PEOPELS'S DEMOCRATIC REPUBLIC OF ALGERIA

MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

Blida 01 University

Institue of Aeronautics and Space Studies

Manuscript

In partial fulfilment of the requirements for the

Degree of Master in Aeronautics

Speciality: Avionics

Multilingual voice recognition using deep learning for human-drone interaction

Made by

- Mebarkia Nihad
- Kacel Yasmine

Advisor:

Mr. Choutri

Vice-Advisor:

Mr. Lagha

Blida; 2020-2021

Acknowledgments

First and foremost, praise and thanks to God, the Almighty, for his showers of blessings throughout my research work to complete the research successfully.

I would like to express my special thanks and gratitude to my supervisor, Dr.Choutri Kheireddine, who gave me the golden opportunity to do this wonderful project, and who also helped me to do a lot of research. And especially to Professor Lagha Mohand for his guidance during this project, it truly has been a very good time in the laboratory.

I am grateful to my parents, Mebarkia Youcef and Belmokhtar Rafika, whose constant love and support keep me motivated and confident. My accomplishments and success are because they believed in me. Thank you to my sister, Yasmine, for always being there for me.

A warm word for my binome who is also my best friend and a wonderful person. I feel so blessed that I got the chance to work with her during the last five years.

I also would like to thank all my respected teachers in the Institute of Aeronautics and Space Studies and all the other members of the department. Last but not least, I would also like to thank my friends: Zahira, Fairouz, Nouha, Akila, Lotfi, Merouane, Rayan, Nadim, and other friends who directly and indirectly provided me with inspiration and valuable suggestions during the course of this study.

. . . Nihad

Acknowledgments

At the end of my work, I concede my thankfulness to Allah all mighty for always responding to my prayers by providing me with an environment to learn patience, discipline, perseverance, and consistency. Furthermore being supervised by both Mr.Choutri Kheireddine and Mr. Lagha Mohand, I'm so thankful for their patience with us and for guiding us throughout all our work without mentioning supplying our needs.

I could not be more thankful to my beloved friend and colleague Nihad Mebarkia for making this experience such an enjoyable and overwhelming adventure alongside our teammates Akila Keddous, Fairouz Khettal, and Doumi Nouha Wissem. Not forgetting my friends, who first of all, believe in me; cheer me up, motivate me with their countless services. Thank you.

Last but not least, I grant my gratitude to my sweet family for their unfailing support and grave trust in me, especially my mother. Without her kindest words and her limitless love, I wouldn't be who or where I am now.

Thank you infinitely.

. . . Yasmine

Contents

ملخص

على مدى السنوات العديدة الماضية ، نال التفاعل بين الإنسان و الطائرة بدون طيار على جزء هام من البحوث العلمية. عند التفاعل مع طائرة بدون طيار ، يتحمل البشر العديد من المسؤوليات. يتم تحديد دورهم من خلال تطبيق الطائرة بدون طيار ومقدار الاستقلالية. الغرض من هذا العمل هو التحكم في حركة المركبة الجوية بدون طيار باستخدام نظام التعرف على الكلام متعدد اللغات. لذلك ، يتم تدريب الشبكة العصبية العميقة على التعرف على كلام المستخدم من العديد من اللغات ومن ثم إنشاء أمر التحكم المطلوب. أجرينا تجارب شملت مشاركين أعطوا أوامر صوتية من أجل مقارنة فعالية كل قاعدة بيانات.يبرهن تنفيذ النظام المصمم باستخدام الأجهزة على درجة عالية من الدقة في التعرف عليه و بساطة الرقابة عليه.

الكلمات الأساسية: التعرف على الكلام ، شبكة عصبية عميقة، المركبة الجوية بدون طيار، التحكم ، تفاعل الإنسان مع الطائرة بدون طيار.

Abstract

Over the past several years, human-drone interaction have got an important part from the scientific researchs. When interacting with a drone, humans take on many responsibilities. Their role is determined by the drone's application and the amount of autonomy. The purpose of this work is to control the movement of the unmanned aerial vehicle (UAV) using a multilingual speech recognition system. For that, a deep neural network (DNN) is trained to recognize the user's speech from many languages and then generate the desired control command. We conducted experiments involving participants giving voice commands in order to compare the effectivencess of each database. Hardware implementation of the designed system proof its high accuracy recognition and control simplicity.

Key words: Speech recognition, DNN, UAV, Control, Human-Drone Interaction.

Résumé

Au cours des dernières années, l'interaction homme-drone a pris une part importante dans les recherches scientifiques. Lorsqu'ils interagissent avec un drone, les humains assument de nombreuses responsabilités. Leur rôle est déterminé par l'application du drone et le degré d'autonomie. Le but de ce travail est de contrôler le mouvement du véhicule aérien sans pilote àl'aide d'un système de reconnaissance vocale multilingue. Pour cela, un réseau de neurones profonds est entraîné à reconnaître la parole de l'utilisateur dans de nombreuses langues, puis à générer la commande de contrôle souhaitée. Nous avons mené des expériences impliquant des participants donnant des commandes vocales afin de comparer l'efficacité de chaque base de données. La mise en œuvre matérielle du système conçu prouve sa reconnaissance de haute précision et sa simplicité de contrôle.

Mots clés : Reconnaissance vocale, réseau de neurones profonds, véhicule aérien sans pilote, Contrôle, Interaction Homme-Drone.

List of figures

Chapter One

Chapter Two

Chapter Three

Chapter Four

List of tables

Chapter Three

Chapter Four

Abbreviations or (Acronyms)

AM: Acoustic Model

ASR: Automatic Speech Recognition

BCI: Brain Computer Interaction

BLE: Bleutooth Low Energy

CNN: Convolutional neural network

CPU: Central Processins Unit

DNN: Deep Neural Networks

EEG: Electoencephalography.

ESC: Electronic Speed Controller

FAA: Federal Aviation Administration

FFT: Fast Fourier transform

FS: Sampling frequency

GHZ: Gigahertz

GUI: Graphical User Interface

HDI: Human Drone interaction

HRI: Human Robot Interaction

ISM: Industrial, Scientific and Medical.

KBPS: KiloBytes per second

LM: Language Model

MBPS: Megabits per second

MFCC: Mel-frequency Cepestral Coefficient

NUI: Natural User Interface

MQTT: MQ Telemetry Transport

ReLU: Rectified Linear Unit

RF: Radio frequency

RX: Receiver

SPI: Serial Peripheral Interface

SR: Speech Recognition

STFT: Short time Fourier transform

TX: Transmitter

UART: Universal Asynchronous Receiver Transmitter

UAV: Unnamed aerial vehicle

UI: User Interface

ULP: Ultra low power

USART: Universal Synchrone and Asynchronous Receiver Transmitter

USB: Universal Serial Bus

VR: Virtual reality

WI-FI: Wireless Fidelity

GENERAL INTRODUCTION

General Introduction

UAVs, commonly known as drones, are aircraft that can perform flight missions without a human pilot on board. With recent technical advancements, drones have emerged as a new form of robot.Personal drones are becoming popular.It's challenging to design how to interact with these flying robots.

Thus, in order to comprehend this new developing field, we aimed to discuss not only what social drone companions can accomplish, but also their design and prespective domain areas in which they may be employed. [1]

Drones are changing our daily lives in extremely beneficial ways. They make our lives easier in different sectors.Drones are an incredibly important innovation, and their prospective applications imply that they will alter our lives by introducing a completely new field of technology known as human drone interaction. [2]

Nowadays, it is common to see more people with no previous knowledge of the subject owning a drone, either to accomplish a specific purpose or to entertain themselves. According to the FAA, the number of registered drones in its database might reach 3.8 million by 2022.[3]. This may be improved further with the introduction of natural user interfaces (NUIs) such as body gestures and voice commands [4] which have been tested in the state of the art in [3]. Most results appear to imply that the implementation of a NUI facilitates human drone interaction.

The goal of this work is to respond to the following query: How can people operate and control a UAV with their voice instead of joysticks? Or can the drone understand Arabic and Amazigh commands?

For this, we were particularly interested in the design of a voice-controlled quadcopter.Our work is divided into four chapters:

We introduce the topic of human drone interaction in the first chapter in order to understand the value of drones in our lives and how to engage with them.

The second chapter covers some basic concepts in voice recognition, as well as the deep neural network, which has shown significant improvement in speech feature extraction and recognition.

The third chapter discusses multilingual speech recognition and the steps to follow to create the Arabic and Amazigh multilingual database. We test it on a variety of people in order to see the performance of this dataset.

In the fourth chapter, we look at the implementation side, or how we applied the simulation from chapter 3 to quadrotors while displaying the feedback in a graphical interface.

We conclude our work with a conclusion that summarizes the findings and suggests suggestions for future works in order to maintain the suggested subject's continuity and performance.

CHAPTER ONE

Chapter I Human drone interaction

I.1. Introduction

The area of human-robot interaction (HRI) has emerged as a result of robot research focusing on understanding and creating interactions with human users during the last decade.

With recent technological advances, UAVs have appeared as a new type of robot that has captivated the interest of HRI research, leading to a whole new field of human-drone interaction (HDI) research. [1]

The term "drone" simply refers to an unmanned aerial vehicle. Drone technology emerged following World War I. Despite the fact that they were designed for military uses. Drones are currently being employed in a wide range of applications, shifting from the military to the civilian sphere. [2]

Drone use has risen at an exponential rate in recent years. This rapid growth is both exciting and frightening.

On the one hand, drones open up new opportunities, with applications ranging from entertainment to delivery, assistance to people with special needs, sports, agriculture, and even rescue.

On the other hand, there are several risks to the use of drones in our environment. [3]

Therefore, it is important to study the field of HDI (Human-Drone-Interaction) to understand how the interaction between humans and drones can be extended to more areas of use.

I.2. Definition of HDI

Human-drone interaction is a diverse field of research. It can be defined as a field of study focused on the understanding and evaluation of the interaction distance and the development of new use cases. The four main areas of HDI are illustrated in figure (I-1).

.

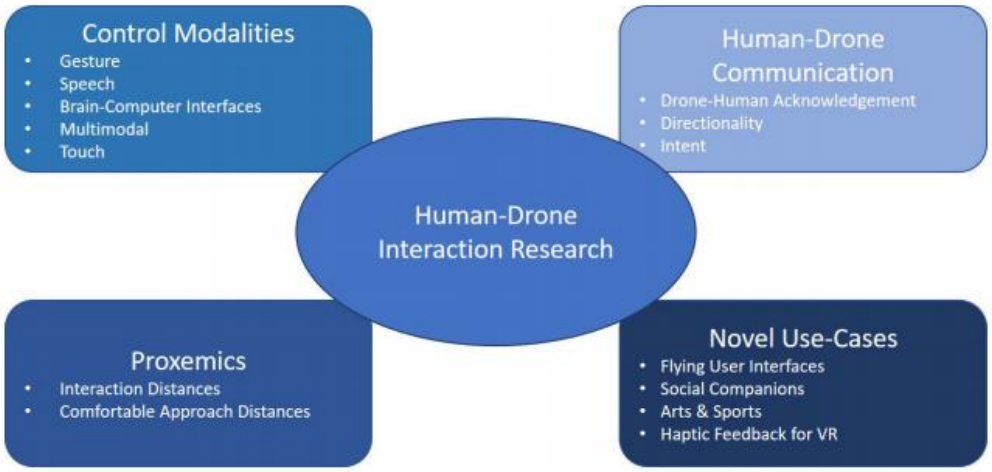


Figure (I-1): *The four major fields of Human drone interaction research [3]*

I.3. Research on Human-Drone Interactions

I.3.1. Role of humans in HDI

When interacting with a drone, humans take on several responsibilities. Their role is determined by the drone's application and the amount of autonomy.

A human can act as an **'active controller'**, controlling the drone directly with a control interface to complete a task. Take a picture of a landscape, for example.

Figure (I-2): The UAV took a picture of a landscape.

Another role is that of a **'recipient'**, in which the user does not operate the drone but benefits from its interaction. Consider the case of a user who receives an item delivered by a drone.

Figure (I-3): *Delivery via drone.*

Humans can also engage with drones as **'social companions'**, in which case the user interacts with the drone. The Joggobot is an example of such an engagement, in which the drone flies alongside individuals who are jogging.

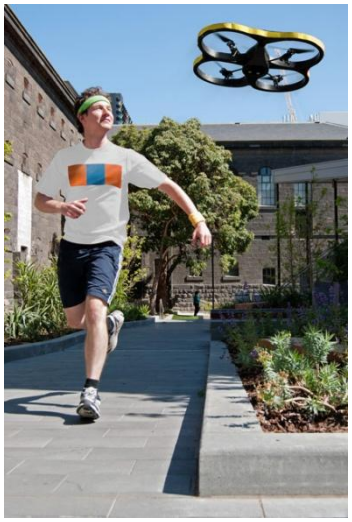


Figure (I-4): *Joggobot..*

Finally, in the case of autonomous drones, there is the task of **'supervisor'**. Despite the fact that most drones may fly autonomously, a person is still necessary to either pre-program the drone's behaviour (for example, planning the flight) or to supervise the flight itself (for example, monitoring a flight for autonomous real-time inspection).

Figure (I-5):*Drone as a supervisor*

I.3.2. HDI over time

HDI is a new field. Figure (I-6) demonstrates the number of publications per year on Google Scholar using human drone interaction in search to demonstrate how this field has developed over time.

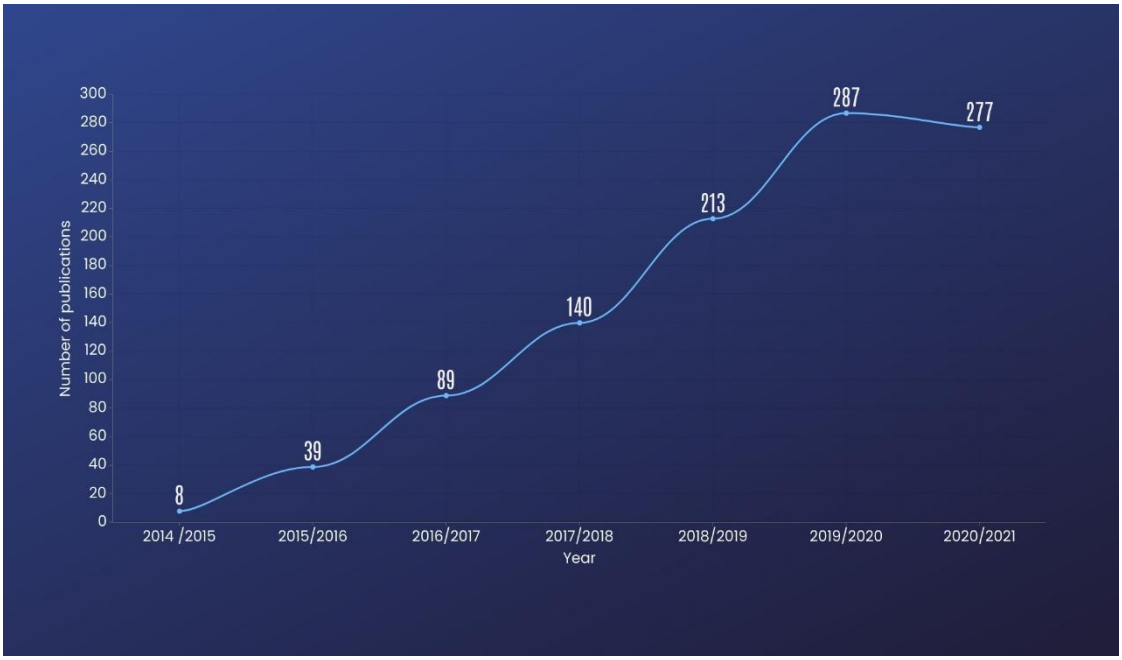


Figure (I-6): *HDI over time.[3]*

The degree of research in this field has surpassed all expectations. The University of South Australia is collaborating with Dragonfly, a Canadian company, to create a drone that can detect a group's temperature, heart rate, respiration rate, as well as coughs and sneezes. Its creators want to employ this technology to fight against Covid-19. [7]

I.3.3. User interfaces

The user interface (UI) is the point at which a computer, website, or application interacts with humans. The purpose of a good UI is to make the user's experience simple and straightforward, requiring the least amount of work from the user to get the maximum desired results. The most important formats are GUI and NUI.

I.3.3.1. Graphical User interfaces (GUI)

The graphical interface, which is still in use today, generates a predictable manner of interaction using WIMP (Windows, Icons, Menu, and Pointer).

The graphical interface provides interaction with the vehicle, as well as observation of states and dynamics, and also the presentation of graphical views and pictures to assist the user in understanding the vehicle's internal behaviour.

The following tasks are often performed by the operator using a graphical interface:

- Pre-define the drone's behavior (vehicle configuration).
- Keep an eye on drone behavior throughout a mission.
- Operate manually with basic motions.
- Gather information to be used later. [4]

I.3.3.2. Natural User interfaces (NUI)

Natural User Interfaces (NUIs) are the next level in user interface evolution, compared to graphical user interfaces (GUIs). [5]

These novel approaches allow users to interact with drones via gesture, speech, touch, and even brain-computer interfaces (BCIs) like electroencephalography (EEG). [3]

Many applications may now be found in our daily lives. Voice assistants, Alexa, and Siri, for example, respond to a voice-activated NUI. [5]

Natural user interfaces' major goal is to provide an easy method of control, which is described as an interface that operates as expected by the user. Non-expert users can interact with natural user interfaces. [3]

I.3.3.2.1. Gesture :

When users are asked to interact with a drone without any instructions, studies demonstrate that gesture interaction takes priority. According to previous research, there are four design criteria for gestures:

1. Gestures must be natural and simple to execute.
2. The motions must be connected to the information in the recorded photos.
3. There must be a clear separation between the background and the moving body.
4. Data processing must be completed as quickly as possible.

Cameras may also be used to recognize gestures. Many drones already have an onboard camera that can be used for gesture recognition without the need for extra payload-heavy sensors.

This study on gesture control shows that it is a natural control method, with the benefits of ease of use and reduced training periods. It also has the advantage of not requiring the user to hold any external devices, such as a joystick. However, for applications that demand delicate and exact control, this method may not be the ideal option, as it has a larger latency and poorer accuracy than other approaches.

Figure (I-7): *Controlling the UAV using gestures.*

I.3.3.2.2. Speech:

Speech is used as a method of interaction by 38% of American users and 58% of Chinese users, according to studies on natural user interfaces. Speech control is considered to be simpler than other approaches, resulting in shorter training times. However, voice recognition, like gesture control, can add delays to the system, limiting its uses.

Furthermore, operating drones with speech has some unique problems, since the propellers produce a loud noise that can decrease voice recognition accuracy.

Another problem is that if voice recognition is conducted on board, users are confined to collocated interaction because the drone must be close to the user to receive spoken instructions. This issue does not affect systems where a ground-control station is used to decode voice commands and control the drone.

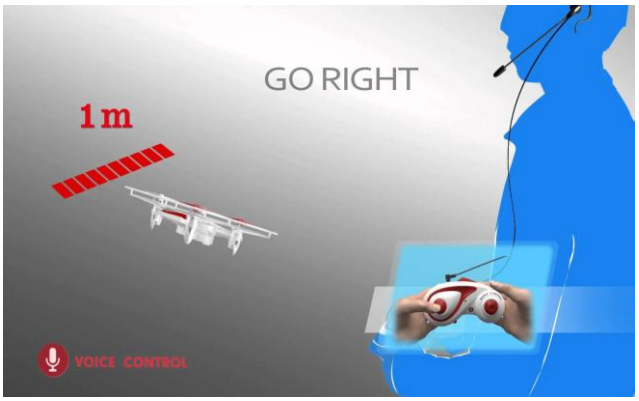


Figure (I-8): *Controlling drones via voice.*

I.3.3.2.3. Brain-Computer Interaction (BCI) :

Brain-computer interface devices have broad potential as assistive technologies and novelty control methods.

Researchers have been investigating the use of brain-computer interfaces (BCIs) to control unmanned aircraft (fixed wing) since 2010, and the first brain-controlled multi-rotor project was reported in 2013. In 2016, the University of Florida hosted the first brain-controlled drone race, which was followed by a race at the University of Alabama.

The pilot needs to use a BCI headset. The most popular of these are Electroencephalography (EEG) headsets, which operate drones via brain signals.

These devices use machine learning algorithms to analyse the brain's electrical activity on the human scalp, which is then used to operate physical systems using brain-waves.

As a result, interactions with BCIs are currently restricted compared to conventional control interfaces, and additional research is needed to improve the fidelity and reliability of these systems before they are deployed in users' homes.

Hands-free interaction and accessibility for disabled people will be possible if BCI reliability and accuracy reach levels similar to traditional control modalities.

Figure (I-9): *Flying drones with brains.*

I.3.3.2.4. Multimodal :

Combining the benefits of several interaction modalities is possible. Furthermore, earlier research showed that 45% of Chinese users and 26% of American users automatically interact with drones in multimodal ways. To develop direct contact with drones, a multimodal method may be employed, such as taking off and landing using voice and controlling movement with gestures.

I.3.3.3 Human-Drone Communication

I.3.3.3.1. Interaction distance

For a good social connection, the interaction distance between the drone and the human must be considered. [8] In the previous experience, 37% of US users stayed in the intimate space of the drone (45cm), 47% stayed in the personal space (1.2m) and the remaining 16% interacted in the social space (3.7 m).

However, the Chinese participants showed more comfortable and closer interaction: 50% in the intimate space, 38% in the personal space and 6% in the social space. Another study of users where drones approached users at different heights (1.80m and 2.13m) concluded that height did not have a significant impact on comfortable approach distance. [9]

I.3.3.3.2. Drone feedback

Studies have previously explored methods of recognizing the mutual attention between a drone and its users, and effective communication to avoid misinterpretation of the system that can even lead to accidents.

In [10], the subject was a comparison of four different drone recognition gestures. The results show that users prefer rotation in the yaw axis to indicate recognition.

The ability of a drone to express its intention to users was investigated in [11], where the expression concerned the manipulation of primitive movements using arc trajectories and the input and output of speed profiles.

The prototyped drones built with these manipulations test the following tasks with the participants: getting close to a person (easy input and output), avoiding a person (arc + easy entry and exit), approaching an object (anticipation) and moving away from an object (arc + easy input and output).

The results revealed that users prefer to work with a drone using manipulated flight paths rather than basic paths, a matter of safety, natural and intuitive interaction.

I.3.3.4. Novel Use Cases:

Drones are currently used for a wide range of applications, but researchers continue to explore new ways in which these systems can be of use.

I.3.3.4.1. Flying user interfaces:

This subsection presents prototypes of drones designed to improve and add mobility to user interfaces. These drones can be used to control crowds in emergency situations, provide information and advice to athletes during sporting activities, or even serve as a tour guide for outdoor activities [14],[15].

Previous work has explored the use of two drones as flying screens, one carrying a projector and the other a projection screen [13].This approach can be used as a new model of public display in urban environments, as it allows the display to gain attention by approaching the user, interacting and leaving. The relationship between the drones was based on a master-slave relationship, with the projector drone following the path of the screen drone using visual markers and computer vision to position itself in order to display the image correctly.

An octocopter (a helicopter with eight rotors) equipped with a smartphone and video projector has been used successfully to display images and text messages on arbitrary surfaces [16]. For evaluation purposes, a flight experiment was performed outdoors, displaying the received messages on the wall of a building. The flight lasted 7 minutes and about 40 people were 15 meters away. During the experiment, 23 messages in total were

displayed. Users found that the system was a fun experience capable of attracting attention and considered cases of use as interactive storytelling.

Figure (I-10): *Life-saving drones.*

Figure* (I-11*): *A drone against forest fires.*

Figure (I-12): *Medical drone.*

I.3.3.4.2. Social companions:

This subsection deals with prototype drones that explore social interaction with users.

They can be used as companions for visually impaired people to provide them with a navigation aid. [17] This study envisions a drone system that stands on hold on a wearable wristband until its help is needed. A blind user would command the drone, and the drone would guide them until the target is reached. The user can follow the drone thanks to the auditory feedback provided by the sound of the rotating propellers. Once a command is received, the drone calculates the distance to the target location and guides the user by flying a set distance in front of them, avoiding obstacles. A Bluetooth connection with the bracelet would allow the drone to adapt its distance and speed to the user.

Researchers have also considered the use of drones as a support agent in a clean environment [18].In this application, the drone finds trash on the ground, persuades users to pick it up, and guides it to the nearest trash can. The drone had different persuasion techniques: visual, audio, and a combination of the two.

Although the analysis of the results did not reveal any effect of the modality of interaction (visual, audio, the combination of the two) on user compliance, other factors were observed, such as the culture of the country and the user's gender.

Figure (I-13): *Accessory for drone to help blind people*

Figure (I-14): *Drones for the blind.*

I.3.3.4.3. Arts and sports:

This subsection shows how drones can be used to create works of art and new sports.

Drawing landscapes has always sparked people's interest, but creating such amazing works of art takes a lot of planning and time, but the use of drones makes it an easy and quick task, as described in [19]. A user can draw a sketch on the screen of a mobile phone which displays the video feed of a drone camera. The user then flies the drone over the area where the artwork will be created. As the drone hovers over the area, the user follows the drawing on the screen by walking and placing markers on the ground. At this point, the user can land the drone and mow the grass following the previous marks, creating the

landscape artwork. In just 30 minutes, a smiley face is implanted on the grass field with two trees incorporated into the design, defining the eyes of the smiley face.

A project known as HoverBall envisions increasing sports by using a drone like a ball capable of changing its physical dynamics [20]. The drone is enclosed in a circular cage and can alter physical dynamics such as gravity, speed, and the trajectory of the ball. For example, HoverBall could decrease ball speed or the effect of gravity during a volleyball game to allow unskilled players, or even children, to play.

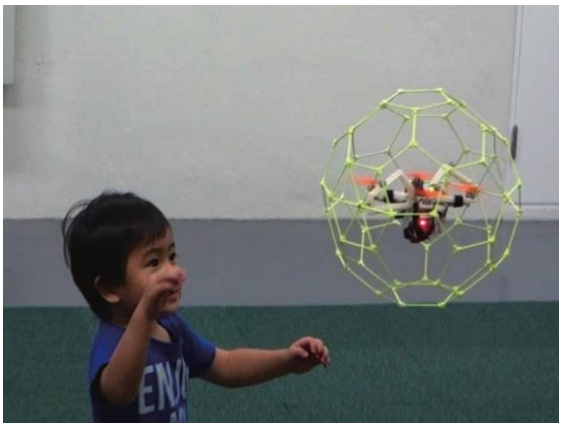


Figure (I-15): *HoverBall.*

I.3.3.4.4. Haptic feedback for virtual reality:

The term “haptic” concerns the use of tactile sensations in interfaces. The haptic feedback is the science and technology of transmission and understanding of information by the sense of touch.

Current VR systems can provide immersive visual and sound experiences, but they lack the ability to provide tactile feedback. As drones can fly in 3D space, they can be used to provide tactile feedback by touching the user at any location and at any speed to provide an adequate experience.

Small quad rotors have been used to provide haptic feedback in virtual reality games [21]. In this project, drones are used to fly towards users at varying speeds while they are immersed in a virtual environment system. Different tips can be attached to the drone, depending on the virtual environment, in order to provide adequate feedback. The prototypical game consists of a Mayan city in the jungle. The drones provide feedback in three scenarios: they play the roles of drones that attack the user, arrows shot by creatures at the player, and bricks and wood that fall on the user when the ruins collapse.

Figure (I-16): *Virtual reality haptic drones.*

I.4. Conclusion:

With HDI research leading to new uses, it's also worth mentioning that advancements in hardware and software technology will allow drones to be used in applications not yet envisioned.

Drones can also be used as companions for people with disabilities to provide navigation assistance.

The two envisaged use cases are: guiding users to a specific location or to helping them find placed objects using computer vision algorithms. Even though the system is not yet operational, a preliminary study was carried out with a blind user. The participants were able to successfully follow the drone as planned and provided positive feedback on the idea of the project.

The researchers also envisioned the use of drones as agents to support a clean environment. In this application, the drone would find trash on the ground, persuade users to pick it up and guide it to the nearest trash can.

In the near future, drones will be widely used in the fields of public advertising, deliveries, sports, emergency response, and to increase human capacity.

In addition, drone popularity will increase once we understand better how society accepts these systems, so future researchers could contribute by studying how society and cultures see drones.

Chapter I Human drone interaction

CHAPTER TWO

Deep learning for speech recognition

II.1. Introduction

Speech recognition is a method that analyses sounds captured by a microphone and transcribes them into a series of words that machines can understand.

Automatic speech recognition has progressed rapidly since its start in the 1950s, especially with the help of phoneticians, linguists, mathematicians, and engineers, who have defined the acoustic and linguistic knowledge necessary to fully understand a human's speech.

However, the result isn't great and is dependent on a number of factors. Favourable conditions for speech recognition involve native speech, belonging to a single speaker with proper diction (not presenting a voice pathology), recorded in a quiet and noiseless environment, and based on a common vocabulary (known words by the system).

When faced with non-native accents, various dialects, speakers with voice pathology, words unknown by the system (usually proper names), and noisy audio signals (low signal/noise ratio), the system's performance suffers. [22]

Voice communication has become an increasingly essential element of our smart gadgets in recent years. We can now activate our smartphone with a simple voice command, tell our automobile where we want to go, and even ask a voice assistant to place an order for us. And this is only the tip of the iceberg because there are so many applications.

II.2. Deep Neural Networks (DNN)

II.2.1. Definition

A Deep neural network (DNN) is a multilayer perceptron (MLP) with many hidden layers (typically more than two). A DNN architecture with an input layer, three hidden layers, and an output layer is shown in Figure (II-1). This allows it to process data in a complex way, using advanced mathematical models.

- The input layer: takes data from the user and sends it on to the first hidden layer.
- The hidden layers : use our inputs to execute mathematical calculations.
- The output layer: is responsible for returning the output data.

Each layer is recognized for extracting data in a unique way. In image recognition, for example, the first layer will look for edges, lines, and other features. Second layer: eyes, ears, nose, and so forth.

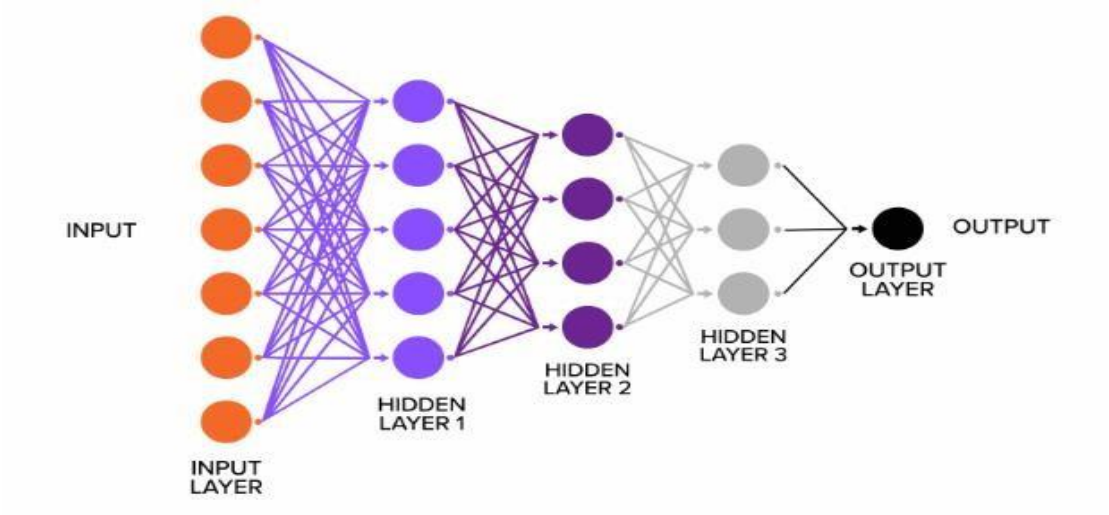


Figure (II-1): *Deep neural network[23]*

Deep networks, in general, are still neural networks (trained by back propagation, learning hierarchical abstractions of the input, and optimizied using gradient-based learning), but with additional layers We are particularly interested in activation functions, error functions, optimization methods, and regularization approaches. [24]

II.2.2. Deep learning architectures:

II.2.2.1. Convolutional Neuron Network:

For automatic speech recognition, there are a variety of deep learning architectures. CNNs are a subtype of the discriminative deep architecture and have shown satisfactory performance in processing two-dimensional data. [25]

CNN, or convolutional neural network, is a type of deep neural network architecture designed for specific tasks like image classification. It has certain unique properties that make it helpful for processing certain type of data, like images, audio and video.

CNNs get their name from the type of hidden layers they consist of. The hidden layers of a CNN generally consist of:

- Convolutional layers: is the most significant, which works by applying a filter to an array of picture pixels.

- Pooling Layers: Reduces the sample size of a certain feature map, which also speeds up processing by reducing the number of parameters the network must process.
- Fully-Connected Layers: help us to classify our data.
- The ReLu layer: serves as an activation function, ensuring non-linearity as data passes through the network's layers.

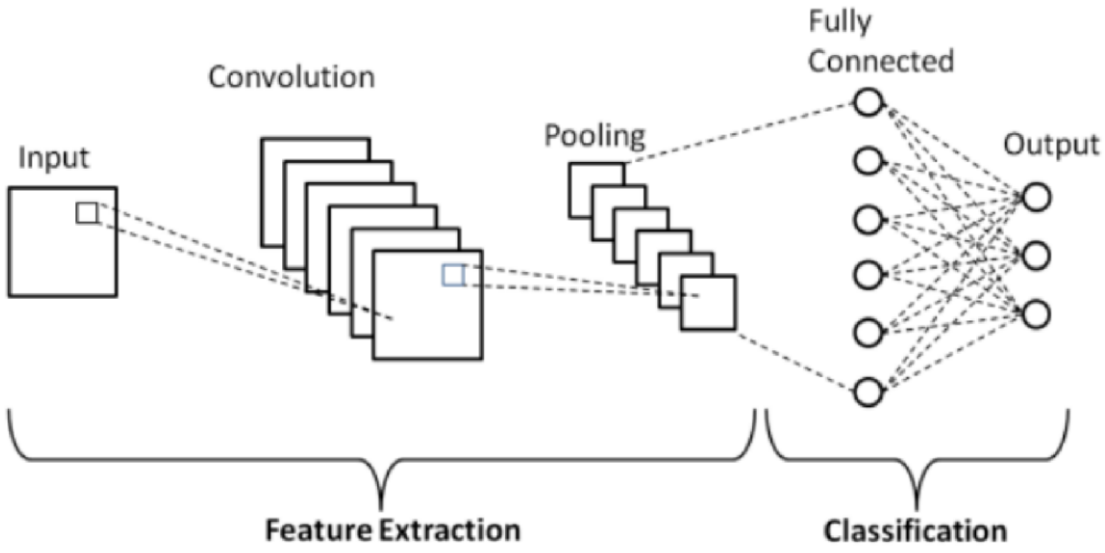


Figure (II-2): *The basic architecture of CNN [26]*

II.3.1.3. Deep learning characteristics:

Most of the time, it's not practical to feed all of the training data into an algorithm in one pass. Some terminology is therefore necessary to improve the understanding of how smaller pieces of data are used. As seen in figure (II-3).

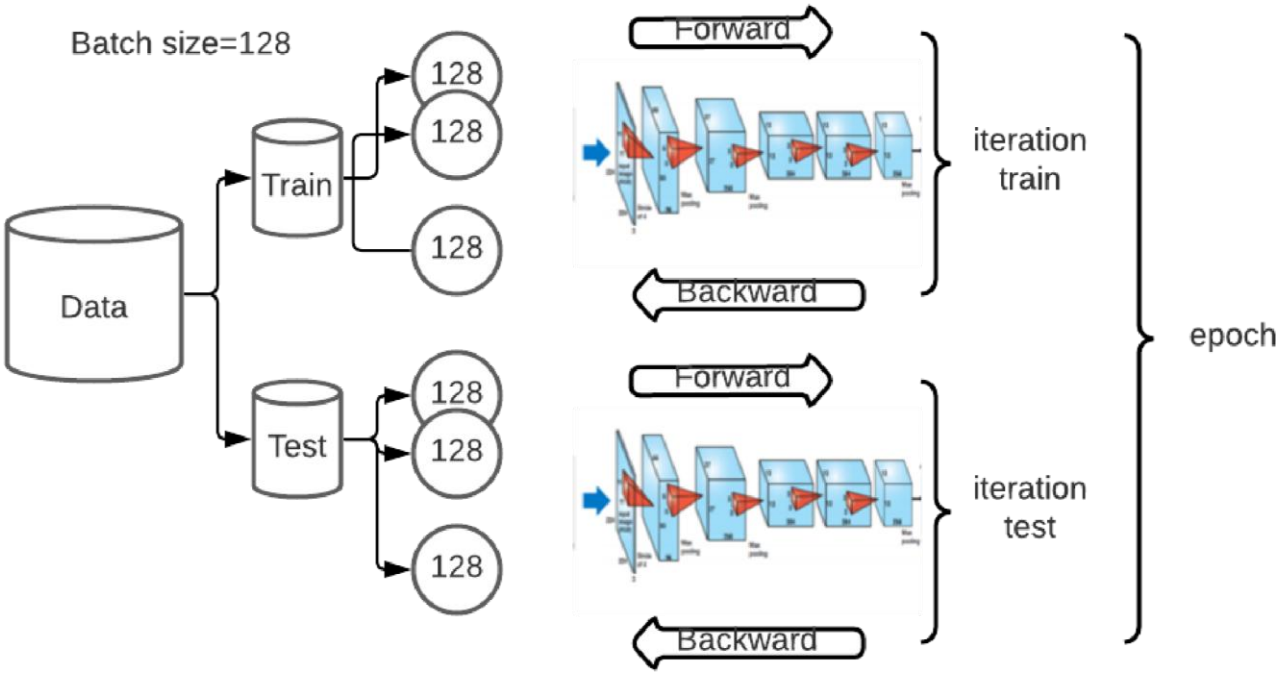


Figure (II-3): *Epoch, batch size and iteration.*

- **Epoch:** The time it takes for an entire dataset to be passed forward and backward through the neural network exactly once. If the entire dataset can not be passed into the algorithm at once, It must be divided into mini batches.

  Epoch = iteration train + iteration test

- **Batch size:** is the total number of training samples present in a single mini-batch
- **Iteration:** is updated throughout training with a single gradient. The iteration number is equal to the number of lots required for one epoch.[27]

  Iteration train= data train / batch size

  Iteration test = data test / batch size

- **Gradient descent:** an optimizing iterative process used to decrease the loss function in Deep Learning.
- **Loss function:** indicates how well in the present set of parameters the model will perform (weights and biases).

- **Learning rate**: has a significant impact on the gradient descent algorithm's efficacy. A very high learning rate can cause the loss function to start to increase after a few iterations, while a moderately high rate causes the loss to plateau at a high value after an initial rapid decrease. A very low learning rate, on the other hand, can be identified by a slow decrease in the loss function over training epochs. A good learning rate, on the other hand, combines a quick decrease during the initial epochs with a lower steady value. [28]

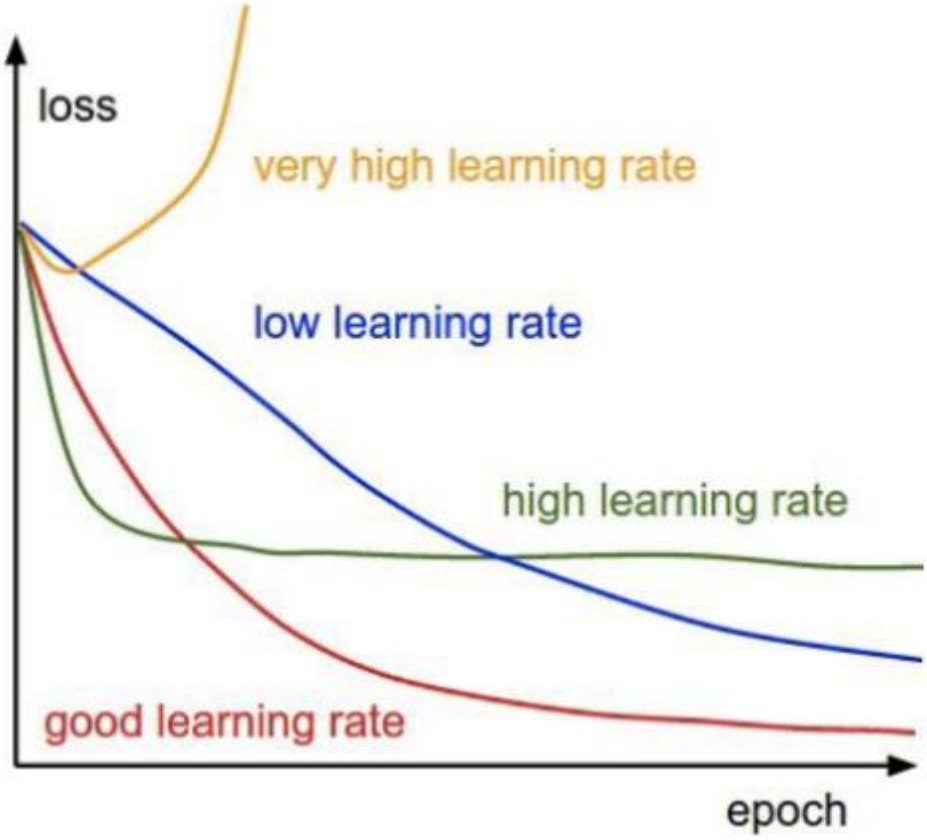


Figure (II-4): *Effect of learning rate on the loss function.*

II.2.3.1. Activation functions:

II.2.3.1.1. Sigmoid:

For a variety of reasons, the sigmoid function is a useful activation. As ssen by the graph in figure this functions as a continuous squashing function, limiting its output to the range (0,1). It is comparanle to the step function, but it has a smooth, continuous derivative that makes it excellent for gradient descent methods. It is similary centered on zero.

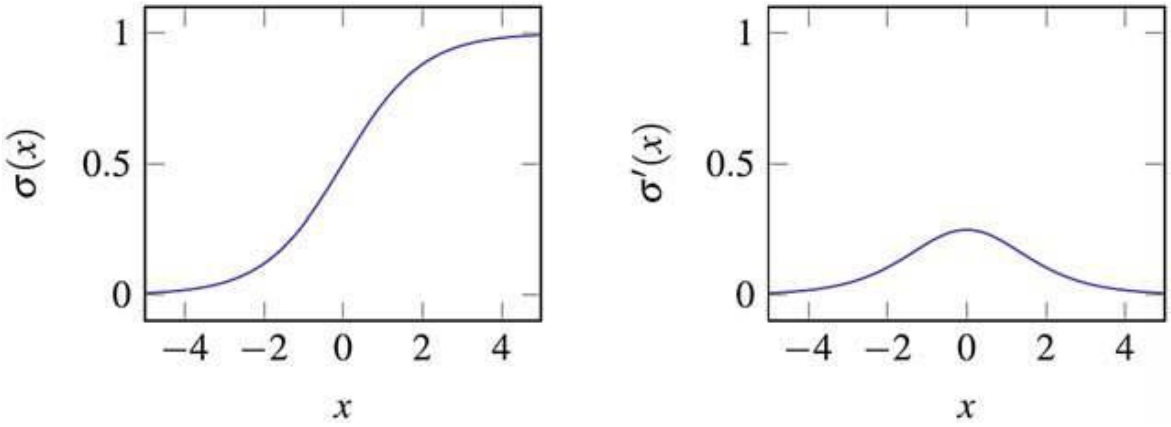


Figure (II-5): *Sigmoid activation function and its derivative.[29]*

II.2.3.1.2. Tanh:

Another popular activation function is the tanh function. It also serves as a sqaushing function, with output limited to the range(1,1). Because it is zero-centered, and the tanh function eliminates one ot the problems associated with sigmoid non-linearity. However, we still have the same problem with gradient saturation at the function's exremes.

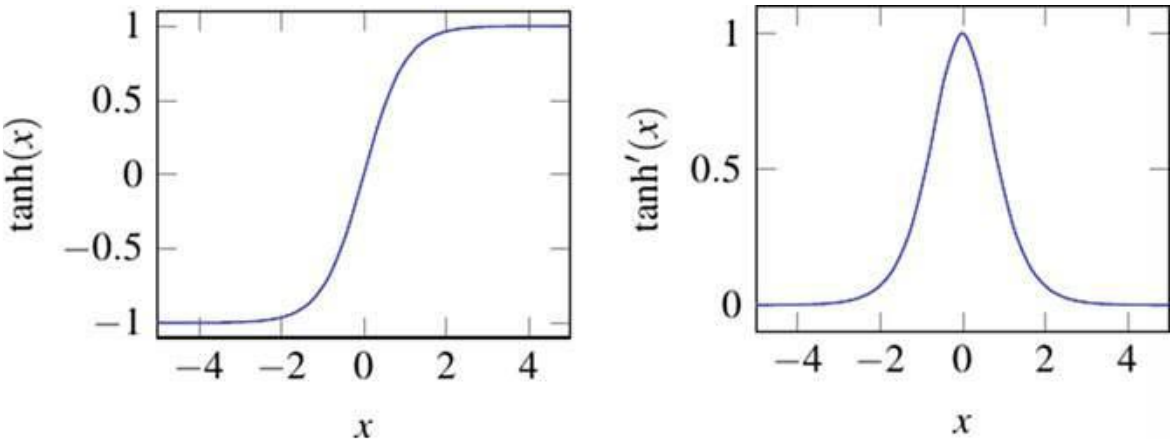


Figure (II-6): *Tanh activation function and its derivative.[29]*

II.2.3.1.3. ReLu:

The rectified linear unit (ReLu) is a simple and quick activation function that is commonly used in the computer vision. This basic function has gained popularity due to its faster convergence. When compared to sigmoid and tanh, it may be because of its non-saturating gradient in the positive direction.

The ReLu function is substantially quicker computationally, in addition to faster convergence. The sigmoid and tanh functions necessitate exponentials, which take significantly longer than a simple max operation. [29]

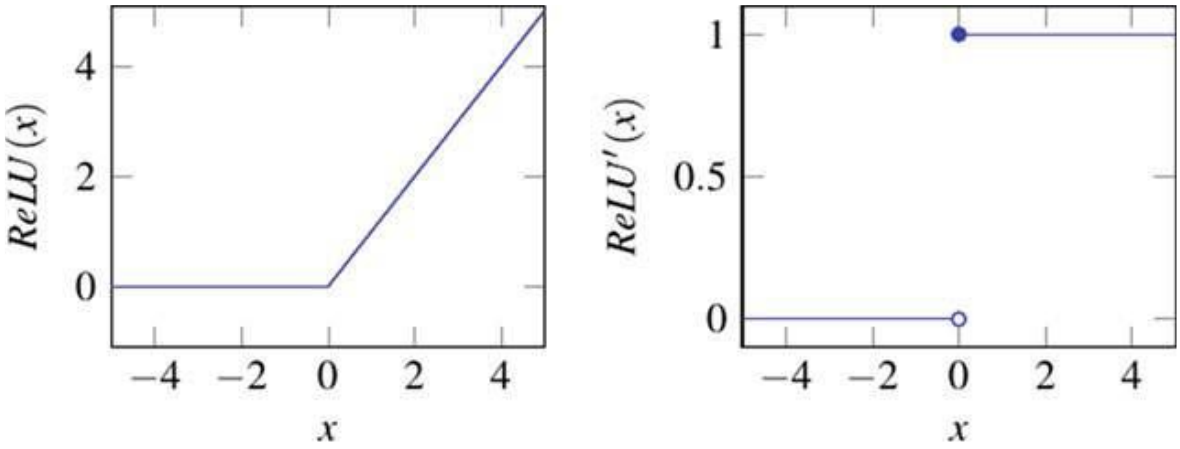


Figure (II-7): *ReLu activation function and its derivative.[29]*

II.4. Speech command recognition using deep learning

The Deep Neural Network demonstrated great improvement in speech feature extraction and recognition.

With the rise of deep learning, the initial layers of deep networks have fundamentally replaced feature extraction. Time-frequency transformations, such as the short-time Fourier transform (STFT), can be used as signal representations for training data in deep learning models.

Convoltional neural networks are commonly used for image data and may learn from 2D signal representations provided by time-frequency transformations. [30].

Deep CNN can be formed by stacking up a CNN with a fully connected DNN or with one or more CNNs where it performs a robust success for image and speech recognition.

- Google Speech command dataset

The TensorFlow and AIT teams collaborated to build the Google Speech Commands Dataset. There are 65.000 one-second clips in the collection. Each clip contains one of the 30 different words spoken by thousands of different subjects.

The clips were recorded in realistic environments with phones and laptops. The 35 words are given below, and they include noise words as well as the 10 command words that are most helpful in a robotics environment:

- Yes
- No
- Up
- Down
- Left
- Right
- On
- Off
- Stop
- Go

It’s organized into more than 30 categories, including instructions like stop and up, as well as other things like numbers and names. There are around 2400 recordings in each folder. [31] We convert the data into the time frequency domain using spectrograms once we get it. Then, as illustrated in Figure (II-6), we employ these spectrograms as an input to our Deep Convolutional Neural Networks.

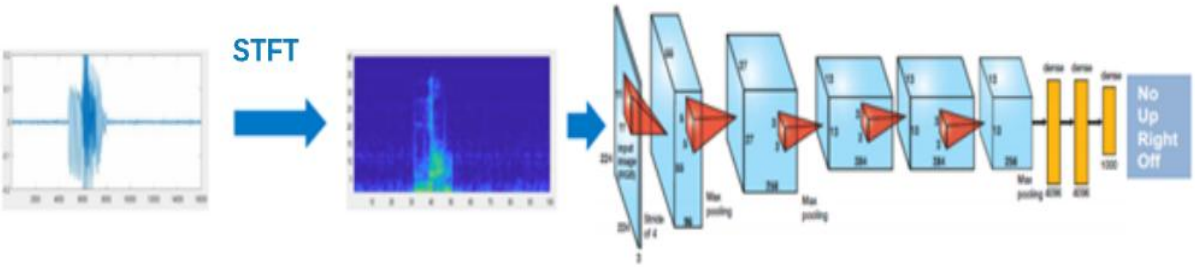


Figure (II-8): *Audio classification usinf deep CNN[32]*

II.4.1.Short time Fourier transform

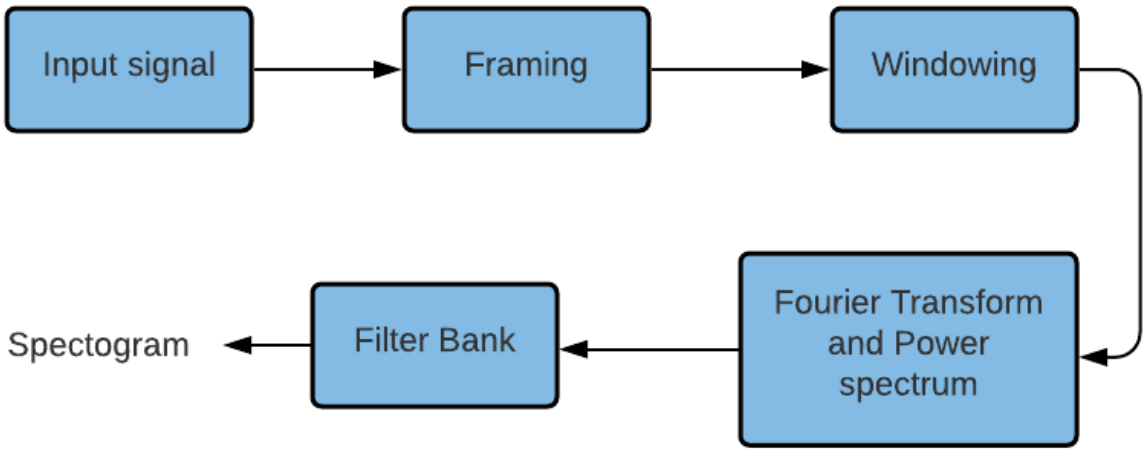


Figure (II-9): *Short time Fourier transform*

- **Input signal:**

Because an analog signal is a continuous time-varying signal, the instantaneous voltage of the signal changes continuously with the pressure of the sound waves in an analog audio transmission. The input voice signal is saved in (.wav) format, as indicated in the diagram. That (.wav) file will be essential for various modifications that are required to extract features from audio sources.

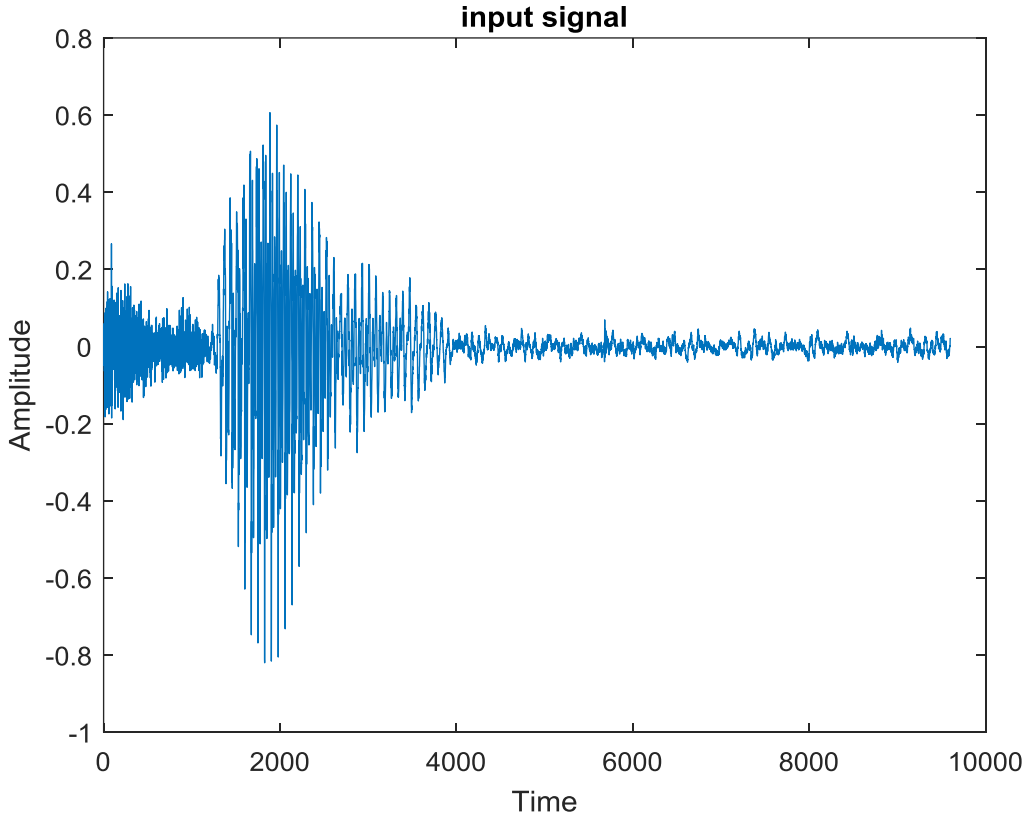


Figure (II-10): *An example of an input speech signal.*

- **Framing:**

An audio signal is constantly changing. So, to make things easier, we assume that the audio signal doesn't vary significantly across short time scales (when we say it doesn't change, we mean statistically). That's why we frame the signal into 20-40 ms frames. Figure (II-9) depicts audio signal frames.

If the frame is much shorter, we don't have enough samples to get a reliable spectral estimate. If it is longer, the signal changes too much throughout the frame.

The numbers of frames are generally 256 (in power of 2) because when FFT is calculated, it would be simple if the numbers of frames were in power of 2.

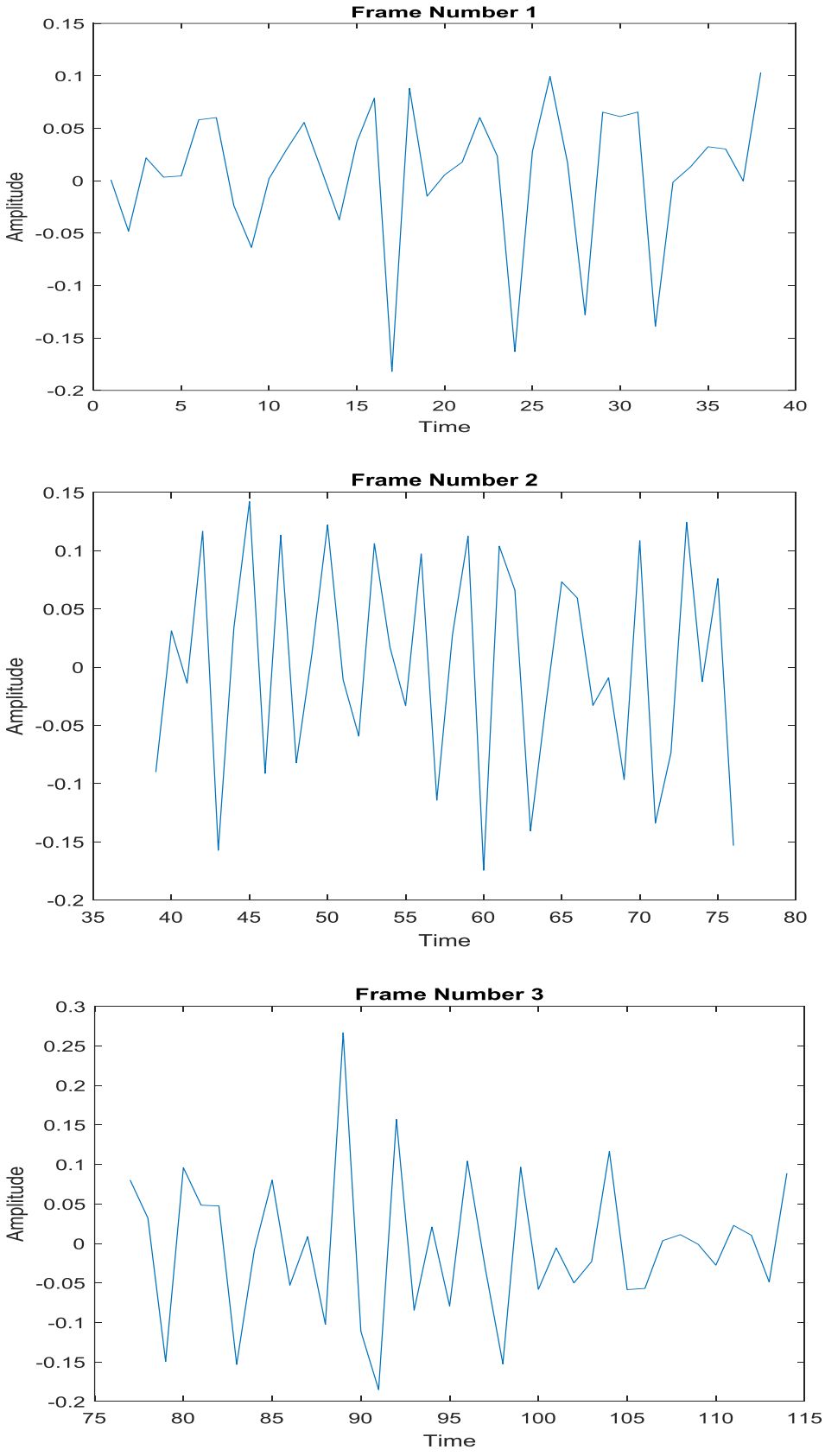


Figure (II-11): *Frames of speech signal.*

- **Windowing:**

The next step after framing is windowing each individual frame. The objective of windowing is to reduce signal discontinuities at the beginning and end of each frame.

The window function is defined as W (n), with n ranging from 0 to N-1. The length of the frame is indicated by the letter N. The result of windowing is the signal given by equation (1).

$$Y(n) = x(n)\, w(n), \quad 0<n<N-1 \qquad \text{(II-1)}$$

The hamming window was considered because the parameter side-lobe is good there. The form of the hamming window function is shown in Figure (II-10).

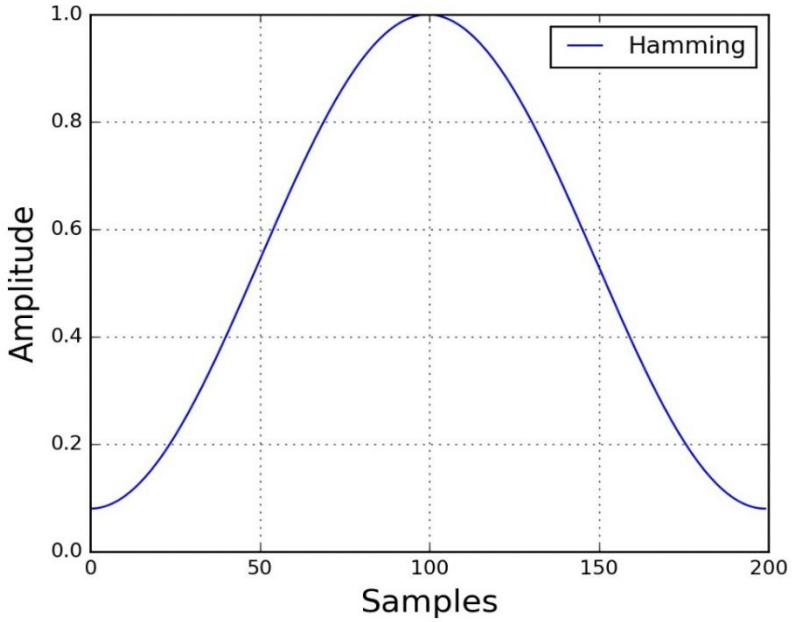


Figure (II-12): *Hamming window.*

Though there are other windows like triangle window function, rectangle window function but hamming window shows the Gaussian characteristics, which has the form:

$$W(n) = 0.54 - 0.46 \cos\left(\frac{2\pi N}{N-1}\right), \quad 0<n<N-1 \qquad \text{(II-2)}$$

- **Fourier Transform and power spectrum**

After that, each frame is exposed to a Fast Fourier transform. As a result, each frame of N samples must be represented in the frequency domain by transforming it from the time domain to the frequency domain using the Fast Fourier transform, which is extensively used in engineering, mathematics, and science for many purposes.

The frequency domain is more effective than the time domain in audio signals because the frequency of a particular individual is an effective better way to represent that person than the amplitude of the signal.

Figure (II-11) illustrates both the domain of the analog signal domain and the digital signal domain.

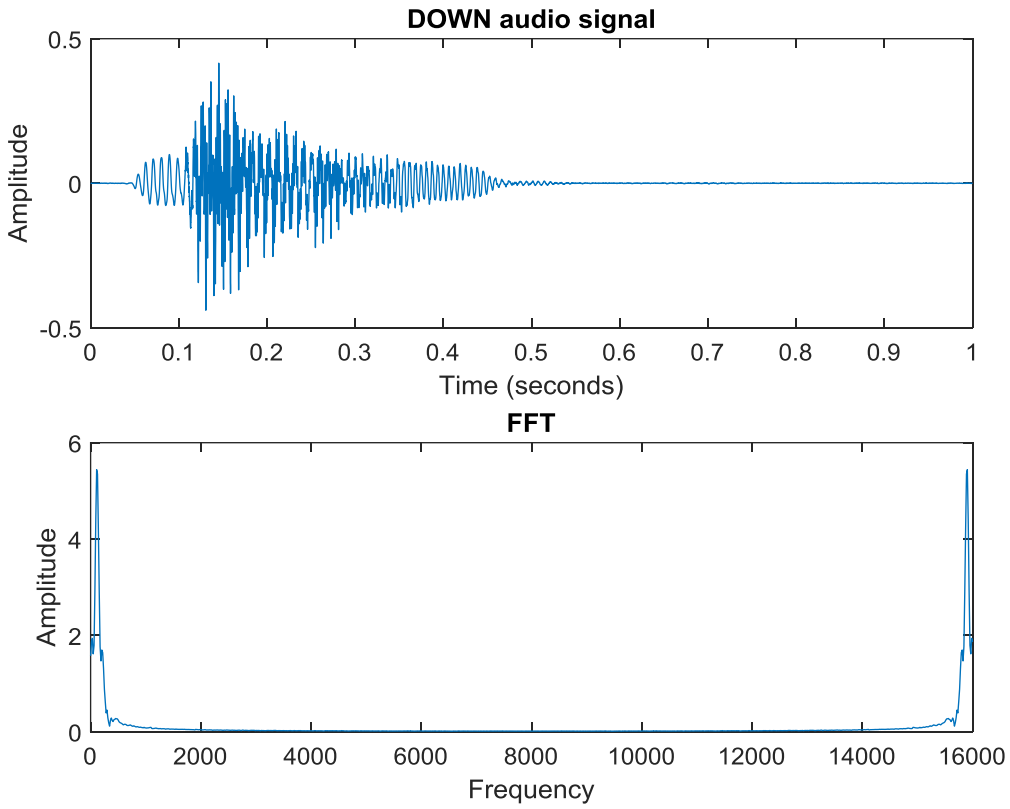


Figure (II-13): *The time and frequency domain.*

And then, to compute the power spectrum, we use the following equation :

$$\boldsymbol{P} = \frac{|\boldsymbol{FFT}(\boldsymbol{xi})|^2}{\boldsymbol{N}} \qquad (\boldsymbol{II}\text{-}3)$$

Where xi, is the ith frame of the signal x.

- **Mel scale filter bank**

Mel is an abbreviation of melody, and melody is very connected with the concept of pitch.

The Mel scale relates the perceived frequency, or pitch, of a pure tone to its actual measured frequency. At low frequencies, humans are significantly better at detecting tiny variations in pitch than they are at high frequencies. By including this scale, our features become more closely aligned with what humans hear.

The formula used to calculate the Mels frequency for any frequency is:

$$\text{Mel}(f) = 2595 \times \log(1 + f/700) \qquad \text{(II-4)}$$

Mel (f): the frequency (mels) and f the frequency (HZ)

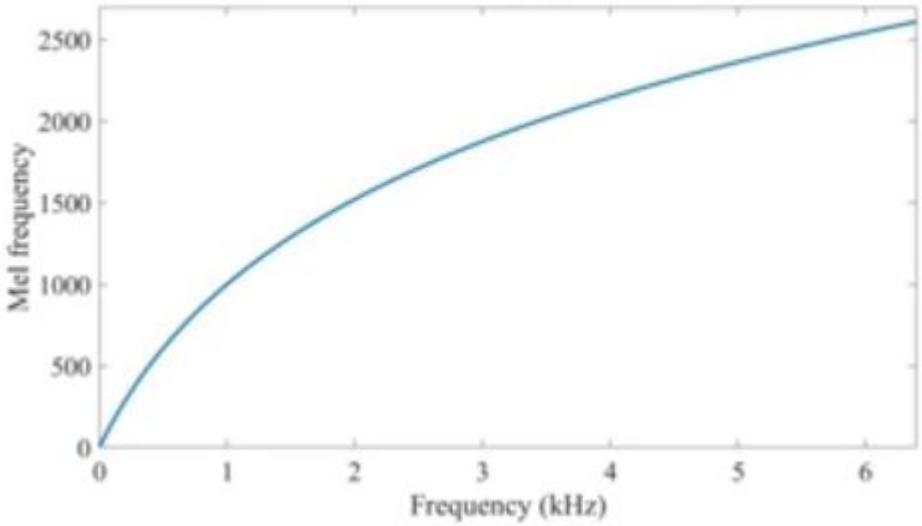


Figure (II-14): *Mel-frequency scale.*

The final step to computing filter banks is applying triangular filters, typically 40 filters.As illustrated in Figure (II-13), each filter in the filter bank is triangular, with a response of 1 at the center frequency and decreasing lineray towards 0 until it reaches the center frequencies of the two neighboring filters, when the response is 0.

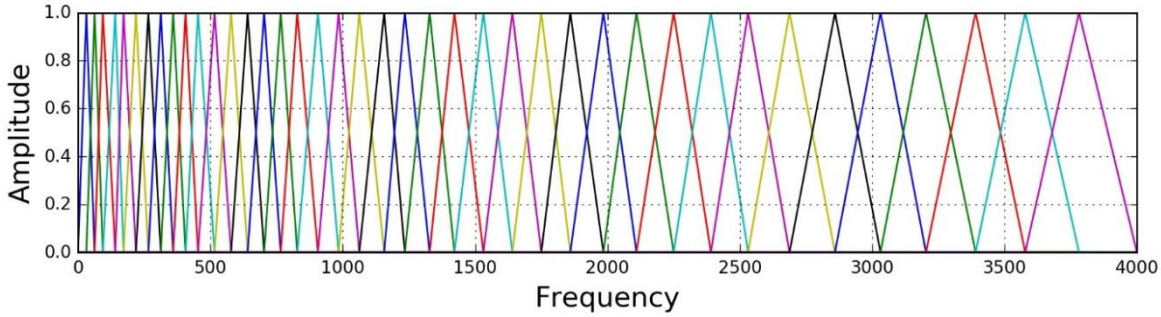


Figure (II-15): *Filter bank in mel-frequency scale*

After applying the filter bank to the power spectrum of the signal, we obtained the following spectogram: [33]

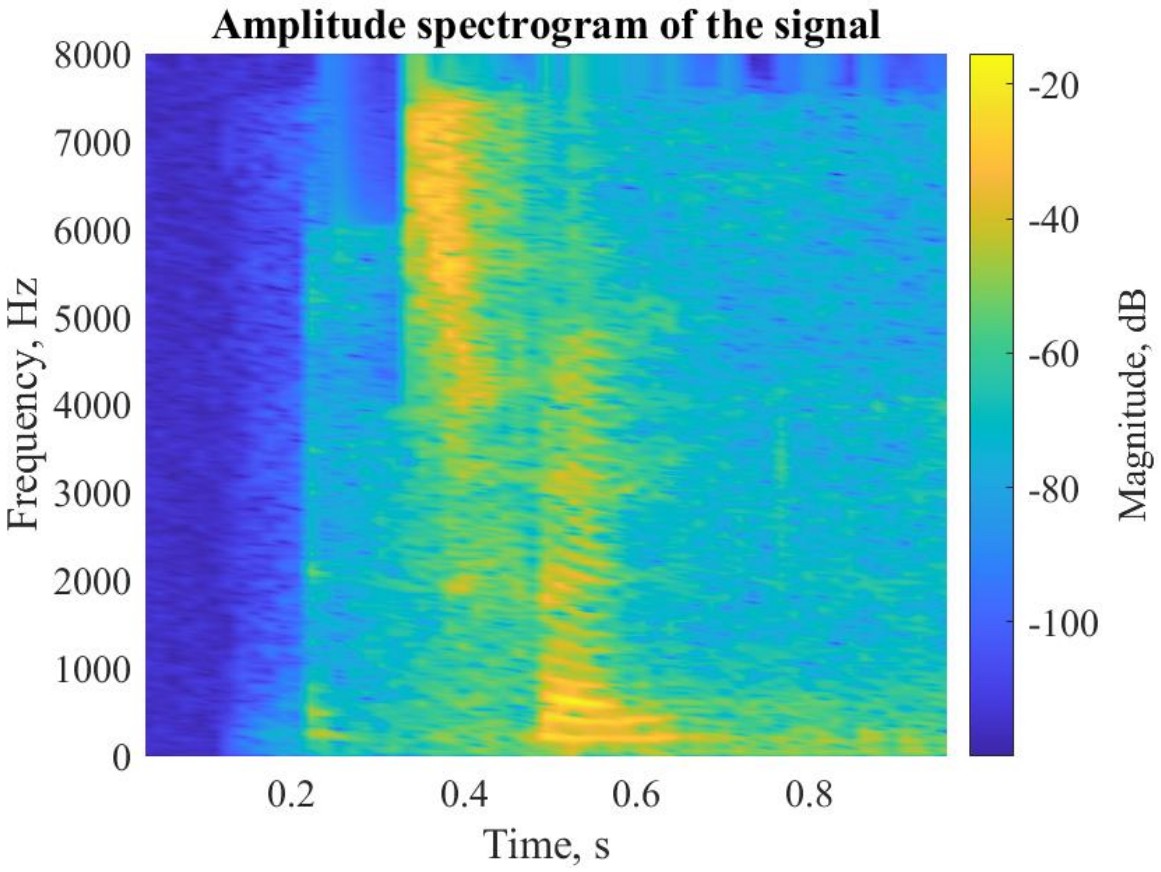


Figure (II-16): *Spectogram of the signal.*

A spectogram of a signal depicts its spectrum over time and is similar to a signal's picture.Time is plotted on the x-axis, while frequency is plotted on the y-axis.It's as if we took the spectrum at various points in time and then stitched them all together into a single plot. It employs a variety of colors to represent the amplitude or intensity of each frequency.The brighter the color, the higher the energy of the signal.

Now, we use these spectograms as an input to our Deep Neural Network to train and test the model as shwon in figure (II-17) and (II-18) [32]

- **Training:**

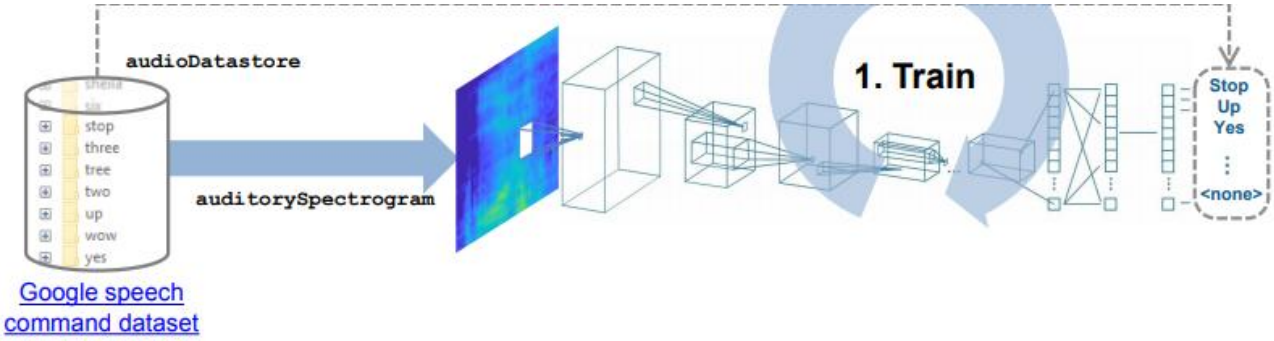


Figure (II-17) : *Training the model[29]*

- Test of recognition:

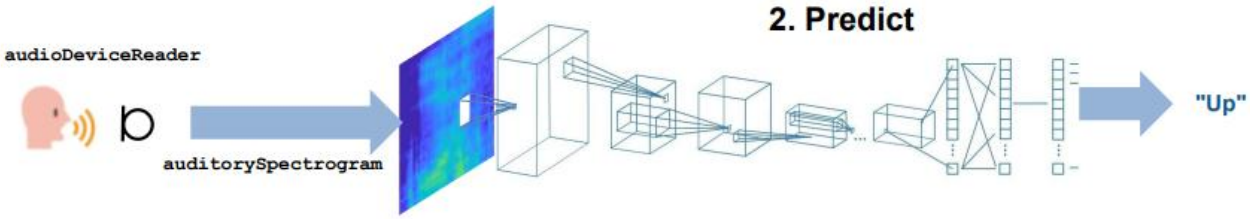


Figure (II-18): *Speech command recognition using Deep Learning for Audio Classificatio[29]*

II.5. Conclusion

Speech processing is essential in any speech system, whether it is Automatic Speech Recognition (ASR), speaker recognition, or anything else... As an example, Mel-Frequency Cepstral Coefficients (MFCCs) were popular features for a long time, but filter banks have lately gained popularity.

With the rise of deep learning, the initial layers of deep networks have essentially replaced feature extraction, but mostly for image data.

CHAPTER THREE

Multilingual speech recognition

III.1. Introduction

There has recently been an increase in rapidly developing high performance automatic speech recognition (ASR) systems for a variety of languages. Speech recognition systems built with deep neural networks (DNNs) have been shown to provide consistent advantages, especially for low-resource languages. [34]

First, we'll work on Google's monolingual database (in English), and then we'll add our multilingual database (in Arabic and Amazigh).

At least, we'll use this simulation as a UAV application.

III.2. Approaches to multilingual speech recognition

Depending on the application's aim, we describe three techniques for multilingual speech recognition: porting, cross-lingual, and simultaneous multilingual speech recognition.

These various techniques are determined by the application's aim, i.e., which and how many languages will be recognized at any one moment.

In addition, the technique used is determined by the available training data in terms of the spoken language, the speaker, and the recording settings.

III.2.1. Porting

Porting is the first method for multilingual speech recognition. A speech recognition system created for one language is ported to another language to be used in that language.

The recognition system is for the new language's sale, and the training data is just for the new language.

The old and new language systems are distinct, as seen in Figure (III-1). The new language's algorithms and principals are derived from the original language's recognizer, and only small adjustments are made to the algorithms in order to obtain optimal performance in the new language.

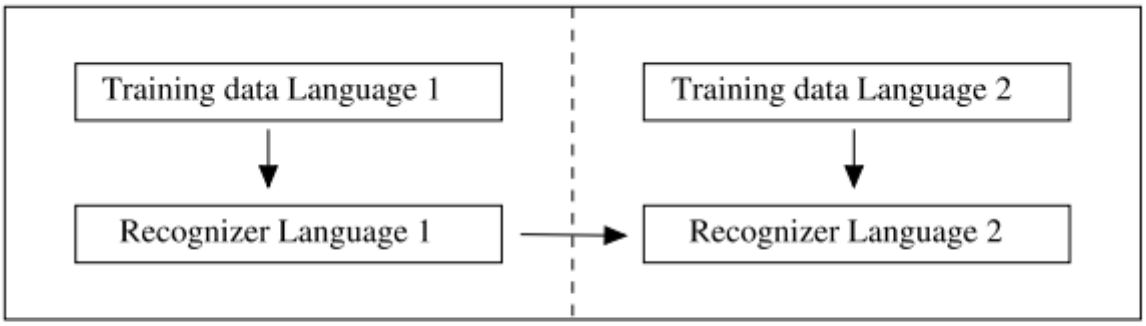


Fig. 1. Sketch of the porting scenario.

Figure (III-1): *A sketch of the porting scenario.*

III.2.2. Cross-lingual recognition

This method achieves the same purpose as the porting method. The difference from the previous technique is that there isn't enough training materials to train the recognizer in the new language. As a result, techniques for using training material of acoustic characteristics in cross-lingual recognition must be developed.

The languages that will be used to train the recognizer must be determined. It is necessary to identify the languages that lead to the best recognition performance in the new languages. It is necessary to choose a relationship between the languages used for training and the language to be recognized.

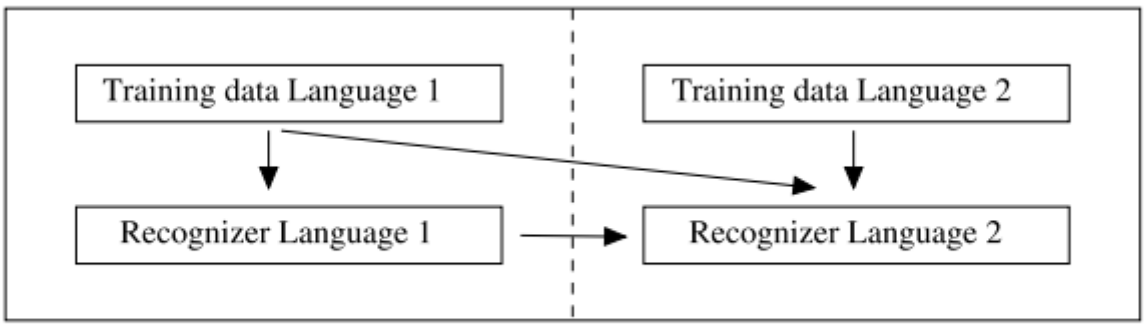


Fig. 2. Sketch of the cross-language scenario.

Figure (III-2): *A sketch of the cross- language scenario*

III.2.3. Simultaneous multilingual speech recognition

The simultaneous multilingual recognition technique is the third cluster of approaches. This technique allows applications to recognize speech in several languages at the same time. Technology has no way of knowing which language is being spoken. Figure

(III-3) is a rough drawing of this method. Each language has its own set of training data. As a result, there is now a single recognizer for all of the languages concerned.

For simultaneous multilingual speech recognition, there are two primary strategies: explicit language identification and implicit language identification.

The first approach uses voice signal to identify the language. The speech recognition system for the specified language is engaged when the language is identified, and the utterance is recognized. The strategy's benefit is that it produces results that are equivalent to monolingual recognition as long as the language identification stage is completed correctly.

The other method involves language identification that is done implicitly. The words in all of the languages concerned can be recognized by the distribution of language models. It is possible to switch between the languages.

The recognized words can be used to determine the spoken language. The same acoustic models may be used for all of the languages included in the strategy.

Furthermore, instead of a cluster of monolingual language models sharing a common start and end node, a single multilingual language model may be used. Depending on the languages and data available, the optimum technique within this approach may differ. If there is little data for one language, for example, acoustics units may be shared between languages. Multilingual units may improve performance if the languages are comparable or if the speakers include non-natives. If languages are similar, on the other hand, it may be more advantageous to keep them as distinct as possible in order to avoid confusion. [35]

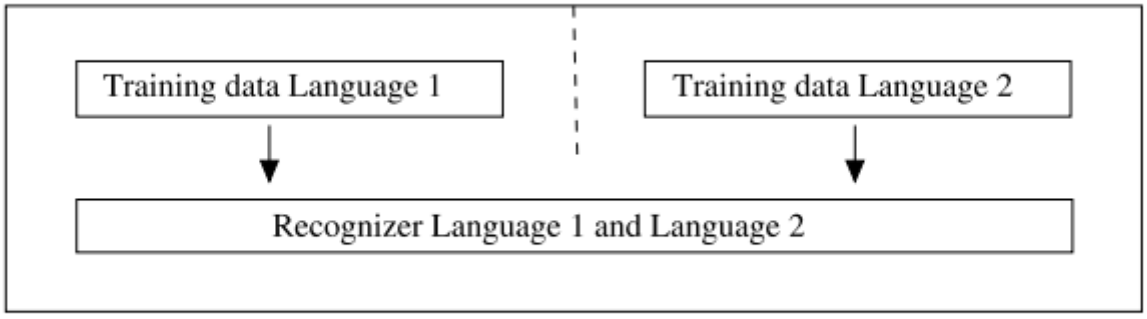


Fig. 3. Sketch of the simultaneous multilingual scenario.

Figure (III-3): *A sketch of the simultaneous multilingual scenario.*

III.3. Multilingual speech command recognition using deep learning

Figure (III-4): *An overview to the Multilingual step.*

III.3.1. Working environment

The development of the material environment is characterized by:

- Operating system : Windows 10 Professional
- Processor : Intel ® Xeon ® CPU ES_1650 v2 ® 3.50 GHz
- Memory : 64 GB

III.3.2. Implementation

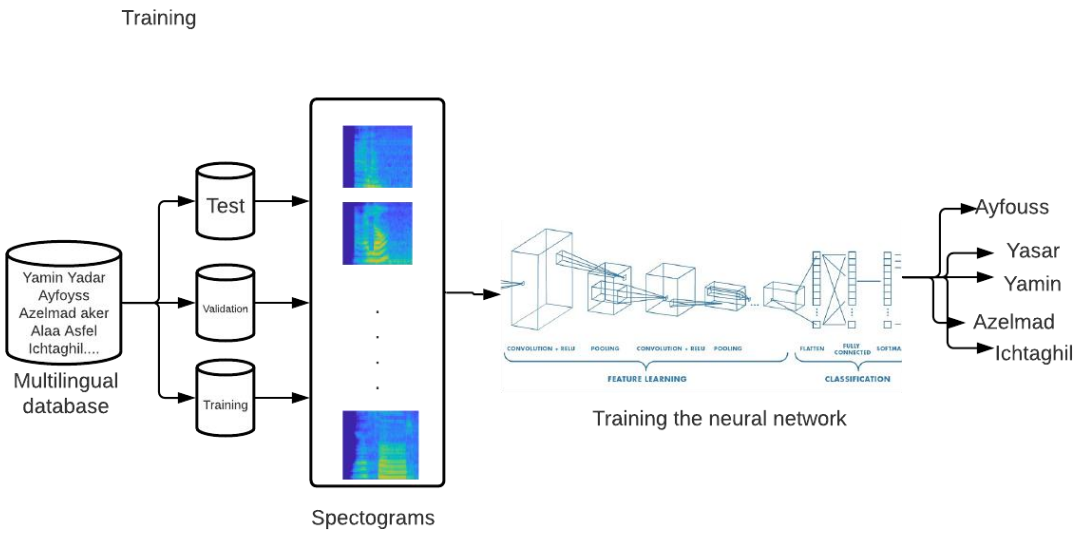


Figure (III-5): *The process of multilingual speech recognition.*

III.3.2.1. Development of the multilingual "Arabic and Amazigh" database

In this part, we will present the different phases of the realization of our database using screenshots.

III.3.2.2. Database registration

For this first step, we choose to create the database using a simple program in MATLAB R2020 with the following parameters.

The recordings have been recorded using a computer via a simple MATLAB program with a sampling frequency of 16 KHz, in (.wav) format,including 12 commands (six for each) and unknow words. Each folder has around 12000 records.

Table (III-1): *Database characteristic*

	Arabic database	Amazigh database
Sampling frequency (Fs)	16KHz	16KHz
Audio Format	.wav	.wav
Speakers	12(5M + 7F)	10(3M + 7F)
Number of commands	6	6
Number of words (Unknown)	20	20
Size of commands	12000	12000

The database of multilingual records is assembled all together as one command such as "up" means the same to the program even it address to it in different languages.

The following table summarizes the drone commands in Arabic and Amazigh.

Table (III-2): *The drone commands in Arabic and Amzaigh*

The command in English	The command in Arabic	The command in Amazigh	Action
UP	Aala	Oussawen	Increase the UAV's altitude
DOWN	Asfel	Oukser	Decrease the UAV's altitude
RIGHT	Yamin	Ayfouss	Move the UAV to the right
LEFT	Yasar	Azelmad	Move the UAV to the left
ON	Ichtaghil	Akker	Turn on the motors
OFF	Tawakef	Ekhsi	Turn off the motors

III.3.2.3. Separation of test data, validation and training

We separate speakers between training, validation and test sets. So, to efficiently find the best values for our algorithm, the best approach is to split our dataset into three independent sets:

- A training dataset for our algorithm.(80% of the dataset)
- A validation dataset to evaluate our trained algorithm which the training algorithm does not observe. (10% of the dataset)
- A test dataset for the evaluation of the final algorithm. (10% of the dataset)

Training dataset: A set of examples used to fit the model.

Validation dataset: The sample of data used to provide an unbiased evaluation of a model fit on the training dataset while tuning model hyper parameters. The evaluation becomes more biased as skills from the validation dataset are incorporated into the model configuration.

Test dataset: The sample of data used to provide an unbiased evaluation of the final model fit on the training dataset.

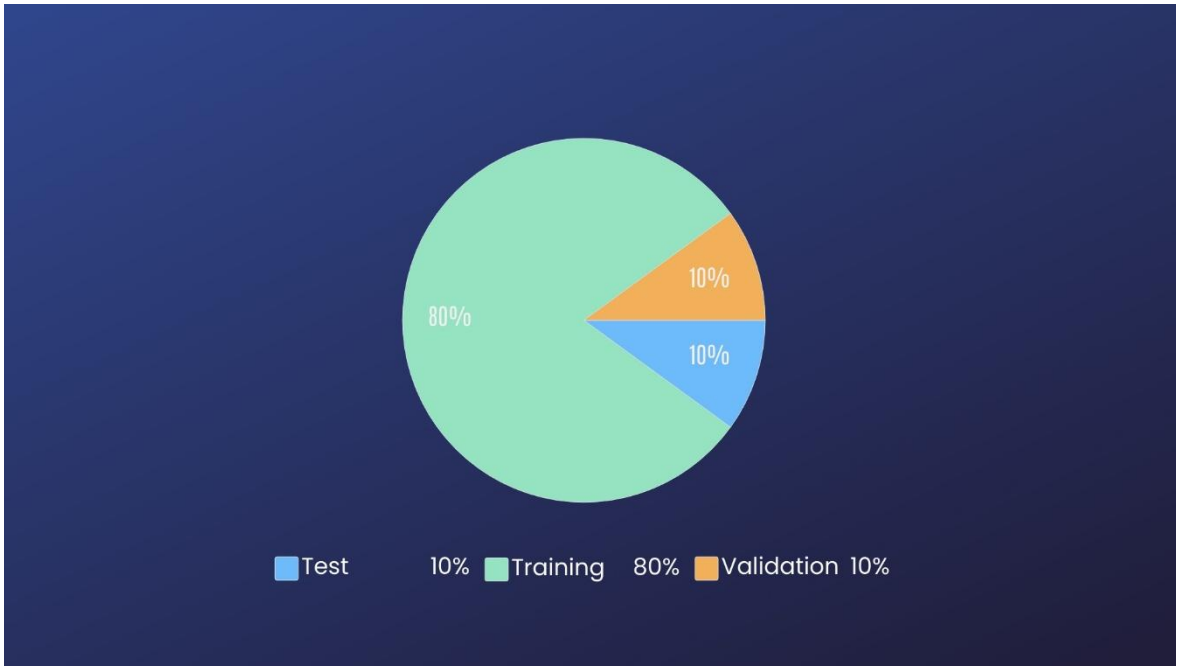


Figure (III-6): *Test,validation and training data.*

III.3.2.4. Converting signals to spectrograms

Once we have the data, we want to transform it into Mel-spectrograms. Figure (III-7) shows a random sample of three files in the data set and the spectrogram is produced from them.

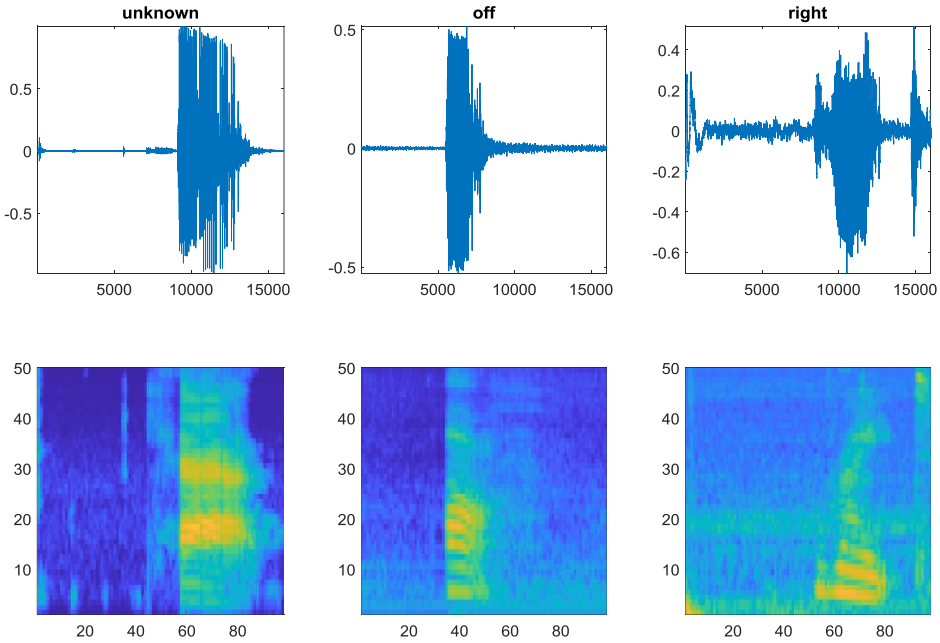


Figure (III-7): *From signals to spectograms.*

Now, we feed it to the network.

III.3.2.5. Neural network architecture

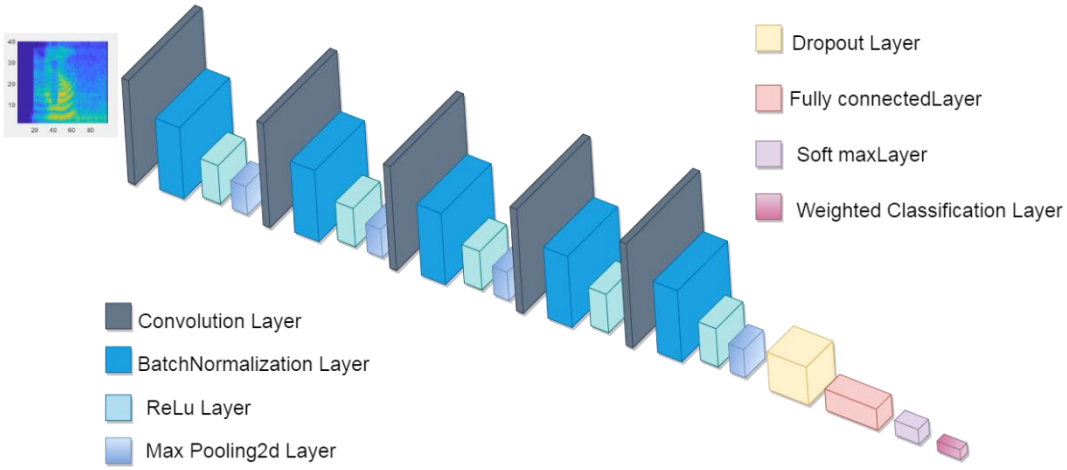


Figure (III-8): *Neural network architecture from deep network designer.*

As illustated in Figure (III-8), the network is made up of:

- 5 convolution layers: each layer acts as a feature extractor.
- 5 batch Normalization layers: have the effect of stabilizing the learning process and dramatically reducing the number of training epochs required to train deep networks.
- 5 ReLu Layer: acts as an activation function.
- 3 Max pooling layers: Downsample the feature maps (in time and frequency).
- Max pooling2dLayer: pools the input features map globally over time.
- Dropout Layers: to reduce the possibility of the network memorizing specific features of the training data.
- Fully connected Layer: allows us to perform classification on our dataset.
- Softmax Layer: we find it just after the fully connected layer in order to predict classes.
- Weighted Classification Layer: calculates the cross entropy loss.
- numF: controls the number of filters in the convolutional layers.

So, to increase the accuracy of the network, we can increase the network depth by adding identical blocks of convolutional, batch normalization, and ReLu layers, or we can increase the number of filters (numF).

Once we have the network ready, we can proceed to train it.

III.3.2.6. Training the network

Training is the hardest part of Deep Learning because we need a large data set and a large amount of computational power.

Training a network is a classical optimization problem. There's a cost function that requires the network to produce outputs as close as possible to the prescribed ones, and then an algorithm to find the values of the network weights that minimize the cost function (it's called back propagation).

III.3.2.6.1. Comparison between the monolingual and the multilingual training

Figure (III-9) represents the training with only the English database (monolingual).

Figure (III-10) represents the training with the English, Arabic and Amazigh databases.(multilingual).

For our training, we use the Adam optimizer with a mini batch size of 128. Train for 25 epochs and reduce the learning rate by a factor of 10 after 20 epochs.

In Matlab, all you need to do is call this train function.

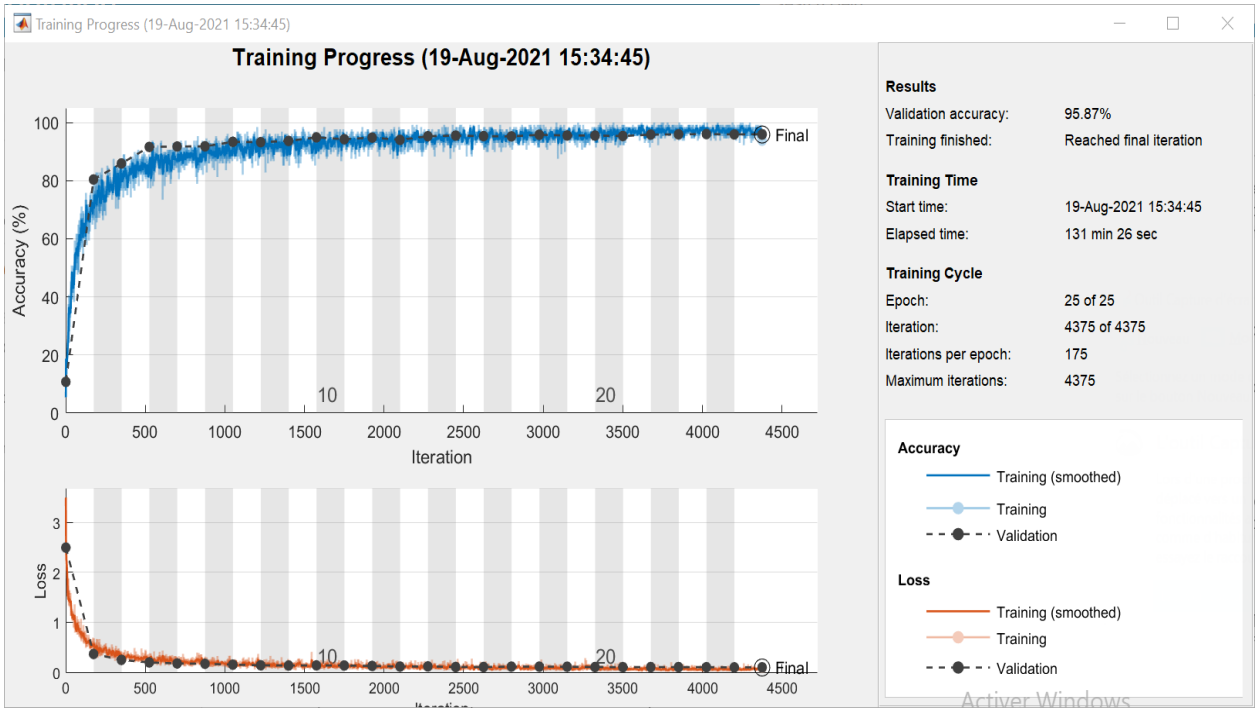


Figure (III-9): *Training progress for monolingual dataset (English)*

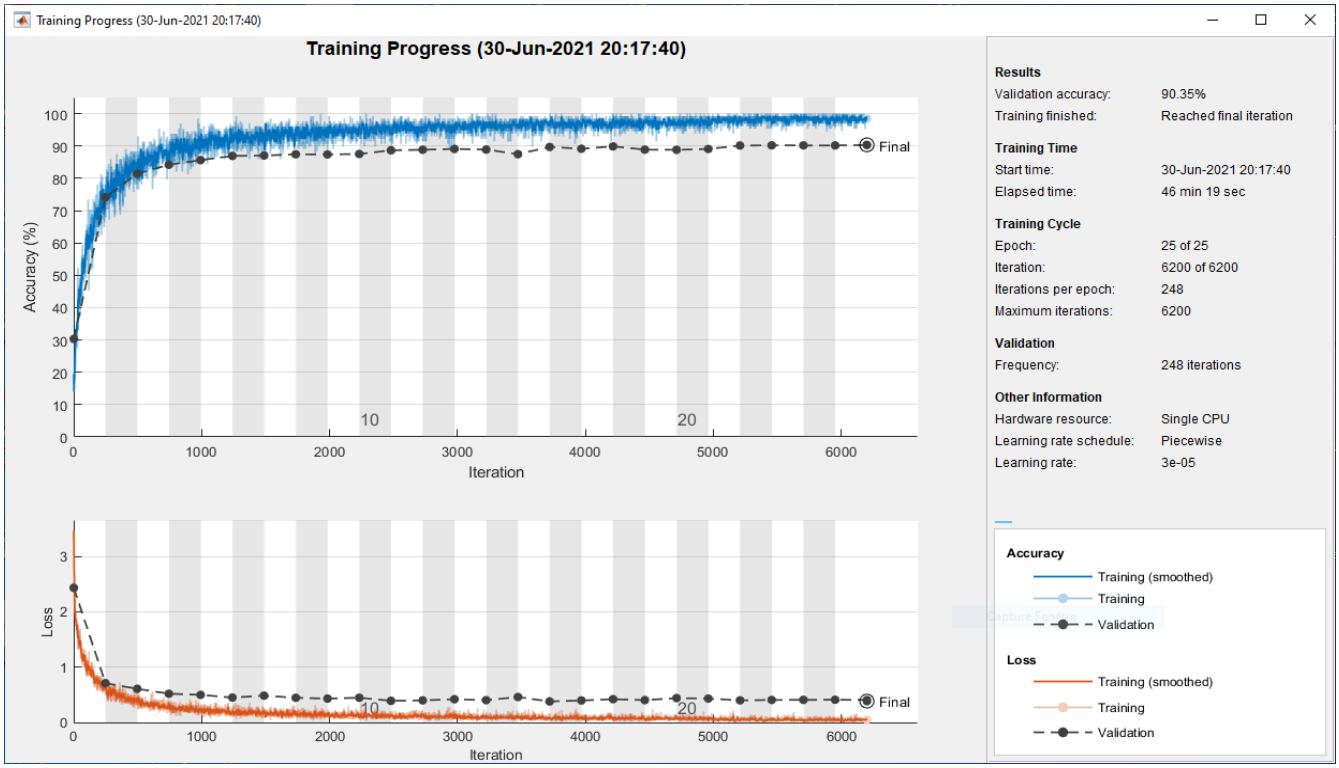


Figure (III-10): Training progress for multilingual dataset (English, Aabich, and Amazigh).

The continuous blue line represents the accuracy reached on the training data and the dashed black line is updated less frequently, which is the accuracy reached on the validation data.

The validation data is not used to optimize the network weights during training, but only to check that it is obstructing well enough what it learns from the training data.

Ideally, we want the black dashed line to be very close to the blue line.

Evaluated trained Network:

We can also check that our network does as well as expected on the training data.

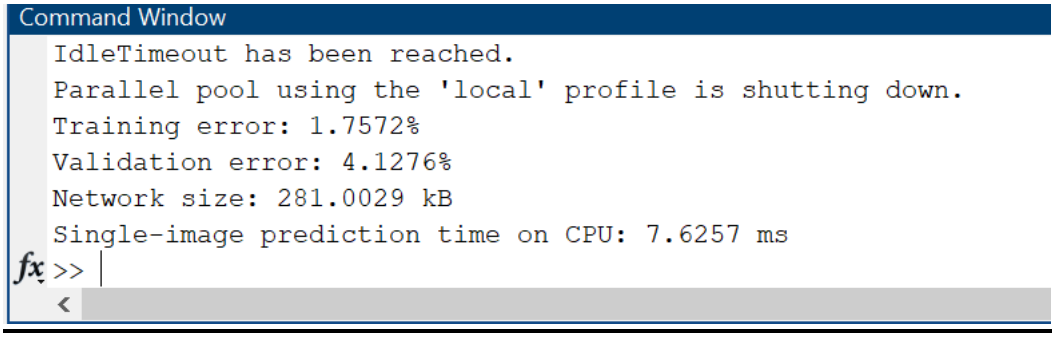


Figure (III-11): *Training results in a monolingual dataset (English)*

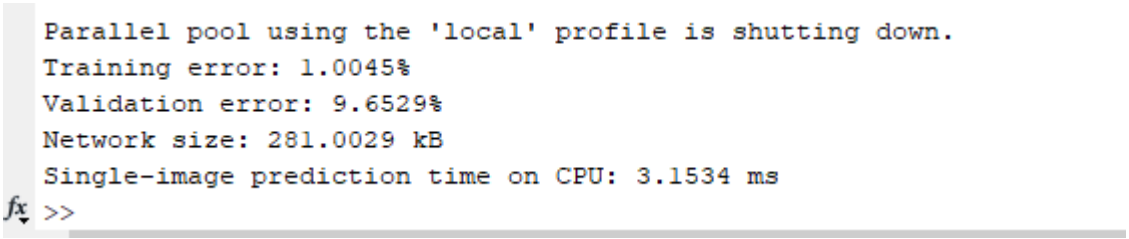


Figure (III-12): *Training results in multilingual dataset (English, Aabich, and Amazigh).*

Table (III-3) : *Comparison between the monolingual and the multilingual training*

	Monolingual	Multilingual
Training error	1.7572 %	1.0045 %
Validation error	4.1276 %	9.6529%
Validation accuracy	95.87 %	90.35%
Learning rate	3e-05	3e-05
Iteration	4375	6200
Epoch	25	25
Iterations per epoch	175	248

III.3.2.6.2. Results and discussion

Accuracy and loss are the most well-known and discussed metrics in deep learning. During the training process, the goal is to minimize this value. Loss is often used in the training process to find the best parameter values for the model.

Accuracy is a method of measuring a classification model's performance. It is typically expressed as a percentage. And it is easier to interpret than loss.

Most of the time, we would observe that accuracy increase with the decrease in loss (as it is shown in figures (III-9) and (III-10), but accuracy and loss have different definitions and measure different things. They often appear to be inversely proportional, but there is no mathematical relationship between them.

As mentioned in table (III-3), training error (multilingual) is less than training error (monolingual).

The validation error (multilingual) is greater than the validation error (monolingual).

The validation accuracy is 94.46% (monolingual) and 90.35% (multilingual), so we can say the monolingual model is more efficient than the multilingual model.

III.3.2.7. Confusion Matrix

You may also be more analytical and utilize a confusion matrix to evaluate the network's performance on test data,as shwon in figure (III-13) and (III-14).

The first table means that when given 248 recordings of "right" commands, the network got 248 of them correctly which means 96.9% of the "right" command got predicted correctly. However, there were only 248 predicted as "right", thus those same 248 correct predictions represent 95.0% of all"right" predictions.

The second table can suggest us things to improve. For example, in the confusion matrix for the multilingual dataset, six of the true "right", "Yamin" or "Ayfouss" were wrongly recognized as "left","Yassar" or "Azelmad" because of the sounds that were confused.

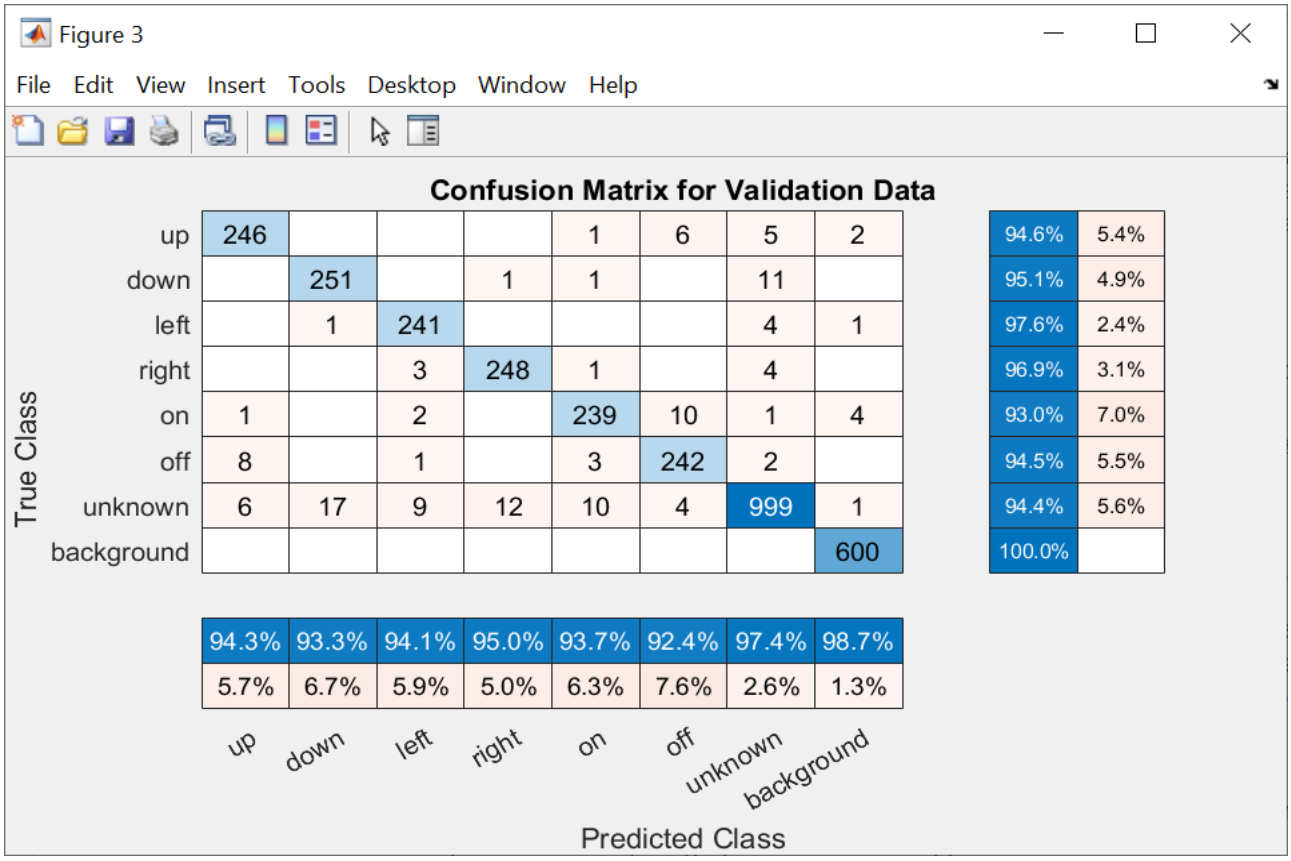


Figure (III-13): *Confusion matrix for monolingual dataset*

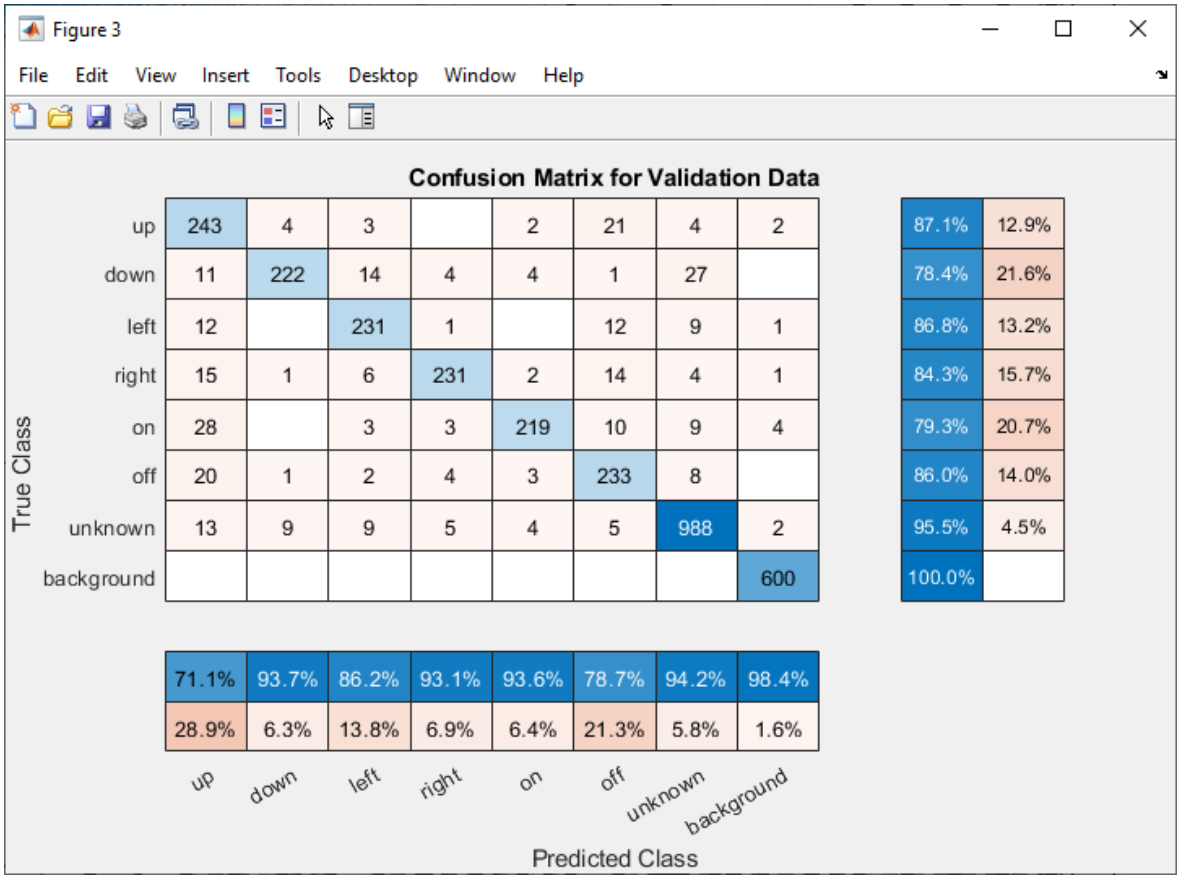


Figure (III-14): Confusion matrix for multilingual dataset

III.3.2.8. Test

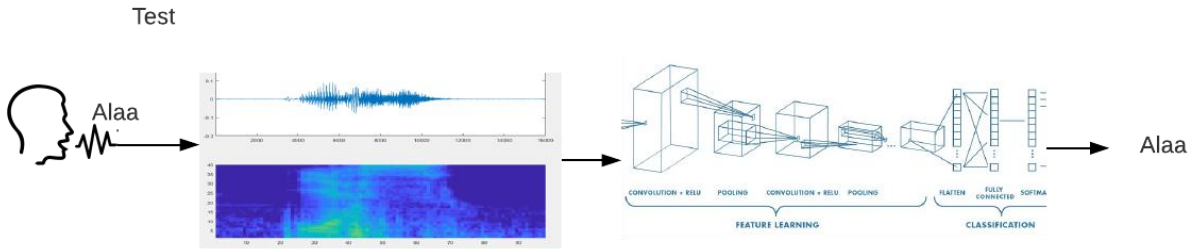


Figure (III-15): *Test of recognition.*

The test is done in real time through a simple MATLAB program which displays figures (III-16).

The first part represents the signal in real time and the second part represents the spectogram of the spoken signal.

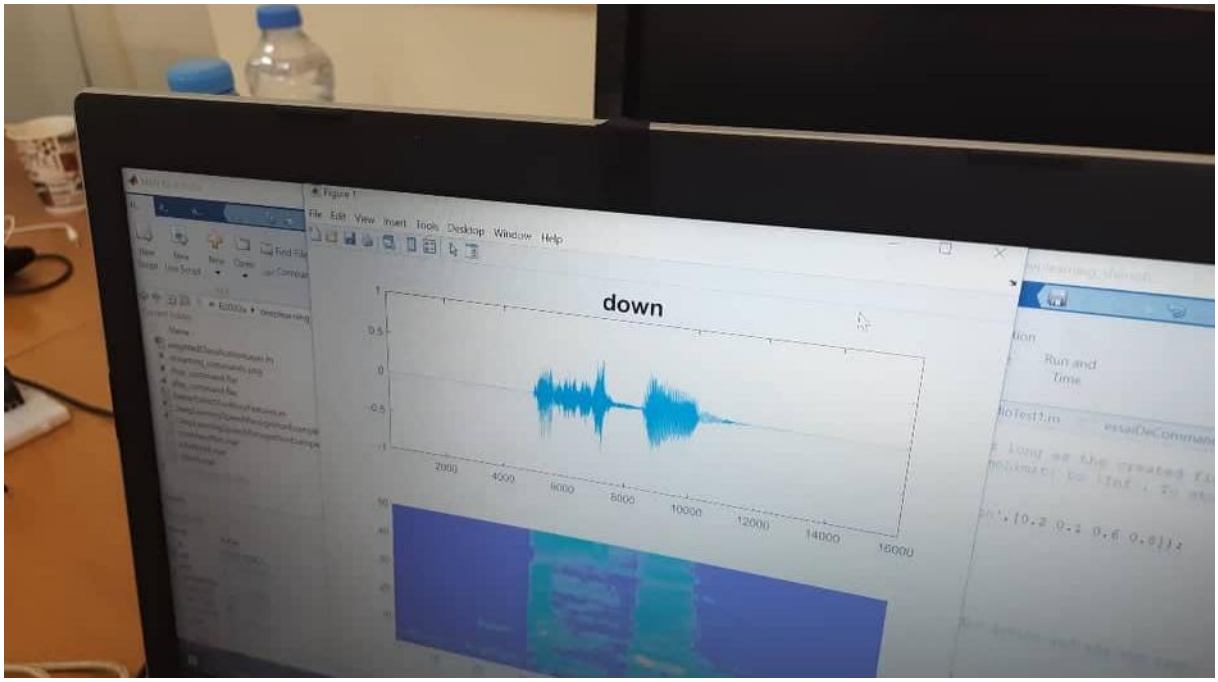


Figure (III-16): *Command detected in real time.*

We did the test on a variety of people in order to see the performance of our database. For this, we used 16 people, as it is shown in figures (III-17) and (III-18).

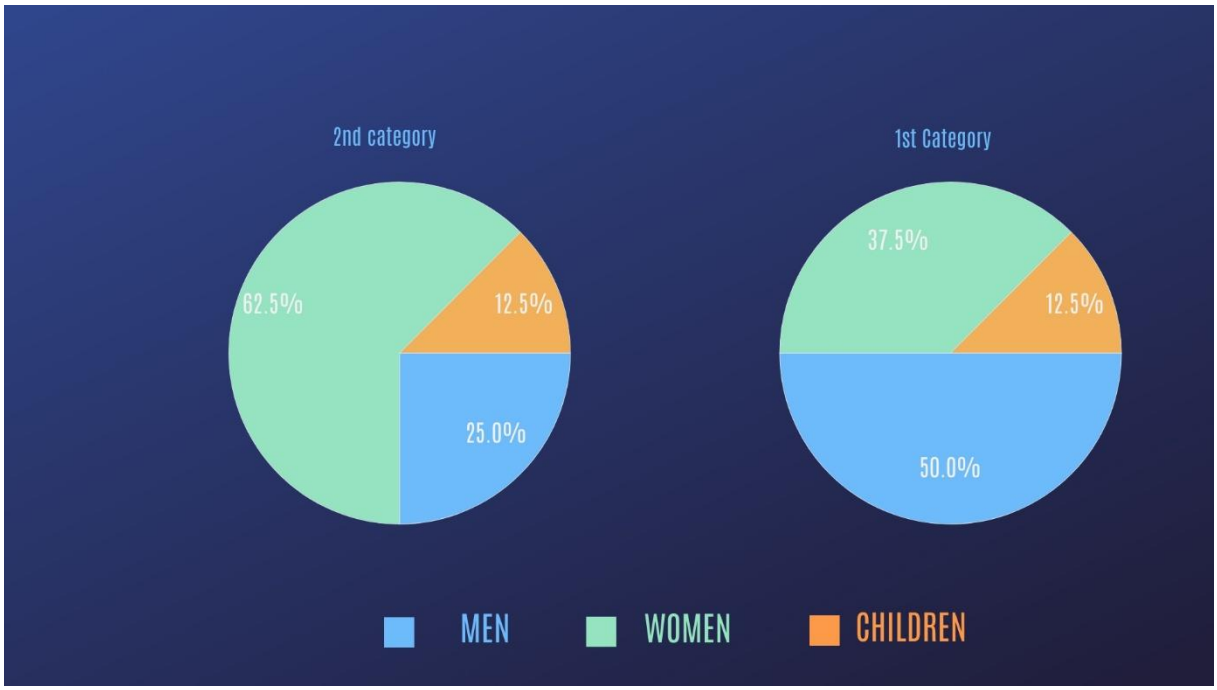


Figure (III-17): *The two groups.*

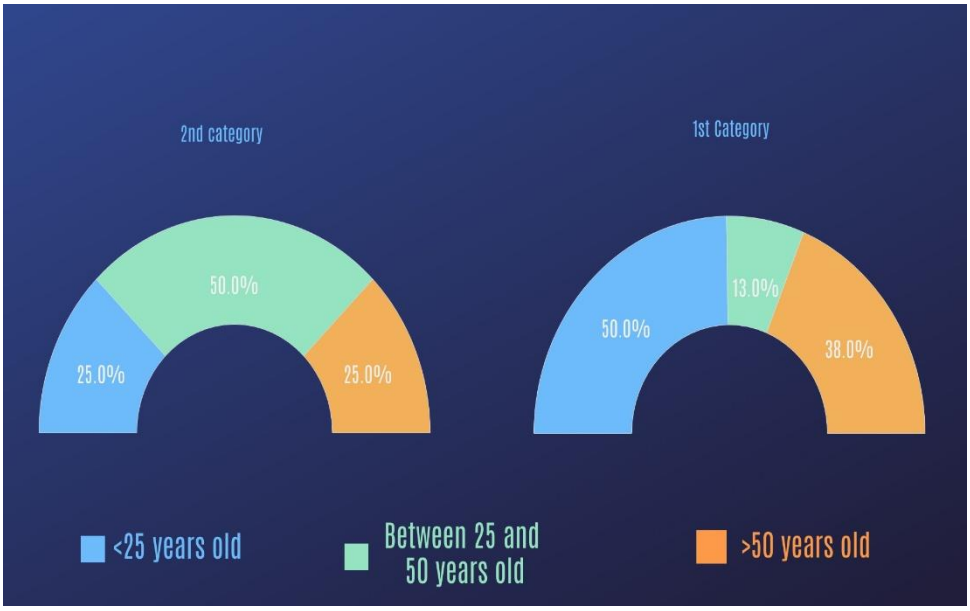


Figure (III-18): *The age of the categories.*

III.3.2.8.1. Test of monolingual dataset

Test of English command:

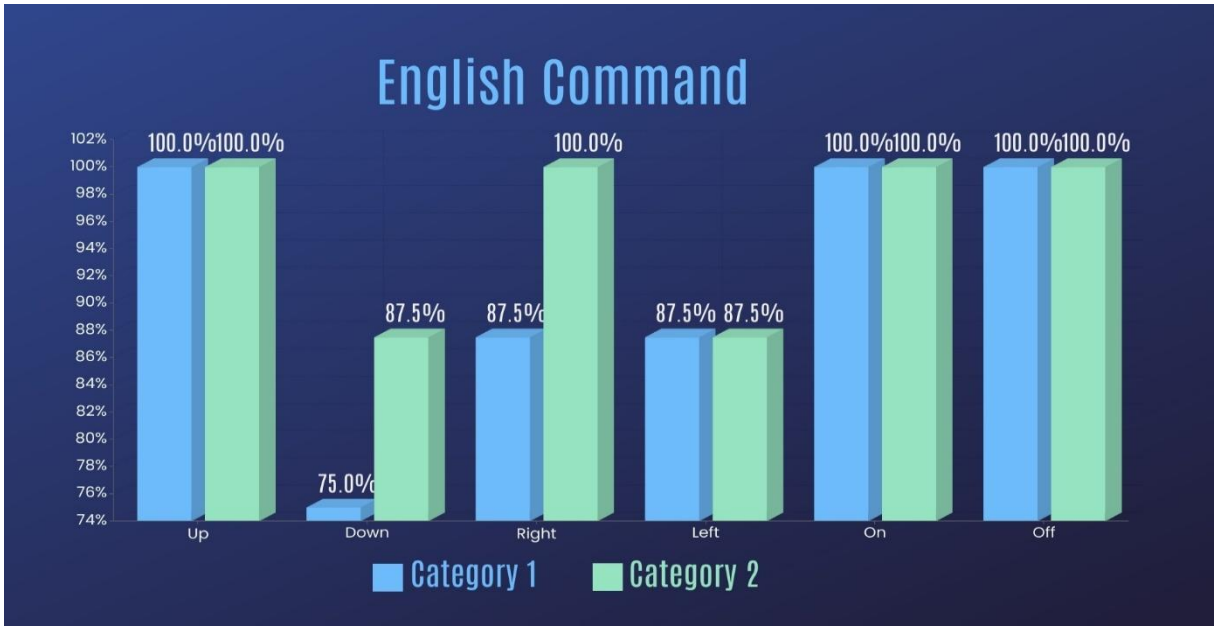


Figure (III-19): *Test of English command*

III.3.2.8.2. Test of multilingual dataset

1 st Test : Test of Amazigh command :

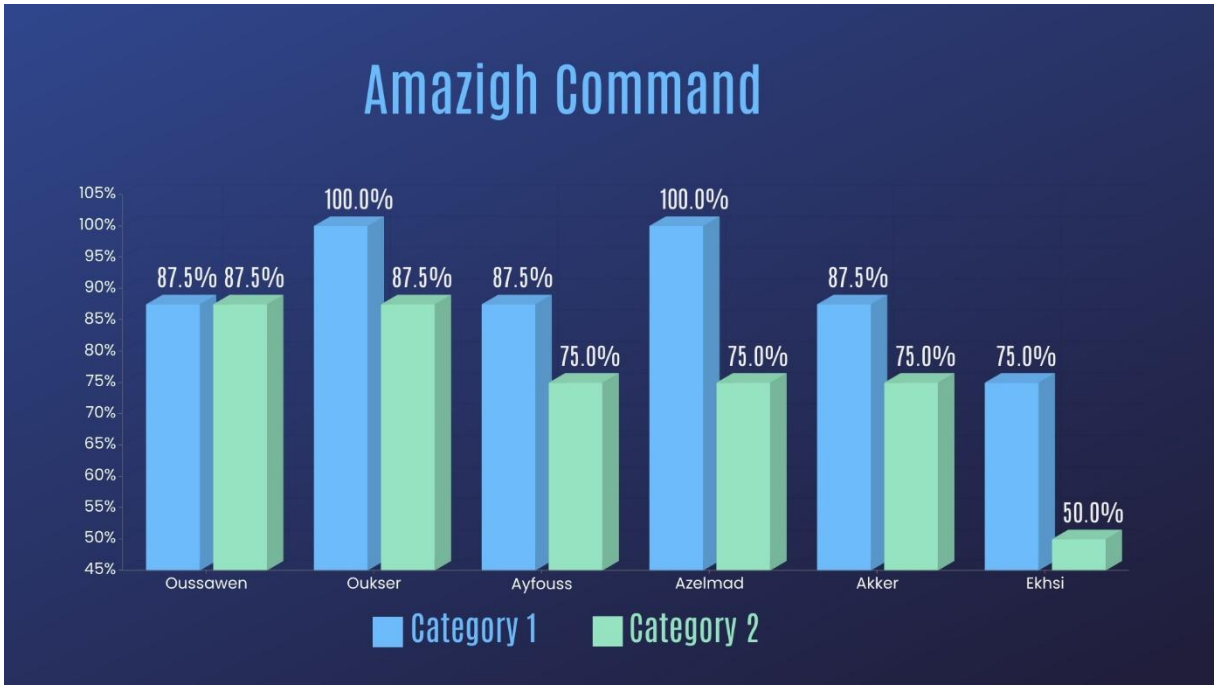


Figure (III-20): *Test of Amazigh command.*

2 nd Test : Test of Arabic command:

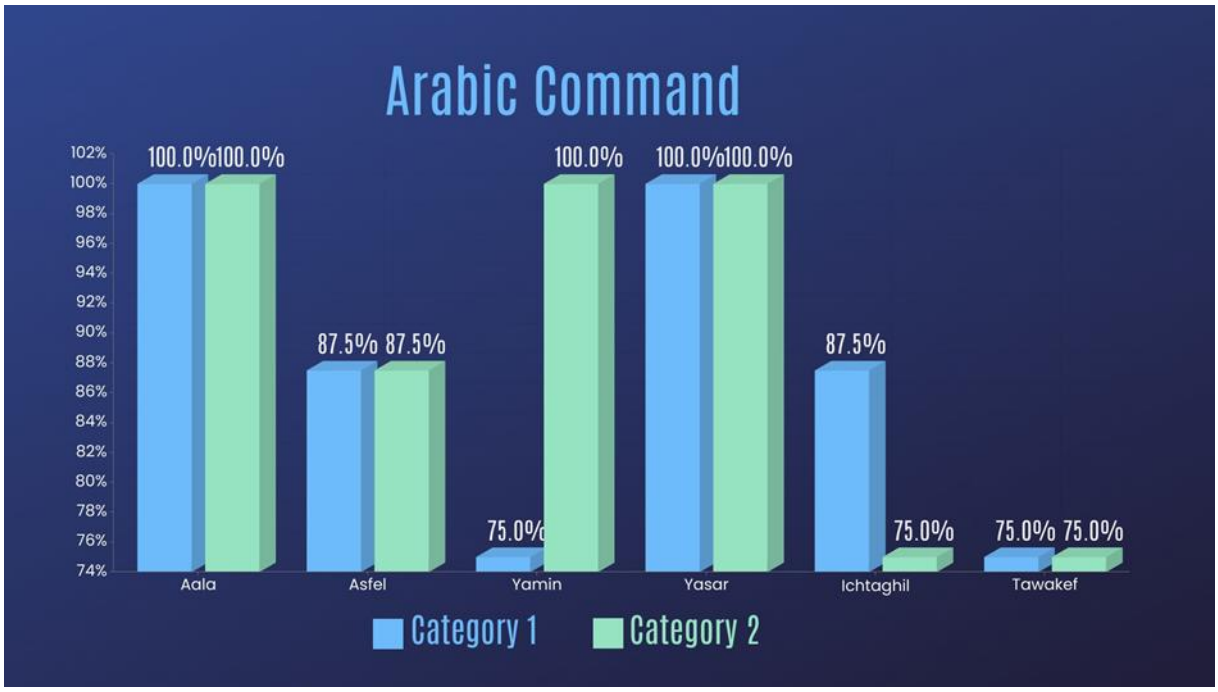


Figure (III-21): *Test of Arabic command*

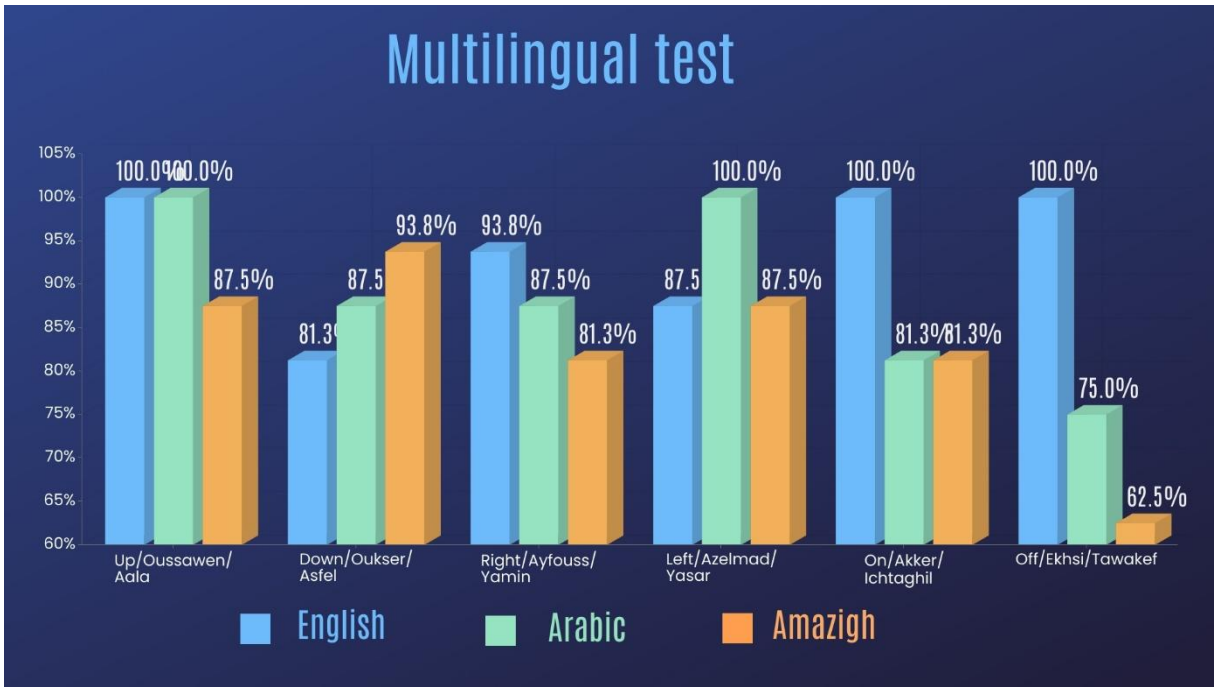


Figure (III-22): *Test of Multilingual dataset*

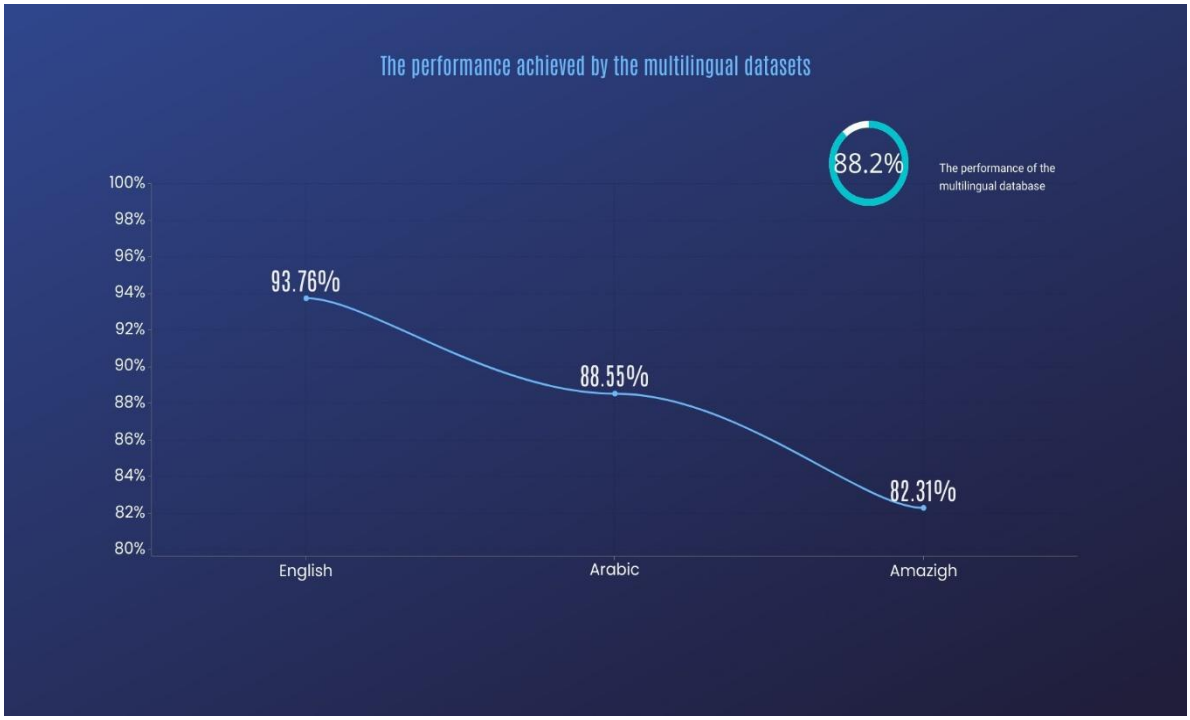


Figure (III-23): *The performance achieved by the Multilingual datasets..*

III.3.2.8.3. Results and discussion

As demonstrated in figures (III-19), (III-20), (III-21), (III-22) and (III-23), we may conclude that :

- ✓ Amazigh database is better compared to English and Arabic for the 1st category. We can say that the Amazigh language is their base language.
- ✓ Arabic database is better for the 2nd category, because they are native speakers. For both categories, Arabic is efficient in terms of percentage compared to the Amazigh database because the commands in Arabic are less complex than in Amzaigh.
- ✓ English database works well with both categories because the commands are short (one sylable) and easy to pronounce.
- ✓ Commands in Amazigh are complex. They contain several sylables, so they have to pronounce the word well in order to obtain good recognition.

- ✓ We can explain the errors made sometimes by category one in Amazigh command by the fact that Amazigh is a language while there are several dialects in this language, which makes the accent different from one person to another.
- ✓ Even in a noisy environement, the multilingual test outperforms the monolingual test.
- ✓ We detect a confusion between left, asfel and ayfouss in the multilingual test.

III.4.Conclusion

The multilingual database implementation produced a reasonably high performance percentage, with 94.46 % validation accuracy for the monolingual dataset and 90.35 % validation accuracy for the multilingual database.

In addition, for the two categories of individuals, we have a performance test of 82.31 %, 88.55 %, and 93.76 % for Amazigh, Arabic, and English, respectively.

For that, we will use this implementation in the next chapter as an application on the UAV.

CHAPTER FOUR

IV.1. Introduction

In this work, as shown in figure (IV-1), it presents a simple scheme of a bidirectional transmission half-duplex type, transporting commands first from the laptop to the drone then interfacing with the real-time parameters through Matlab (or laptop) by receiving motion parameters from the UAV.

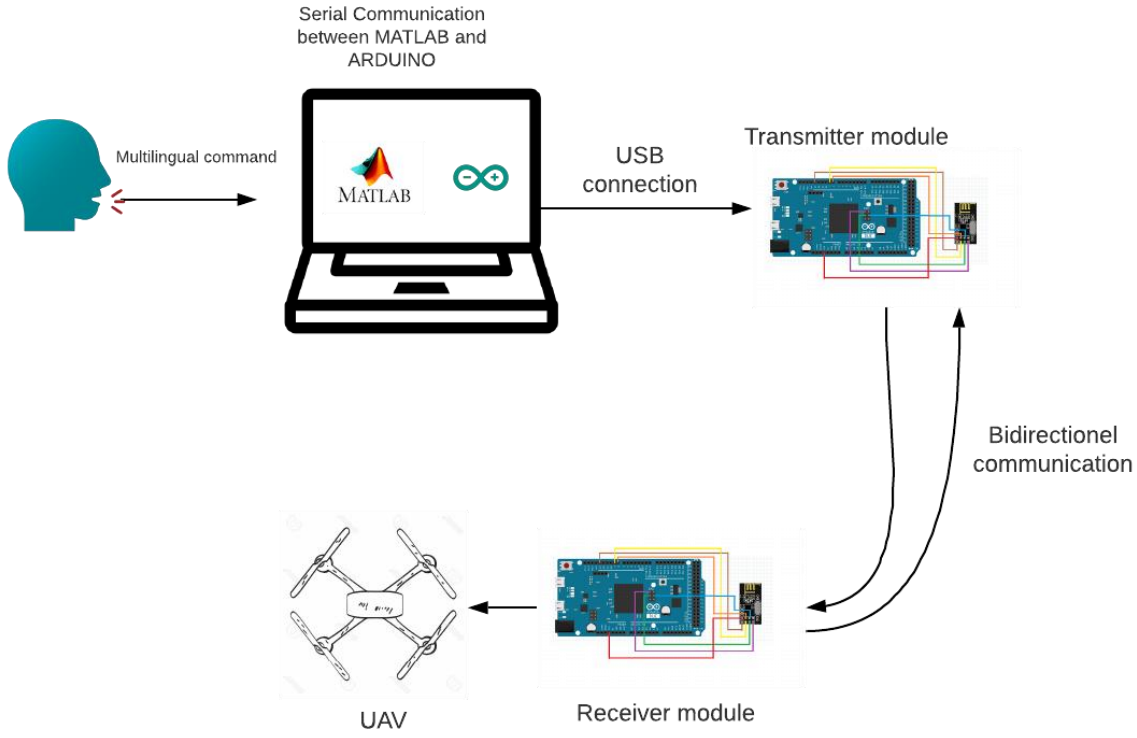


Figure (IV-1): *Multilingual command recognition to control a UAV*

IV.2. Wireless communication

Communication first starts with initializing the microphone by setting its parameters. Then the microphone gives the entering voice as input to Matlab. Matlab tests the recognition as commands voice or unknown words, then send them to Arduino. The flow chart of the communication algorithm is shown in figure (IV-2).

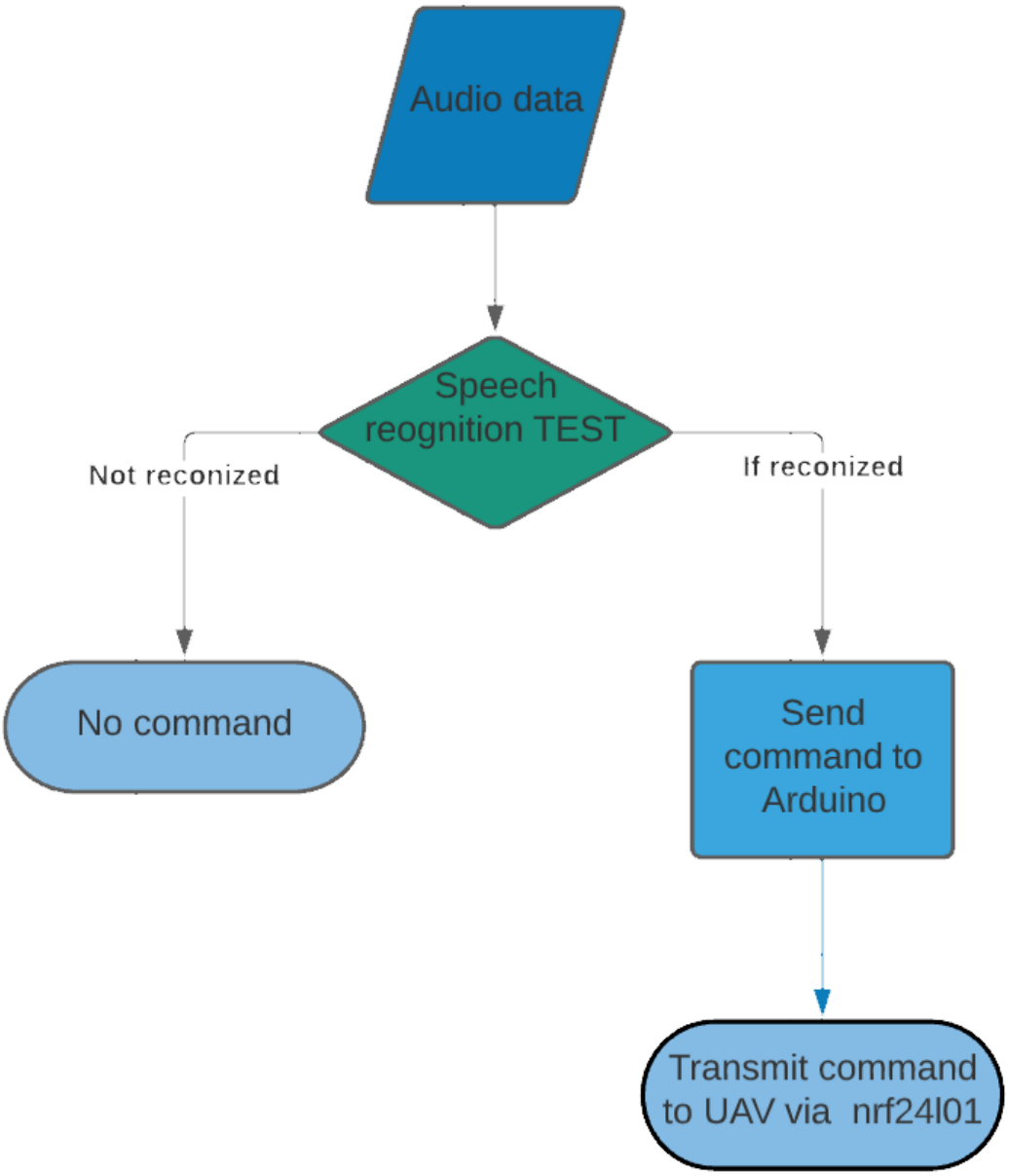


Figure (IV-2): *flowchart of speech recognition algorithm*

The microphone of the laptop will be set its sensitivity to a lower amplification according to the noise making by the surrounding, so it could allow the test to be more accurate.

The figure (IV-3) shows the result of the speech recognition test in Matlab. It demonstrates an analogic signal of the voice input and up above its signification as commands or unknown words.

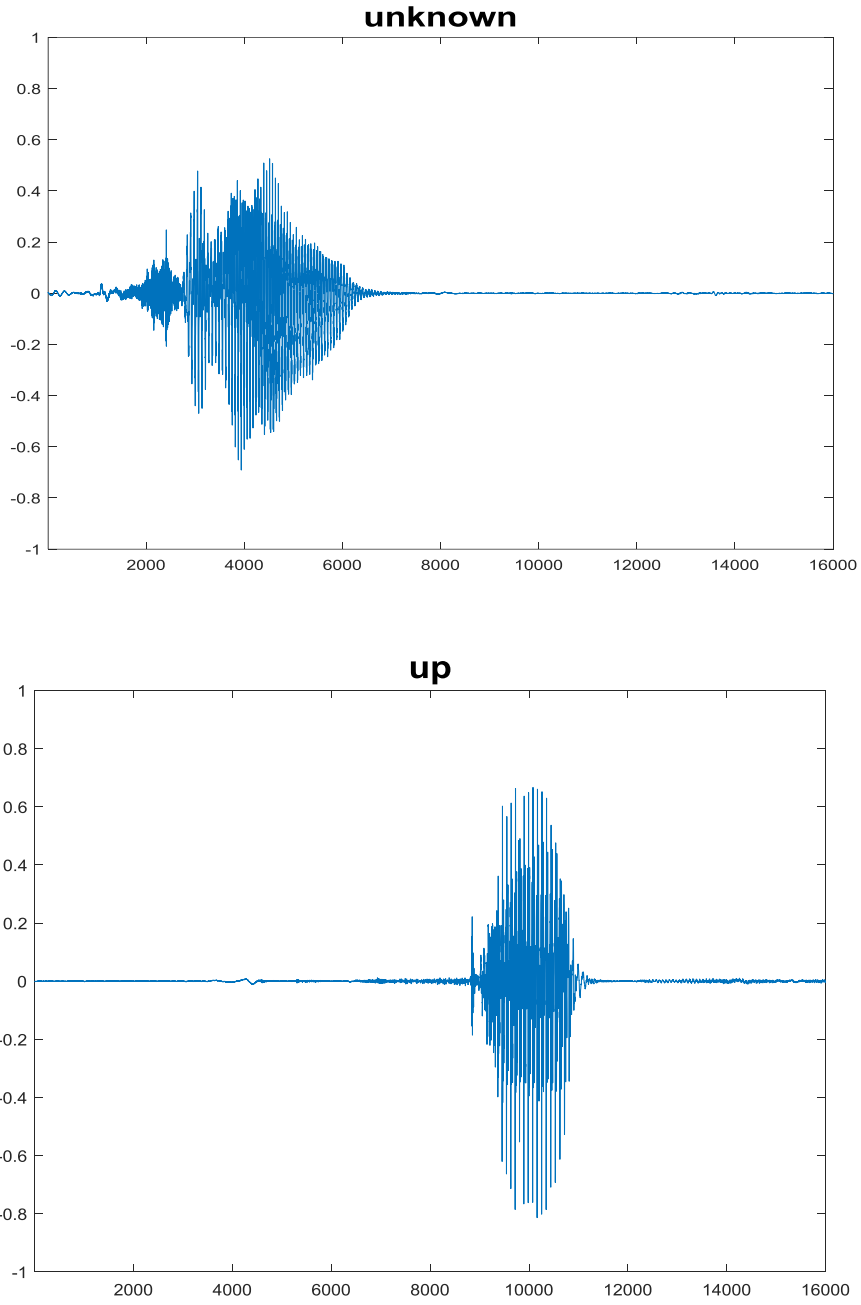


Figure (IV-3): *the result of the speech recognition test*

IV.2.1. UART serial communication

The commands up, down, right, left, on, off are sent to Arduino through serial communication as 1, 2, 3, 4, 5, 6 instructions respectively to facilitate the decoding of data received, because this form of communication stands for the process of sending data one bit

at time, sequentially, through the bus or communication channel as shown in figure (IV-4). Then those instructions are translated into control signals of the UAV in the Arduino software IDE.

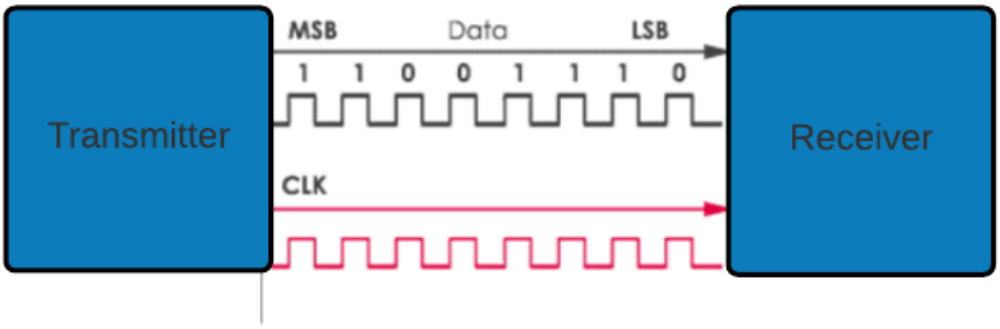


Figure (IV-4): *Serial communication*

A UART is a hardware peripheral that is present inside a microcontroller, performs a serial communication protocol. It is mostly used for short-distance, low-speed, low-cost data exchange between computers and peripherals [36].

The interface of UART is shown in figure (IV-5).

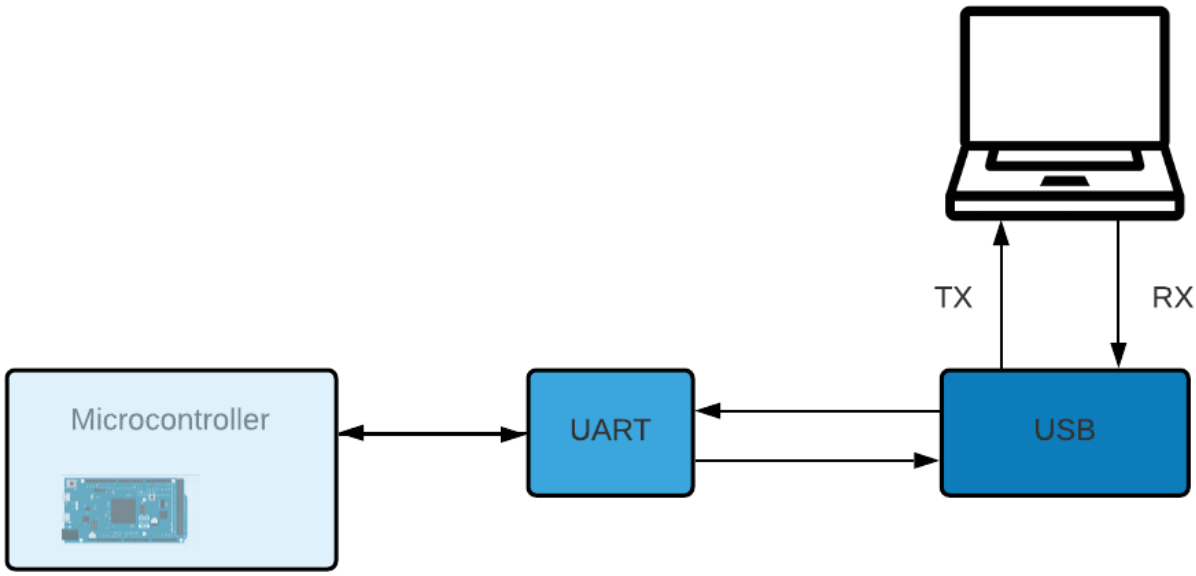


Figure (IV-5): *UART interface*

➢ ***The Arduino Due*** is a microcontroller board based on the Atmel SAM3X8E ARM Cortex-M3 CPU. It is the first Arduino board based on a 32-bit ARM core microcontroller [37]. Table 6 shows its characteristics.

Table (IV-1): *the characteristic of the Arduino DUE*

Microcontroller type	AT91SAM3X8E
Operating voltage	3.3V
Length	101.52 mm
Width	53.3 mm
Weight	36 g
Clock speed	84 MHZ
Size of flash memory	512 KB available to user for application

Simply connect it to a computer with a micro-USB cable (the transmitter one) or power it with a battery (the receiver one) to get started.

The SAM3X provides one hardware UART (pins RX0 and TX0) and three hardware USARTs for TTL (3.3V) serial communication. Serial on pins RX0 and TX0 as shown in figure (IV-6) provides Serial-to-USB communication for programming the board through the ATmega16U2 microcontroller.

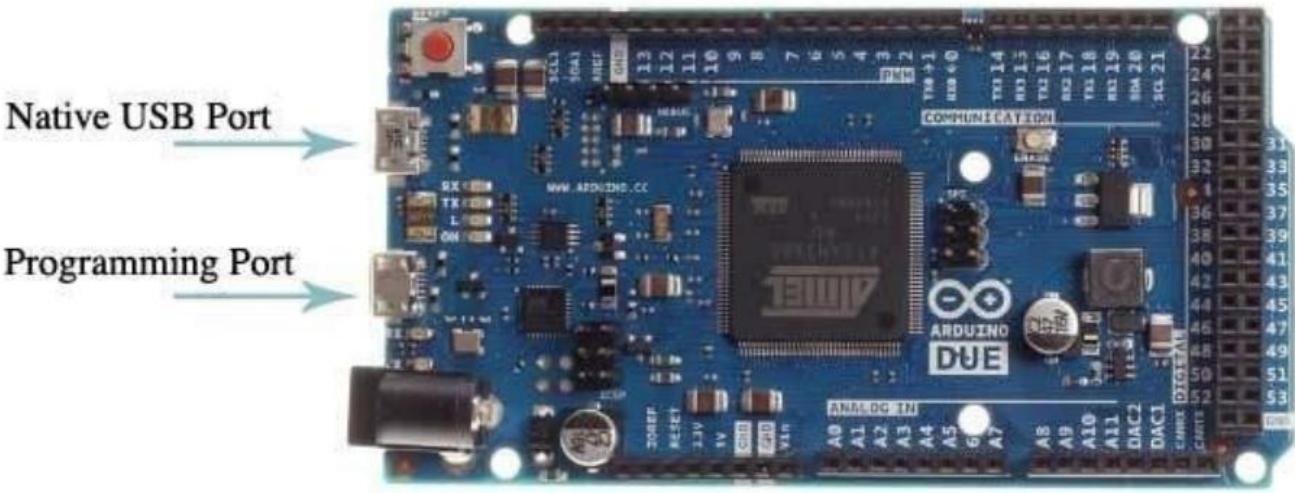


Figure (IV-6): *Arduino DUE board*

Among the common wireless communication modules like Bluetooth and Wi-Fi, thr nRF24L01+ radio modules as shown in figure (IV-7) are cheap and powerful, highly integrated, ultra-low power (ULP) 2Mbps RF transceiver ICs for the 2.4GHz ISM (Industrial, Scientific, and Medical) band.

Figure (IV-7): *NRF24L01+ module*

➢ ***The NRF24L01+ module*** integrates a complete 2.4GHz RF transceiver with a 250 kbps transmission rate. In the open air, it can reach 800 to 1K meters in communication distance, supporting a high-speed ubiquitous SPI for the application controller [38]. It is thus able to operate in conjunction with Arduino DUE without the addition of any external hardware, as long as it supports the SPI protocol. A corollary to this is that it has to be used in conjunction with a microcontroller since it can't work on its own, resulting in a bulky unit; not an ideal fit for a wireless sensor network. Since it works on the 2.4GHz ISM band, it is possible to emulate standard radio protocols working at that bandwidth, such as Bluetooth Low Energy (BLE) or the 2.4GHz IEEE 802.11 Wi-Fi standards using this transceiver with a supporting microcontroller. Its features and specifications are shown in figure (IV-8) and table 2.

➢ ***The NRF24L01+ Pinout***

The NRF24L01 Transceiver module has a total of 8 pins.

Table (IV-2): *The Pinout specification of NRF24L01+*

Pin	Symbol	Description
1	GND	Ground
2	VCC	Supply voltage(1.9V-3.6V)
3	CE	It's an input used to control data transmission and reception in TX and RX modes, respectively.
4	CSN	SPI chip select
5	SCK	Is the serial clock for the SPI bus
6	MOSI	SPI slave data input
7	MISO	SPI slave data output
8	IRQ	Maskable interrupt pin

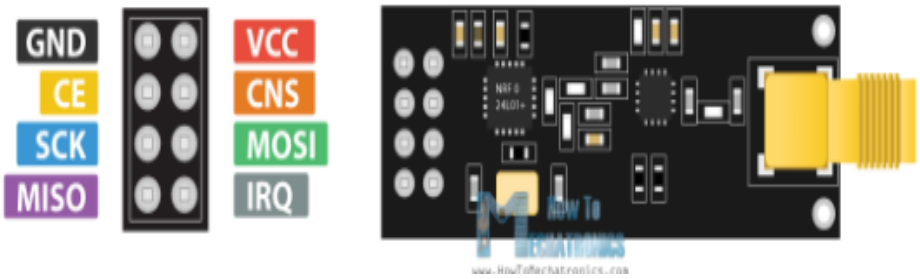


Figure (IV-8): *NRF24L01+ Pinout*

The SPI protocol sends and receives data in a continuous stream without any interruption. This protocol is recommended for high speed data communication is required. The maximum speed it can provide is 10 Mbps.

The nRF operates on the 2.4 GHz ISM band with a channel spacing of 1 MHz, which leads to 125 possible channels from 2.4 GHz to 2.525 GHz.

For all point to point transfer test experiments, we have used only the RF24 library. The library exposes only the bare transmission functionality through member functions of the RF24 class. For point-to-point communication, tests were carried out by sending messages of size 4 bytes through 32 bytes with an increment of 4 bytes. The network is comprised of Arduino with attached nRF modules as illustrated in figure (IV-9).

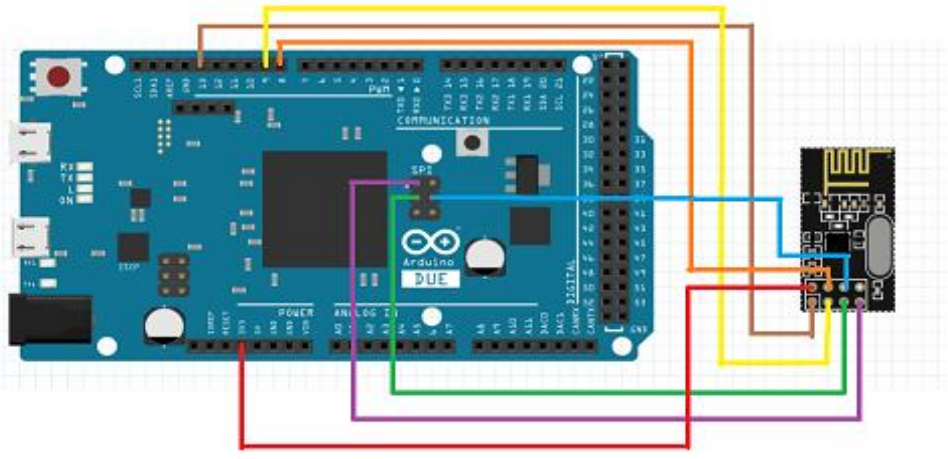


Figure (IV-9): *Arduino & NRF24L01+*

This circuit must be produced in duplicate, in order to obtain a transmitter circuit and a receiver circuit, as illustrated in figure (IV-10).

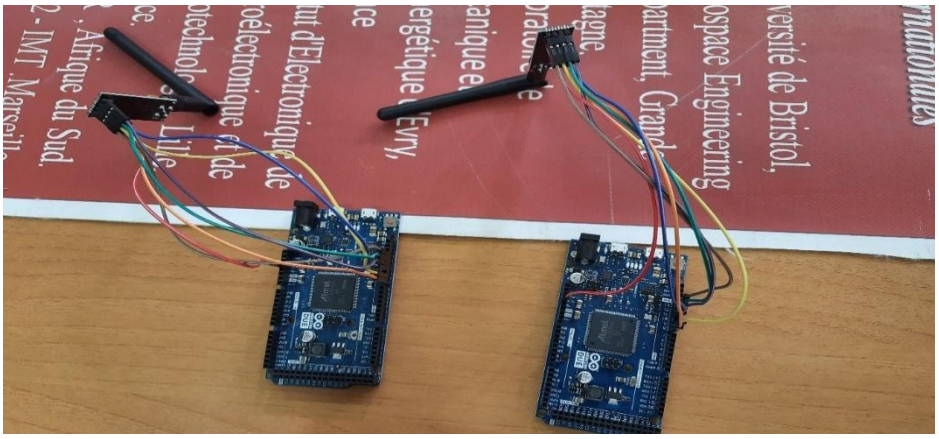


Figure (IV-10): *transmitter & receiver module*

IV.3.Transmitter part

The emission part consists of entering a voice input through a laptop microphone which Matlab will test for recognition, then through serial communication with the Arduino board, the commands will be sent via module nrf24l01+.

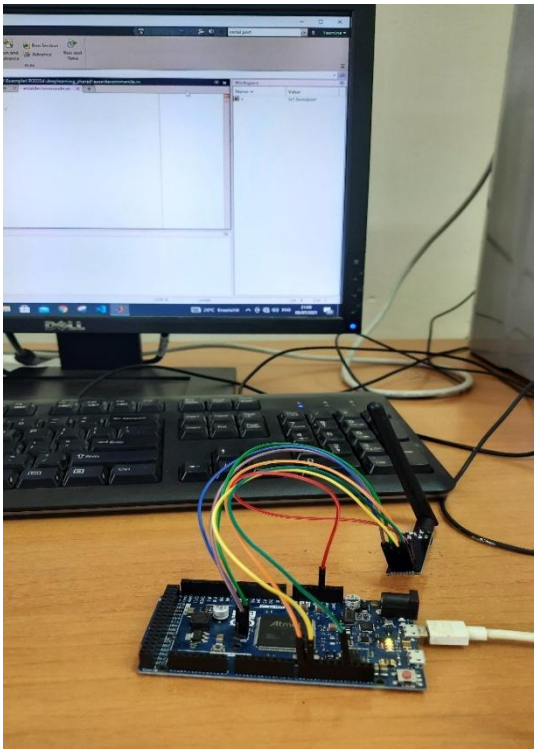


Figure (IV-11): *transmitter side*

IV.4.Reception part

In the reception part, the commands will be received via nrf24l01+ to the Arduino board placed in the UAV as control signals corresponding to the manipulation of the motor speed. Therefore, the control signals will be sent back to the emission part as real-time parameters to interface it in Matlab as the axis motion of the UAV.

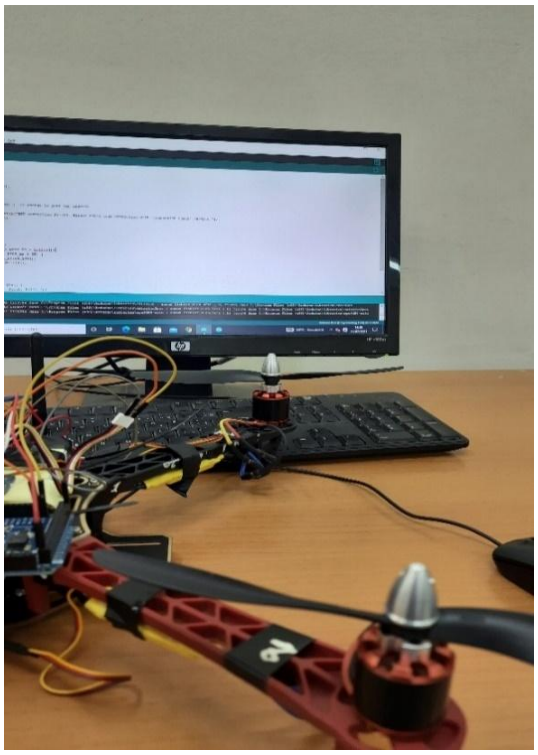


Figure (IV-12): *reception side*

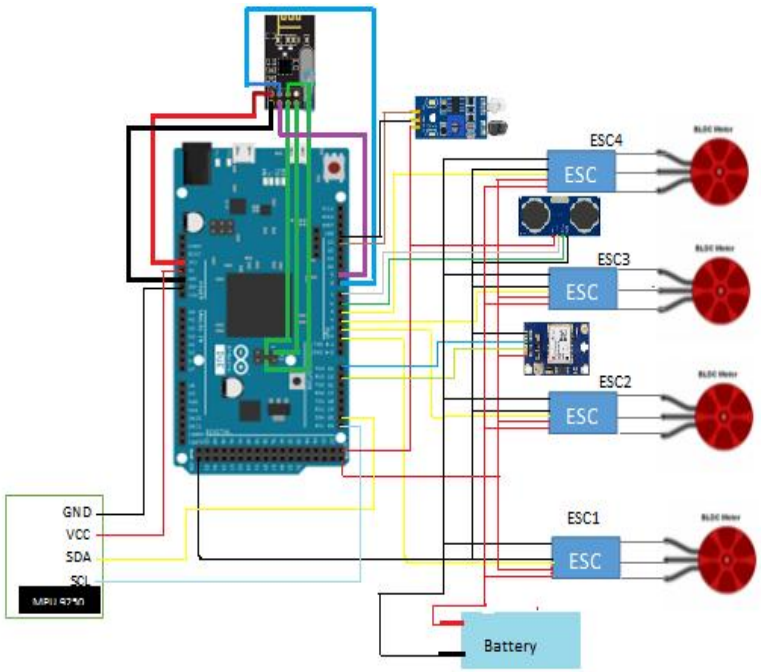


Figure (IV-13): *diagram illustrating the connection of all the components in thereceiving part*

IV.5. Graphical Interface

To present the variable speed of each of the UAV motors interacting with each of the voice commands in a GUI using the Node-RED platform. Therefore, operating with MQTT to exchange data between the Node-RED dashboard and Matlab, subsequently making use of the serial communication function in Node-RED to connect it to the Arduino board. The figure (IV-14) shows the interface for controlling UAVs through voice commands.

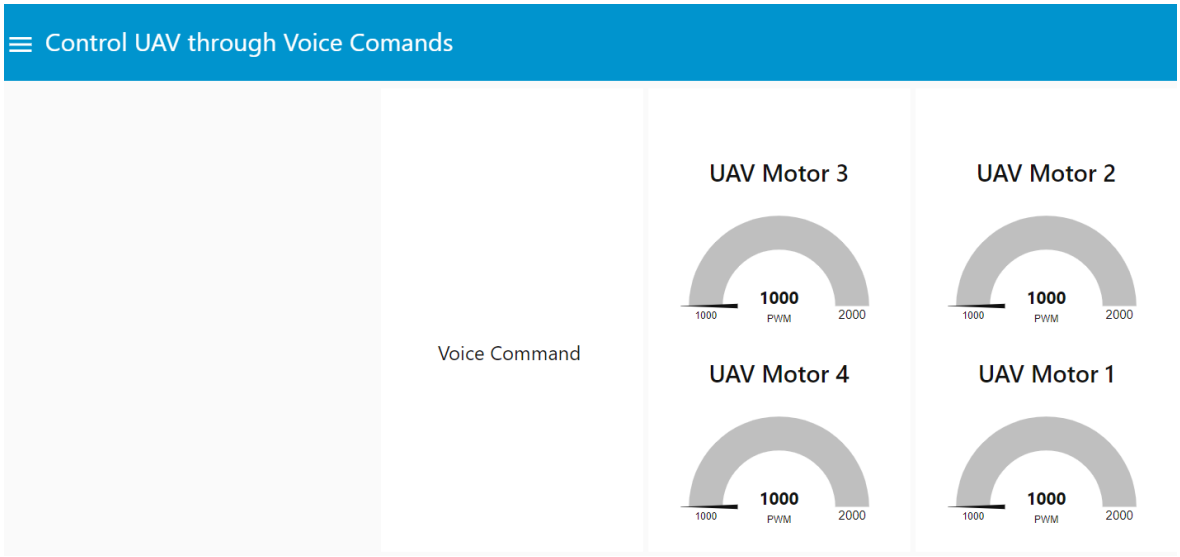
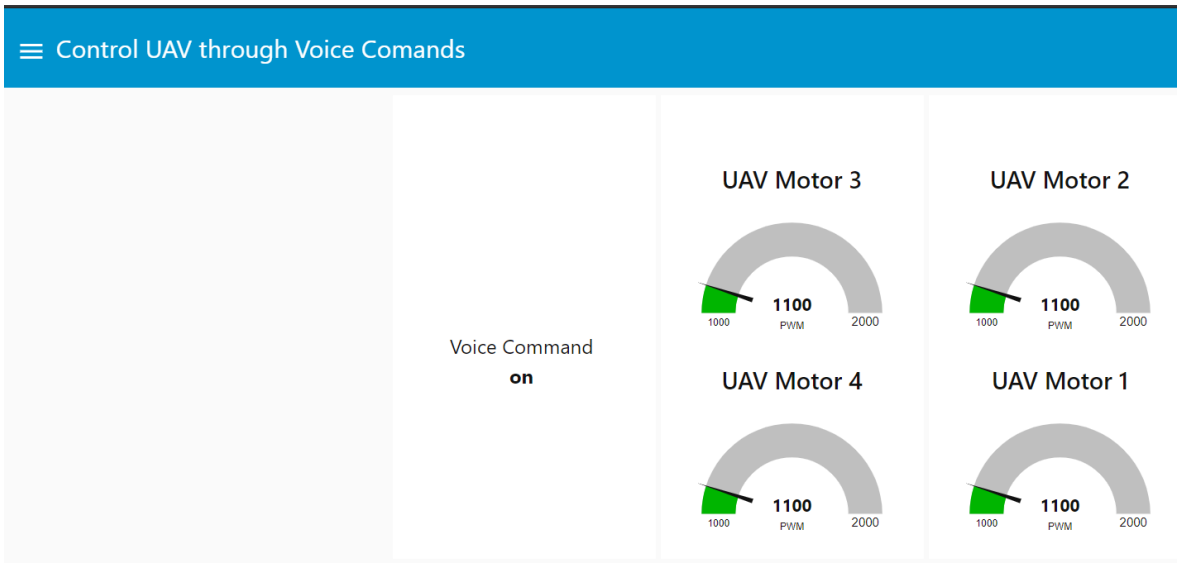


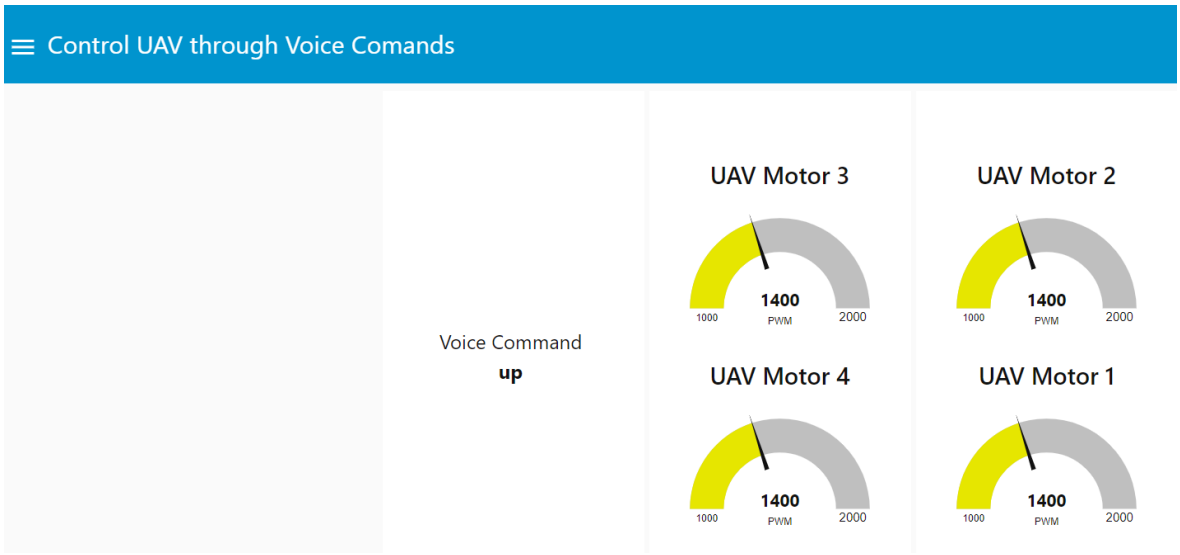
Figure (IV-14): *Interface of UAV controlled through voice commands*

The four gauges express the speed of UAV motors in PWM units. Besides them, a text edit identifies the voice command entered.

The figures (IV-15) interpret the response of the test of multilingual speech recognition of the commands "ON", "UP", "LEFT', "RIGHT", "DOWN" and "OFF" respectively as a reaction of the UAV.

Control via voice commands





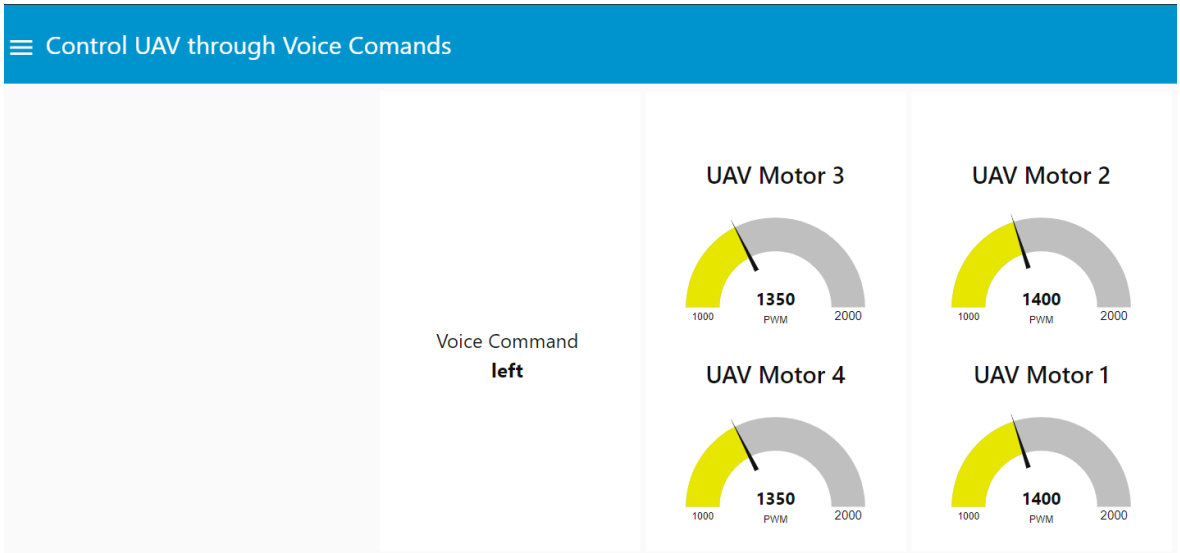


***Figure* (IV-15):** *The test of all the Commands.*

Figure (IV-16): *UAV with servo motors on it*

IV.6. Conclusion

The use of the implementation of a multilingual database as an application on the UAV so as to send command voice in any language, the ones trained for.

The process of communication between reception and transmission needs to step into multiple communication phases, such as the serial communication between Matlab and Arduino, as well as Matlab and the user interface Node-Red platform.

Even though the clock speed of the Arduino is 84 MHz, there's asynchronization between reception and transmission because the command will pass through those phases, so to send it finally to the UAV and receive back the motor speed in realtime interface, to confirm the command was recognized.

GENERAL CONCLUSION

General conclusion

Our work was a response to the question asked in the general introduction, so in this work we were interested in establishing in Arabic and Amazigh a multilingual speech command recognition system in order to use it as an application to control UAV movements.

To achieve this, we first described the interaction between humans and drones to really understand this type of interaction, because to master something you have to understand the details and the basics.

Then we moved on to the design of this recognition system using MATLAB 2020 software. Good simulation results were obtained.

In the context of more research, it would be interesting to expand the database to increase its efficiency. Therefore, we suggest creating a website to record commands by a large number of people in order to cover all existing dialects. A command confirmation system should complement this application.We may even go into a lexical field, or just go for other complex commands for other purposes.

REFERENCES

REFERENCES:

[1] Karjalainen, K., & Romell, A. (2017). *Human-Drone Interaction: Drone as a companion? An explorative study between Sweden and Japan* (Master's thesis).

[2] https://www.digitalistmag.com/digital-economy/2019/11/05/are-drones-changing-way-we-live-06201367/

[3] Tezza, D., & Andujar, M. (2019). The state-of-the-art of human–drone interaction: A survey. *IEEE Access*, *7*, 167438-167454.

[4] Fernandez, R. A. S., Sanchez-Lopez, J. L., Sampedro, C., Bavle, H., Molina, M., & Campoy, P. (2016, June). Natural user interfaces for human-drone multi-modal interaction. In *2016 International Conference on Unmanned Aircraft Systems (ICUAS)* (pp. 1013-1022). IEEE.

[5] Yam-Viramontes, B. A., & Mercado-Ravell, D. (2020, September). Implementation of a Natural User Interface to Command a Drone. In *2020 International Conference on Unmanned Aircraft Systems (ICUAS)* (pp. 1139-1144). IEEE.

[6] Funk, M. (2018). Human-drone interaction: let's get ready for flying user interfaces!. *Interactions*, *25*(3), 78-81.

[7] https://insideunmannedsystems.com/draganfly-selected-to-develop-pandemic-drone/

[8] Yeh, A., Ratsamee, P., Kiyokawa, K., Uranishi, Y., Mashita, T., Takemura, H., ... & Obaid, M. (2017, October). Exploring proxemics for human-drone interaction. In *Proceedings of the 5th international conference on human agent interaction* (pp. 81-88).

[9] Duncan, B. A., & Murphy, R. R. (2013, August). Comfortable approach distance with small unmanned aerial vehicles. In *2013 IEEE RO-MAN* (pp. 786-792). IEEE.

[10] Jensen, W., Hansen, S., & Knoche, H. (2018, April). Knowing you, seeing me: investigating user preferences in Drone-Human acknowledgement. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (pp. 1-12).

[11] Szafir, D., Mutlu, B., & Fong, T. (2014, March). Communication of intent in assistive free flyers. In *Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction* (pp. 358-365).

[12] Szafir, D., Mutlu, B., & Fong, T. (2015, March). Communicating directionality in flying robots. In *2015 10th ACM/IEEE International Conference on Human-Robot Interaction (HRI)* (pp. 19-26). IEEE.

[13] Nozaki, H. (2014). Flying display: a movable display pairing projector and screen in the air. In *CHI'14 Extended Abstracts on Human Factors in Computing Systems* (pp. 909-914).

[14] Schneegass, S., Alt, F., Scheible, J., & Schmidt, A. (2014, June). Midair displays: Concept and first experiences with free-floating pervasive displays. In *Proceedings of The International Symposium on Pervasive Displays* (pp. 27-31).

[15] Schneegass, S., Alt, F., Scheible, J., Schmidt, A., & Su, H. (2014). Midair displays: Exploring the concept of free-floating public displays. In *CHI'14 Extended Abstracts on Human Factors in Computing Systems* (pp. 2035-2040).

[16] Scheible, J., Hoth, A., Saal, J., & Su, H. (2013, June). Displaydrone: a flying robot based interactive display. In *Proceedings of the 2nd ACM International Symposium on Pervasive Displays* (pp. 49-54).

[17] Avila, M., Funk, M., & Henze, N. (2015, October). Dronenavigator: Using drones for navigating visually impaired persons. In *Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility* (pp. 327-328).

[18] Obaid, M., Mubin, O., Basedow, C. A., Ünlüer, A. A., Bergström, M. J., & Fjeld, M. (2015, October). A drone agent to support a clean environment. In *Proceedings of the 3rd International Conference on Human-Agent Interaction* (pp. 55-61).

[19] Scheible, J., & Funk, M. (2016, June). DroneLandArt: landscape as organic pervasive display. In *Proceedings of the 5th ACM International Symposium on Pervasive Displays* (pp. 255-256).

[20] Nitta, K., Higuchi, K., & Rekimoto, J. (2014, March). HoverBall: augmented sports with a flying ball. In *Proceedings of the 5th Augmented Human International Conference* (pp. 1-4).

[21] Knierim, P., Kosch, T., Schwind, V., Funk, M., Kiss, F., Schneegass, S., & Henze, N. (2017, May). Tactile drones-providing immersive tactile feedback in virtual reality through quadcopters. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems* (pp. 433-436).

[22] Orosanu, L. (2015). *Reconnaissance de la parole pour l'aide à la communication pour les sourds et malentendants* (Doctoral dissertation, Université de Lorraine).

[23] https://smartboost.com/blog/deep-learning-vs-neural-network/

[24] Kamath, U., Liu, J., & Whitaker, J. (2019). *Deep learning for NLP and speech recognition* (Vol. 84). Cham: Springer.

[25] Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y., & Alsaadi, F. E. (2017). A survey of deep neural network architectures and their applications. *Neurocomputing*, *234*, 11-26.

[26]https://www.researchgate.net/figure/Schematic-diagram-of-a-basic-convolutional-neural-network-CNN-architecture-26_fig1_336805909

[27] https://srdas.github.io/DLBook/

[28] Varma, S., & Das, S.(2018). Introduction to deep learning.

[29] Kamath, U., Liu, J., & Whitaker, J. (2019). *Deep learning for NLP and speech recognition* (Vol. 84). Cham: Springer.

[30] https://www.mathworks.com/discovery/feature-extraction.html

[31] Warden, P. (2017). Launching the speech commands dataset. *Google Research Blog*.

[32]https://www.matlabexpo.com/content/dam/mathworks/mathworks-dot-com/images/events/matlabexpo/us/2018/master-class-deep-learning-for-signals.pdf

[33]Fayek, H. (2016). Speech processing for machine learning: Filter banks, mel-frequency cepstral coefficients (mfccs) and what's in-between.

[34] Tong, S., Garner, P. N., & Bourlard, H. (2017). An investigation of deep neural networks for multilingual speech recognition training and adaptation. In *Proc. of INTERSPEECH* (No. CONF).

[35] Uebler, U. (2001). Multilingual speech recognition in seven languages. *Speech communication*, *35*(1-2), 53-69.

[36] https://www.codrey.com/embedded-system/uart-serial-communication-rs232/

[37] https://www.arduino.cc/en/pmwiki.php?n=Main/ArduinoBoardDue.

[38] Semiconductor, ―nRF24L01 overview‖ available at http://www.nordicsemi.com/eng/Products/2.4GHz-RF/ nRF24L01 (2007).

[39] Choutri, K., Lagha, M., & Dala, L. (2021). A Fully Autonomous Search and Rescue System Using Quadrotor UAV. International Journal of Computing and Digital systems, 10,2-12.

[40] Choutri, K., Lagha, M., & Dala, L. (2019). Multi-layered optimal navigation system for quadrotor UAV. Aircraft Engineering and Aerospace Technology.