

장애인을 위한 음성인식 엘리베이터

Voice Recognition Elevator for Handicapped People

오 용 재* 김 정 래* 정 익 주**
Oh, Yong-Jae Kim, Jeong-Rae Chung, Ik-Joo

Abstract

In this paper, we proposed an efficient method for implementing a voice recognition elevator. Unlike the existing ones, the proposed system is based on the bluetooth communication and smartphones equipped with the google speech recognition software, which makes it possible that the speech recognition capability can be added to the previously installed elevators. In order to improve the recognition accuracy, instead of using the result of the google recognizer, we built a web server where the user data are accumulated and they are used for recognition error correction.

키워드 : 음성인식 엘리베이터, 장애인, 스마트 장치
Keywords : Voice recognition elevator, The handicapped, Smart devices

1. 서론

현대의 엘리베이터는 단순한 이동 수단이 아닌 또 다른 생활의 한 부분으로 인식되어지고 있다. 최근 엘리베이터 내부 컨셉은 쾌적함을 위해 공기 청정기, 살균기, 에어컨과 같은 친환경 시스템이 주류를 이루고 있다. 하지만 그것은 상류층의 우아한 생활이라는 모티브를 가지고 만든 것이다. 이번 논문에서는 더 많은 사람들이 '우아한' 생활뿐만 아니라 '스마트한' 생활, '편리한' 생활을 누릴 수 있도록 하는 엘리베이터를 구상하였다.

장애인에 대한 권리와 편의에 대한 관심이 증가함에 따라 그와 관련된 사업 또한 발전 가능성이

보이고 있다. 음성인식 엘리베이터는 그런 가능성 중 하나라고 할 수 있다[1]. 음성인식 엘리베이터는 단순히 장애인이나 신체적 부자유자만을 위한 것은 아니다. 일반인들이 남들보다 스마트한 생활을 한다는 인식을 갖게 함으로써 현대 사회에서 소속감과 만족감을 선사 할 수 있을 것이라 생각한다. 물론 근본적인 목적은 신체적 부자유자들의 권익과 편의를 위한 것이지만 사업성이 있어야 실현 될 수 있기 때문에 사업적인 가능성도 고려해야 한다.

사회 복지 차원에서 과거와 다르게 장애인들의 자립심이 커지고, 사회 생활에 대한 욕구가 증대되면서 여러 문제점들이 드러난다. 그중 하나가 바로 '이동'의 불편함이다. 엘리베이터는 1층부터 적게는 5층 많게는 60층 이상까지 가는 버튼이 있다. 하지만 신체가 불편한 이들에게는 건물의 높이보다 더 높게 느껴질 수도 있다. 장애인을 배려하기 위해 높이를 낮춘 버튼들도 있지만, 그 버튼을 누를 수

* 강원대학교 전자공학과 학사과정

** 강원대학교 전자공학과 교수, 공학박사, 교신저자

없는 사람도 분명 존재한다. 본 논문에서는 그러한 문제점을 극복하고자 기존의 푸쉬형 버튼 대체할 수 있는 스마트폰 기반의 음성인식 기술을 기존의 엘리베이터에 접목하는 방안을 제시하고 데모 시스템을 구현하였다.

본 논문은 1장의 서론에 이어, 2장에서 주요 기술과 기존 기술과의 비교를 하고 3장에서 기본 구조의 설계 및 알고리즘에 대해 언급한다. 4장에서 엘리베이터와 외부 서버와의 통신 방법, 음성인식 기의 구현에 대해 서술했고, 5장에서 실제 구현 모습을 확인하고 6장에서 결론을 맺는다.

2. 기존 기술과 본 연구의 주요 기술

설계에 앞서 실제 엘리베이터에 대해 알아본다. 기본적으로 엘리베이터는 도르레, 줄, 카, 평형추 등 3만여 개의 부품으로 구성되어 있다.

용도에 따라서 승객용, 화물용, 자동차용 등 여러 가지가 있다. 속도에 따라 저속(15~45m/min), 중속(60~105m/min), 고속(200~300m/min)으로 구분되고, 구동 방식에 따라 로프식, 유압식 등으로 나눌 수 있다. 이번 논문에서 구현된 데모 시스템은 로프식으로 구동되며 간략화된 구조로써 실제 엘리베이터와 다르게 과속 방지 장치나 안전 장치는 모두 제외되었다. 총 층수는 7층이므로 저속형 엘리베이터이다. 기존에 제안된 음성인식 엘리베이터는 엘리베이터 카에 자체적으로 내장된 음성인식부와 마이크, 모터 제어부로 구성되어 있다.

음성인식 방식을 살펴보면, 대화 내용에 따라 오작동이 발생하지 않기 위하여 특정한 키워드가 스위치 역할을 하게끔 설정되어 있다. 이 특정 키워드가 인식된 후에 원하는 층수를 말하면 해당 동작을 수행 하도록 되어있다. 이 방식은 대화중 오작동을 일으킬 확률이 적지만 매번 키워드를 말하는 불편함이 있고, 관리자가 별도의 업데이트를 하지 않는 한, 최초 인식률에서 벗어나지 못한다는 한계가 있다[2]. 또한 처음 건물을 시공하고 엘리베이터를 설계할 때 음성인식 엘리베이터로 설계하지 않으면 이후 교체가 어렵다. 따라서 최초 설계할 시 엘리베이터의 설계를 음성인식 엘리베이터 목적으로 설계해야만 하므로 가격이 비교적 비싸고 적용범위가 새로 완공한 빌딩이나 아파트 등, 제한적이다.

본 논문에서 제안한 음성인식 엘리베이터는 스마트폰이 마이크 역할을 하며 사용자의 말투와 억양에 따라 데이터가 업데이트 되어 인식률이 향상되며, 가격이 저렴하고 쉽게 설치할 수 있으며, 조작이 가능한 블루투스를 사용하므로 뛰어난 가성비와 범용성을 지니고 있다. 또한, 이미 설치되어 있는 일반 엘리베이터에도 적용이 가능하다는 장점이 있다.

3. 설계

3.1 기본 구조 설계

Digital Signal Controller(DSC)인 TMS320F28035를 이용하여 센서와 모터, 블루투스와의 연동을 구현하였다. TMS320F28035 DSC는 가격이 저렴하면서도 모터를 제어하는데 있어서 매우 우수한 성능을 보이고, 19개의 GPIO 핀이 있어서 많은 센서와 LED를 사용해야 하는 데모 시스템에 매우 적합한 마이크로프로세서이다. 제어용 모터로는 스텝 모터를 사용하였고, 센서는 적외선 거리센서를 사용했다. 음성인식 엘리베이터를 만들기 위해 그림 1과 같이 구성했다.

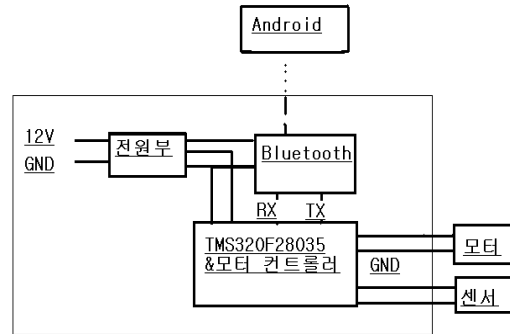


그림 1. 하드웨어 블록 다이어그램

모터를 제어하는데 사용한 방법은 일반적인 PWM을 이용하여 딜레이와 움직이는 거리를 제어했다. LED의 점등과 센서의 인식은 GPIO 포트와 TMS320F28035에 내장된 AD 변환기를 이용했다. 음성인식부는 스마트폰 상의 구글 인식기 및 인식률 향상을 위해 구축한 서버와 통신하여 인식 결과를 얻게 된다. TMS320F28035와 스마트폰의 통신은 블루투스를 이용하여 수행된다.

3.2 알고리즘

그림 2와 같은 방식으로 동작하도록 엘리베이터 프로그램을 제작했다. 알고리즘을 구상할 때 해당 프로세서가 최소한의 동작으로 오류 없이 작업을 수행하도록 했으며, 프로그램 소스에서도 하나의 배열로 모든 동작을 수행 할 수 있도록 코드 최적화를 하였다. 원하는 층수에 따라 배열의 순서가 바뀌면서 우선 순위를 결정한다. 0부터 13 (총 14개)의 배열에 순서대로 들어가고, Up, Down의 순서에 따라 다시 배열 번호가 바뀌게 된다.

사실 상 엘리베이터는 고려해야할 경우의 수와 변수가 상당히 많지만, 특수한 상황은 배제하였다. 예를 들면, 전력이 차단되어 엘리베이터가 정지한 경우, 다시 복구되었을 때 위치의 재설정, 복수의

장소에서 이용자가 동시에 감지되었을 때와 같은 경우이다.

복수의 장소에서 이용자가 감지되는 것은 센서의 인식 반응 속도보다 빠르게 감지되면 현재 층수에서 가까운 곳의 이용자를 먼저 인식할 것인지, 아니면 먼 곳을 인식할 것인지의 문제인데, 이번 데모 시스템에서는 경우 그럴 가능성은 없다고 가정하였다.

그림 2에는 나타나 있지 않지만, 이동 중에도 그림 2와 같이 동작한다. 즉, 1층에서 7층을 가는 도중에 4층에서 이용자가 감지되었을 경우 현재 위치가 4층보다 이전이면 4층을 경유하지만 4층 이상이면 무시하고 진행한다.

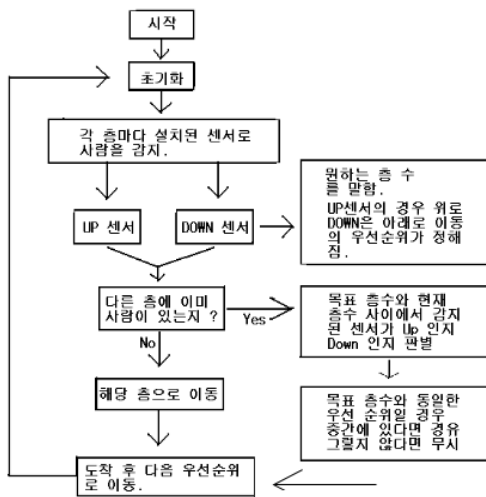


그림 2. 동작 흐름도

4. 하드웨어 구성

4.1 Android 와 TMS320F28035 간의 통신



그림 3. 블루투스 모듈(FBZx5XX)

Bluetooth 칩은 통상적으로 많이 쓰이는 그림 3의 저가형 칩을 사용했다. Android와 프로세서를 연결하는 부분으로 수시로 연결 신호를 송출 한다. 연결 방법은 1 to N 방식으로 N to 1 방식과 구조적으로 차이점이 있다. 엘리베이터의 이용자는 한 명이 아닌 여러 명이므로 다수의 Master를 두느냐 Slave를 두느냐의 선택이 있다. 하지만 다수의 Slave를 두는 경우(스마트폰이 Slave인 경우), Master에서 보내온 신호에 따라 연결 순서가 정해지며 그때 그때 연결 승인을 해야 한다. 따라서 엘리베이터 구동에 문제가 있으며 이용자가 직접 연결을 시도해야 하는 불편함이 있다.

본 논문에서는 다수의 Master를 두어서(스마트폰이 Master) 원활한 연결 상태를 유지하고자 1 to N 방식을 이용했다. Android 이용자의 블루투스 장치가 Master가 되고 엘리베이터에 내장된 블루투스 칩이 Slave가 된다. 하나의 Slave에 다수의 마스터가 연결되는 방식인데 동시에 연결이 아닌 순차적인 연결, 즉 일정 주기를 두고 돌아가며 데이터를 수신한다. 딜레이는 대략 40~100ms 정도의 주기로 데이터를 주고받는다. 따라서 연결 상태는 다수의 마스터가 동시에 연결되어 있는 상태로 확인되어 여러 명의 사용자가 연결되는 순서를 기다릴 필요 없이 동시에 연결 된 것처럼 느껴진다. 다음은 위에서 언급한 1 to N 방식으로 블루투스 연결 구현을 위한 핵심 소스의 일부분이다.

```
public boolean
onOptionsItemSelected(MenuItem item)
{
    switch (item.getItemId()) {
        case R.id.scan:
            if(D) Log.e(TAG, "+++++++25+++++++");
            Intent serverIntent = new Intent(this,
                DeviceListActivity.class);
            startActivityForResult(serverIntent,
                REQUEST_CONNECT_DEVICE);
            return true;
        case R.id.discoverable:
            if(D) Log.e(TAG, "+++++++26+++++++");
            ensureDiscoverable();
            return true;
    }
    return false;
}
```

하지만 실제로 데모 시스템을 구현할 때 사용자의 스마트폰 기종에 따라 프로세서와의 연결 속도, 서버와의 통신 속도가 제각기 다르므로 연결 대기 시간을 0.82초 이상으로 조정했다. 이것은 서버모터가 동작을 완료하기 위한 최소한의 대기시간인데, 본 논문에서 구현한 데모 시스템의 엘리베이터 문이 열리고 닫히는 최대 시간이다.

그림 4는 Android와 프로세서 간의 연결 상태를 확인할 수 있는 화면이다.



그림 4. 어플리케이션 사용자 인터페이스 화면

4.2 Android 와 서버간의 통신

Android에서 지원해주는 TTS(Text To Speech)기능과 구글에서 지원 해주는 STT(Speech To Text) API를 이용하여 음성 제어 어플리케이션 제작 및 서버와의 연동을 구현했다. APMsetup7 프로그램을 이용하여, Apache 웹서버, PHP 서버 개발 언어, Mysql 데이터베이스를 컴퓨터에 구축하고 데이터베이스는 엘리베이터에서 일반적으로 사용되는 층수 단어와 해당 숫자를 기본적으로 저장 해놓았다.

사용자가 엘리베이터의 일정거리 내로 접근하면 엘리베이터의 거리센서에 감지가 되고 TMS320F28035 프로세서는 bluetooth 통신을 통해 이를 Android 단말기에 알린다. Android 단말기는 "층수를 말하세요." 라는 텍스트를 출력하고, 이 텍스트를 읽어서 사용자에게 알려준다. Android의 TTS 기능이 종료되면, 구글 음성인식기가 자동으로 호출되어 사용자의 명령을 기다리는 상태가 되고 사용자가 명령하면 어플리케이션으로부터 얻은 단어를 웹 서버(DB)에서 조회하고 숫자와 적절한 멘트를 웹 서버에 띄우면 이것이 어플리케이션으로 전송된다. 전송된 데이터가 다시 블루투스를 통하여 TMS320F28035 프로세서로 전송되면 그에 따라 엘리베이터가 목표 층수로 이동한다. 여기까지 걸리는 시간은 대략 0.5초에서 1초 정도이며 이는 관리자가 임의로 조절 할 수 있다.

데이터의 송수신에 걸리는 시간을 조절하는 이유는 상황에 따른 회선의 상태, 즉 블루투스의 연결 상태, 인터넷의 상태에 따라 지연이 발생하면

우선 순위 오류가 발생할 수 있으므로 충분한 대기시간을 줘야 하는 경우가 생길 수 있기 때문이고, 본 논문에서는 최대 2초까지 대기 시간으로 설정할 수 있도록 하였다.

4.3 음성인식기 구현

구글 인식기는 개발자를 위한 API를 지원하므로 특수한 경우가 아니라면 음성 인식기를 자체 제작할 필요는 없을 것이다. 본 논문에서도 구글에서 지원하는 음성 인식 API를 이용하여 기본적인 단어 인식기를 구현했다. 구글 인식기의 경우 단어에 대한 인식률이 매우 높으나 문장 속에 단어가 들어갈 경우, 인식기 자체가 문장을 고려하지 않고 있기 때문에 상당한 오류가 발생한다. 따라서 본 논문에서는 예를 들어 "7층으로 가라."와 같이 사용자가 자연스럽게 연속된 문장으로 명령을 내리는 경우는 고려하지 않았다. 한편, 개개인의 발음상 편차로 인하여 목표 층수의 표현이 모호하면 오작동을 일으킬 수 있는데, 이 문제점을 해결할 수 있는 한 가지 방법은 스마트폰을 사용하는 개개인의 발음으로 인식된 결과를 수집하여 서버에 저장해 놓는 것이다. 예를 들어 특정 명령어를 발음해도 계속해서 다른 단어들로 인식이 된다면 그런 단어들을 모아서 관리하고 해당 사용자의 명령에 오인식이 발생하면 서버에 저장된 오인식 발생 패턴을 이용하여 오인식 결과를 바로 잡는 것이다. 다른 사람에게는 일반적인 인식률을 보이겠지만 해당 사용자에게는 매우 높은 인식률을 점차적으로 보여줄 수 있는 방식이나 해당 화자를 인식하고 별도로 서버를 관리해야하는 어려움이 따른다.

기본적인 구글 음성 인식기는 이용자가 원하는 층수를 말하면 음성 인식기를 통해 구글 데이터베이스를 거쳐, 95%정도 일치하는 단어들을 유추하여 복수의 샘플을 다시 전송해 준다. 그중에 원하는 값에 최대한 일치하는 결과를 선택한다. 여기까지는 화자 독립, 즉 누가 이용하던지 거의 일정한 신뢰도를 보인다. 하지만 이번 데모 시스템에서 제안한 방식은 이용자가 말을 했을 때 구글에서 보내온 데이터 중에 신뢰도 90%이상의 데이터를 별도로 구축한 서버에 저장하여 DB의 양을 늘리면서 확실적인 신뢰도의 향상을 기대하는 방식이다. 비록 앞서 언급한 화자별 DB를 이용하지는 않지만 점차적으로 DB의 크기가 커짐에 따라 이용자에게 점점 맞춰지는 '화자 중속'과 유사한 방식으로 변하게 된다. DB의 크기가 무한정 커지는 것을 막기 위해 기존에 저장된 데이터와 중복되는 것은 자체적으로 삭제한다.

한 가지 문제점은, DB의 크기가 무한정 커지지 않도록 제한하고 있지만 서버에 축적되는 데이터는 사용할수록 계속 쌓여간다. DB의 크기가 단순히 커지기만 하면 잘못된 단어가 입력되어 오작동

을 일으킬 가능성이 있다. 예를 들어 7층을 가기 위해 '7층' 이라고 말했지만 이용자의 발음상 "7층" 과 "시청"의 발음상 차이가 모호하다면 Android는 두 단어의 사전적 관계성이 아닌 발음으로만 구분하므로, "시청"이라는 단어도 "7층"으로 인식 할 수도 있다는 것이다.

이 문제를 해결하기 위한 방법은 두 가지 방법이 있다. 사전적 의미가 판이하게 다른 경우는 배제하는 경우와 관리자가 서버 DB를 업데이트 하는 방법이다.

만약 "칭층", "칭층", "칠층" 과 같이 의미는 없으나 발음이 비슷한 경우는 "7층"으로 인정하고, "시청", "칠층"과 같이 어느 정도 발음이 비슷하지만 사전적 의미가 판이하게 다를 경우 배제하는 방법이다. 자동적으로 사전적 의미를 검색하며 배제하기 위해서는 공개된 의미 사전이 필요한데 공개된 자료가 없어 이번 구현에서는 적용하지 못하였다. 의미 사전이 이용 가능하다면 이 부분은 언제든 적용이 가능하다. 현재 데모 시스템에 구현된 방법은 저장된 DB를 확인하여 판이하게 다른 단어는 자체적으로 삭제하여 관리한다.

5. 구현

5.1 엘리베이터



그림 5. 내부 배선 및 센서와 Led

그림 5와 같이 엘리베이터 내부에 센서와 LED를 설치했다. 내부 강판을 제거하고 촬영하여 복잡하게 보이지만 완전히 조립된 데모 시스템에서는 카의 움직임에 영향을 끼치지 않게끔 정리된다. 단락의 위험이 있는 부분은 모두 절연 테이프를 마감되어 있으며 센서와 LED가 좌우 각각 6개 총 24개로 구성되어 있어 실제 엘리베이터처럼 up, down 버튼이 구별 되어 있다.

LED와 적외선 거리센서의 컨트롤은 기본적으로 A/D 변환기를 이용하였다. 센서에 사람 또는 물체가 감지되면 해당 층의 LED가 점등되어 목표 층 수로 설정 되었다는 것을 알려주고, 카가 도착하면 점등되었던 LED는 소등된다. 여러 층에서 중복 감지되었을 때도 해당 층의 LED가 모두 점등 되어 있다가 도착하는 순서대로 소등된다.

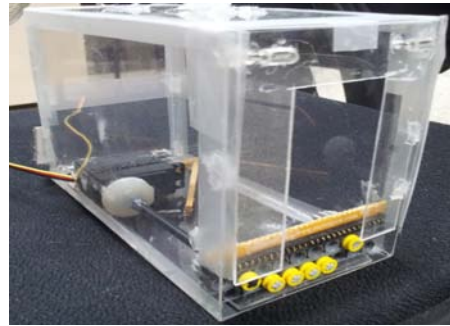


그림 6. 내부 카

엘리베이터 내부에서 움직이는 카는 상층부에 있는 모터에 연결된 줄을 통해 도르레 방식으로 상하 운동을 한다. 문의 개폐를 위해 PWM 방식을 이용하여 서브모터를 제어한다. 일반적인 좌, 우로 열리는 자동문 형식으로 목표층에 도달하고 완전히 정지한 후에 열리고 다시 닫힌 다음 나머지 동작을 수행한다.

5.2 엘리베이터 외부

그림 7은 완성된 엘리베이터의 외관이다. 앞서 언급했던 외부 강판과 센서, LED, 칩 등이 모두 조립되어 있고, 복잡하게 보였던 선이 모두 정리되어 보이지 않는 것을 확인할 수 있다. 측면과 후면은 내부가 보이지 않는다.

5.3 성능 시험 및 결과

구현한 음성인식 엘리베이터가 정상적으로 동작되는지, 정확도와 인식률을 테스트 했다. 오류가 일어날 가능성이 있는 경우와 실험 횟수가 거듭될 수록 인식률이 향상 되는지 확인하기 위해 각 경우마다 50번 정도 동작시킨 결과를 성능 시험 결과로 하였다. 인터넷과 블루투스의 연결 상태도 고려해야 하므로 3개의 각기 다른 스마트폰을 일정 거리 떨어진 곳에서 사용했다. 실험에 사용한 스마트폰은 갤럭시노트, 갤럭시S2, 갤럭시S3 이다.

표 1에서 인식 실패의 경우 구글 DB를 거쳐 와야 하는 시스템의 특성상 통신 상태가 불량하면 나타날 수 있는 오류이다. 실제 단어의 인식을 실패한 경우는 총 250회의 시험 운행 중 3번 있었다. 오차율은 1.2% 정도로 높은 인식률을 보여줬다.

물론 시험에 사용한 핸드폰은 사용자의 데이터가 충분히 모이지 않은 샘플이었으므로 자주 사용할 수록 오차는 더욱 줄어들 것으로 보인다. 시험 중 목표 층수에 도달하지 못한 경우가 1회 있었는데 주변 소음에 영향을 받아 잘못된 층수가 인식된 경우로 핸드폰의 특성상 주변 소음에 취약하다.

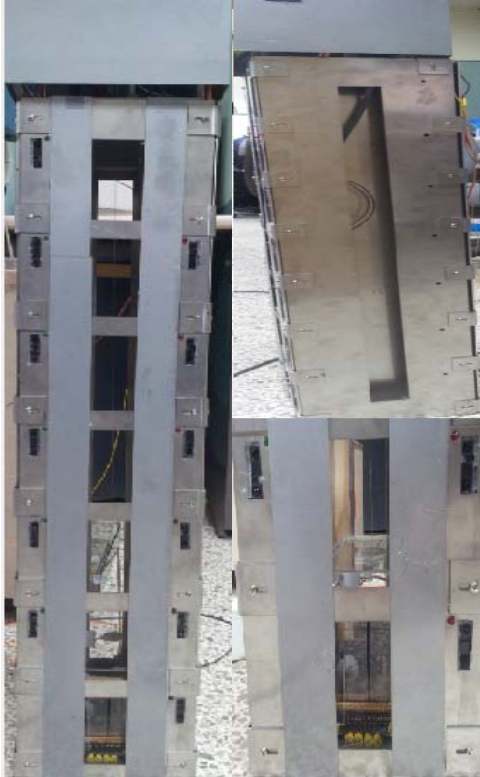


그림 7. 엘리베이터 외부 조립 완성사진

위의 결과는 데모 시스템 제작 전에 예상 했던 성능보다 좋은 결과인데, 최초 인식률을 구글인식기의 일반적인 인식률인 95%을 다소 상회할 것으로 예상했지만 실제 시험에서는 더 높은 결과가 나왔다. 이는 단순히 구글의 인식결과를 이용하지 않고 앞서 언급한 바와 같이 별도의 서버에 축적된 사용자의 오류 패턴을 고려하여 오류가 수정되었기 때문이다. 하지만 주변 상황, 즉 소음이나 밀폐된 장소와 같은 특수한 상황에서의 시험이 결여되어 그 부분에 대한 결과는 95%보다 더 낮을 것으로 예상된다.

표 1. 실험 결과

실험 층수	실험 횟수	실패 횟수
1층에서 무작위 층으로	50	2회 (인식 실패, 목표층수의 오류)
5층에서 무작위 층으로	50	1회(인식 단어의 오류)
7층에서 무작위 층으로	50	1회(인식 실패)
1층에서 7층으로 이동 중	50	1회(인식 단어의 오류)
7층에서 1층으로 이동 중	50	1회 (인식 단어의 오류)

6. 결론

이번 논문에서는 장애인 혹은 어떠한 이유라도 버튼을 누를 수 없는 사람들을 위해 음성으로 조작이 가능한 엘리베이터를 구현하였다. 장애인을 위한 편의 시설은 복지, 사업적인 측면으로도 매우 유용한 아이템이라 생각된다. 본 논문에서 제안한 방식은 이미 설치된 엘리베이터를 그대로 활용하기 위해 블루투스 장치와 스마트 폰을 이용하였기 때문에 기존의 음성인식 엘리베이터보다 경제적이다.

음성인식은 단순히 구글의 인식 결과를 이용하지 않고, 별도의 서버를 설치하여 사용자의 오류 패턴 데이터를 이용하여 인식 오류를 개선하도록 하였다.

본 논문에서 제안한 방식은 병원이나 복지 시설 처럼 장애인이나 노약자 분들이 유용하게 사용할 수 있을 것으로 사료된다.

참 고 문 헌

- [1] 송도선, 정동규, “음성인식 기반 엘리베이터 제어시스템 설계”, *한국정보기술학회 하계 종합 학술대회 논문집*, 2010.
- [2] L.Rabiner, B. Juang, *Fundamentals of Speech Recognition*, Prentice Hall, 1993.
- [3] TMS320F29035 Datasheet, Texas Instruments.
- [4] Code Composer Studio v.4 User's Guide, Texas Instruments.