JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 26 No. 9, pp. 57-64, September 2021
https://doi.org/10.9708/jksci.2021.26.09.057

Implementation of Speech Recognition and Flight Controller Based on Deep Learning for Control to Primary Control Surface of Aircraft

Hwa-La Hur*, Tae-Sun Kim**, Myeong-Chul Park**

*Professor, Dept. of Aeronautical Software Engineering, Kyungwoon University, Gumi, Korea
**Professor, Dept. of Avionics Engineering, Kyungwoon University, Gumi, Korea
**Professor, Dept. of Avionics Engineering, Kyungwoon University, Gumi, Korea

[Abstract]

In this paper, we propose a device that can control the primary control surface of an aircraft by recognizing speech commands. The speech command consists of 19 commands, and a learning model is constructed based on a total of 2,500 datasets. The training model is composed of a CNN model using the Sequential library of the TensorFlow-based Keras model, and the speech file used for training uses the MFCC algorithm to extract features. The learning model consists of two convolution layers for feature recognition and Fully Connected Layer for classification consists of two dense layers. The accuracy of the validation dataset was 98.4%, and the performance evaluation of the test dataset showed an accuracy of 97.6%. In addition, it was confirmed that the operation was performed normally by designing and implementing a Raspberry Pi-based control device. In the future, it can be used as a virtual training environment in the field of voice recognition automatic flight and aviation maintenance.

▸ **Key words**: Speech Recognition, CNN, MFCC, Flight Controller, TensorFlow

[요 약]

본 논문에서는 음성 명령을 인식하여 비행기의 1차 조종면을 제어할 수 있는 장치를 제안한다. 음성 명령어는 19개의 명령어로 구성되며 총 2,500개의 데이터셋을 근간으로 학습 모델을 구성한다. 학습 모델은 TensorFlow 기반의 Keras 모델의 Sequential 라이브러리를 이용하여 CNN 모델로 구성되며, 학습에 사용되는 음성 파일은 MFCC 알고리즘을 이용하여 특징을 추출한다. 특징을 인식하기 위한 2단계의 Convolution layer 와 분류를 위한 Fully Connected layer는 2개의 dense 층으로 구성하였다. 검증 데이터셋의 정확도는 98.4%이며 테스트 데이터셋의 성능평가에서는 97.6%의 정확도를 보였다. 또한, 라즈베리 파이 기반의 제어장치를 설계 및 구현하여 동작이 정상적으로 이루어짐을 확인하였다. 향후, 음성인식 자동 비행 및 항공정비 분야의 가상 훈련환경으로 활용될 수 있을 것이다.

▸ **주제어**: 음성 인식, 합성곱 신경망, MFCC, 비행 제어장치, 텐서플로우

• First Author: Hwa-La Hur, Corresponding Author: Myeong-Chul Park
*Hwa-La Hur (haru@ikw.ac.kr), Dept. of Aeronautical Software Engineering, Kyungwoon University
**Tae-Sun Kim (tskim@ikw.ac.kr), Dept. of Avionics Engineering, Kyungwoon University
**Myeong-Chul Park (africa@ikw.ac.kr), Dept. of Avionics Engineering, Kyungwoon University
• Received: 2021. 08. 25, Revised: 2021. 09. 23, Accepted: 2021. 09. 24.

http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945

I. Introduction

자동 음성 인식 기반 인터페이스를 통한 동작 에이전트는 일상생활의 일부가 되어 다양한 영역에서 활용되고 있으며 구동 장치의 조작 없이 음성을 통하여 각종 장치를 제어할 수 있게 되었다[1]. 또한, 일정한 대화 패턴을 유지하는 상담 및 예약 콜 센터 등의 상업적 용도에서 구글의 어시스턴트, 삼성의 빅스비, 애플의 시리, 아마존의 알렉사 등과 같이 개인용 모바일 단말장치를 통한 명령 인식 등에 사용되는 등 광범위하게 응용되고 있다. 대부분의 동작은 인식 모델의 복잡성과 방대한 연산으로 인하여 클라우드 환경에서 수행되는 것이 일반적이다. 사용자의 오디오는 “하이 빅스비”와 같이 명시적인 동작의 시작만의 인지하고 후속되는 음성 명령어나 지시어는 백엔드 서버로 전송되어 기록되고 분석되어진다. 이러한 자동 음성 인식 인터페이스는 구동 장치를 제어할 수 없는 장애우나 전문적인 감각을 통한 조작이 수반되는 다양한 조종장치을 손쉽게 제어할 수 있는 장점을 가진다[2]. 특히, 정해진 명령 분류를 위한 음성 인식 인터페이스는 입력된 음성 신호의 전체를 사용하지 않고 대표 특징을 추출하여 딥러닝 기술을 사용하는 것이 일반적이며 최근 각광받고 있는 합성곱 신경망(CNN : Convolutional Neural Networks)[3,4]과 오픈소스 기반의 딥러닝 프레임워크를 사용한다[5]. 자동 음성 인식분야의 특징 추출 방법으로 MFCC(Mel-Frequency Cepstral Coefficients)[6] 알고리즘이 널리 사용되고 있으며, MFCC는 멜 스펙트럼(Mel Spectrum)에서 켑스트럴(Cepstral) 분석을 통해 추출한 값으로 특징벡터(Feature)화 한다. 머신러닝이나 딥러닝의 학습을 위해서는 데이터를 벡터화해야 한다는 점에서 음성 자동인식분야에서 널리 사용되고 있는 알고리즘이다. 다양한 자동 제어 기능을 탑재한 이동 수단이 많지만 아직 항공기의 경우에는 음성인식을 통한 자동 비행에 대한 연구가 미흡한 것이 사실이다. 항공기 조종은 기존의 케이블 방식에서 전자 조종 방식을 통해 보다 손쉽게 조종할 수 있었으며 조종실의 조작 인터페이스도 간편화되어 조종사의 시각을 안정화 및 오토파일럿 등 여러 자동화 기술들이 도입되어 조종의 용이성을 확보하고 있다. 이에 본 논문은 조종 명령에 따른 자동 인식을 통한 각종 조종 계기의 동작을 구현하여 음성인식을 통한 자동 비행에 대한 가능성을 타진해 보고자 한다. 또한, 연구 결과물을 활용하여 항공정비 분야에서 훈련생의 가상환경을 통한 교구로 활용될 것을 기대한다. 전체적인 시스템의 구성은 Fig. 1과 같다.

본 연구에서는 라즈베리 파이 기반에서 음성으로 항공기 1차 조종면 제어를 위한 음성인식 조종 장치를 구현하고자 한다. 음성 인식을 위한 학습은 TensorFlow기반 합성곱 신경망 학습 모델을 구축하여 라즈베리 파이에서 모델 추론을 원활하게 실행하기 위하여 TensorFlow Lite 형태로 변환하여 사용한다. 조종 명령어는 4개의 영역에 총 19개의 명령어로 설계되며 7개(Left, Right, Stop, Up, Down, On, Off)의 오디오 데이터셋은 구글의 Speech Commands Dataset[7]을 이용하고 나머지 오디오 데이터셋은 자체 제작하여 음성 데이터셋을 구성하였다. 논문의 구성은 2장에서 음성인식 기술과 MFCC 알고리즘에 대해 살펴보고 3장에서 파이썬으로 구현된 CNN 학습모듈과 라즈베리 파이 기반의 음성인식 비행 제어장치의 구현과정을 설명하고 4장에서 결과에 관해 기술한다.

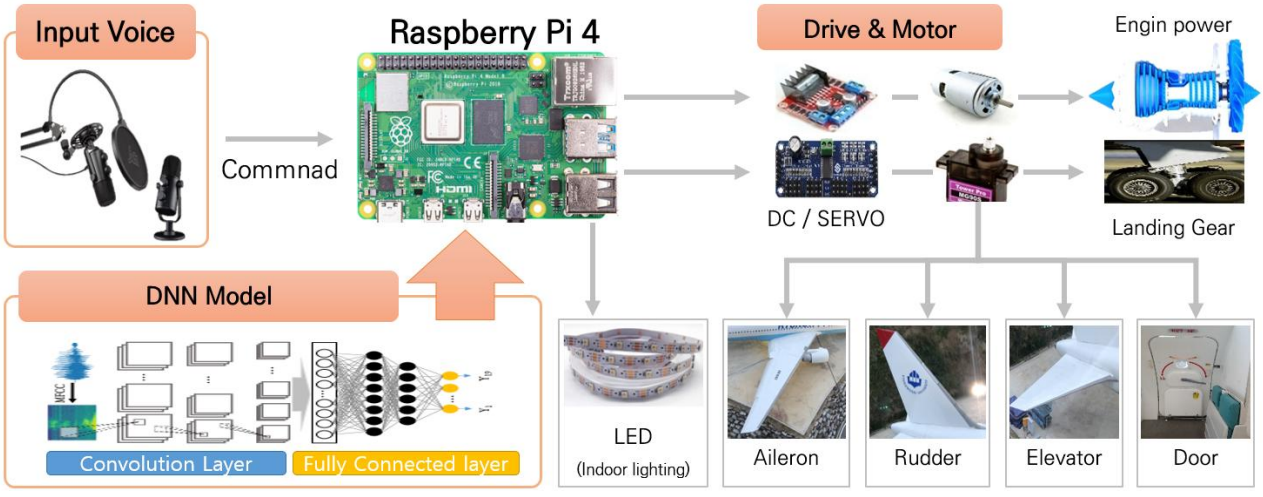

Fig. 1. Overview of The Proposed System

II. Background

1. Speech Recognition Technology

음성인식은 머신러닝이나 딥러닝을 이용하여 문제해결을 위한 명시적 프로그래밍 없이 입력 데이터셋의 학습을 통하여 내부의 반복되는 패턴과 통찰력을 이용하여 식별하는 과정으로 이루어진다[8]. 장동열[9]은 로봇을 대상으로 음성 명령을 통하여 모바일 매니플레이터와 연동하여 특정 장소에 있는 큐브를 컨테이너로 이동시키는 시스템을 개발하여 신체적 불편함이 있는 사람들을 대신하여 물건의 이송하는 식당, 카페 등의 배달 로봇과 제조업 현장에서 사용할 수 있는 이송 로봇의 가능성을 보였다. 소순원[10]은 한국어 음성 파일에서 MFCC을 추출한 데이터셋을 심층 인공 신경망에서 학습하여 화자의 연련을 분류하는 모델을 제안하였다. Tomasz[11]은 Fig. 2와 같이 조종사가 비행기를 제어하는데 사용하는 음성 명령어의 유형을 제시하고 주요 요구사항에 대해 정의하였다. 하지만 이는 조종 장치의 관점이 아닌 3축에 관한 회전 운동을 제어하는 동작에 국한되어 있다.

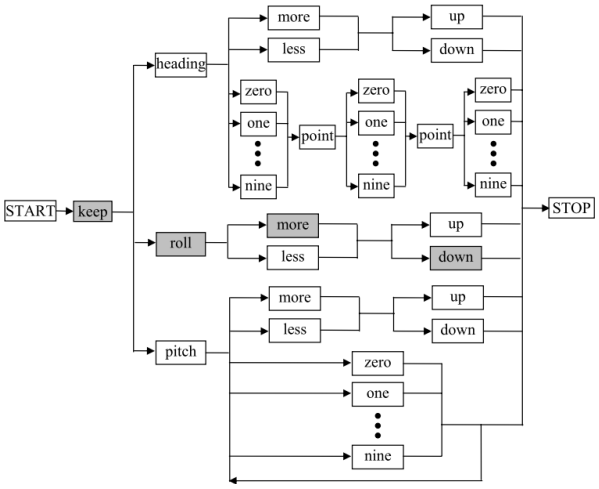

Fig. 2. Exemplary Word Net for Voice Controlled Aviation System[37]

김성우[12]는 MFCC 특징 추출을 통한 HMM(Hidden Markov Model) 알고리즘 적용으로 전투기용 음성명령 시스템을 제안하였지만, 구문 트리가 복잡하고 후처리 과정을 수행해야하는 제한점을 가진다. Siyaev[13]는 항공기의 가상 환경 제어를 위한 스마트 안경을 활용한 CNN모델 기반의 상호 작용 시스템을 제안하였다. 특히, 영어와 한국어가 혼합된 요청에 대한 피드백을 제공하는 특징을 가진다. Lin[14]은 음성에 장애를 가진 대상자의 음성 명령어를 정확히 인식하기 위하여 CNN 모델과 PPG(Phonetic Posteriorgram)을 결합한 시스템을 제안하였다. Ruben[15]는 도메인 기반의 영어와 스페인어로 구성된 9개의 음성 명령어를 해석하여 드론을 제어하는 시스템을 제안하였다

2. MFCC(Mel-Frequency Cepstral Coefficients)

본 논문의 CNN 모델의 입력으로 사용되는 오디오의 특징은 최종적으로 MFCC(Mel Frequency Cepstral Coefficients) 정보이다. MFCC는 음성 인식을 위해서 중요 특징만을 남기고 정보를 정제하는 방법으로 모델의 학습 데이터 셋으로 이용하게 된다. 입력된 오디오 파일(Wave)을 MFCC 형식으로 변환하는 과정은 Fig. 3과 같다. 실제 데이터 셋을 구성하기 위하여 녹음된 음성파일을 대상으로 각 타임 슬라이스(25ms)에 대한 FFT(Fast Fourier Transform)를 계산하여 해당 타임 슬라이스에 대한 주파수 정보를 추출하게 된다. 그리고 추출된 스펙트럼을 대상으로 인간의 음성 인식은 1khz 이하의 저주파수 대역에 민감한 특징을 이용하여 저주파수 대역의 세밀한 분석을 위한 Mel-spaced filterbank 필터를 적용하여 Mal 스펙트럼을 생성하는데 각 에너지가 누적된 일차원 수치 벡터를 가지게 된다.

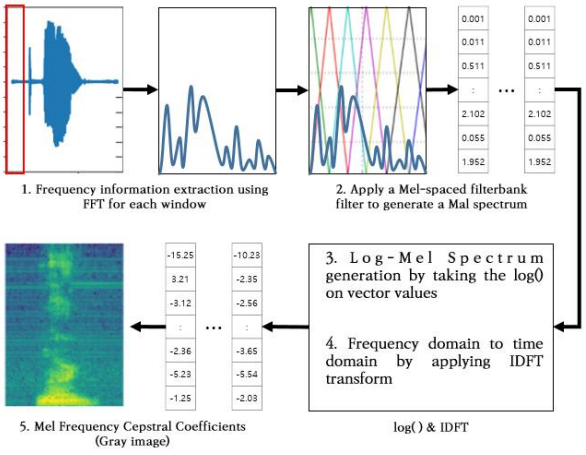

Fig. 3. Block Diagram of Feature Extraction Process in Audio File(MFCC)

이 벡터 값에 로그를 취하여 로그 멜 스펙트럼(log-Mel Spectrum)을 만든다. 로그를 취하는 이유는 인간의 음성 인식이 로그 스케일에 가깝기 때문으로 소리 인식을 위해 에너지가 100배일 경우, 두 배의 소리로 인식한다는 의미이다. 이 결과에 IDFT(Inverse Fourier Transform) 변환을 적용하여 주파수 도메인을 시간 도메인 정보로 바꾸고 내부 변수 간의 상관관계를 해소하기 위한 결과를 MFCC라 하며 124*129 사이즈의 이차원 배열 구조를 가

지게 된다. 예측 모델의 학습 용이성을 위하여 그레이스케일 이미지로 변환하여 모델의 입력 값으로 사용한다. Fig. 4는 데이터셋으로 입력된 Waveform을 MFCC로 변환한 코드와 결과를 보인 것이다.

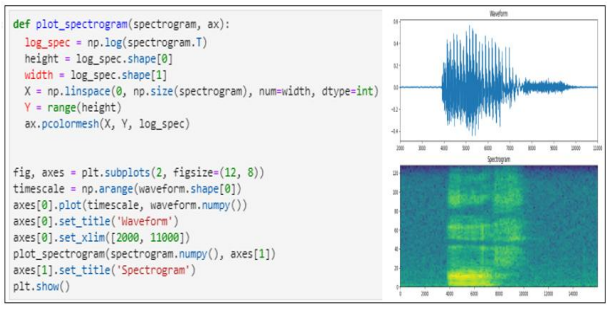

Fig. 4. Code and Result to Convert Waveform to MFCC

III. Design & Implementation

1. Configuration of Datasets

항공기 조종면(Fight Control Surface)은 1차 조종면과 2차 조종면으로 구분할 수 있다. 1차 조종면은 항공기의 3축의 회전운동을 발생시키는 역할에 해당되며 보조 날개에 해당하는 에일러론(Aileron)은 세로축(Longitudinal axis)의 롤링(Rolling) 동작으로 양 날개에 부착되어 있다. 방향타에 해당하는 러더(Rudder)는 Z 축(Vertical axis)의 요잉(Yawing) 동작으로 수직 꼬리 날개의 조정을 의미한다. 마지막으로 승강타에 해당하는 엘리베이터(Elevator)는 Y 축(Lateral axis)의 피칭(Pitching) 동작으로 수평 꼬리 날개의 조정을 의미한다. 2차 조종면은 슬랫(Slat), 플랩(Flap), 스포일러(Spoiler)로 구성되면 슬랫 동작은 앞날개 앞면부에 부착되며 공기 흐름의 박리를 방지하여 양력을 증가시키는 용도이며 플랩은 동일한 받음각에서 양력 발생과 양력 증가를 담당하며 주로 착륙 시에 양력을 유지하며 낮은 속도로 활주로에 접근하기 위한 용도이다. 스포일러는 착륙 시에 날개의 윗면이 위로 들리는 동작으로 큰 저항을 발생시켜 주로 착륙 시 접지력을 높이는 용도로 사용된다. 훈련을 위한 데이터 셋은 기본적으로 Google Speech Commands 데이터 셋을 사용하였으며 해당 데이터 셋에 수록되지 않은 명령에 대해서는 단어별 50명의 음성을 1초 단위로 5가지 내외에 장단, 억양들을 고려하여 다수의 음성으로 녹음하여 수집하였다. 남성과 여성의 비율을 동일하게 했으며, 개인별 두 가지 음색(High Tone, Low Tone)으로 데이터 셋을 구성하였다. 음성 식별 대상인 19개 명령어에 대하여 총 2,500개의 데이터 셋을 대상으로 훈련 데이터 80%, 검증 데이터셋을 10%, 테스트 데이터셋을 10%로 분할하여 사용한다.

Table 1. Configuration of Datasets

Part	Command	Number	Source
PILOTING	Piloting	200	Self-recording
	Left	200	Google
	Right	200	Google
	Rise	200	Self-recording
	Descent	200	Self-recording
DOOR	Door	100	Self-recording
	Open	200	Self-recording
	Close	100	Self-recording
LANDING	Landing	100	Self-recording
	Takeoff	100	Self-recording
	Land	100	Self-recording
ENGIN	Engin	100	Self-recording
	Start	100	Self-recording
	Stop	100	Google
	Up	100	Google
	Down	100	Google
LIGHTING	Lighting	100	Self-recording
	On	100	Google
	Off	100	Google

2. Structure of CNN Model

학습 모델 구성의 주요 절차는 Fig. 5와 같이 오디오 파일에서 특징을 추출하고 CNN 모델을 통하여 음성 단어를 식별하는 학습을 훈련한다.

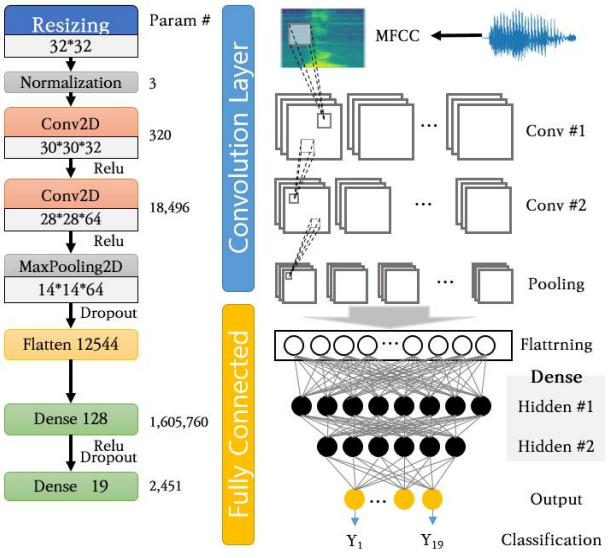

Fig. 5. The Proposed CNN Model Structure

완성된 모델은 라즈베리 파이에서 로드하여 마이크로 입력되는 음성을 추론 엔진을 통하여 실시간으로 예측하게 된다. CNN 모델의 구성은 크게 두 가지 영역으로 구분

된다. Convolution 레이어는 특징을 추출하는 영역이고 Fully Connected 레이어 실질적인 분류를 위한 영역이다. 실질적 CNN 모델의 구축은 파이썬으로 구현된 Keras 라이브러리를 사용하였다. 먼저, Fig. 6과 같이 신경망 모델은 Keras 모델의 Sequential 라이브러리를 이용하여 레이어를 선형으로 연결하여 적층하는데 학습 모델은 CNN 모델을 사용하였으며 빠른 학습을 위하여 입력 세이프를 다운 샘플링(Resizing)하여 평균과 표준편차를 기반으로 이미지를 정규화(Normalization)하였다.

```
for spectrogram, _ in spectrogram_ds.take(1):
  input_shape = spectrogram.shape
print('Input shape:', input_shape)
num_labels = len(commands)

norm_layer = preprocessing.Normalization()
norm_layer.adapt(spectrogram_ds.map(lambda x, _: x))

model = models.Sequential([
    layers.Input(shape=input_shape),
    preprocessing.Resizing(32, 32),
    norm_layer,
    layers.Conv2D(32, 3, activation='relu'),
    layers.Conv2D(64, 3, activation='relu'),
    layers.MaxPooling2D(),
    layers.Dropout(0.25),
    layers.Flatten(),
    layers.Dense(128, activation='relu'),
    layers.Dropout(0.5),
    layers.Dense(num_labels),
])

model.summary()
```

```
Input shape: (124, 129, 1)
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
resizing (Resizing)          (None, 32, 32, 1)         0
_________________________________________________________________
normalization (Normalization (None, 32, 32, 1)         3
_________________________________________________________________
conv2d (Conv2D)              (None, 30, 30, 32)        320
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 28, 28, 64)        18496
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 14, 14, 64)        0
_________________________________________________________________
dropout (Dropout)            (None, 14, 14, 64)        0
_________________________________________________________________
flatten (Flatten)            (None, 12544)             0
_________________________________________________________________
dense (Dense)                (None, 128)               1605760
_________________________________________________________________
dropout_1 (Dropout)          (None, 128)               0
_________________________________________________________________
dense_1 (Dense)              (None, 19)                2451
=================================================================
Total params: 1,627,030
Trainable params: 1,627,027
Non-trainable params: 3
_________________________________________________________________
```

Fig. 6. Model Generation Code and Summary

입력된 2차원 이미지를 대상으로 첫 번째 컨볼루션 레이어를 추가하는데 초기 입력 세이퍼는 32*32 해상도의 이미지이며 3*3 크기의 필터 32개를 적용하고 활성화 함수(Activation function)는 ReLU를 사용하였다. 필터는 이미지에서 특징을 분리해내는 기능을 담당하는데 출력 공간의 차원이 결정되고 필터의 크기이며 합성곱 연산의 결과로 32*32의 이미지 사이즈가 30*30이 된다. 그래서 출력 형태(Output Shape)는 30*30*32가 된다. 두 번째 컴볼루션 레이어는 64개의 필터를 사용하며 출력 형태는 28*28*64가 되고 이미지의 손실을 줄이기 위해 폴링(Pooling)작업을 추가로 수행한다. 폴링작업은 합성곱에서 얻어진 특징 맵(Feature map)을 대상으로 값을 샘플링하여 정보를 압축하는 과정이다. 구현에서는 Max-pooling을 이용하는데 특정 영역에서 가장 큰 값을 샘플링하며 기본적인 필터는 2*2 크기로 출력 형태는 14*14*64로 압축되게 된다. 폴링된 이미지를 연속 1차원 단일 벡터로 변환하기 위하여 평탄화(Flattening) 작업을 수행하여 학습을 위한 Fully-Connected layers의 입력 유형으로 변경한다. 그리고 분류를 위한 Fully-Connected layers의 Dense 층은 두 개를 추가하는데, 첫 번째 Hidden 레이어의 노드 수는 128개로 정의하고 활성화 함수는 ReLU을 사용한다. 그리고 신경망 모델의 가정 일반적인 문제점 중에 하나인 과적합(Overfitting) 문제를 해결하기 위하여 Dropout 함수를 이용하여 정규화를 취한다. 이는 학습 단계에서만 적용되면 본 연구에서는 폴링작업 이후와 첫 번째 Hidden 레이어의 결과물을 대상으로 드롭아웃을 적용하였다. Dropout(0.25)는 입력된 노드에 대하여 25%를 무작위로 0으로 만드는 작업을 의미한다. 19가지의 명령어를 식별하는 것이 목적이기 때문에 마지막 Hidden 레이어 에는 출력 노드가 19개가 된다.

손실 함수를 통해 얻은 손실값으로부터 모델을 업데이트하기 위한 옵티마이저(Optimizer)는 Adam 알고리즘을 이용한다. 최적화 방법은 경사 하강법을 어떤 방법으로 사용할지를 결정하는 것으로 관성 부여(Momentum)와 상황에 따라 이동 거리를 조정(Adagrad)하는 두 방식을 합친 것이 Adam 방식이다. 손실 함수는 레이블 형태가 3개 이상의 클래스로 구성되어 있기 때문에 단순 범주형에 해당하는 Sparse_categorical_crossentropy 함수를 이용한다. 평가지표에 해당하는 메트릭(Metrics)은 훈련을 모니터링하기 위해 사용하는 것으로 분류 메트릭에 해당하는 accuracy 로 정의한다. 이렇게 학습과정을 설정하고 모델의 훈련을 위하여 에포크를 20으로 설정하고 Overfitting을 방지하기 위하여 Early Stopping을 설정하였다. verbose를 1로 지정하여 언제 학습을 멈추었는지를 화면에 표시하게 하였고 patience는 2로 설정하여 성능이 증가하지 않는 에포크를 2번까지 허용하게 설정하였다. 다양한 값으로 테스트해 본 결과, 본 모델에서 가장 적절한 에포크는 20으로 지정하였다.

```
Epoch 1/20
40/40 [==========] val_loss: 2.4509 - val_accuracy: 0.2720
Epoch 2/20
40/40 [==========] val_loss: 1.9942 - val_accuracy: 0.4640
Epoch 3/20
40/40 [==========] val_loss: 1.6738 - val_accuracy: 0.5640
Epoch 4/20
40/40 [==========] val_loss: 1.3545 - val_accuracy: 0.6280
Epoch 5/20
40/40 [==========] val_loss: 1.1489 - val_accuracy: 0.6840
Epoch 6/20
                    ⋮
40/40 [==========] val_loss: 0.2079 - val_accuracy: 0.9640
Epoch 17/20
40/40 [==========] val_loss: 0.1789 - val_accuracy: 0.9640
Epoch 18/20
40/40 [==========] val_loss: 0.1683 - val_accuracy: 0.9800
Epoch 19/20
40/40 [==========] val_loss: 0.1508 - val_accuracy: 0.9760
Epoch 20/20
40/40 [==========] val_loss: 0.1294 - val_accuracy: 0.9840
```

Fig. 7. Model Training Process (epoch = 20)

3. Implementation of Flight Controller

제안하는 비행 제어장치의 회로도는 Fig. 8과 같다. 전체적인 구성은 메인부를 중심으로 입력부, 출력부, 구동부로 구성되어 있다. 입력부는 마이크를 통해 음성을 메인부

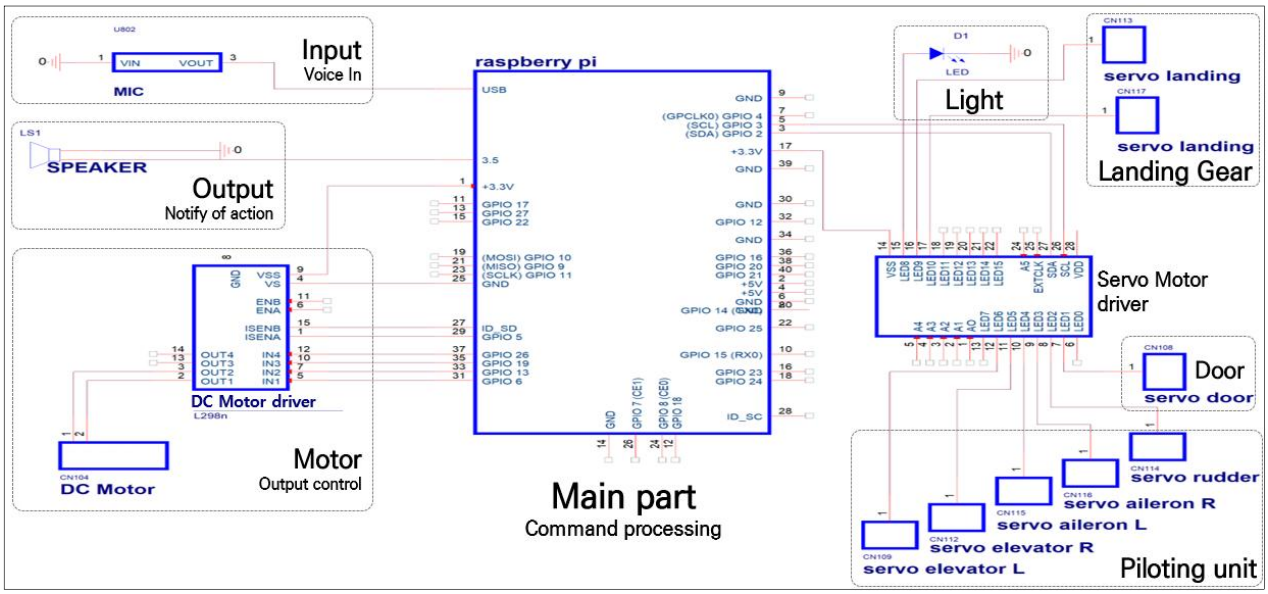

Fig. 8. Circuit Design of Flight Controller

인 MCU(라즈베리 파이4B)로 전달해준다. 입력부에서는 동작(조종면, 엔진출력, 출입문, 조명, 착륙장치)과 작동 값의 두 개의 입력을 받는다. 메인부는 입력부에서 들어온 명령을 전달받아 학습모델을 통하여 명령을 식별하고 해당 동작을 조종면, 출입문, 엔진출력, 실내등, 착륙장치로 구분하여 분류하고 작동 값을 구동부로 전달한다. 이 때 구동부의 각 모터들은 값에 맞게 동작하고 충분한 작동 전압을 받기 위해 라즈베리파이와 모터들 사이에 모터 드라이버를 통한다. 구동부가 명령에 맞게 작동되면 스피커를 통해 명령이 정상적으로 작동되었음을 조종사에게 알린다. 출력부의 스피커는 구동부의 동작을 보고하기도 하지만 질문을 하여 조종사의 다음 명령을 유도한다. 제어 장치의 동작이 시작되면 전체 구성요소에 대한 초기화가 이루어진다. 초기화가 완료되면 사용자 음성 입력을 위한 대기 음성이 출력된다. 대기 입력중 명령 음성을 받아들이면 동작의 주요 분류를 식별한다. 식별되는 분류는 조종면 제어(PILOTING), 출입문 개폐(DOOR), 착륙장치(LANDING), 엔진출력(ENGIN), 실내조명(LIGHTING)으로 분류된다. Fig. 9는 조종면 제어 부분을 간략히 보인 흐름도 이다. A는 다른 분류의 명령으로 인식되어 별도의 흐름을 따른다. 조종면 제어가 인식되면 모션 식별을 위한 2차 음성을 식별한다. 식별되는 모션은 기체의 선회(TURNING(Left,Right)), 고도상승(RISE)과 하강(DESCENT)이며 실제 동작은 지시되는 각도에 따라 서보모터를 동작시킨다. 출입문의 동작은 열림(OPEN)과 닫힘(CLOSE)으로만 식별되며 착륙장치의 동작은 착륙과 이륙에 해당하는 랜딩 기어의 올림(TAKEOFF)과 내림(LAND)으로 식별되어 동작한다. 엔진출력 동작은 엔진 시동(START), 엔진 정지(STOP), 엔진 출력 상승(UP), 엔진 출력 감소(DOWN)의 네 가지로 구분된다. 마지막으로 실내조명은 조명의 켜짐(ON)과 꺼짐(OFF)으로 동작한다.

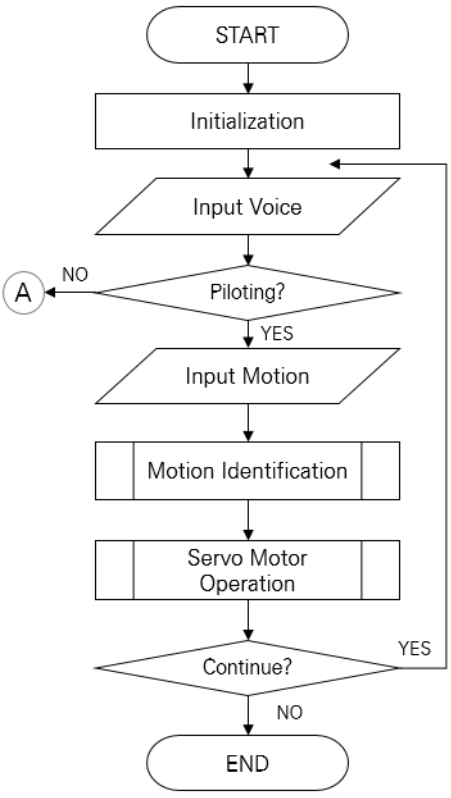

Fig. 9. Flow Chart for Controlling the Control Surface

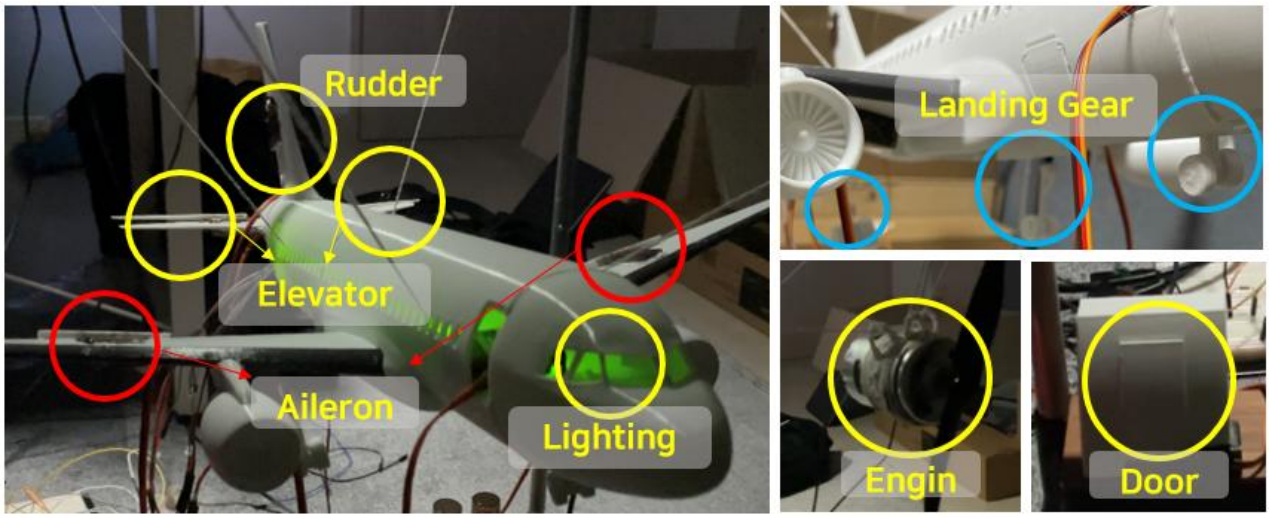

Fig. 10. Prototype of Flight Controller

Fig. 10은 실제 구현된 비행 제어장치를 보인 것이다. Fig.8의 설계도에 따라 제작되었으며 음성 명령 입력 시 각 요소별 동작을 확인하기 위한 용도이다.

IV. Conclusions

본 논문은 음성 명령을 통하여 항공기의 1차 조종면을 제어하는 장치를 구현하였다. 음성 명령 인식은 TensorFlow 기반의 CNN 학습 모델을 구현하고 실제 동작은 라즈베리 파이에서 모델 추론을 통하여 동작한다. 조종 명령은 총 19개로 구성되며 데이터셋은 2,500개의 음성 파일로 구성하였다. 훈련이 완료된 모델에 대한 검증 데이터셋의 정확도는 Fig. 7에서 98.4%로 확인하였으며, 학습 및 검증 손실 곡선이 Fig. 11과 같이 적절히 개선되고 수렴됨을 확인하였다. 이는 모델이 과적합(Overfitting)되지 않았음을 의미하며 안정적인 학습결과를 보인 것이다.

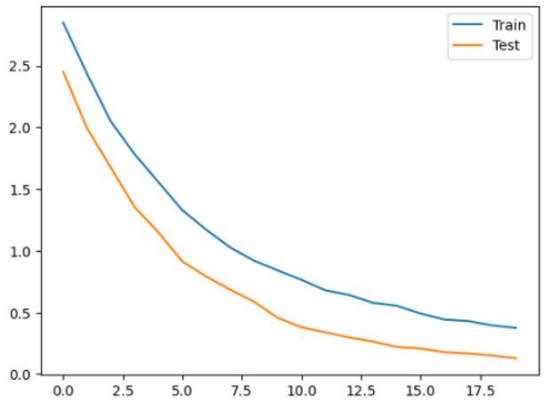

Fig. 11. Learning Curve of Model

또한 테스트 데이터셋을 통한 성능평가에서는 97.6%의 정확도를 보였고 테스트 데이터셋의 각 명령어 대하여 모델이 얼마나 정확하게 수행되었는지 확인하기 위한 오차 행렬(Confusin Matrix)의 결과에서도 Fig. 12와 같이 우수한 성능을 보였다. lighting과 piloting 명령이 1건 정도의 오류를 보였고 전반적으로 모든 명령에 대한 식별력이 매우 높은 것을 알 수 있다.

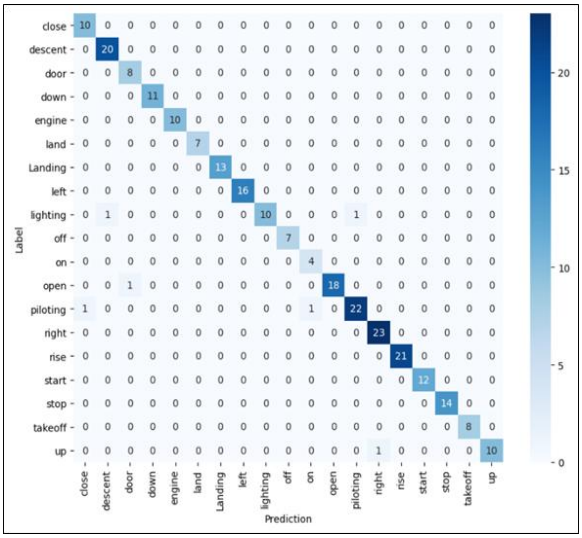

Fig. 12. Confusin Matrix via Heatmap

연구의 결과는 음성 조종 명령의 자동 인식을 통한 각종 조종 계기의 동작을 통하여 음성인식 자동 비행에 대한 가능성을 확인하였고 항공정비 분야에서 훈련생의 가상 환경을 통한 훈련 교구로 활용될 수 있을 것으로 사료된다. 향후, 연속적 명령에 따른 구문 트리를 해석하고 인식하는 모델을 개발하여 실제 비행 훈련에 적용할 예정이다.

REFERENCES

[1] M. A. Anusuya and S. K. Katti, "Speech Recognition by Machine, A Review," International Journal of Computer Science and Information Security, IJCSIS, Vol. 6, No. 3, pp. 181-205, December 2009.

[2] Myeong-Chul Park et al., "Drone controller using motion imagery brainwave and voice recognition," Proceedings of the Korean Society of Computer Information Conference 28(2), pp. 257-258, July 2020.

[3] T. N. Sainath, A. Mohamed, B. Kingsbury and B. Ramabhadran, "Deep convolutional neural networks for LVCSR," 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 8614-8618, May 2013. DOI: 10.1109/ICASSP.2013.6639347

[4] Mitra. Vikramjit et al., "Evaluating robust features on Deep Neural Networks for speech recognition in noisy and channel mismatched conditions," Proceedings of the Annual Conference of the International Speech Communication Association, pp. 895-899, September 2014.

[5] V. Pratap et al., "Wav2Letter++: A Fast Open-source Speech Recognition System," IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 6460-6464, 2019. DOI: 10.1109/ICASSP.2019.8683535

[6] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 28, No. 4, pp. 357-366, August 1980. DOI: 10.1109/TASSP.1980.1163420

[7] Warden. Pete, "Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition," ArXiv Preprint ArXiv:1804.03209, 2018.

[8] A. B. Nassif, I. Shahin, I. Attili, M. Azzeh and K. Shaalan, "Speech Recognition Using Deep Neural Networks: A Systematic Review," IEEE Access, Vol. 7, pp. 19143-19165, 2019. DOI: 10.1109/ACCESS.2019.2896880

[9] Dongyeol Jang, Seungryeol Yoo, "Integrated System of Mobile Manipulator with Speech Recognition and Deep Learning-based Object Detection," The Journal of Korea Robotics Society, Vol. 16(3), pp. 270-275, Sep. 2021. DOI : 10.7746/jkros.2021.16.3.270

[10] Soonwon So et al., "Development of Age Classification Deep Learning Algorithm Using Korean Speech," Journal of Biomedical Engineering Research, Vol. 39, pp. 63-68, Apr. 2018. DOI : 10.9718/JBER.2018.39.2.63

[11] Tomasz Rogalskia and Robert Wielgatb, "A concept of voice guided general aviation aircraft," Aerospace Science and Technology, Vol. 14, pp. 321–328, Feb. 2010. DOI : 10.1016/j.ast.2010.02.006

[12] Seongwoo Kim, Mingi Seo, Yunghwan Oh and Bonggyu Kim, "A Study on Cockpit Voice Command System for Fighter Aircraft," KSAS, Vol. 41(12), pp. 1011-1017, Dec. 2013. DOI : 10.5139/JKSAS.2013.41.12.1011

[13] Aziz Siyaev, Geun-Sik Jo, "Towards Aircraft Maintenance Metaverse Using Speech Interactions with Virtual Objects in Mixed Reality," Sensors, 21(6), Mar. 2021. DOI : 10.3390/s21062066

[14] Yu-Yi Lin et al., "A Speech Command Control-Based Recognition System for Dysarthric Patients Based on Deep Learning Technology," Applied Sciences, Vol. 11(6), Mar. 2021. DOI : 10.3390/app11062477

[15] Ruben Contreras, Angel Ayala and Francisco Cruz, "Unmanned Aerial Vehicle Control through Domain-Based Automatic Speech Recognition," Computers, Vol. 9(3), Sep. 2020. DOI : 10.3390/computers9030075

Authors

Hwa-La Hur received a M.S. degree in Computer Engineering from Dong-a Univeristy in 1992, a Ph.D. degrees in Electronic Engineering from Pusan National Univeristy in 2001.

He is currently a Professor in the Department of Aeronautical Software Engineering, KyungWoon University. He is interested in Time-Dealy, Model predictive control, Remote control robot.

Tae-Sun Kim, he is a professor of department of avionics engineering at Kyungwoon University. He holds a doctorate degree in Electronic Engineering from Yeungnam University. From 1991 to 1995, he was a researcher in the TV Research Institute at LG Electronics. He is currently a Professor in the Department of Avionics Engineering, KyungWoon University. His research interests include image analysis, image system and signal processing.

Myeong-Chul Park received a B.S. degree in Computer Science from Korea National Open University in 1999, and the M.S. and Ph.D. degrees in Computer Science from GyeongSang National University in 2002 and 2007, respectively. He is currently a Professor in the Department of Avionics Engineering, KyungWoon University. He is interested in Visualization, Simulation, Education of Software, Virtual Reality, and Parallel Programming.