FRR test

단어명	인증횟수	거부횟수	평균 Distance
정성훈	29	1	23.5
김정훈	28	2	25.9
김재완	28	2	24.7

화자인식의 FRR 테스트 한 결과는 표 4.6과 같다. 화자 인식 FRR 테스트는 총 3명이 30번씩 발성 테스트를 한 결과를 보여준다. FRR 테스트 결과 5.56%거부률을 나타냈다. 본 논문에서는 Distance 값이 30이하일 때만을 화자인식에 성공하고, 30이상일 때는 등록되지 않은 음성이라 판단하여 화자인식을 거부하게끔 동작한다.

표 4.7 화자인식 FAR 테스트

Talbe 4.7 Speaker recognition FAR test

단어명	인증횟수	거부횟수	평균 Distance
정성훈	1	29	97.5
김정훈	0	30	120.4
김재완	0	30	130.8

표 4.7에서는 타인이 발음하여 인식을 시도한 경우의 FAR 테스트 결과를 나타낸 것이다. 실험은 각 단어별로 본인을 제외한 나머지

사람들로 구성되어 랜덤하게 테스트를 수행하였다. FAR 테스트 결과 1.1% 타인 수락률을 나타냈다.

제 5 장 결 론

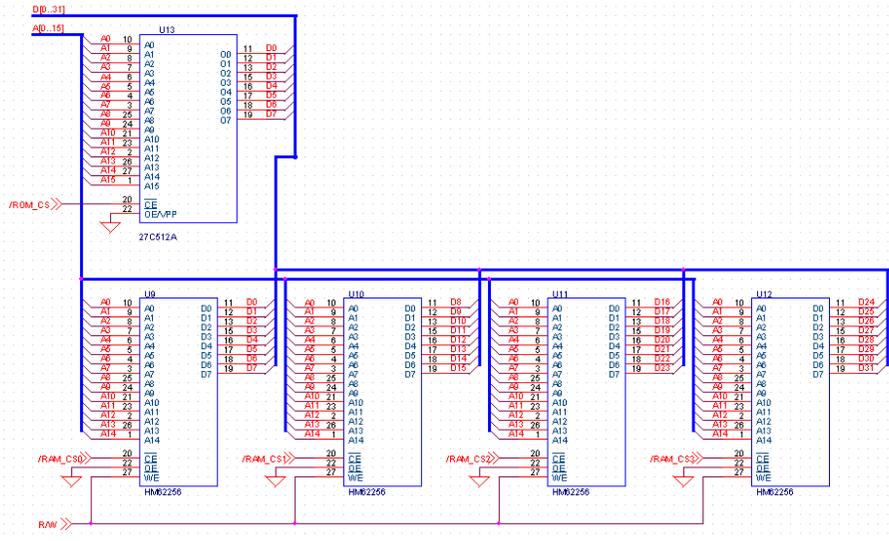
본 논문에서는 화자 종속형 음성인식 시스템을 구현하기 위한 방법 중 가장 많이 사용되는 DTW 알고리즘을 이용하여 임베디드 시스템(Embedded System)을 구현하였다. 본 논문에서 제안한 시스템은 잡음에도 높은 인식률과 정확성, 빠른 수렴속도를 가지는 RLS (Recursive Least Square) 적응 필터를 사용하였다. 이 알고리즘은 필터계수가 최적의 값에 수렴하도록 파라미터 값이 바뀔 때 마다 계산을 하여 에러를 최소화하였고, FIR 필터 구조 중 트랜스버설(직접형)을 사용하여 계산량의 감소와 수렴속도의 개선을 가져왔다.

또, DTW 인식 알고리즘을 이용하여 실제 DSP(TMS320C32)보드인 SITK-32C를 이용하여 화자 종속형 음성인식 알고리즘을 도어락(Doorlock)시스템에 적용하여 실험을 하였으며, 그 결과 기존의 종속형 시스템 보다 FRR(본인거부률)이 2.74%가 향상되었고, 음성 인식시스템에서 가장 중요한 요소인 FAR(타인수락률)은 1.2%의 개선으로 안정된 보안 시스템을 입증하였다. 앞으로의 연구방향으로는 음성인식의 궁극적인 목표인 화자독립형 과 화자 종속형을 결합한 연속어(continuous speech)인식 시스템을 임베디드 시스템에 적용하기 위해서는 효율적인 알고리즘 및 하드웨어 부분을 좀 더 보강해야 할 것이다.

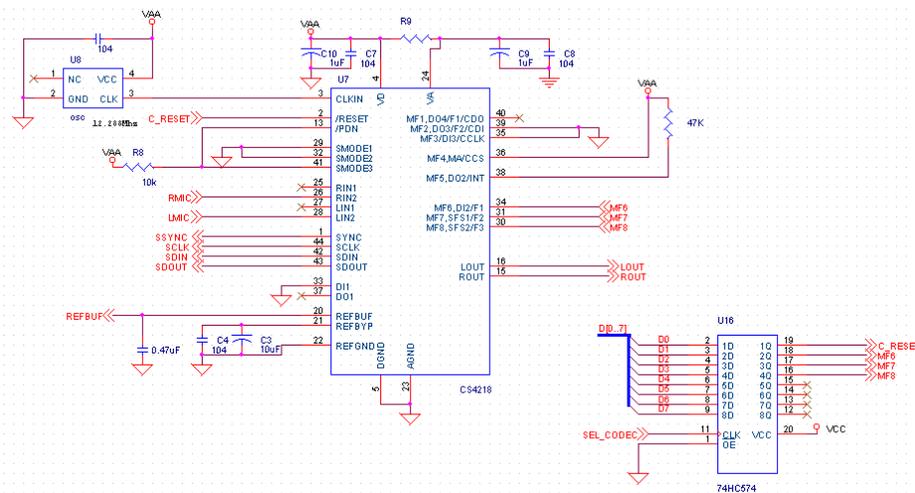
참고 문헌

- [1] Lawrence Rabiner and Biing Hwang Juang, "Fundamentals of Speech Recognition", Prentice Hall, 1993
- [2] Steven L.Gay, Jacob Benesty, "Acoustic Signal Processing for Telecommunication", Kluwer Academic Publishers, 2000.
- [3] Claudio Becchetti, "Speech Recognition Theory and C++ Implementation ", John Wiley & Sons Ltd, 1999.
- [4] 이채욱, "최신 디지털 신호처리", (주)북스힐, 2002.
- [5] 우경호, "자동차 소음환경에서의 음성 인식 시스템의 성능 향상 및 실시간 구현", 연세대학교 공학석사학위 논문, 2001
- [6] 이지홍, 서일 DSP 기술연구소 공저, "DSP chip의 활용", 서일 DSP 주식회사
- [7] "TMS320C32 General Purpose User's Guide", Texas Instrument
- [8] 윤덕용, " TMS320C32 마스터", Ohm사, 1999
- [9] 김창근, 한학용, "TMS320C32를 이용한 실시간 음성인식 무선자동차의 구현", 신호처리 시스템학회, 2001.
- [10] 함영준, 권혁재, 최수영, 정익주, "DSP Processor(TMS320C32)를 이용한 화자인증 보안시스템이 구현", 강원대학교 산업기술연구소 논문집, 제21권 B호, 2001
- [11] 이태한, "범용 DSP를 이용한 차량 항법용 음성 인식 시스템의 성능 개선", 연세대학교 공학석사학위 논문, 1999
- [12] 김정훈, "음성인식처리용 임베디드 시스템의 설계 및 구현에 관한 연구", 한국해양대학교 공학석사학위 논문, 2003

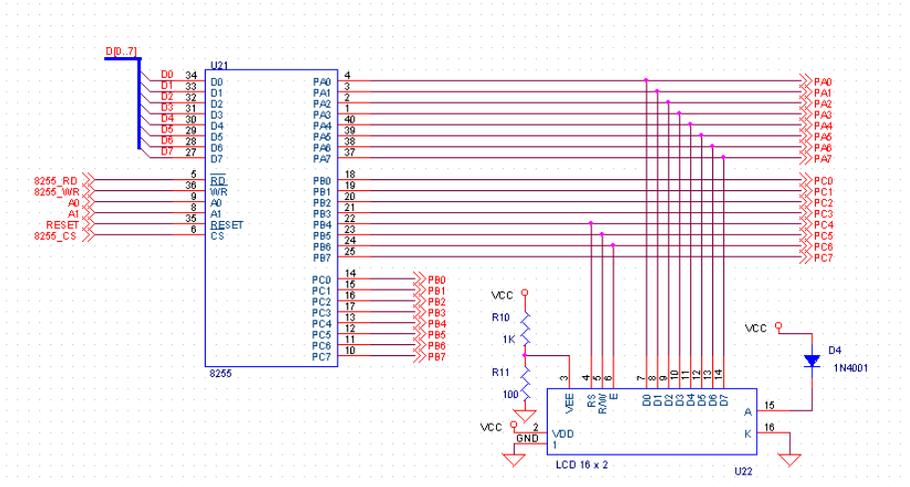
- [13] 류홍석, “ART2 적용 임베디드 음성인식 시스템의 설계 및 구현에 관한 연구”, 한국해양대학교 공학석사학위 논문, 2004
- [14] 장원일, “人工知能 技法을 利用한 임베이드형 多重生體 認識 시스템 具現에 관한 研究”, 한국해양대학교 공학석사학위 논문, 2005
- [15] 오경환, “음성언어정보처리”, 홍릉과학출판사, 1997
- [16] 한진수, “음성신호처리”, 오성미디어, 2000
- [17] Vinay K. Ingle, John G. Proakis, “Digital signal Processing Using MATLAB”, Brooks/Cole Publishing Company, 2000
- [18] 이상배, “퍼지-뉴로 제어 시스템”, (주)교학사, 1999
- [19] 김정훈, 류홍석, 강재명, 강성인, 이상배, “화자 독립형 음성인식 모듈 설계에 관한 연구” 제어자동화 시스템 공학회 추계 학술대회, pp.109 - 112, 2002. 12
- [20] 김정훈, 류홍석, 강성인, 강재명, 김관형, 이상배 “음성인식을 적용한 다기능 휠체어 시스템의 설계 및 구현”, 한국퍼지 및 지능 시스템 학회집 제12권 제1호, pp.1-5, 2002
- [21] 김정훈, 류홍석, 강성인, 강재명, 이상배 “다기능 전동휠체어의 음성인식 모듈에 관한 연구”, 대한전자공학회 컴퓨터소사이어티 제25권 1호, pp.83-86, 2002



부록 2. 외부 메모리 인터페이스 회로도
Appendix 2. External memory interface circuit diagram

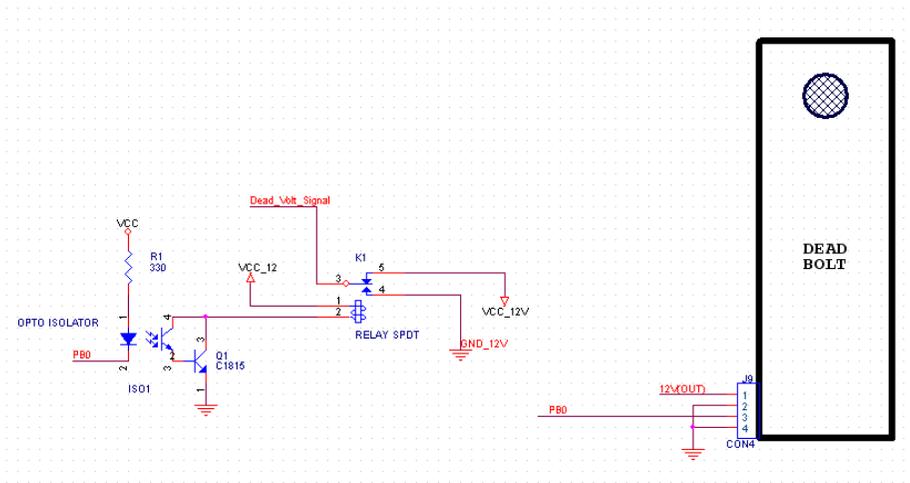


부록 3. 음성 코덱칩 인터페이스
Appendix 3. Speech codec-chip interface



부록 4. LCD 주변 회로도

Appendix 4. LCD peripheral circuit diagram



부록 5. 전원부 및 데드볼트 회로도

Appendix 5. Power supply and dead bolt circuit



부록 6. 화자 인증 시스템
Appendix 6. Speaker verification system

```

function [xi,w]=rls(lambda,M,u,d,delta)
%      Recursive Least Squares
%      Call :
%      [xi,w]=rls(lambda,M,u,d,delta);
%      Input arguments:
%      lambda      = forgetting factor, dim 1x1
%      M           = filter length, dim 1x1
%      u           = input signal, dim Nx1
%      d           = desired signal, dim Nx1
%      delta       = initial value, P(0)=delta^-1*I, dim 1x1

%      Output arguments:
%      xi          = a priori estimation error, dim Nx1
%      w           = final filter coefficients, dim Mx1

% initial values
w=zeros(M,1);
P=eye(M)/delta;
% make sure that u and d are column vectors
u=u(:);
d=d(:);

% error vector
xi=d;

% Loop, RLS
for n=M:N
    uvec=u(n:-1:n-M+1);
    k=lambda^(-1)*P*uvec/(1+lambda^(-1)*uvec'*P*uvec);
    xi(n)=d(n)-w'*uvec;
    w=w+k*conj(xi(n));
    P=lambda^(-1)*P-lambda^(-1)*k*uvec'*P;
end

```

부록 7. RLS 알고리즘 Matlab 소스
Appendix 7. Matlab source of RLS algorithm

감사의 글

연구실에 들어와서 본 논문이 있기까지 세심한 배려와 깊은 관심으로 지도해 주신 이상배 교수님께 진심으로 감사드립니다. 그리고 바쁜 가운데도 논문 심사과정에서 부족한 점들을 조언해 주신 양규식 교수님, 박동국 교수님과 항상 학부 때부터 많은 가르침을 주신 임재홍 교수님, 심준환 교수님과 잘 뵙지는 못했지만 손경락 교수님께도 감사드립니다.

고등학교에서 갓 올라와 아무것도 모르는 저에게 많은 전공지식을 접하게 해 주시고 앞으로 내가 나아가야 할 방향을 제시해 주신 강길범 교수님께 감사드립니다.

비록, 연구실에서 같이 보낸 시간은 짧았지만 부족한 후배에게 항상 많은 조언과 격려를 아끼지 않으신 이재현, 강성인, 공석민, 이창규 선배님에게 감사드리며, 같이 프로젝트를 하면서 많은 조언과 가르침을 주신 김정훈, 김동한, 김영탁 선배님과 사회의 일선에서 맡은바 일에 최선을 다하시면서 후배의 취업과 진로에 대해서 많은 조언과 격려를 아끼지 않으신 류홍석, 강종윤, 김수정 선배님에게 감사하다는 말을 전합니다.

그리고 짧지만 1년이라는 기간 동안 같이 연구실 생활을 하면서 동고동락을 함께 한 동생이자 후배인 김재완 에게도 고맙다는 말을 전합니다.

저에게 힘든 시련이 찾아 왔을 때 저에게 많은 격려와 조언으로 저에게 힘을 불어주신 저의 친형인 이승형 에게도 감사의 말을 전하며, 친동생이지도 않은 저에게 많은 격려와 조언으로 인생의 방향을 제시해 주신 희간형 에게도 감사의 말을 전합니다. 중학교 때 제가 부족한 점을 같이 해결해주시느라 고생하신 우리 은순 누나에게도 마음속 깊이 감사의 말을 전하며, 제가 어려운 시련과 좌절에 빠졌을 때 저의 옆에서 힘이 되어준 용미에게도 고마운 마음을 전합니다. 군복무 시절 저에게 따뜻한 편지 한 통으로 저에게 힘을 불어 넣어주신 둘째 형수께도 감사드리며, 어릴 때부터 지금까지 나와 희로애락을 같이 했던 친구인 광수, 이경과 동생인 문석에게도 고마운 마음을 전하며, 대학 때 알게 된 친구인 태상, 종렬, 봉석, 철우, 동욱, 홍석, 권삼에게 고맙다는 말을 전합니다.

초등학교를 입학하면서 대학원까지 제가 학업에 매진할 수 있도록 항상 사랑해주시며, 저의 뒤에서 항상 힘이 되어주신 저희 부모님께 진심으로 감사드리며, 저를 친자식처럼 길러주시고 사랑해 주신 부모님께도 고개 숙여 감사하다는 말을 전합니다.

끝으로 지금은 보고 싶어도 찾아뵙지는 못 하지만 누구보다도 더 좋아하시고 반겨주실 사랑하는 우리 친 할머니에게도 감사의 마음과 이 논문을 바칩니다.