electronics

MDPI

Article

A Multi-Lingual Speech Recognition-Based Framework to Human-Drone Interaction

Kheireddine Choutri [1,*], Mohand Lagha [1,*], Souham Meshoul [2,*], Mohamed Batouche [2], Yasmine Kacel [1] and Nihad Mebarkia [1]

[1] Aeronautical Sciences Laboratory, Aeronautical and Spatial Studies Institute, Blida 1 University, Blida 0900, Algeria; kacelyasmine56@gmail.com (Y.K.); mebarkia.nihad@gmail.com (N.M.)
[2] Department of Information Technology, College of Computer and Information Sciences, Princess Nourah Bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia; mabatouche@pnu.edu.sa
* Correspondence: choutri.kheireddine@univ-blida.dz (K.C.); laghamohand@univ-blida.dz (M.L.); sbmeshoul@pnu.edu.sa (S.M.)

Abstract: In recent years, human–drone interaction has received increasing interest from the scientific community. When interacting with a drone, humans assume a variety of roles, the nature of which are determined by the drone's application and degree of autonomy. Common methods of controlling drone movements include by RF remote control and ground control station. These devices are often difficult to manipulate and may even require some training. An alternative is to use innovative methods called natural user interfaces that allow users to interact with drones in an intuitive manner using speech. However, using only one language of interacting may limit the number of users, especially if different languages are spoken in the same region. Moreover, environmental and propellers noise make speech recognition a complicated task. The goal of this work is to use a multilingual speech recognition system that includes English, Arabic, and Amazigh to control the movement of drones. The reason for selecting these languages is that they are widely spoken in many regions, particularly in the Middle East and North Africa (MENA) zone. To achieve this goal, a two-stage approach is proposed. During the first stage, a deep learning based model for multilingual speech recognition is designed. Then, the developed model is deployed in real settings using a quadrotor UAV. The network was trained using 38,850 records including commands and unknown words mixed with noise to improve robustness. An average class accuracy of more than 93% has been achieved. After that, experiments were conducted involving 16 participants giving voice commands in order to test the efficiency of the designed system. The achieved accuracy is about 93.76% for English recognition and 88.55%, 82.31% for Arabic and Amazigh, respectively. Finally, hardware implementation of the designed system on a quadrotor UAV was made. Real time tests have shown that the approach is very promising as an alternative form of human–drone interaction while offering the benefit of control simplicity.

Keywords: speech recognition; UAV; deep learning; human–drone interaction; natural user interfaces

Citation: Choutri, K.; Lagha, M.; Meshoul, S.; Batouche, M.; Kacel, Y.; Mebarkia, N. A Multi-Lingual Speech Recognition-Based Framework to Human-Drone Interaction. *Electronics* **2022**, *11*, 1829. https://doi.org/10.3390/electronics11121829

Academic Editors: Andrea Prati, Luis Javier García Villalba and Vincent A. Cicirello

Received: 1 May 2022
Accepted: 7 June 2022
Published: 9 June 2022

1. Introduction

Unmanned aerial vehicles (UAVs), or drones, are robots that fly autonomously without a human pilot on board [1]. They were originally used solely for military applications, but are now also increasingly being used for civilian purposes impacting our lives in extremely beneficial ways. Drone technology is steadily becoming more prevalent and its utility has been demonstrated in a number of recent applications in various fields. Drones were used to inspect and monitor compliance with restrictions during the Covid-19 outbreak, as well as to deliver food, medications, and other supplies to isolated places. With the use of drones becoming more common, a disruptive shift in the way we use technology is expected, which is being supported by an intensive research activity. One important aspect in the design, development and integration of such flying engines is how to interact

with them. Developing effective and easy-to-use systems for human–drone interaction (HDI) is a challenging task if we consider the technical, social, and regulatory aspects, and the growing range of potential users and activities for which drones are designed. Traditional remote controllers employ two joysticks for flight control, one for pitch and roll and the other for throttle and yaw. This method requires the constant use of both hands, which might be challenging. The drone's camera is controlled via additional wheels and buttons. Moreover, in tough or stressful situations (such as windy places), the absence of intuitive controls increases the mental burden of the pilot, thereby compromising safety and efficiency. Another difficulty is the difference in position and orientation between the drone and its pilot, which can make it difficult to align the camera feed from the drone with the pilot's surroundings. When the drone is far away or out of direct line of sight, it might be difficult to determine its position and orientation as well as the direction its camera is facing.

The type of interaction depends on the application at hand. Methodologies for human–computer interaction (HCI) and human–terrestrial-robot interaction cannot be used directly for HDI because of the flying complexity inherent in drone technology. A plethora of solutions for HDI have been proposed in the literature and alternative control strategies using gesture and speech were proposed. A review of the state of the art in the context of human–drone interaction can be found in [2]. The authors discuss the main open issues and challenges currently highlighted and reported in research projects and papers. The papers were categorized on the basis of the following dimensions: the drone role, the context of use, indoor vs outdoor usage and the interaction mode. Speech control problems were addressed by many works [3,4]. However, the use of English as the only language for control limits the number of users. Furthermore, improving speech recognition in noisy environment (particularly propeller noise) remains a major challenge.

This paper presents an experimental framework for UAVs control using an automatic multi-lingual speech recognition (AMLSR) system based on a deep learning model. In the Middle-East and North Africa (MENA) zone, people speak different languages, basically Arabic and Amazigh. The idea was to use voice commands to interact with a drone using words spoken in English, standard Arabic and Amazigh. To this end, a convolutional Neural Network has been designed and built using the Google speech commands dataset for English and customized datasets for Arabic and Amazigh. Then, 18 possible commands interpreted as instructions for the UAV were collected. The commands are considered as six classes of actions that the UAV can execute from three different languages.

The proposed method contributes to the state of the art by enhancing the English-based speech recognition system. Multilingual recognition is more advantageous because it provides users with additional options and is not limited to a single category. Additionally, the designed method improves the multilingual recognition for all users, regardless of their native language. Additionally, environmental and propeller noise robustness is addressed. In all languages, instruction utterance recognition is unaffected by signal noise. Consequently, high rates of recognition accuracy were attained during tests. Experiments were conducted using a hardware implementation in real time. To this end, a graphical user interface was created to facilitate human–drone interaction. Due to the big training dataset, English language commands were better understood during testing. Arabic and Amazigh were also detected with a high success rate. In general, the developed framework accommodates users from the MENA region fairly well, regardless of their native language and environmental noise. Based on the state of the art in relevant literature, this study introduces several contributions:

- Faster and more efficient voice control recognition. Other interfaces, such as gesture control, delay the system, thereby limiting its utility.
- Multilingual ASR system (English, Arabic, and Amazigh) that enables a broad spectrum of users to interact with the UAV with simplicity.
- Voice recognition with high accuracy of over 93% using deep learning.
- Quadrotor UAV hardware implementation and real-time testing of the designed system.

- Graphical user interface to reduce the user's workload, simplifying the command and interaction with the UAV, and decreasing the impact of background noise on speech recognition.

The paper is organized as follows: First, a review of related work is provided in Section 2. After that, Section 3 describes the proposed framework that suggests two stages, namely the design of a CNN model for AMLSR and the deployment of the model in a physical environment. Experimental results and real time tests and discussions can be found in Section 4. Section 5 presents the hardware implementation and the graphical user interface. Comparative study and discussion are described in Section 6. Finally, a summary of main results and future work are given in Section 7.

2. Related Works

Human–drone interaction (HDI) is an active and growing field of study that focuses on the evaluation and understanding of interaction distance as well as the development of novel use cases. The methodology and best practices for conducting user research in the HDI field are presented in [5]. This research proposes a taxonomy for HDI user studies that analyzes the various approaches found in the literature about human–drone interaction. Existing human–robot and human–computer interaction approaches must be adapted for drone research, as the complexity of flight introduces additional elements. To this end, a road map to further examine autonomous drones and their integration in human areas, as well as to investigate future interaction strategies, with the goal of establishing HDI best practices is presented in [6]. In [7] the authors examine the relationship between autonomy and User Experience (UX) at various perceived workload levels. This work aims to consider both technical and UX-related factors when developing the next generation of assistive flying robots.

Nowadays, it is common to see more people with no prior knowledge of the subject owning a drone, either to accomplish a specific purpose or for entertainment purposes. As drone technologies became more pervasive and affordable, researchers began to shift interface design towards modern user interfaces that no longer restrict drone control to a remote controller or a ground control station. These innovative techniques, also known as natural user interfaces (NUI), enable users to interact with drones via gesture [8], speech [9], touch [10], and even using brain–computer interfaces (BCIs) [11]. These methods have been evaluated in [12]. Most results appear to imply that the implementation of a NUI facilitates human drone interaction. Most findings suggest that implementing a NUI facilitates human–drone interaction. In addition, as the roles and operations of military UAVs have expanded, along with the need for UAV group control, the traditional "mouse–keyboard" single-mode NUI technology has proven incapable of meeting the requirements of future unmanned warfare. The authors of [13] created a new multi-mode UAV interactive system based on cutting-edge virtual reality and artificial intelligence hardware and software.

According to studies on natural user interfaces [12], speech is used as a method of interaction by 38% of American users and 58% of Chinese users. The Natural Language Processing (NLP) task of real-time computational transcription of spoken language is known as automatic speech recognition (ASR). The authors of [14] provide an overview of the various techniques and approaches used to perform the task of speech recognition. The research provides a thorough comparison of cutting-edge techniques currently being used in this field, with a particular emphasis on the various deep learning methods. In [15], the authors present a statistical analysis of the use of deep learning in speech-related applications, whereas authors of [16] present two cases of successful speech recognition implementations based on Deep Neural Network (DNN) models. The first is a DNN model created by Apple for its personal assistant Siri, and the second is region-based convolutional recurrent neural network (R-CRNN) designed by Amazon for rare sound detection in home speakers. The work described in [17] provides a conceptual understanding of CNN, as well as its three most common architectures and learning algorithms. DNN has shown great promise in speech recognition systems with multiple languages. In [18], the authors propose

a deep learning speech recognition algorithm that combines speech features and speech attributes in the context of English speech. In [19], a combination of deep belief network (DBN) and Deep Bidirectional Long Short-Term Memory (DBLSTM) with Connectionist Temporal Classification (CTC) output layer to create an acoustic model on the Farsdat Persian speech data set is proposed. A contribution to the Amazigh language is also provided in [20]. The paper investigated and implemented an automatic speech recognition system in an Amazigh-Tarift-based environment. In [21], the authors investigate the use of cutting-edge end-to-end deep learning approaches to build a robust diacritised Arabic ASR for the Arabic language. These approaches rely on the Mel-frequency Cepstral Coefficients and the log Mel-Scale Filter Bank energies as acoustic features.

Speech control is widely used in a variety of applications. The work in [22]. details the research and application of human–computer interaction technology based on voice control in UAV ground control stations. Moreover, authors of [23] propose a system for locating and rescuing victims buried beneath debris. A speaker installed on the UAV causes victims to react, and their voices are recorded to detect them. In [24], the authors investigate a speech-based natural language interface for defining UAV trajectories. To determine the effectiveness of this interface, a user study comparing its performance to that of a conventional mouse-based interface is also presented. However, previous works were able to control UAVs via speech, but voice recognition accuracy was insufficient. In [9], the authors designed a speech control scheme for UAVs based on Hidden Markov Model (HMM) and Recurrent Neural Networks (RNN) to address this issue. HMM is utilized to identify error commands and RNN is utilized to train the sets of UAVs commands, with the subsequent command predicted based on the training result. The recognition rate of incorrect commands is as high as 61.90%, while the overall error rate is reduced to 1.43%. A further enhancement can be found in [25] with Speech Recognition Engine based on Convolutional Neural Network (CNN), a Voice Command Controller for Fixed Wing UAV. According to the classification report, the model achieved a quantitative evaluation with an average of 87% for precision and 83% for recall. An alternate approach to speech recognition for robotics applications based on a combination of spectrograms, MEL and MFCC features, and a deep neural network-based classification is presented in [26]. The algorithm's overall validation accuracy is as high as 97%, whereas the testing accuracy of the system is 95.4%. Since this is a classification algorithm, results have been presented using custom datasets for voice classification. In [27], the authors presented a multi-modal evaluation dataset for UAV control, comprised of spoken commands and associated images that represent the visual context of what the UAV sees when the pilot utters the command. For previous works, even with high-rate accuracy, robustness over noise was not considered. In [28], the authors conducted a discriminant analysis of voice commands in the presence of an unmanned aerial vehicle with four rotating propellers, in addition to measuring background sound levels and speech intelligibility. For male speakers, classification of speech commands based on mel-frequency spectral coefficients showed a promising classification rate of 76.2%. Deep Xi, a deep learning approach to a priori SNR estimation proposed in [29], is capable of producing speech enhancements of higher quality and intelligibility than recent deep learning speech enhancement approaches. The method was evaluated using both real-world nonstationary and colored noise sources, at multiple SNR levels. The problem with previous publications is that voice recognition is performed on board. To receive verbal commands, the drone must be close to the user (no more than two to three meters away). Users are restricted to collocated interaction.

3. Proposed Framework

To achieve human–drone interaction using natural language, a two-stage framework is proposed. Overall, as can be seen on the block diagram shown in Figure 1 below, the first stage aims to develop a system that can recognize commands spoken in three different languages namely Arabic, English, and Amazigh using a deep learning model. During this stage, various data are first collected from many sources, then they are preprocessed during

a data preparation phase. Then a deep learning model is designed, built, and fine-tuned using training and validation sets. The performance of the developed deep learning model for automatic Multi-Lingual Speech Recognition (AMLSR) is evaluated using a test set. In the second stage the developed (AMLSR) system is deployed to control the movements of a quadrotor UAV. During this stage, input audio commands are acquired, preprocessed, and fed to the AMLSR system. If the command is recognized, the corresponding instruction is sent directly to the quadrotor UAV; otherwise, the command is rejected.

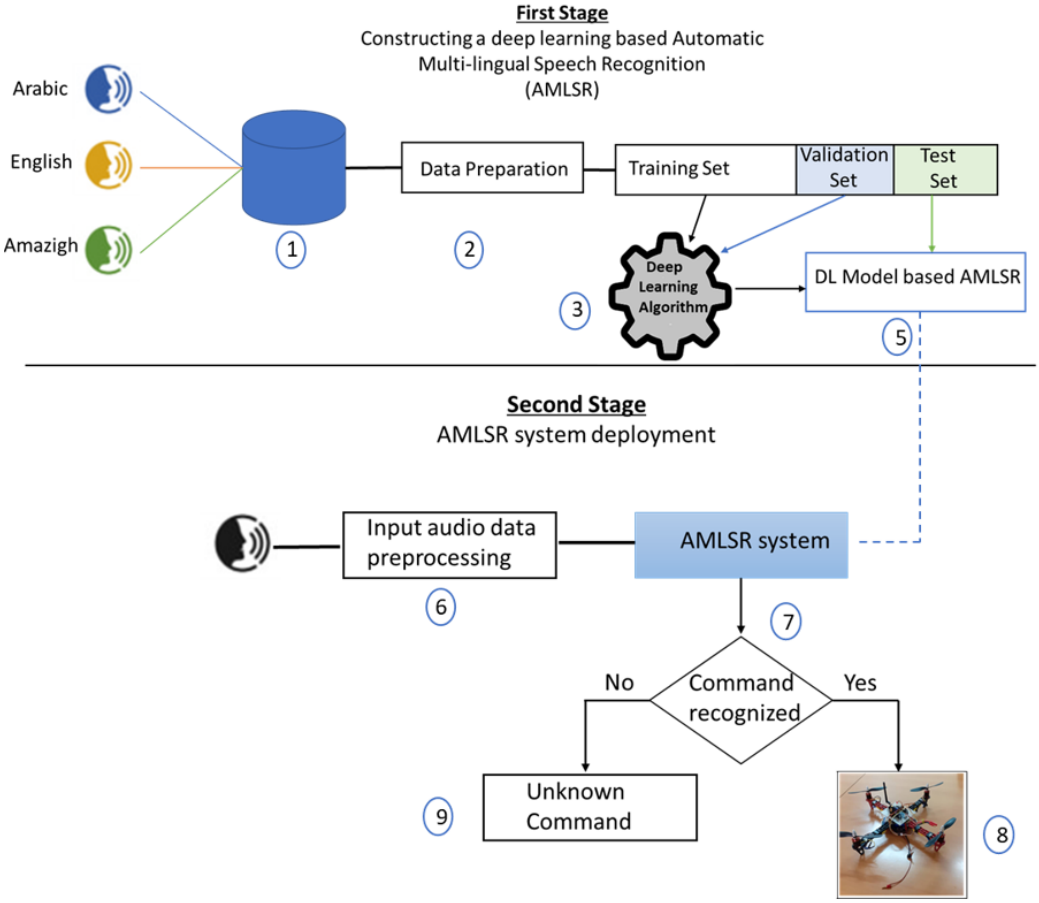


Figure 1. Block diagram of the proposed two-stage framework.

3.1. First Stage: AMLSR System Design

3.1.1. Data Collection and Preparation

During this first step, data records are first collected from the Google Speech Commands Dataset for English language, and from our personnel records for Arabic and Amazigh languages. The English dataset contains 65,000 one-second clips for 30 different words spoken by thousands of different subjects including the desired commands like up and stop, as well as other objects like numbers and names categorized as unknown words. The other language clips were recorded with phones and laptops in realistic environments. Each record has a sampling frequency of 16 KHz, in (.wav) format, including 12 commands (six for each) and unknown words. Table 1 summarizes the record characteristics for our personnel database.

Table 1. Clips characteristics in Arabic and Amazigh databases

Parameters	Arabic Database	Amazigh Database
Sampling Frequency	16 KHz	16 KHz
Audio Format	.wav	.wav
Speakers	12 (5 M + 7 F)	12 (5 M + 7 F)
Number of commands	6	6
Number of unknown words	20	20
Size of commands	5220	960

Before extracting features, it is important to ensure that the network is robust enough to deal with background noise. Background noise has been mixed into the instruction records to achieve this. A variety of types from various sources are employed, and the background samples in the various data sets are highly correlated. Figure 2 shows an example of the signal to noise ratio (SNR) of one of the commands (Akker (On-Am)) correlated with white noise. The obtained SNR level is −8.61 dB, showing the noisy conditions in which the experiments are done.

In this work, six possible commands with three possible languages under various conditions and spoken by different speakers are considered. Hence, there is more than one command for a specific action, as summarized in table Table 2. For example the action "up" can be carried out by saying the word "Aala" in Arabic or "Oussawen" in Amazigh. The collected data undergo a preparation phase where audio records represented as speech waveforms are transformed into spectrograms before being used in the training, validation, and testing of the machine learning model as shown on Figure 3.

Table 2. Commands in English, Arabic and Amazigh.

English Commands	Arabic Commands	Amazigh Commands	Actions
Up	Aala (Up-Ar)	Oussawen (Up-Am)	Increase altitude
Down	Asfal (Down-Ar)	Oukser (Down-Am)	Decrease altitude
Right	Yamine (Right-Ar)	Ayfouss (Right-Am)	Move to the right
Left	Yassar (Left-Ar)	Azelmad (Left-Am)	Move to the left
On	Ichtaghil (On-Ar)	Akker (On-Am)	Turn motors on
Off	Tawakef (Off-Ar)	Ekhsi (Off-Am)	Turn motors off

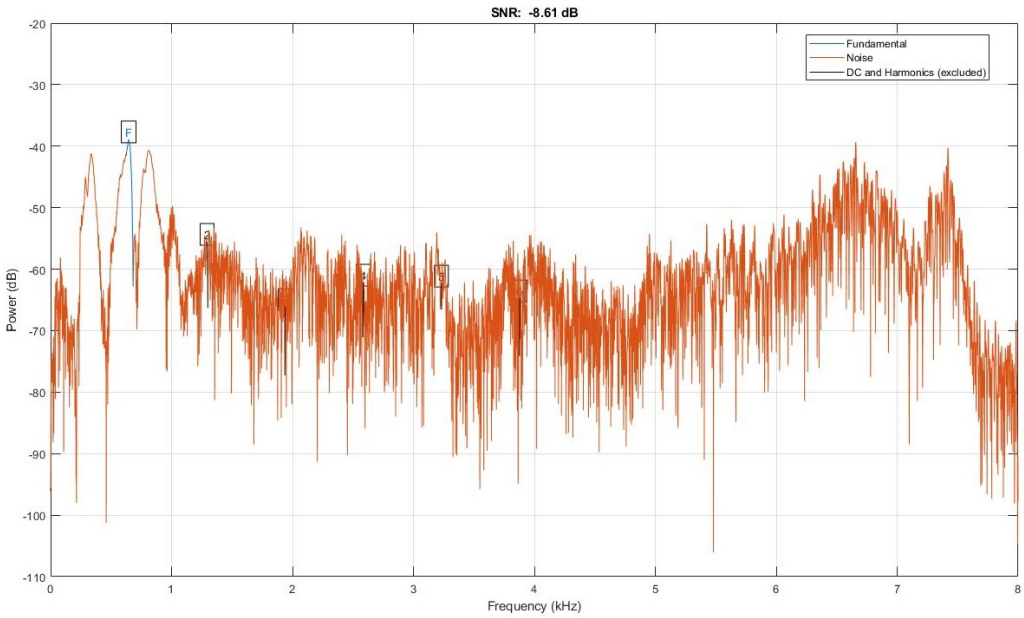


Figure 2. Signal to noise ratio (SNR) for Akker (On-Am).

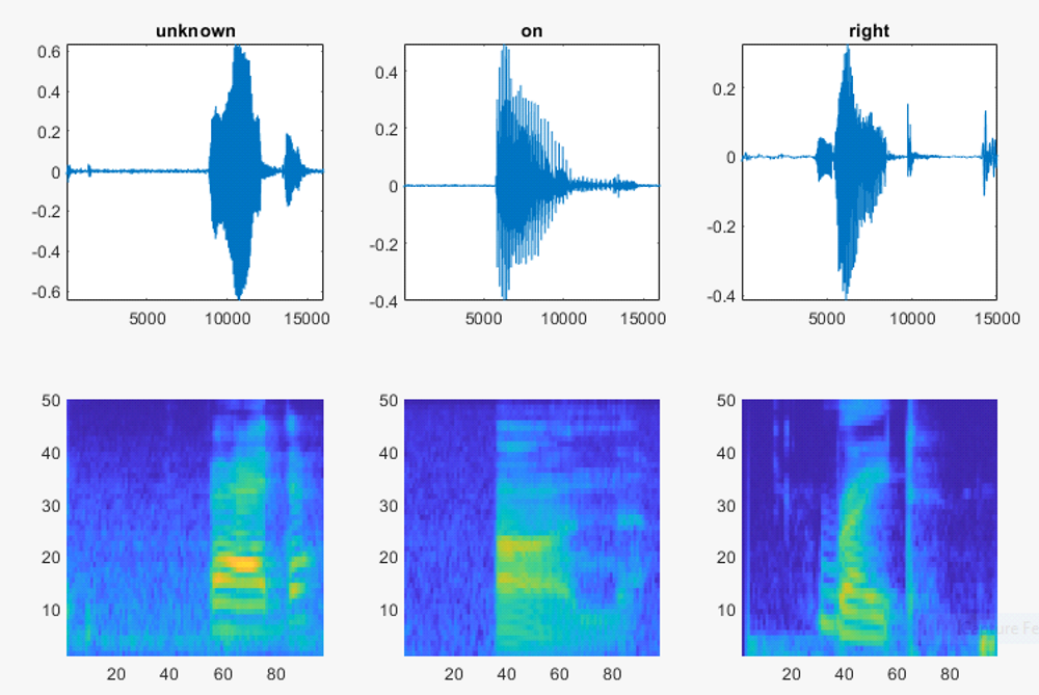


Figure 3. From audio signals to spectrograms.

3.1.2. Deep Learning Model for AMLSR

The command recognition problem is cast as a multi-class classification task. Various architectures have been designed and implemented in convolutional neural networks (CNNs) to handle this task. Ref. [30] presented LeNet-5 as the first development of CNNs that demonstrates impressive results for solving handwritten digit recognition problem. AlexNet [31], is another CNN architecture similar to LeNet-5, but with a more complex network and more parameters to learn. Simonyan and Zisserman in [32] have designed the VGG-16 network that is well-known for its consistent design and which has been successful in a variety of fields. Finally, GoogleNet is a model proposed by [33]. The main goal was to create a model with a lower budget that could reduce the amount of power needed, the number of trainable parameters employed, and the amount of memory used. The number of trainable parameters in the network was drastically reduced by the model.

As shown on Figure 4, the process of developing AMLSR is divided into two stages: training and testing. First, the database that includes UAVs instructions is collected from different users records that speak the desired languages. The dataset is then separated into test, validation and training. The neural network is trained using the extracted features of training data to set the model parameters. Finally, the obtained model is evaluated using the test data.

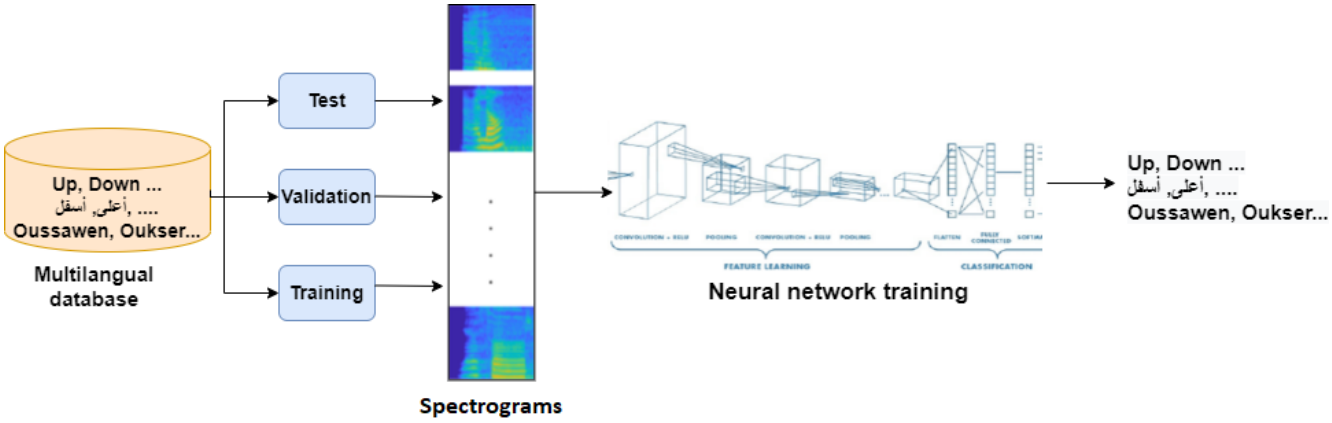


Figure 4. Deep Learning Model for AMLSR.

As illustrated in Figure 5, the architecture of the developed CNN includes:

- 5 convolution layers: each layer acts as a feature extractor.
- 5 By normalizing the outputs of intermediary layers during training, batch normalization attempts to reduce the internal covariate shift in a network. This accelerates the training process and enables faster learning rates without increasing the risk of divergence. Batch normalization does this by normalizing the output of the preceding hidden layer using the mean and variance of the batch (mini-batch). For an input mini-batch $\beta = x_{1:m}$, we learn parameters γ and β via [34]:

$$\begin{aligned} \mu_\beta &= \frac{1}{m}\sum_{i=1}^{m} x_i \\ \sigma_\beta^2 &= \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_\beta)^2 \\ \widehat{x}_i &= \frac{x_i - \mu_\beta}{\sqrt{\sigma_\beta^2 + \epsilon}} \\ y_i &= \gamma \widehat{x}_i + \beta \end{aligned} \tag{1}$$

- 5 ReLu Layer: The rectified linear unit (ReLU) is a simple, fast activation function typically found in computer vision. The function is a linear threshold, defined as:

$$f(x) = max(0, x) \tag{2}$$

- 4 max pooling layers: As the name suggests, the max pooling operation chooses the maximum value of neurons from its inputs and thus contributes to the invariance property. Formally, for a 2d output from the detection stage, a max pooling layer performs the transformation in order to downsample the feature maps (in time and frequency) [34].

$$h_{i,j}^{l} = \max_{p,q} h_{i+p,j+q}^{l-1} \tag{3}$$

where the function h is generally known as a kernel or filter transformation, p and q denote the coordinates of the neuron in its local neighborhood and l represents the layer. In $k - max$ pooling, k values are returned instead of a single value in the max pooling operation.

- Dropout Layers: Applying dropout to a network involves applying a random mask sampled from a Bernoulli distribution with a probability of P. This mask matrix is applied elementwise (multiplication by 0) during the feed-forward operation. During the backpropagation step, the gradients for each parameter and the parameters that were masked in the gradient are set to 0 and other gradients are scaled up by $\frac{1}{1-P}$.

- Fully connected Layer: allows us to perform classification on the dataset.
- Softmax Layer: we find it just after the fully connected layer in order to predict classes. The output unit activation function is the softmax function [35]:

$$y_r(x) = \frac{exp(a_r(x))}{\sum_{j=1}^{k} exp(a_j(x))} \tag{4}$$

where $0 \leq y_r \leq 1$ and $\sum_{j=1}^{k} y_j = 1$

- Weighted Classification Layer: For typical classification networks, the classification layer usually follows a softmax layer. In the classification layer, trainNetwork takes the values from the softmax function and assigns each input to one of the K mutually exclusive classes using the cross entropy function for a 1-of-K coding scheme [35]:

$$loss = -\frac{1}{N} \sum_{n=1}^{N} \sum_{i=1}^{K} w_i t_{ni} ln(y_{ni}) \tag{5}$$

where N is the number of samples, K is the number of classes, w_i is the weight for class i, t_{ni} is the indicator that the nth sample belongs to the ith class, and y_{ni} is the output for sample n for class i, which in this case, is the value from the softmax function. In other words, y_{ni} is the probability that the network associates the nth input with class i.

- numC: controls the number of channels in the convolutional layers. It's worth mentioning at this stage that the network's accuracy is linked to its depth. Increasing the number of filters (numC) or adding identical blocks of convolutional, batch normalization, and ReLu layers may improve overall performance. Once the network is created, its training can be started.

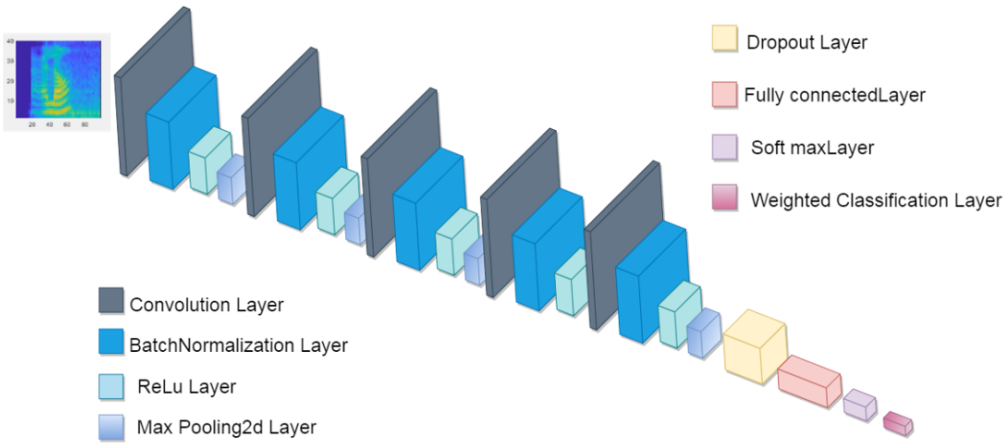


Figure 5. Deep neural networks architecture.

3.2. Second Stage: AMLSR System Deployment

The aim of this stage is to conduct real time tests of the designed system using hardware implementation. First, a graphical user interface is designed to simplify the interaction between the users and the UAV. The interface is designed with the multi-lingual recognition system to detect and recognize user instructions. Next, the instructions are send to the UAV by mean of a communication link to be executed. The transmission/reception system allows the users to control the drone for a range of 1Km. All experiments were made under a quadrotor UAV equipped with all the necessary sensors as shown in Figure 6.

Figure 6. Quadrotors used during test.

4. Implementation Setup, Results and Discussion

This section includes two main parts. In the first part, the implementation details and results of the AMLSR system's performance evaluation will be presented and discussed. Then, the second part will be devoted to the hardware implementation for the deployment of the system.

4.1. AMLSR System Testing Results

In order to build the CNN, the data set has been split into:

- Training set: 80% of the data set was used to fit the model by identifying the set of weights and biases that, on average, cause the least loss across all input examples. The most difficult step in building any machine learning model, particularly a deep learning model, is training the model, which necessitates massive data sets and computational power. It is, indeed, an optimization task as it seeks to minimize the loss associated with incorrect predictions.
- Validation set: 10% of the data examples was used to fine tune the model's hyperparameters and provide an unbiased evaluation of the model fit on the training set.
- Test set: 10% of the data examples was used to evaluate the model performance.

The material environment in which the work was done is characterized by the following features: Windows 10 Professional operating system, Intel Core i7-ES1650 3.50 GHz processor, 64 GB DDR4 RAM, and NVIDIA GeForce GTX 1050Ti GPU.

Figure 7 represents the training with the English, Arabic, and Amazigh databases. For the training phase, the Adam optimizer is used with a mini batch size of 128. Trained for 30 epochs and with learning rate reduced by a factor of 10 after 20 epochs. The most well-known and discussed metrics in deep learning are accuracy and loss. The goal of the training process is to reduce the loss. In the training process, loss is frequently used to find the best parameter values for the model. Accuracy is a metric used to assess the performance of a classification model. It is most commonly expressed as a percentage. It is less difficult to interpret than loss. Most of the time, we would see an increase in accuracy as the loss decreases (as shown in Figure 7), but accuracy and loss have different definitions and measure different things. They frequently appear to be inversely proportional, but no mathematical relationship exists between them.

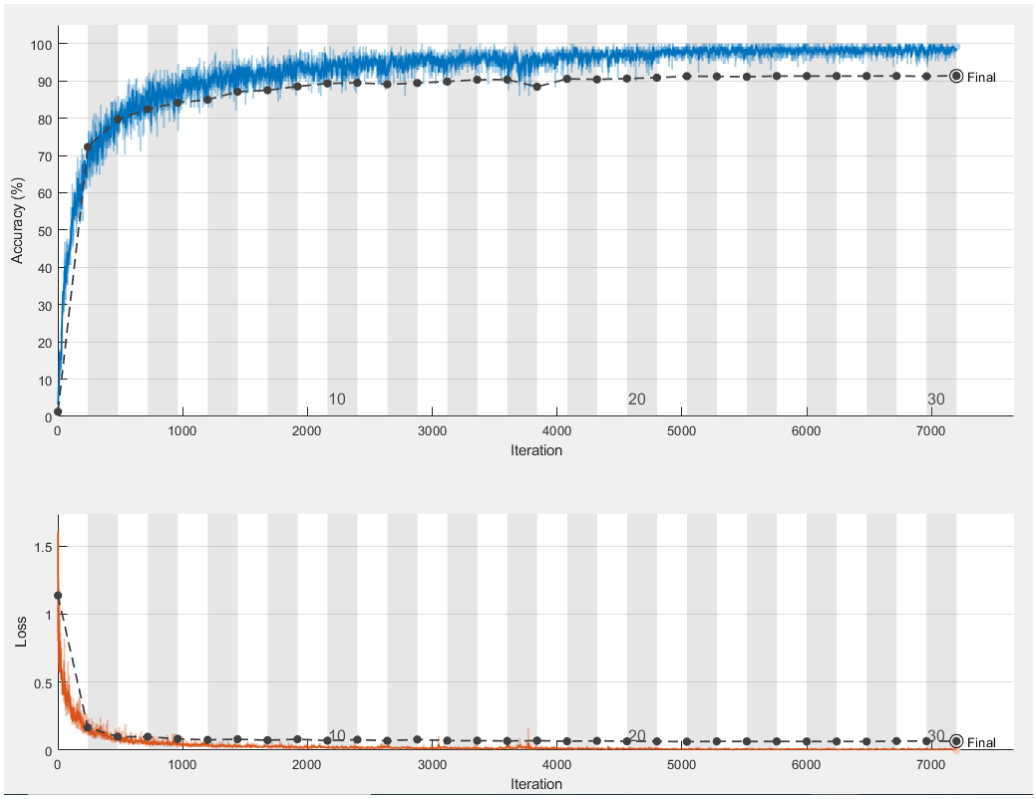


Figure 7. Neural network training progress.

The continuous blue line represents the accuracy achieved on the training data, while the dashed black line, which represents the accuracy achieved on the validation data, is updated less frequently. The validation set is used to ensure that the network is sufficiently abstracting what it learns from the training data, giving an indication on whether the training of the model is progressing well. The training error is about only 1% while the validation error reached 9%. The total validation accuracy is 91.31% , which means that the model is highly accurate and can be used for the desired application.

4.2. Performance Measures and Evaluation

The network's performance on test data can also be evaluated using a confusion matrix as shown in Figure 8. As the target variable is multinomial, we are dealing with a multi-class classification problem. It has 20 levels, 18 of which are related to the spoken words "up", "down", "right", "left", "on" and "off" in the three languages, and two to the classes "unknown" and "background". As a result of the testing phase, a 20×20 confusion matrix was derived, from which several performance measures were considered to evaluate the performance of the proposed CNN model for multilingual recognition of spoken commands. The ONE-vs-ALL strategy was used to calculate the values of the performance measures, with each level being considered as the positive class at a time and the others as the negative class. The performance measures that are considered in the experimental study are:

- Recall: This metric assesses how confident we can be that the model will find all spoken words related to the selected positive class. In other words, it indicates the proportion of true positives that are correctly classified. Recall is calculated for each target level (l_i) as follows:

$$Recall(l_i) = \frac{TP(l_i)}{TP(l_i) + FN(l_i)} \qquad i = 1..20 \tag{6}$$

- Precision: This metric helps in assessing how confident we can be that a spoken command identified as having a positive target level actually has that level. In other words, it indicates the proportion of predicted positives that are actually positive.

$$Precision(l_i) = \frac{TP(l_i)}{TP(l_i) + FP(l_i)} \qquad i = 1..20 \tag{7}$$

- F1-measure (or F1 score): The F1 measure, which is the harmonic mean of precision and recall, provides an alternative to assessing the misclassification rate and aids in concentrating on minimizing both false positives and false negatives equally.

$$F1_{score}(l_i) = 2 \times \frac{Precision(l_i) \times Recall(l_i)}{Precision(l_i) + Recall(l_i)} \qquad i = 1..20 \tag{8}$$

- Average class accuracy (ACA): Because the data set is imbalanced, using raw classification accuracy to evaluate overall performance can be misleading. Instead, we propose using the harmonic mean to calculate the average class accuracy, as shown in the equation below:

$$ACA_{HM} = \frac{1}{\frac{1}{20}\sum_{i=1}^{20}\frac{1}{Recall(l_i)}} \tag{9}$$

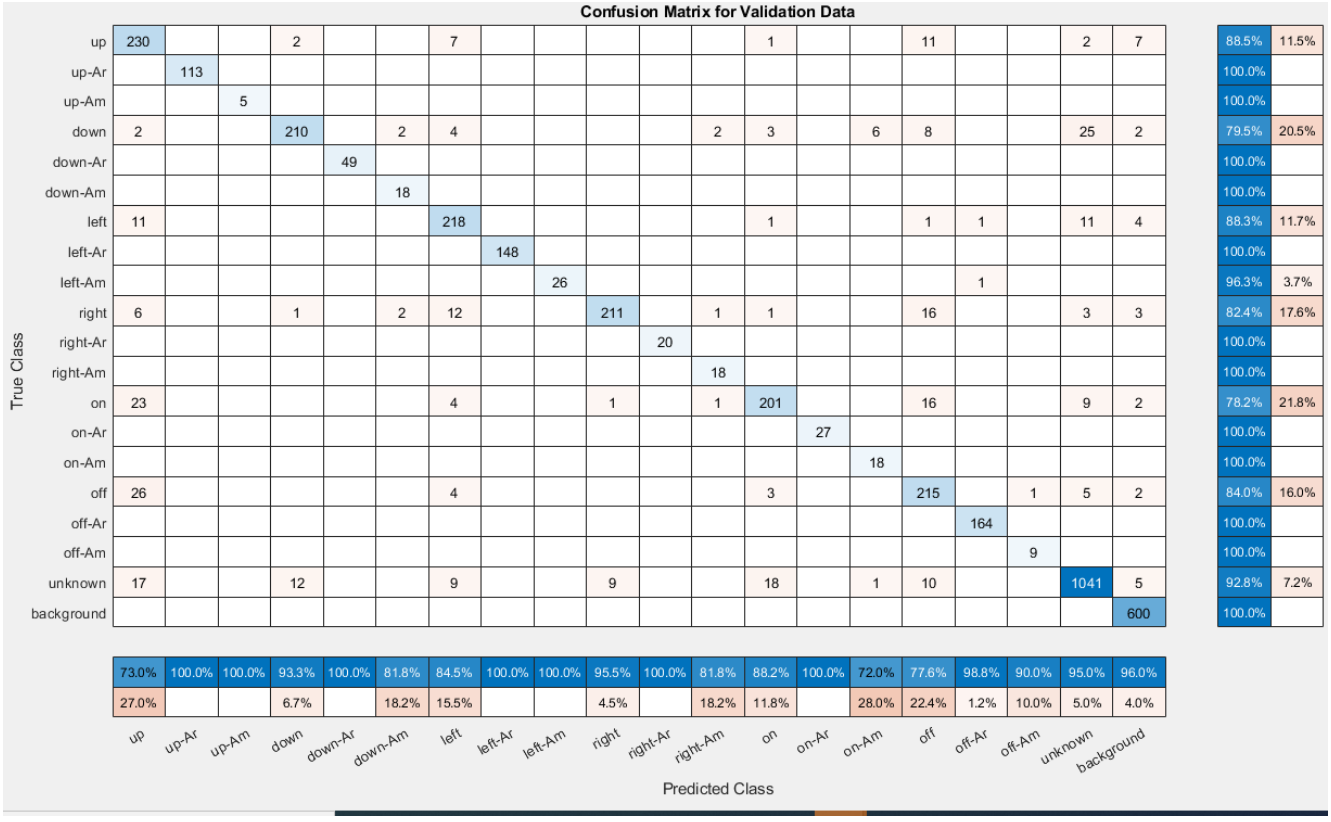


Figure 8. Obtained confusion matrix.

As indicated in Figure 9, the model's average class accuracy was 0.93, indicating its promising potential. Figure 9 also shows the obtained results in terms of recall, precision, and recall for each of the classes under consideration. Precision and recall metrics had values ranging from 0.7 to 1 and above 0.8, respectively. F1 scores, on the other hand, vary from 0.8 to 1. Despite the size of the English dataset, the results show poor performance for English words as compared to Arabic and Amazigh words. This can be explained by the

completely distinct word pronunciation, as well as the presence of a considerable portion of unknown English words.

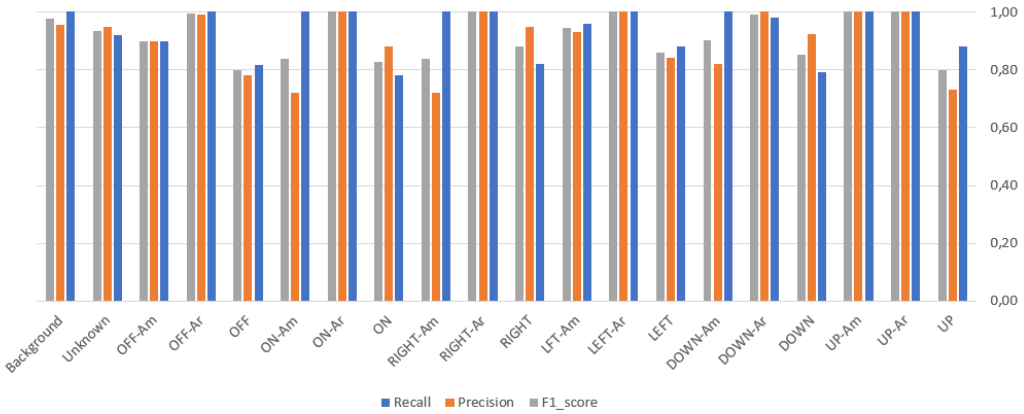


Figure 9. Performance measures and results.

4.3. Real Time Tests of the Proposed AMLSR

After training and evaluating the network, the test phase of the AMLSR system is performed. First the audio signal is extracted from the users in real time. The spectrogram is generated and transferred to the network in order to recognize the desired command.

The data was collected from 16 persons divided into two categories: the first one formed from Amazigh native speakers and the second composed of Arabic native speakers. Figure 10 shows speakers' age and gender distribution.

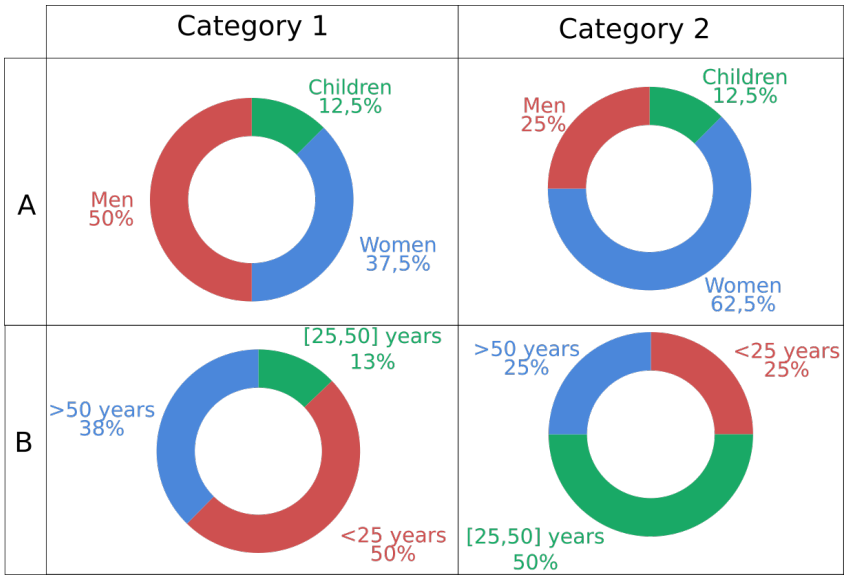


Figure 10. Speakers categories (**A**) Gender distribution. (**B**) Age distribution.

Overall, Figure 11 depicts the AMLSR model's achieved performance following this test phase. The AMLSR system performed reasonably well, with 93.76% for English word recognition, 88.55% for Arabic word recognition, and 82.31% for Amazigh word recognition, respectively. This time, English terms were easily identified because there was no unknown word introduced in the test set, and non-native speakers, unlike native speakers, have an accentuated and clear pronunciation of these English words.

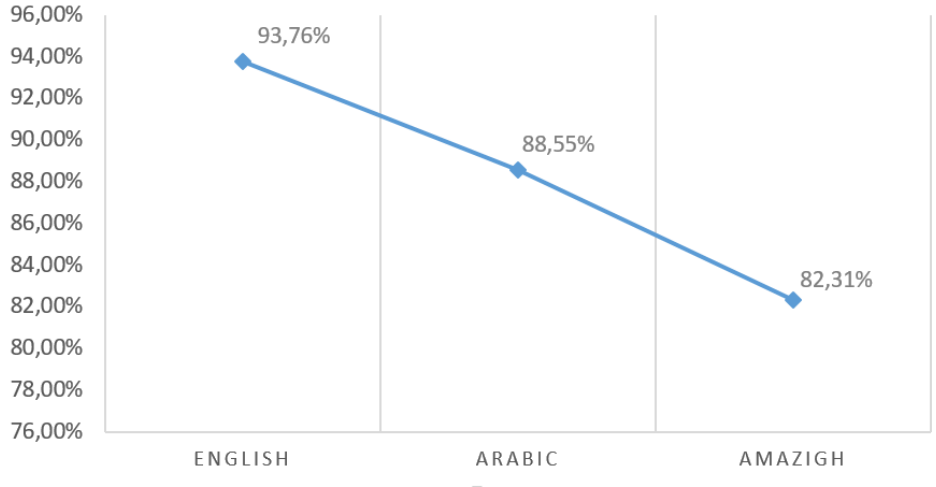


Figure 11. Achieved performance during test phase for each data set.

5. Hardware Implementation for AMLSR Deployment

The UAV used for the real time test is a quadrotor drone manufactured in our laboratory for academic purposes. The drone can support an additional payload of 500 g in order to execute different kind of missions. It is equipped with an Atmel processor on-board connected to different sensors such as the Inertial measurement unit for attitude estimation, GPS for positioning and ultra-sound for obstacles detection. Radio communication and data transmission is assured via TX/RX module with a 1 km transmit rang. Furthermore, communication with the interface is granted via USB. The quadrotor drone has a flight autonomy of approximately ten minutes depending on its usage.

The main goal of creating the graphical interface is to reduce the user's workload by simplifying command and interaction with the UAV. The interface was created so that a single operator could communicate with the UAV and issue the required movement commands using the proposed languages. In addition, as shown in Figure 12, the following factors were taken into consideration when developing the design:

- The interface is able to automatically detect the available UAVs.
- The interface displays the information related to the motors rotation speed as PWM signals. This information is displayed with different-colored text in order to help the operator to avoid overload.
- The detected command is displayed as 'Text' with the language of the user.
- The interface is offered in a variety of languages depending on the user's language. English is primarily suggested.
- The interface displays visual alerts in reaction to various events that may occur during the system's functioning.

In the implemented system, once a command is transmitted to the UAV, it is carried out until another command is delivered. As a result, to stop the UAV's engine, an explicit instruction for the action "stop" is required. For safety and to avoid collisions, the UAV will automatically stop when it reaches 0.1 m above the ground.

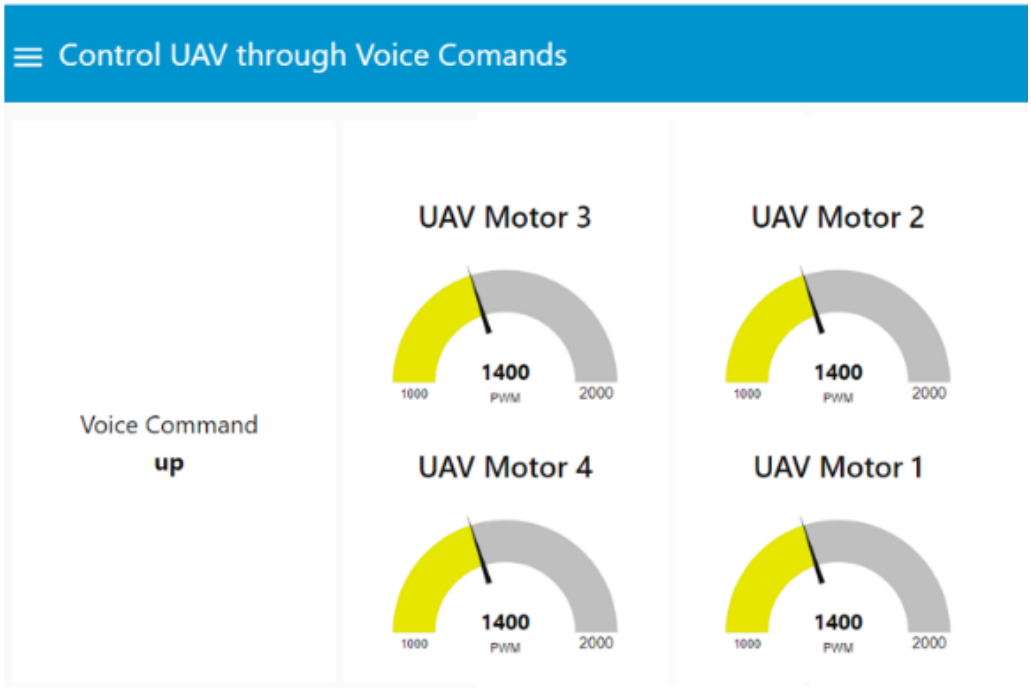


Figure 12. Graphical interface for AMLSR system.

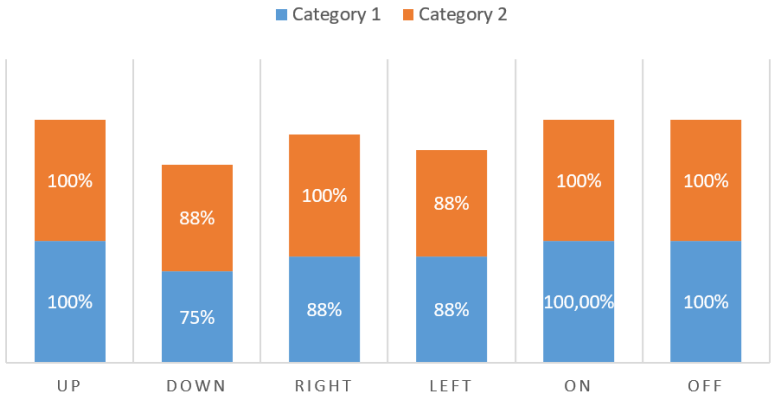
6. Comparative Study and Discussion

Figure 13 shows that being a native speaker has an impact on the recognition rate. For example, in category 1, Amazigh words were recognized better than English and Arabic terms, whereas Arabic words were recognized best in category 2. However, because Arabic commands are less complex than Amazigh commands, the results for Arabic words appear to be better than those for Amazigh terms in both categories. Furthermore, English recognition works well with both categories because the commands are short and easy to pronounce. Because Amazigh commands have many syllables, users must correctly pronounce the word in order to be recognized. The fact that Amazigh is a language with several dialects, therefore the accent changes from person to person, explains category 1 errors in Amazigh command. Finally, there was considerable ambiguity in the multilingual test between the terms "left", "asfel", and "ayfouss". Moreover, even in a noisy environment, the results are very encouraging.

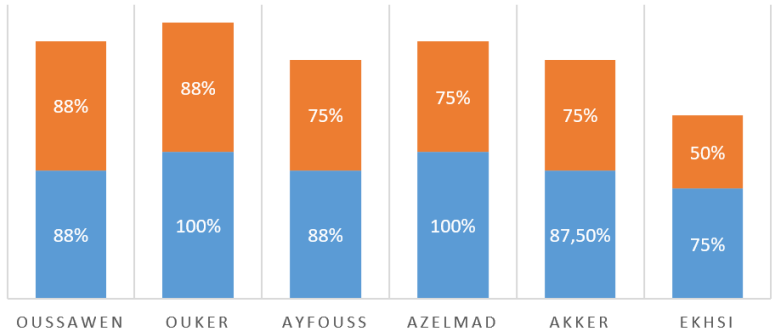
This work can be considered as the first attempt that uses the most commonly spoken languages in the MENA region, including standard Arabic, English, and Amazigh to develop an AMLSR system for HDI. Table 3 compares our work to some similar studies in the ASR related-literature based on the following criteria: Languages used for ASR, speech signal (SS) representation used, average accuracy, robustness, and system hardware implementation. Several features can be considered to design an ASR system, including the Mel-frequency cepstral coefficients (MFCC) and spectrograms.

Table 3. Comparison with similar studies

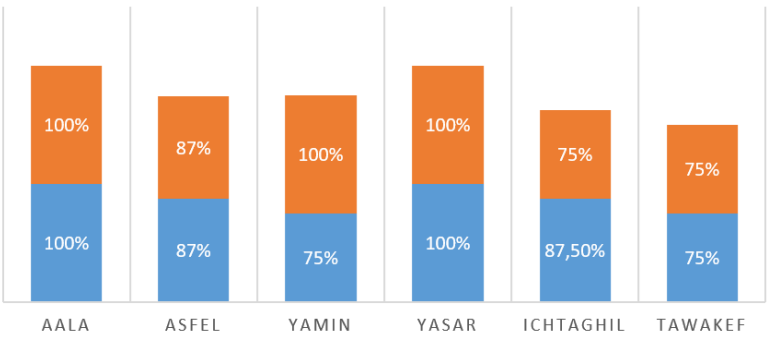
Criteria	[36]	[28]	[3]	[4]	AMSLR
Used Languages	En, Ar	En	En, Sp	En	En, Ar, Am
Used representation	MFCC	MFCC	GCS API	Spectrograms	Spectrograms
Average Accuracy	89.85%	76.2%	96.67%	79%	93.76%
Robustness	Not Considered	Considered	Considered	Considered	Considered
System Hardware Implementation	No	Yes	Simulation	Yes	Yes



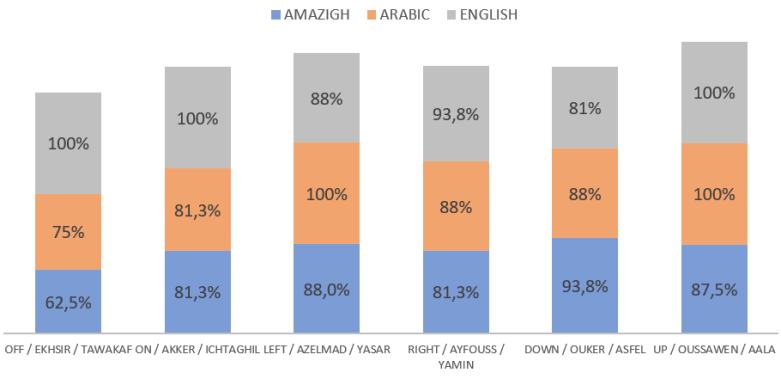
(**a**)



(**b**)



(**c**)



(**d**)

Figure 13. Accuracy results on test data for different spoken words: (**a**) English, (**b**) Amazigh, (**c**) Arabic, and (**d**) Multilingual.

Considering the entire hardware implementation, the prototype's performance was quite promising. However, our system is limited to a small set of commands. A more extensive instruction dictionary is needed. Furthermore, additional instructions to manage the drone's position and camera should also be considered. The developed system must account for communication disruptions and environmental limits in order to avoid collisions and obstacles.

Extending the dataset while ensuring good class-balance in target feature distribution is required to improve the ASR system. To avoid the burden of further data collection and labelling, a Self-Supervised Learning (SSL) approach, as described in [37], could be used for this purpose. Semi-supervised learning can also be investigated. Furthermore, one of the most recent aspects to be explored is multitask language learning as it could improve speech interaction with drones [38]. Another component that might be included to explain the predictions produced by algorithms, as stated in [39], is the explainability of the settings. Moreover, the system can be extended to handle audiovisual interaction techniques, in which the drone can predict the user's movements using an integrated camera and background invariant detection [40]. As the altitude of drones changes, this last strategy must be supplemented with multiscale object detection [41].

7. Conclusions

The goal of this work was to develop a speech recognition system that can be used to interact with unmanned aerial vehicles using words spoken in English, Arabic, and Amazigh. As a first step, the architecture of a CNN has been defined, and the data required for training, validation, and testing has been collected and pre-processed. A rate of more than 90% accuracy was achieved. A quadrotor drone was built and used during the model's deployment in a real environment. A user interface has also been designed to make it simple to control the engine. This step of the system evaluation involved a sample of 16 people of various ages and genders. The AMLSR system performed well in terms of recognition accuracy, with an accuracy of 93.76% for English interaction, 88.5% for Arabic interaction, and 82.31% for Amazigh interaction. The AMLSR system was able to understand the user's voice instructions with an average accuracy of 88.2% during tests. The proposed prototype can be improved in a variety of ways as part of future work plans. It would be interesting to investigate other DL models and to expand the database in order to improve efficiency. As suggestions, we propose creating a website where a large number of people can record commands in order to cover other existing dialects. Furthermore, the graphical interface could be improved to show more aspects such as UAV attitude sphere, battery health, and so on. We may even enter a lexical field to broaden the command dictionary to include all possible words used for control and even a navigation tool.

Author Contributions: Conceptualization, K.C.; Data curation, Y.K.; Methodology, S.M.; Software, N.M.; Supervision, M.L. and M.B.; Validation, K.C., M.L., S.M. and M.B.; Writing—original draft, K.C.; Writing—review & editing, S.M. and M.B. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by Princess Nourah Bint Abdulrahman University Researchers Supporting Project number (PNURSP2022R196), Princess Nourah Bint Abdulrahman University, Riyadh, Saudi Arabia.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Acknowledgments: The authors would like to acknowledge the Princess Nourah Bint Abdulrahman University Researchers Supporting Project number (PNURSP2022R196), Princess Nourah Bint Abdulrahman University, Riyadh, Saudi Arabia.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Castrillo, V.U.; Manco, A.; Pascarella, D.; Gigante, G. A Review of Counter-UAS Technologies for Cooperative Defensive Teams of Drones. *Drones* **2022**, *6*, 65. [CrossRef]
2. Mirri, S.; Prandi, C.; Salomoni, P. Human-Drone Interaction: State of the art, open issues and challenges. In Proceedings of the ACM SIGCOMM 2019 Workshop on Mobile AirGround Edge Computing, Systems, Networks, and Applications, Beijing, China, 19 August 2019; pp. 43–48.
3. Contreras, R.; Ayala, A.; Cruz, F. Unmanned aerial vehicle control through domain-based automatic speech recognition. *Computers* **2020**, *9*, 75. [CrossRef]
4. Park, J.S.; Na, H.J. Front-end of vehicle-embedded speech recognition for voice-driven multi-UAVs control. *Appl. Sci.* **2020**, *10*, 6876. [CrossRef]
5. Wojciechowska, A.; Frey, J.; Sass, S.; Shafir, R.; Cauchard, J.R. Collocated human–drone interaction: Methodology and approach strategy. In Proceedings of the 2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI), Daegu, Korea, 11–14 March 2019; pp. 172–181.
6. Cauchard, J.R.; Khamis, M.; Garcia, J.; Kljun, M.; Brock, A.M. Toward a roadmap for human–drone interaction. *Interactions* **2021**, *28*, 76–81. [CrossRef]
7. Christ, P.F.; Lachner, F.; Hösl, A.; Menze, B.; Diepold, K.; Butz, A. Human-drone-interaction: A case study to investigate the relation between autonomy and user experience. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–10 October 2016; pp. 238–253.
8. Liu, C.; Szirányi, T. Real-Time Human Detection and Gesture Recognition for On-Board UAV Rescue. *Sensors* **2021**, *21*, 2180. [CrossRef]
9. Nan, Z.; Jianliang, A. Speech Control Scheme Design and Simulation for UAV Based on HMM and RNN. *J. Syst. Simul.* **2020**, *32*, 464.
10. Kim, D.; Oh, P.Y. Human-drone interaction for aerially manipulated drilling using haptic feedback. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 25–29 October 2020; pp. 9774–9780.
11. Tezza, D.; Garcia, S.; Hossain, T.; Andujar, M. Brain eRacing: An exploratory study on virtual brain-controlled drones. In Proceedings of the International Conference on Human-Computer Interaction, Orlando, FL, USA, 26–31 July 2019; pp. 150–162.
12. Tezza, D.; Andujar, M. The state-of-the-art of human–drone interaction: A survey. *IEEE Access* **2019**, *7*, 167438–167454. [CrossRef]
13. Jie, L.; Jian, C.; Lei, W. Design of multi-mode UAV human-computer interaction system. In Proceedings of the 2017 IEEE International Conference on Unmanned Systems (ICUS), Beijing, China, 27–29 October 2017; pp. 353–357.
14. Malik, M.; Malik, M.K.; Mehmood, K.; Makhdoom, I. Automatic speech recognition: A survey. *Multimed. Tools Appl.* **2021**, *80*, 9411–9457. [CrossRef]
15. Nassif, A.B.; Shahin, I.; Attili, I.; Azzeh, M.; Shaalan, K. Speech recognition using deep neural networks: A systematic review. *IEEE Access* **2019**, *7*, 19143–19165. [CrossRef]
16. Izbassarova, A.; Duisembay, A.; James, A.P. Speech recognition application using deep learning neural network. In *Deep Learning Classifiers with Memristive Networks*; Springer: Cham, Switzerland, 2020; pp. 69–79.
17. Indolia, S.; Goswami, A.K.; Mishra, S.P.; Asopa, P. Conceptual understanding of convolutional neural network-a deep learning approach. *Procedia Comput. Sci.* **2018**, *132*, 679–688. [CrossRef]
18. Song, Z. English speech recognition based on deep learning with multiple features. *Computing* **2020**, *102*, 663–682. [CrossRef]
19. Veisi, H.; Mani, A.H. Persian speech recognition using deep learning. *Int. J. Speech Technol.* **2020**, *23*, 893–905. [CrossRef]
20. El Ouahabi, S.; Atounti, M.; Bellouki, M. Toward an automatic speech recognition system for amazigh-tarifit language. *Int. J. Speech Technol.* **2019**, *22*, 421–432. [CrossRef]
21. Alsayadi, H.A.; Abdelhamid, A.A.; Hegazy, I.; Fayed, Z.T. Arabic speech recognition using end-to-end deep learning. *IET Signal Process.* **2021**, *15*, 521–534. [CrossRef]
22. Zhou, Y.; Hou, J.; Gong, Y. Research and Application of Human-computer Interaction Technology based on Voice Control in Ground Control Station of UAV. In Proceedings of the 2020 IEEE 6th International Conference on Computer and Communications (ICCC), Chengdu, China, 11–14 December 2020; pp. 1257–1262.
23. Yamazaki, Y.; Tamaki, M.; Premachandra, C.; Perera, C.; Sumathipala, S.; Sudantha, B. Victim detection using UAV with on-board voice recognition system. In Proceedings of the 2019 Third IEEE International Conference on Robotic Computing (IRC), Naples, Italy, 25–27 February 2019; pp. 555–559.
24. Meszaros, E.L.; Chandarana, M.; Trujillo, A.; Allen, B.D. Speech-based natural language interface for UAV trajectory generation. In Proceedings of the 2017 International Conference on Unmanned Aircraft Systems (ICUAS), Miami, FL, USA, 13–16 June 2017; pp. 46–55.
25. Galangque, C.M.J.; Guirnaldo, S.A. Speech Recognition Engine using ConvNet for the development of a Voice Command Controller for Fixed Wing Unmanned Aerial Vehicle (UAV). In Proceedings of the 2019 12th International Conference on Information & Communication Technology and System (ICTS), Surabaya, Indonesia, 18 July 2019; pp. 93–97.
26. Kumaar, S.; Bazaz, T.; Kour, S.; Gupta, D.; Vishwanath, R.M.; Omkar, S. A Deep Learning Approach to Speech Based Control of Unmanned Aerial Vehicles (UAVs). *CS & IT Conf. Proc.* **2018**, *8*. [CrossRef]
27. Oneata, D.; Cucu, H. Kite: Automatic speech recognition for unmanned aerial vehicles. *arXiv* **2019**, arXiv:1907.01195.

28. Mięsikowska, M. Discriminant Analysis of Voice Commands in the Presence of an Unmanned Aerial Vehicle. *Information* **2021**, *12*, 23. [CrossRef]
29. Nicolson, A.; Paliwal, K.K. Deep Xi as a front-end for robust automatic speech recognition. In Proceedings of the 2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE), Gold Coast, Australia, 16–18 December 2020; pp. 1–6.
30. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
31. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems 25, Lake Tahoe, NV, USA, 3–6 December 2012.
32. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
33. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
34. Kamath, U.; Liu, J.; Whitaker, J. *Deep Learning for NLP and Speech Recognition*; Springer: Cham, Switzerland, 2019; Volume 84.
35. Bishop, C.M.; Nasrabadi, N.M. *Pattern Recognition and Machine Learning*; Springer: New York, NY, USA, 2006; Volume 4.
36. Nassif, A.B.; Shahin, I.; Elnagar, A.; Velayudhan, D.; Alhudhaif, A.; Polat, K. Emotional Speaker Identification using a Novel Capsule Nets Model. *Expert Syst. Appl.* **2022**, *193*, 116469. [CrossRef]
37. Jing, L.; Tian, Y. Self-supervised visual feature learning with deep neural networks: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *43*, 4037–4058. [CrossRef] [PubMed]
38. Samant, R.M.; Bachute, M.; Gite, S.; Kotecha, K. Framework for Deep Learning-Based Language Models using Multi-task Learning in Natural Language Understanding: A Systematic Literature Review and Future Directions. *IEEE Access* **2022**, *10*, 17078–17097. [CrossRef]
39. Joshi, G.; Walambe, R.; Kotecha, K. A review on explainability in multimodal deep neural nets. *IEEE Access* **2021**, *9*, 59800–59821. [CrossRef]
40. Kotecha, K.; Garg, D.; Mishra, B.; Narang, P.; Mishra, V.K. Background Invariant Faster Motion Modeling for Drone Action Recognition. *Drones* **2021**, *5*, 87. [CrossRef]
41. Walambe, R.; Marathe, A.; Kotecha, K. Multiscale object detection from drone imagery using ensemble transfer learning. *Drones* **2021**, *5*, 66. [CrossRef]