# A DEEP LEARNING APPROACH TO SPEECH BASED CONTROL OF UNMANNED AERIAL VEHICLES (UAVs)

Saumya Kumaar[1], Toshit Bazaz[2], Sumeet Kour[2], Disha Gupta[2],
Ravi M. Vishwanath[1] and S N Omkar[1]

[1]Indian Institute of Science, Bengaluru
[2]National Institute of Technology, Srinagar

## ABSTRACT

*Speech recognition has been one of the key research domains in computational signal processing. Despite high levels of computational complexity associated with achieving speech recognition in real-time, promising progress has been made under the umbrella of voice controlled robotics. This paper proposes an alternate approach to speech recognition for robotics applications, without adding on external hardware. We use a combination of spectrograms, MEL and MFCC features and a neural network based classification which is usually done offline, whereas the proposed method offers a remote real-time control of the robot that can be used to survey terrains that are otherwise impervious for humans, or monitor activities inside huge structures like wind-mills, gas pipelines etc. The trained model occupies lesser than 4MB on the storage medium of the platform and it also displays metrics of confidence and accuracy of prediction. The overall validation accuracy of the algorithm goes as high as 97% while the testing accuracy of the system is 95.4%. Since this is a classification algorithm, results have been presented on custom voice classification datasets.*

## KEYWORDS

*Deep Learning, Signal Processing, Unmanned Aerial Vehicles, Speech Recognition*

## 1. INTRODUCTION

Most speech recognition applications in robotics rely heavily on either hardware based systems (like VRBot, GeeTech, RKI-1199 etc.) or Googles Speech API. In both these cases, additional requirements come into picture in the form of extra hardware or the need for an internet connection.

Now most of the commercial/hobbyist robotic applications are built using System-On-Chip (SOCs) like Raspberry Pi, Odroid-XU4, Beaglebone etc. which run on Linux-based RTOS platforms and have reasonable computational capabilities. This paper proposes an alternate approach to speech recognition for robotics applications, without adding on external hardware. We use deep neural networks with only fully connected layers for recognizing different possible speech commands given to the drone, via spectrogram classification, in real time. Most of the

Natarajan Meghanathan et al. (Eds) : SAI, ICAITA, CSITA, ISPR, Signal - 2018
pp. 19–30, 2018. © CS & IT-CSCP 2018 DOI : 10.5121/csit.2018.81003

research done in spectrogram and other features based classification is usually done offline, whereas the proposed method offers a remote real-time control of the robot that can be used to survey terrains that are otherwise impervious for humans, or monitor activities inside huge structures like wind-mills, gas pipelines etc.

The primary contributions of this paper are listed below :

- In this study, we worked with 8 control commands. Histogram equalization was applied to the spectrograms before feeding them to the network in order to enhance features for the network to learn. Since only 8 words are taken into consideration, speech recognition problem turns into a simple 8-class classification problem.

- A novel deep net architecture with a very small memory footprint, which further gives decent classification accuracy on custom voice/speech dataset.

## 2. RELATED WORK

Beard *et al.* [1] have created several alternative UAV interfaces in which users operate physical controllers to generate the requisite numerical commands. These interfaces are built using PDAs, full-size computers, a voice-recognition system, a force-feedback attitude joystick, a force-sensing interface using an IBM TrackPointTM, and a novel physical icon interaction scheme. Real-world tests with this interface have demonstrated that ambient wind noise and conversation can wreak havoc on the reliability of the voice-recognition system. A method of muting the microphone input is required, but even with 2461 such a system in place, considerable difficulties arise in environments with strong winds or loud background noises. However, our experience bas shown the voice interface to be very valuable, especially under favorable weather conditions.

UAV control stations feature multiple menu pages with systems accessed by keyboard presses as presented by Draper *et al.* [2]. Use of speech-based input may enable operators to navigate through menus and select options more quickly. This experiment processed and presented the utility of conventional manual input against the speech input for actions performed by UAV operators on the control station at two different levels of mission difficulty. Pilots performed a continuous flight/navigation control job while keeping in mind to complete eight different data input/entry tasks types with each input modality. Results from the experiment have proven that speech input and speech recognition based control was significantly better than manual input or RC control in terms of task completion time, task accuracy, flight/navigation measures, and pilot ratings. Across all the given tasks, data entry time was drastically reduced by approximately 40% with speech input.

The AirSTAR testbed developed by Jordan *et. al* [3] has been developed to provide an in-flight capability to validate various flight critical technologies. The testbed is composed of three elements: a 5.5% dynamically scaled, turbine powered generic transport model (GTM), a Mobile Operations Station (MOS) and associated ground based facilities, and a test range. This research capability, along with wind tunnel testing, full scale flight testing, and flight simulation, provides the methods and tools to develop and test the technologies demanded by the AvSP. The expanded flight envelope of the GTM and the requirements to gather large amounts of data (at high rates) presented unique challenges to the development of the AirSTAR testbed. Because the GTM will be operating outside of the normal benign flight envelope of full scale transport aircraft and most

UAVs, additional measures had to be taken, both on the plane and in the control station, to mitigate the risks associated with this type of flight.

McLain *et. al* [4] UAV research interests have been revolving around cooperative and coordinated control of multiple vehicles and real-time trajectory generation and optimization. Their primary objectives for experimental testing of their research are to validate the feasibility of practical implementation of their methods and to foster innovation to overcome implementation challenges. For the control of UAVs, real-world issues such as sensor noise, communication dropout, communication delay, and computation latency can degrade performance and lead to catastrophic failures. Sensors that are inherently asynchronous with varied sample rates can pose challenges for estimation and coordination. Airframe payload capacity influences the choice of sensors and onboard computers and thus the inherent capabilities of the vehicle. Environmental factors, such as wind, weather, and lighting can adversely affect sensor and control system performance. Field tests often expose the unanticipated challenges that must be dealt with in a real-world scenario. Furthermore, these challenges often force significant innovations to occur to enable success.

Prodeus*et. al* [5] compared ix noise reduction algorithms with the use of a set of indicators. Among them are popular noise reduction algorithms such as spectral subtraction, Wiener filtering, MMSE and logMMSE, and two less well-known Wiener-TSNR and Wiener-HRNR algorithms. It has been proven that when the noise reduction system is used as preprocessing or of automatic/autonomous speech recognition (ASR) system, only a small amount of speech signal quality indicators is in significant consensus with the recognition accuracy or classification rate. In specific, these include Log-Likelihood Ratio (LLR) and Signal Composite Index (SCI) indicators. Furthermore, no single algorithm amongst al of the considered noise reduction algorithms, is the top-most in terms of maximum recognition rate for a very huge variety range of input signal-to-noise ratio all ranging from -10 dB to +30 dB.

They reviewed the theory of discrete Markov chains and showed how the concept of hidden states, can be effectively used. They illustrate the theory with two simple examples, namely coin-tossing, and the classic balls-in-urns system. They discuss the three fundamental problems of HMMs, and give several practical techniques for solving these problems. They also discussed the various types of HMMs that have been studied including ergodic as well as left-right models. They discussed state density function, onservation duration density, and optimization criterion for choosing optimal HMM parameter values. They also discuss the issues that arise in implementing HMMs including the topics of scaling, initial parameter estimates, model size, model form, missing data, and multiple observation sequences. They described an isolated word speech recognizer, that was implemented with HMM. They extend the ideas presented before to the problem of recognizing a string of spoken words based on concatenating individual HMMs of each word in the vocabulary. They briefly outlined how the large vocabulary speech recognizer use ideas of HMM.

A database as well as a recognition experiment was presented in this paper by Hirsch *et. al* [7] to obtain comparable recognition results for the speaker-independent recognition of connected words in the presence of additive background noise and for the combination of additive and convolutional distortion. The distortions are artificially added to the clean TIDigits database. The noisy database together with the definition of training and test sets can be taken to determine the performance of a complete recognition system. In combination with a predefined set-up of a

HTK(Hidden Markov Model Tool Kit) based recognizer it can be taken to evaluate the performance of a feature extraction scheme only.

Hinton *et al.* [8] reviewed exploratory experiments on TIMIT database and used them to demonstrate the power of a two-stage training procedure for acoustic modeling. The DNNs that worked well on TIMIT database were then applied to five different large-vocabulary continuous speech recognition tasks. Their DNNs worked well on all the tasks and on some the tasks it outperformed the state of the art.

According to Graves *et. al* [9], it is possible to train RNNs end-to-end for speech recognition. This approach exploits the larger state-space and richer dynamics of RNNs compared to HMMs, and avoids the problem of using potentially incorrect alignments as training targets. The question that inspired their paper was whether RNNs could benefit from depth in space.

In this paper by Itakura *et. al* [10], an approach to the problem was described from a statistical point of view, and it was shown that the log likelihood ratio, which is the best criterion to test the hypothesis, was reduced to the logarithm of the ratio of prediction residuals, and can be used as a powerful distance measure. This result of their research was applied to automatic recognition of isolated words, where the sequential likelihood ratio test was adopted to reduce the amount of computation.

## 3. METHODOLOGY

The system was trained on the features of the voice samples (MEL and MFCC) and corresponding spectrograms of 15 subjects from 19-22 years of age speaking 8 different words that were Takeoff, Land, Forward, Backward, Left, Right, Up & Down in 10 different pitches. Among these 15 subjects 9 were male and 6 were female.

An open-source code was used to collect the voice samples and at the same time to create spectrograms corresponding to each sample and then all samples were subjected to 9 pitch variation. The voice samples were recorded in random order, and there was a 5s hint before each sample was collected to tell the subject which word to say. Among the recorded samples, only the samples with noise below a particular level were used.

Then MEL and MFCC features were extracted from these voice samples and a batch generator was used to extract all 1200 samples at a time. These 1200 samples were split into training and test sets. The training and test sets consist of labels of voice samples, the spectrograms and the MEL/MFCC features corresponding to each voice sample.

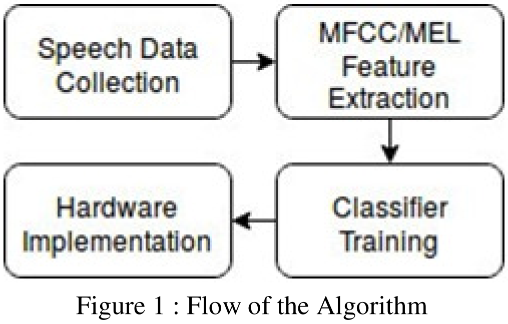


Figure 1 : Flow of the Algorithm

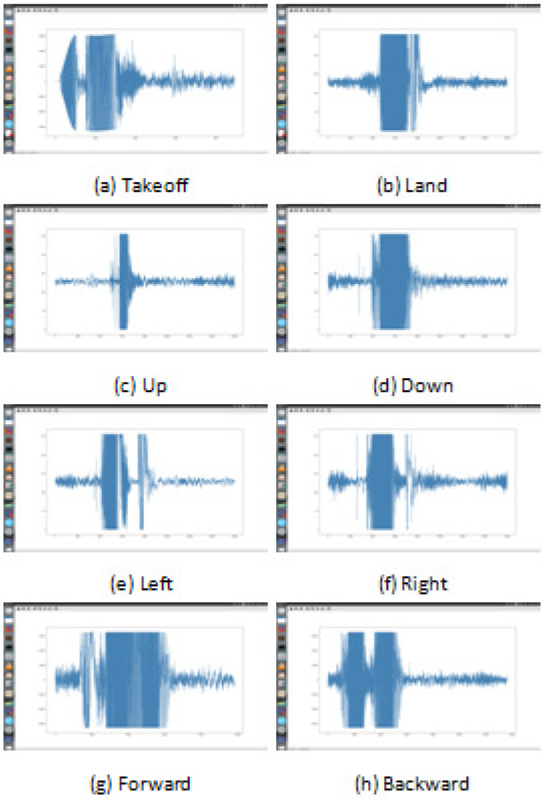
### 3.1. Dataset

The dataset (Fig. 2) consists of a total of 1200 recorded voice samples and 1200 spectrograms of the subjects from 19-22 years of age speaking 8 different words in 10 different pitches. The words were Takeoff, Land, Forward, Backward, Left, Right, Up & Down. The words were marked with numbers from ranging from 0-7 (0-Takeoff, 1-Land, 2-Forward, 3-Backward, 4-Up, 5-Down, 6-Left & 7-Right). The above mentioned words were chosen specifically for UAV control because UAVs or drones can execute these commands only, so we do not need an extensive speech recognition system for controlling a robot.

A total of almost 3000 voice samples were recorded among which 1200 were marked as correct (having noise below the particular level. All the voice samples recorded were in English and each of the recorded voice samples last for a period of 5 seconds. Sample Dataset is shown in the figure below.

### 3.2. Feature Extraction

For our research we observed that MFCC and MEL feautre sets to be appropriate for speech classification. Also, spectrograms have been made. The extraction methods are explained as follows.



(a) Takeoff

(b) Land

(c) Up

(d) Down

(e) Left

(f) Right

(g) Forward

(h) Backward

Figure 2.: Sample Data Audio Plots

### 3.2.1. The Spectrogram

If $x$ is signal of length $N$, and considering consecutive clips of $x$ of length $m$ where $m$ <<$n$ and let X ∈ $R^{m(Nm+1)}$ be the matrix with the consecutive segments as consecutive columns. In other words, $[x[0],x[1],...,x[m1]]^T$ is the first column, $[x[1],x[2],...,x[m]]^T$ is the second column, and so forth. Both the rows and columns of $X$ are indexed by time. We see that $X$ is a not a mathematically useful representation of $x$, whose columns are the Discrete Fourier Transforms of the columns

$$\hat{X} = \overline{F}X \tag{1}$$

$$X = \frac{1}{m}F\hat{X} \tag{2}$$

The spectrogram of $x$ with window size $m$ is the matrix $\hat{X}$ are indexed by frequency and the columns are indexed by time. Each location on $\hat{X}$ Note that the rows of $\hat{X}$ corresponds to a point in frequency and time. So $\hat{X}$ˆ is a mixed time-frequency representation of $x$. Because the conversion and transformation between X and $\hat{X}$ is also highly redundant.

The spectrogram is a matrix. To visualize it we can view the matrix as an image with the $i$, $j$–$th$ entry in the matrix corresponding to the intensity or color of the $i$, $j$–$th$ pixel in the image.

The spectrograms of various voice samples have been plotted and shown (Fig. 3) with post histogram equalization. Histogram equalization has been done to enhance the features (contrast basically) in the spectrograms.

### 3.2.2. MEL Frequency Cepstral Coefficients (MFCC)

The implementation of Mel Frequency Cepstral Coefficients is one of the standard benchmarked method for audio/speech-based feature extraction. There are about 20 coefficients in ASR, although speech encoding could be probably achieved with the help of only 12-13 coefficients. However, a disadvantage of using MFCC features is it's sensitivity to noise due to its' dependence on spectral form. It is therefore recommended to use techniques that extract information from the periodicity of speech data, which could be used to overcome the above mentioned problem, although human speech may also contain aperiodic content.

As an approximation to Mel-frequency scale, the frequency scale that is used here is approximately linear for frequencies below the range of 1 kHz and logarithmic for frequencies higher than 1 kHz. The motivation for this approximation comes from the fact that the human auditory sensory system is comparatively less frequency-selective as frequency increases beyond 1 kHz. The MFCC features correspond to the cepstrum of the log filterbank energies. To calculate them, the log energy is first computed from the filter bank outputs as

$$S_t[m] = \log\left(\sum_{n=0}^{N-1} |X_t[n]|^2 H_m[n]\right) \quad 0 \le m < M \tag{3}$$

where $Xt[n]$ is the DFT of the $t^{th}$ input speech frame, $H_m[n]$ is the frequency response of $m^{th}$ filter in the filterbank, N is the transformation window size and M is the total number of filters. Then, the discrete cosine transform (DCT) of the log energies is computed as follows :

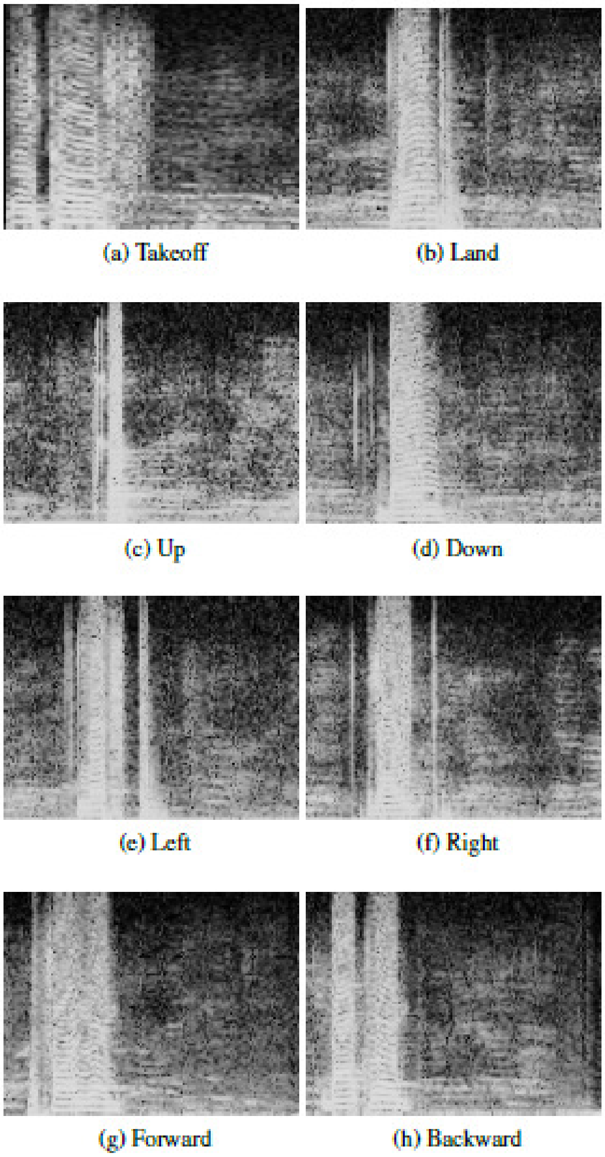

(a) Takeoff
(b) Land
(c) Up
(d) Down
(e) Left
(f) Right
(g) Forward
(h) Backward

Figure 3 : Spectrograms of the Voice Samples

$$c_t[m] = \sum_{n=0}^{N-1} S_t[n] \cos\left(\pi m \left(\frac{n-0.5}{M}\right)\right) \quad 0 \leq m < M \tag{4}$$

Since the human auditory system is dependent on time based evolution of the spectral content of the signal, attempts are often made to include the extraction of this data as part of MFCC feature analysis. In order to capture the changes in the coefficients over time, first and second difference coefficients are computed as respectively.

$$\Delta\vec{c_t} = \vec{c}_{t+2} - \vec{c}_{t-2} \tag{5}$$

$$\Delta\Delta\vec{c_t} = \Delta\vec{c}_{t+1} - \Delta\vec{c}_{t-1} \tag{6}$$

These dynamic coefficients are then concatenated with the static coefficients $\vec{c}_k$ according to making up the final output of feature analysis representing the $t^{th}$ speech frame.

$$\vec{x_t} = [\vec{c_t} \quad \Delta\vec{c}_t \quad \Delta\Delta\vec{c}_t] \tag{7}$$

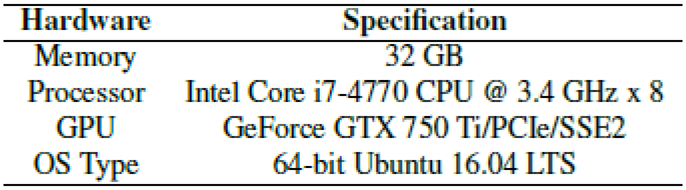
### 3.2.3. MEL Scale Cepstral Analysis (MEL)

Mel scale cepstral analysis is very similar to perceptual perceptual linear predictive coefficients (PLP), where the short term spectrum is modified based on psychophysically based spectral transformations. In this method, however, the spectrum is warped according to the MEL Scale, whereas in PLP the spectrum is warped according to the Bark Scale. The main difference between Mel scale cepstral analysis and perceptual linear prediction is related to the output cepstral coefficients. The PLP model uses an all-pole model to smooth the modified power spectrum. The output cepstral coefficients are then computed based on this model. In contrast Mel scale cepstral analysis uses cepstral smoothing to smooth the modified power spectrum. This is achieved by direct conversion of the log power spectrum to the cepstral domain using the standard algorithm of Inverse Discrete Fourier Transform (iDFT).

## 3.3. The VoiceNet Model

In this study, among the 1200 samples extracted using the batch generator, 1080 samples were used for training of the model, and 120 samples were used for testing of the model.We used a regression neural network that takes an input of size (20,170) consisting of 3 fully connected layers, 3 dropout layers and a softmax activation layer. The neural network uses adam as optimizer and categorical cross entropy as loss function. The network has been visualized in the the following graphic (Fig. 4). The training of the network was carried out on a system with specifications listed in Table I.

Table 1: System Specifications



| Hardware | Specification |
|---|---|
| Memory | 32 GB |
| Processor | Intel Core i7-4770 CPU @ 3.4 GHz x 8 |
| GPU | GeForce GTX 750 Ti/PCIe/SSE2 |
| OS Type | 64-bit Ubuntu 16.04 LTS |

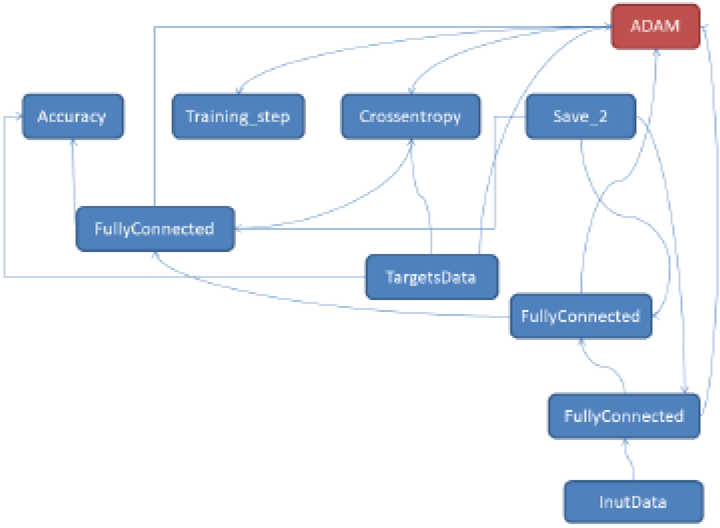


Figure 4.: Network Architecture

## 4. HARDWARE IMPLEMENTATION

Two quadrotors have been tested with this algorithm.

### 4.1. Bebop 2

An off-the-shelf quadrotor, Parrot Bebop 2 (Fig. 5), compatible with Python programming language, was used as hardware platform for algorithm development and testing. WiFi is used for communicating between the systems.

Figure 5: Parrot Bebop 2

### 4.2. Custom Drone

As a common understanding, there is a requirement for a custom-built quadcopter with onboard computational capabilities. The BumbleB (Figure 6), the drone we designed and fabricated, is equipped with a companion ODROID-XU4 single-board-computer (see Table II) which runs the VoiceNet algorithm. The specifications of BumbleB are tabulated in Table III.

Table 2 : ODROID-XU4 Specifications

| Hardware | Specification |
|---|---|
| Memory | 2 GB LPDDR3 PoP Stacked |
| Processors | Samsung Exynos5422 Cortex-A15 @ 2GHz with Cortex-A7 Octa core CPUs |
| GPU | Mali-T628 MP6 |
| Storage | eMMC5.0 HS400 Flash Storage |
| Kernel Type | 32-bit (ARMv7) Linux Kernel 4.9 LTS |
| OS Type | Lubuntu 14.04 LTS |

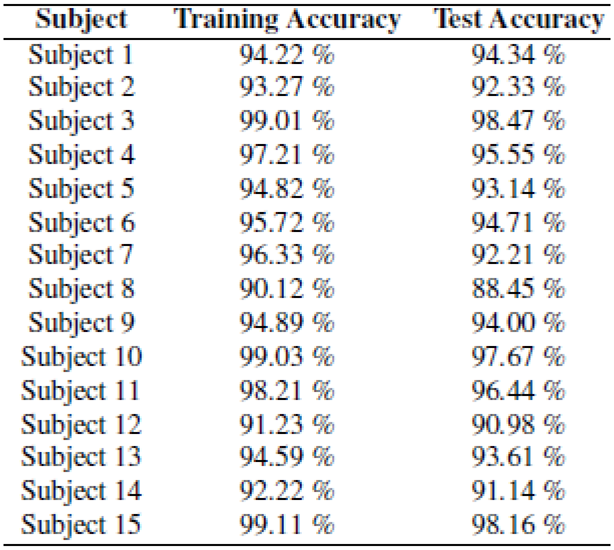
Table 3 : Drone Specifications

| Component | Specification |
|---|---|
| Frame Configuration | H-4 |
| Frame Size (L x B x H) | 0.276 m × 0.63 m × 0.12 m |
| Battery Rating | 5200 mAh, 3S,11.1V LiPo |
| ESC Rating | 30 A Brushless |
| Motor Ratings | 920 KV Brushless |
| Propeller Dimensions | 9 × 4.5 |
| Flight Controller | Pixhawk |
| Avg. Flight Time | 15-20 Minutes |
| On-Board Computer | ODROID-XU4 |
| On-Board Camera | Logitech C270 (30FPS) |

## 5. EVALUATION AND RESULTS

Since there are not many metrics available pertaining to our current problem statement, we report the classification accuracy of our VoiceNet on custom dataset. We also take into consideration the various pitches of the subjects who were involved in the study. The VoiceNet model takes approximately 1.33 seconds to process an audio sample and predict the word said. This is primarily because of the small neural network designed and various features fed into it. The break-up of the timing is 0.34 seconds for feature extraction and 0.99 seconds for prediction.

Table 4 : Individual Accuracies of Subjects



| Subject | Training Accuracy | Test Accuracy |
|---|---|---|
| Subject 1 | 94.22 % | 94.34 % |
| Subject 2 | 93.27 % | 92.33 % |
| Subject 3 | 99.01 % | 98.47 % |
| Subject 4 | 97.21 % | 95.55 % |
| Subject 5 | 94.82 % | 93.14 % |
| Subject 6 | 95.72 % | 94.71 % |
| Subject 7 | 96.33 % | 92.21 % |
| Subject 8 | 90.12 % | 88.45 % |
| Subject 9 | 94.89 % | 94.00 % |
| Subject 10 | 99.03 % | 97.67 % |
| Subject 11 | 98.21 % | 96.44 % |
| Subject 12 | 91.23 % | 90.98 % |
| Subject 13 | 94.59 % | 93.61 % |
| Subject 14 | 92.22 % | 91.14 % |
| Subject 15 | 99.11 % | 98.16 % |

## 6. DISCUSSION AND CONCLUSION

A novel solution to UAV control has been presented in this paper. The fact that a drone does not need an extensive speech recognition system to odentify only some keywords like take-off, forward etc. This calls for a smaller sized deep nets for speech recognition. Further aspects of this research include decreasing the time complexity even further and making the interface more robust so that it could be integrated with robots of different nature.

## REFERENCES

[1] Beard, RandalW., Derek Kingston, Morgan Quigley, Deryl Snyder, Reed Christiansen,Walt Johnson, Timothy McLain, and Michael Goodrich. ”Autonomous vehicle technologies for small fixedwing UAVs.” Journal of Aerospace Computing, Information, and Communication 2, no. 1 (2005): 92-108.

[2] Draper, Mark, Gloria Calhoun, Heath Ruff, David Williamson, and Timothy Barry. ”Manual versus speech input for unmanned aerial vehicle control station operations.” In Proceedings of the Human Factors and Ergonomics Society Annual Meeting, vol. 47, no. 1, pp. 109-113. Sage CA: Los Angeles, CA: SAGE Publications, 2003.

[3] Jordan, Thomas, John Foster, Roger Bailey, and Christine Belcastro. ”AirSTAR: A UAV platform for flight dynamics and control system testing.” In 25th AIAA Aerodynamic Measurement Technology and Ground Testing Conference, p. 3307. 2006.

[4] McLain, TimothyW., and RandalW. Beard. ”Unmanned air vehicle testbed for cooperative control experiments.” In American Control Conference, 2004. Proceedings of the 2004, vol. 6, pp. 5327-5331. IEEE, 2004.

[5] Prodeus, A. M. ”Performance measures of noise reduction algorithms in voice control channels of UAVs.” In Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD), 2015 IEEE International Conference, pp. 189-192. IEEE, 2015.

[6] Rabiner, Lawrence R. ”A tutorial on hidden Markov models and selected applications in speech recognition.” Proceedings of the IEEE 77, no. 2 (1989): 257-286.

[7] Hirsch, Hans-Gnter, and David Pearce. ”The Aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions.” In ASR2000-Automatic Speech 11 Recognition: Challenges for the new Millenium ISCA Tutorial and Research Workshop (ITRW). 2000.

[8] Hinton, Geoffrey, Li Deng, Dong Yu, George E. Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior et al. ”Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups.” IEEE Signal Processing Magazine 29, no. 6 (2012): 82-97.

[9] Graves, Alex, Abdel-rahman Mohamed, and Geoffrey Hinton. ”Speech recognition with deep recurrent neural networks.” In Acoustics, speech and signal processing (icassp), 2013 ieee international conference on, pp. 6645-6649. IEEE, 2013.

[10] Itakura, Fumitada. ”Minimum prediction residual principle applied to speech recognition.” IEEE Transactions on Acoustics, Speech, and Signal Processing 23, no. 1 (1975): 67-72.12