computers

MDPI

Article

Unmanned Aerial Vehicle Control through Domain-Based Automatic Speech Recognition

Ruben Contreras [1,*], Angel Ayala [2,*] and Francisco Cruz [1,3,*]

[1] Escuela de Ingeniería, Universidad Central de Chile, Santiago 8330601, Chile
[2] Escola Politécnica de Pernambuco, Universidade de Pernambuco, Recife 50720-001, Brasil
[3] School of Information Technology, Deakin University, Geelong 3220, Australia
* Correspondence: ruben.contreras@alumnos.ucentral.cl (R.C.); aaam@ecomp.poli.br (A.A.); francisco.cruz@deakin.edu.au (F.C.)

Received: 5 August 2020; Accepted: 17 September 2020; Published: 19 September 2020

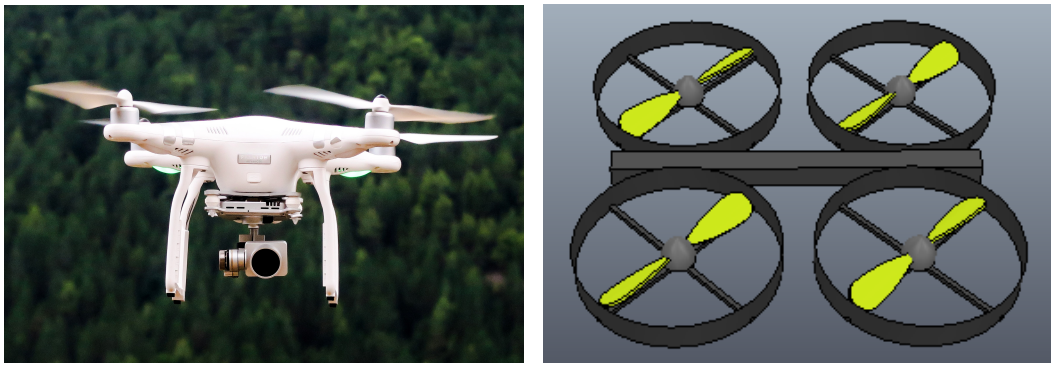
Abstract: Currently, unmanned aerial vehicles, such as drones, are becoming a part of our lives and extend to many areas of society, including the industrialized world. A common alternative for controlling the movements and actions of the drone is through unwired tactile interfaces, for which different remote control devices are used. However, control through such devices is not a natural, human-like communication interface, which sometimes is difficult to master for some users. In this research, we experimented with a domain-based speech recognition architecture to effectively control an unmanned aerial vehicle such as a drone. The drone control was performed in a more natural, human-like way to communicate the instructions. Moreover, we implemented an algorithm for command interpretation using both Spanish and English languages, as well as to control the movements of the drone in a simulated domestic environment. We conducted experiments involving participants giving voice commands to the drone in both languages in order to compare the effectiveness of each, considering the mother tongue of the participants in the experiment. Additionally, different levels of distortion were applied to the voice commands to test the proposed approach when it encountered noisy input signals. The results obtained showed that the unmanned aerial vehicle was capable of interpreting user voice instructions. Speech-to-action recognition improved for both languages with phoneme matching in comparison to only using the cloud-based algorithm without domain-based instructions. Using raw audio inputs, the cloud-based approach achieves 74.81% and 97.04% accuracy for English and Spanish instructions, respectively. However, with our phoneme matching approach the results are improved, yielding 93.33% accuracy for English and 100.00% accuracy for Spanish.

Keywords: drone control; automatic speech recognition; robot simulator

1. Introduction

Presently, unmanned aerial vehicles (UAVs) are more frequently used with a wide variety of applications in many areas such as security, industry, food, and transport, among others [1]. In this regard, it is essential to incorporate solutions that provide UAVs with the ability to be controlled remotely, making understandable the orders communicated that must be executed. A very popular kind of UAV is a drone, which is a mobile robotic structure capable of flying that may be operated remotely. Several types of UAVs exist, and they are categorized according to their frame properties, propellers, engine, power system, electronic control, and communication system [1]. Commonly, UAVs have a built-in camera to capture video during flight. Additionally, others include a thermal infrared camera [2] to record wildlife without disturbing their environment, and some UAVs include a radio frequency sensor to detect hazardous waste in railway accidents [3]. Moreover, UAVs have

become helpful vehicles for the acquisition of data as well as for the transport of elements with no human presence. An example of a quadrotor drone that can be operated in real-world scenarios by radio control is shown in Figure 1a.



(**a**) (**b**)

Figure 1. Example of quadrotor drones. The unmanned aerial vehicles are shown in both real-world and simulated environments. (**a**) DJI Phantom radio controlled quadrotor operating in a real environment; (**b**) Simulated quadrotor drone in V-REP used in the proposed domestic scenario.

A significant part of the UAV functionalities and advantages lies in the sensors [1], which provide an extension to its capacities in order to obtain information about the environment in where it is deployed. Nevertheless, just a few add-ons are focused on extending its remote control capability. For instance, this problem was addressed by Fernandez et al. [4] where voice control was proposed using dictionaries. The proposed technique consisted of 15 commands for UAV control in a given language.

In this paper, we present an experimental approach for drone control through a cloud-based speech recognition system, improved by a domain-based language. The cloud-based automatic speech recognition was carried out using the *Google Cloud Speech* (https://cloud.google.com/speech-to-text/) (GCS) service [5] through the Web Speech API (https://wicg.github.io/speech-api/) [6], which in this context was customized with a domain-based dictionary for the proposed scenario as shown in [7]. Therefore, we combined GCS and a predefined language based on the problem domain to convert the voice into text. Then, this was interpreted as an instruction for the drone. Our domain-based language is a dictionary comprising 48 instructions. Some of these are in Spanish and others in English. These commands were interpreted and mapped into one of the nine available actions that the drone could execute. Experiments were performed in a simulated domestic environment that included chairs, tables, and shelves, among others. This novel approach contributes to the state of the art by improving the automatic speech recognition, in terms of action classification for drone control, by scaffolding the raw voice input signal through domain-based phoneme matching. Additionally, the proposed approach contributes by allowing an enhanced drone control independently of the user's native language. In this regard, even in the presence of a noisy signal and limited English instruction utterances, these did not significantly affect the speech recognition system after phoneme matching, achieving in all tested cases high rates of accuracy.

During the experiments, participants gave voice commands to the drone in a simulated environment. In general, Spanish language instructions were better understood than the English language instructions. This was mainly due to the native language spoken by the participants. However, in both cases, the success rate for recognition of the instructions was improved by using domain-based instructions. The implementation of the simulated scenario, including the drone and the speech recognition features mentioned above, was carried out using the V-REP (From November 2019, V-REP simulator was replaced by CoppeliaSim version 4.0. See https://www.coppeliarobotics.com/) robot simulator, which was previously used in domestic robotic scenarios [8–12]. V-REP [13] is a simulation tool that allows experimentation with virtual robots and provides a graphical interface

for creating and editing different simulated models, as well as designing the environment with the necessary elements. Figure 1b shows the simulated drone that was used within the home environment implemented in this project.

2. Related Works

2.1. Unmanned Aerial Vehicle Control

For some years now, unmanned aerial vehicles (UAVs), such as drones, have been more in demand in the market [14]. Nevertheless, UAVs were created many decades ago, when Archibald Low proposed the first drone in 1916 on a project for the British Air Ministry to develop an unmanned defense aircraft for use against German airships [15]. From Low's development of the first radio controlled UAV to the current drone industry, a plethora of changes have occurred. Presently, drones play an active role in many areas, such as military, agriculture, recreation, and search and rescue [1]. The massive emergence of UAVs and their extensive applications in different fields have led to the development of simpler control forms for non-expert end-users. Implementation of UAV systems was proven to be cost-effective in covering vast area extensions, e.g., for data acquisition tasks [16]. Moreover, drones present greater maneuverability in areas where other traditional unmanned air vehicles have shown inefficacy [17].

Since UAVs are designed to fly with no onboard pilot, a self-driving UAV is, indeed, a desirable characteristic present in these vehicles. Self-controlled drones have achieved a highly autonomous flight level, as specified in [18], and addressed by novelty machine learning techniques [19–24]. Many of these techniques achieved successful performance in online path planning. However, the results still rely on previously presented paths, or in a route tracking pattern. In this regard, when we refer to an autonomous system, it does not mean it is necessarily an intelligent system. The main challenge to enable intelligent UAVs is the automated decision-making process [25]. Other authors have addressed UAV online control as a semi-autonomous system [26] capable of being controlled externally through a hardware interface. The semi-autonomous UAV control technique allows a human operator to intervene in the drone's actions when required to make paths that are more precise during flights [27]. Moreover, related task-specific techniques were proposed in order to use the UAV as a tool that aids the users' tasks [28,29]. Semi-autonomous remotely operated drones for different tasks allow users to make corrections in the flight path during the mission, improving overall results for many goal-specific purpose systems.

A simple UAV controller seeks to become familiar with the use of the drone for all possible users in many different fields. For instance, in Fernandez et al. [4], the authors experimented with several natural user interfaces for human-drone interaction, among them the gesture and speech control. For gestures, experiments included body and hand interaction. Each interface was tested with different users in public areas, achieving overall positive results. A more recent gesture-based control of semi-autonomous vehicles [30], tested in virtual reality and real-world environments, showed users in some situations still preferred to use a joystick device to control the vehicle, mainly because they were more habituated to such technology. However, the authors also concluded that hand gesture-based control is more intuitive and easy for users to learn how to drive the vehicle quickly. Although the actual technology captures a wide range of hand motions with high precision, humans may not be so precise in all circumstances. Therefore, a control system may benefit from supplementary methods to pre-process the inputs and smooth the movements. In terms of speech for controlling the drone, Wuth et al. [31] demonstrated that users find speech-based communication transparent and efficient. This efficient communication relies on knowledge of the context for the specific task addressed by the drone. Overall, the context plays an important role in human-drone environments to achieve effective control of the UAV.

2.2. Speech Control

Currently, some researchers addressed UAV control using more natural interfaces for people, e.g., through automatic speech recognition. For instance, Lavrynenko et al. [32] proposed a radio-based remote control system, in which a semantic identification method based on mel-frequency cepstral (MFC) coefficients was used. The audio captured was translated to an action that, in turn, was transferred to the UAV for its execution. Each time a microphone captured a new voice command, the system computed the cepstral coefficient. The coefficient was compared against a database of cepstral coefficients, using the minimum distance criterion to match the desired command. Nevertheless, the database of cepstral coefficients only comprised four voice commands, corresponding to each direction of the UAV.

A more accurate speech recognition method was presented by Fayjie et al. [33]. In that study, the authors used a hidden Markov model for speech recognition with voice adaptation in order to control the UAV. Their proposal was based on a speech-decoding engine called Pocketsphinx, used with ROS in the Gazebo simulator. The speech decoding worked with the CMU Sphinx Knowledge Base Tool, implemented with seven actions to control altitude, direction, yaw, and landing. However, the CMU Sphinx Knowledge Base Tool is not being actively developed and is considered deprecated when compared to modern neural network-based approaches. Another similar approach was created by Landau et al. [34], where the authors used the Nuance speech recognition service. They proposed a hands-free UAV control with voice commands, to actuate over a DJI Phantom 4 drone, developed with the DJI Mobile SDK for iOS. The proposed architecture was composed of a Bluetooth hands-free for voice capture, and the speech commands were translated and evaluated using regular expressions. The regular expressions were divided into three groups. The first group contained possible words to move the drone in any direction. The second consisted of possible words to move the drone in any direction, but with an established distance. The third group was composed of words for take-off and landing the drone. The implementation of this work was limited to be used through a Bluetooth hands-free device connected to an Android smartphone in order to control only DJI manufactured drones.

Chandarana et al. [35] presented a custom-developed software using speech and gesture recognition for UAV path planning. In their research, the authors performed a comparison of natural language interfaces using a mouse-based interface as a baseline and evaluating individual users who were required to complete a flight path composed of flight trajectories or actions. The authors proposed a software where the users interacted using either a mouse, gesture, or speech, in order to build three specific flight paths. The speech recognition phase was handled using the CMU Sphinx 4-5prealpha speech-to-text software, used with rule-based grammar. This allowed the system to hear compound name formations, e.g., forward-left and backward-right trajectories, among others. Their work also presented an evaluation of the user's response to natural language interfaces. Although the highest performance was achieved with the mouse-based interface, users reported preference in using speech for mission planning.

Additionally, a multi-modal approach considering voice interaction with drones used a word dictionary for speech recognition [4]. However, only 15 different commands in one language were allowed to control the UAV. Quigley et al. proposed a speech controller to recognize commands sent to a semi-autonomous UAV [26]. In their experiments, flight tests were conducted revealing that ambient noise and conversation could considerably affect the reliability of the speech recognition system. A study of gesture and speech interfaces for interaction with drones in simulated environments was conducted by Jones et al. [36]. The results showed that subjects participating generally preferred to use lower-level commands, such as left or right to control the drone.

One of the most recent studies was an extension of [32] developed by Lavrynenko et al. [37]. In this extension, the authors presented a similar radio-based control system with cepstral analysis. However, in this project they also added encrypted communication. The proposed architecture used a voice-control panel that handled the encryption, including the cepstral and wavelet coefficients.

Moreover, the inverse process of coefficient quantization was performed, comparing it to the cepstral database using the minimum distance criterion. Both parts, the encryption and decryption, presented an encryption key, which works with signal filters acquiring the features of the speech.

3. Proposed Architecture

In the last decade, the Natural User Interface (NUI) technology was actively developed [38]. It focuses on communication naturally between digital devices and end-users. These approaches have fostered the idea that humans can interact with a machine in such a way that no control device is needed to generate input information to the machine, such as keyboards, touch screens, or any other device that people require for physical contact [39]. Different types of NUIs exist, and some of these include:

- Voice recognition: The interface must be able to recognize instructions through the user's voice.
- Gesture recognition: The interface can capture gestures from the human body and interpret them.
- Visual marker interaction: Visual markers are added. These are captured by a camera and recognized by the machine.

In the proposed architecture, the drone interaction was not carried out through remote control, but rather it used a Natural-language User Interface (NLUI), interpreting instructions from humans through automatic speech recognition. For instruction interpretation, the person's voice was captured by a microphone connected to a computer that executed the algorithms to process the audio signal received. The microphone can be either a built-in one from a laptop computer or any other external device. However, the quality of the signal captured may considerably vary and, in turn, impact the accuracy of the interpretation [40]. To transform the audio signal into text, Google Cloud Speech (GCS) was used in combination with a domain-based language. Audio streams are received from the microphone and sent to the cloud-based GCS service through the Web Speech API, from where we obtained a recognized sentence as a hypothesis. Next, the hypothesis was compared to our domain-based dictionary by performing a phoneme matching using the Levenshtein distance [41].

The Levenshtein distance $\mathcal{L}$, also known as the edition distance, is the minimal amount of operations needed to transform a sentence s_x into another sentence s_y. We compared the characters inside s_x to the ones inside s_y. The operations considered to transform the sentence comprised substitutions, insertions, and deletions. The cost of each edition operation was equal to 1. The distance was computed recursively as $\mathcal{L}_{s_x,s_y}(|s_x|,|s_y|)$ with $|s_x|$ and $|s_y|$ as the length of the sentences s_x and s_y respectively, and where the i-th segment of the sentence was computed as shown in Equation (1). In the equation, $c_{s_{x_i},s_{y_j}}$ is 0 if $s_{x_i} = s_{y_j}$ and 1 otherwise. Thus, the cost of transforming the sentence $s_1 =$ "to the left" to $s_2 =$ "go to the left" was equal to $\mathcal{L}_{s_1,s_2} = 3$ since it involved the insertion of 3 new characters. Furthermore, the cost of transforming the sentence $s_3 =$ "go right" to $s_4 =$ "go left" was equal to $\mathcal{L}_{s_3,s_4} = 4$ since the number of operations needed was 3 substitutions and 1 deletion.

$$\mathcal{L}_{s_x,s_y}(i,j) = \begin{cases} \max(i,j) & \text{if } \min(i,j) = 0 \\ \min \begin{cases} \mathcal{L}_{s_x,s_y}(i-1,j)+1 \\ \mathcal{L}_{s_x,s_y}(i,j-1)+1 \\ \mathcal{L}_{s_x,s_y}(i-1,j-1)+c_{s_{x_i},s_{y_j}} \end{cases} & \text{if } \min(i,j) \neq 0 \end{cases} \quad (1)$$

To perform the phoneme matching, the Levenshtein distance was computed between the recognized hypothesis and the domain-based dictionary. Afterwards, the instruction showing the minimum distance was selected. Once the voice command was converted into text, the signal was processed and classified as an instruction for the UAV. In our scenario, the drone was within the V-REP robot simulator. Figure 2 shows the proposed architecture. Moreover, Algorithm 1 portrays the operations carried out for the control of the drone through voice commands considering both with and without phoneme matching.

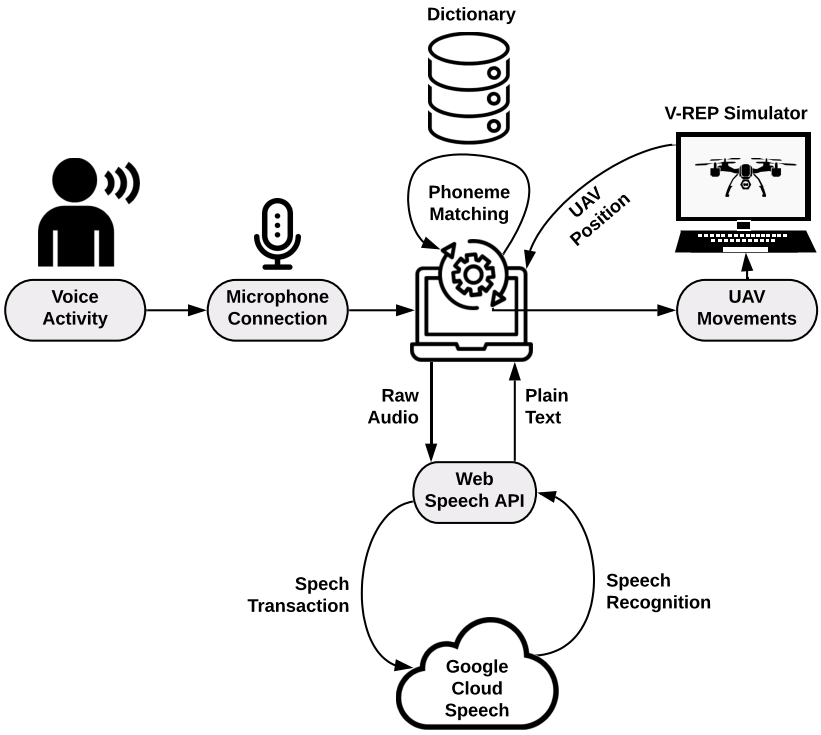


Figure 2. The proposed architecture for UAV control through speech. In this representation, a person speaks the instructions in a microphone, and these are processed by our algorithm. The instruction is then classified using the domain-based dictionary and executes for the UAV.

Algorithm 1 Algorithm implemented for the interpretation of an audio signal into an instruction for the drone. The algorithm comprises two sections for speech recognition with and without phoneme matching.

Initialize dictionary D with instructions i and classes C.
repeat
  Wait for microphone *audio* signal.
  Send *audio* signal to Google Cloud Speech.
  Receive *hypothesis* h.
  if phoneme matching is activated **then**
    for each instruction $i \in D$ **do**
      Compute $\mathcal{L}_{h,i}$.
    end for
    Choose instruction as min $\mathcal{L}_{h,D_i}$.
    Match chosen instruction to action class $a \in C$.
    Execute action class $a \in C$ in the scenario.
  else
    for each instruction $i \in D$ **do**
      Compare h to D_i.
      if $h \in D$ **then**
        Match instruction $i \in D$ to action class $a \in C$.
        Execute action class $a \in C$ in the scenario.
        Exit loop.
      end if
    end for
  end if
until an exit instruction is given

4. Experimental Setup

Different tools were used for developing this project. One of these was V-REP [13], a closed-source simulation software freely available with an educational license for several operating systems, such as Linux, Windows, and iOS, for simulating different types of robots in realistic environments. Additionally, it has a wide range of API libraries to communicate with the simulator through different programming languages [42]. For this project, a simulated scenario was built composed of different types of furniture used daily in a domestic environment. We used the flight stabilization controller provided by the simulator in order to focus on the execution of the commands through voice directions. The experimental scenario is illustrated in Figure 3.

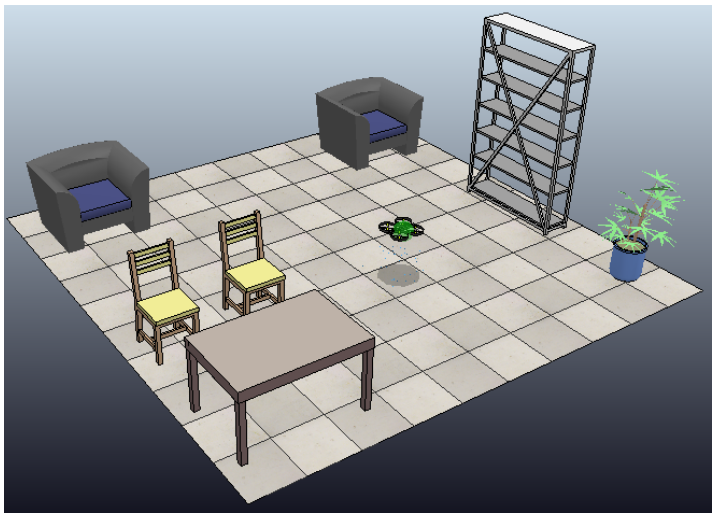


Figure 3. The simulated domestic environment in V-REP with a quadrotor and furniture used daily, such as sofas, chairs, a table, a shelf, and a plant.

In our scenario, once an instruction was given to the drone, it was executed continuously until another action was instructed. Therefore, to stop the vehicle, it was necessary to provide explicit instructions for the action "stop". The only exception to the previous rule was the execution of the action "down", which could be automatically stopped in case the drone reached 0.5 m distance from the ground. In such a case, the movement was stopped to avoid a collision. All nine possible actions defined in the simulation scenario are shown in Table 1. The instructions can be given using both languages: Spanish and English.

Table 1. Description of allowable commands to produce an action to control the UAV.

No.	Action Classes	Description
1	Up	Increase the UAV's altitude
2	Down	Decrease the UAV's altitude
3	Go right	Move the UAV to the right
4	Go left	Move the UAV to the left
5	Go forward	Move the UAV forward
6	Go back	Move the UAV backward
7	Turn right	Turn the UAV 90° clockwise
8	Turn left	Turn the UAV 90° counterclockwise
9	Stop	Stop the UAV

For the architecture implementation, the programming language Python was used and connected to the simulator through the V-REP API, in order to pass the instructions between the automatic speech recognition algorithm and the simulator. As mentioned, words and phrases may be uttered by the users in two languages, Spanish and English. The selection and benefits of using these languages were

twofold. On the one hand, the mother tongue of participants in the experimental stage was Spanish. Therefore, on the other hand, since English is used globally, in this study, it was necessary to conduct a comparison of both Spanish and English for accuracy.

Each action class had more than one way to execute a movement, e.g., the action "down" can be carried out by saying the word "baja" or the sentence "disminuir altura" in Spanish, or also in the form "go down" or simply "down" in English. It is important to note that not all phrases were necessarily grammatically correct either in Spanish or in English. As a result, we did not assume here all the time that an end-user would give an instruction using grammatically correct sentences. Furthermore, it is widely acknowledged that on many occasions, the spoken language is less structured. Therefore, it lacks formality since the users did not follow grammar rules. In this regard, we defined a domain-based dictionary comprising 48 sentences belonging to the nine action classes. It is important to note that the classes "go" and "turn" were differentiated since the former moved the drone to the left or right in x, y coordinates maintaining the drone's orientation, and the latter changed the drone's yaw angle by 90° clockwise or counterclockwise.

The experiments were run in a computer with the following characteristics: Intel Core i7-8750H processor, 8GB DDR4 2666 MHz RAM, NVIDIA GeForce GTX 1050Ti with 4GB of GDDR5, and Windows 10 Home. The Internet connection used was an optical fiber with a 300/100 Mbps download/upload speed.

5. Results

In this section, we present the main results we obtained by testing the proposed algorithm. In our experiments, apart from testing with online instructions uttered by different people, we also used recordings from diverse locations, such as open spaces, offices, and classrooms. Recordings presented an averaged signal to noise ratio (SNR) of -3.09×10^{-4} dB, showing a slightly better ratio for sentences in Spanish. This may also be attributed to the native language of the participants. The SNR values are shown in Table 2 for each action class in both languages.

Table 2. Raw input SNRs (dB) for each action class in both Spanish and English language.

Class	English	Spanish	Average
Up	-4.21×10^{-4}	5.33×10^{-4}	5.57×10^{-5}
Down	-9.06×10^{-4}	-2.37×10^{-5}	-4.65×10^{-4}
Go Right	-6.93×10^{-4}	-2.09×10^{-4}	-4.51×10^{-4}
Go Left	-8.03×10^{-4}	-3.16×10^{-5}	-4.18×10^{-4}
Go Forward	-3.85×10^{-4}	-1.36×10^{-4}	-2.60×10^{-4}
Go Back	-6.86×10^{-4}	-9.70×10^{-6}	-3.48×10^{-4}
Turn Left	-9.54×10^{-4}	-5.55×10^{-5}	-5.05×10^{-4}
Turn Right	-7.79×10^{-4}	2.68×10^{-4}	-2.55×10^{-4}
Stop	-2.94×10^{-4}	3.02×10^{-5}	-1.32×10^{-4}
Average	-6.58×10^{-4}	4.06×10^{-5}	-3.09×10^{-4}

To determine the accuracy of the proposed algorithm, tests in two languages with and without phoneme matching were executed using three different setups, i.e., raw input, 5% noisy input, and 15% noisy input. The two noisy setups were added to test the robustness of the algorithm in the presence of noise and included uniform noise $n_1 = 0.05$ (uniformly distributed $U(-n_1, n_1)$) and $n_2 = 0.15$ (uniformly distributed $U(-n_2, n_2)$) equivalent to 5% and 15% with respect to the original raw input. For each setup, each action class was performed 15 times for each language. Therefore, each class was called a total of 30 times, 15 for English and 15 for Spanish. Overall, 270 instructions were tested for each setup, 135 for each language. A total of 5 people participated in this experimental test. Although we were aware that the number of participants was rather small, we were still able to draw significant conclusions for future experiments. Additionally, this research included people from different age

groups, ranging from 19 years old to 56 years old (mean $M = 35.4$, standard deviation $SD = 18.45$, 3 women, 2 men).

Figure 4 illustrates the accuracy obtained using English and Spanish instructions for all levels of noise. Figure 4a–c show the accuracy without using phoneme matching, i.e., the algorithm compared the text received from GCS directly to our domain-based dictionary trying to find an exact coincidence. Otherwise, it was not recognized or labeled as "no class". When phoneme matching was not used, a considerable accuracy difference occurred between Spanish and English commands, with the former presenting the highest recognition values. In this regard, the users instructing in Spanish, i.e., their native language, achieved better action recognition in comparison to English commands, likely due to accent and pronunciation differences in speaking the words in a foreign language. Figure 4d–f demonstrate the recognition accuracy obtained using the domain-based language for phoneme matching. When using phoneme matching, the difference in the recognition achieved between both languages was attenuated by our algorithm which looks for the most similar instruction to classify the audio input.

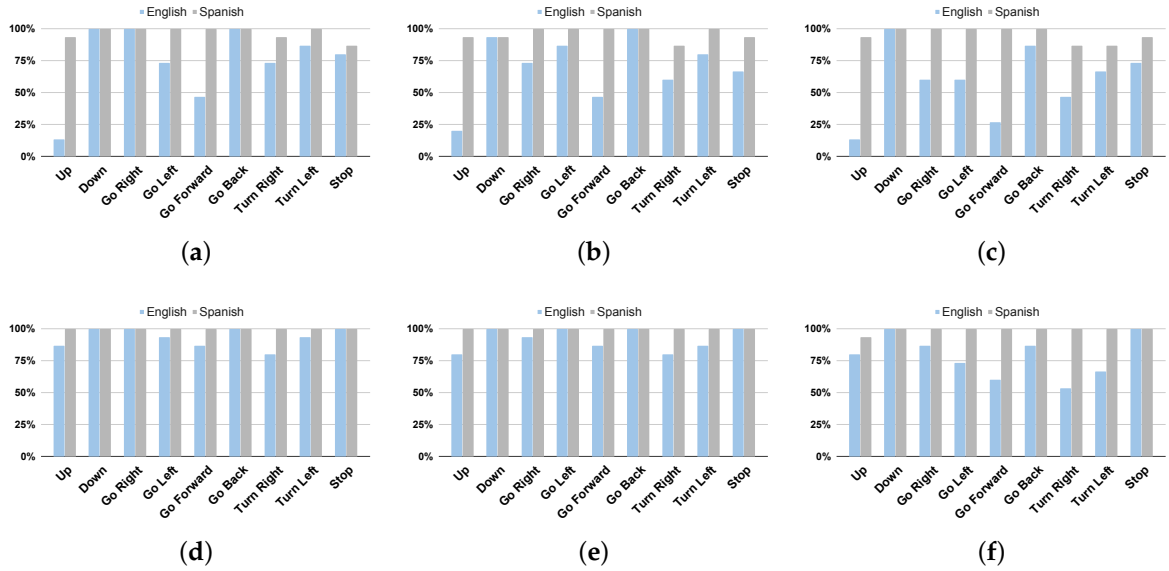


Figure 4. Average recognition accuracy for each action class in Spanish and English languages with different levels of noise in the input signal. Without using phoneme matching, the text received from the cloud-based service was directly transferred to the scenario. This implementation showed a considerable difference between languages due to the user's native language. Using phoneme matching, the text received from the cloud-based service was compared to the instructions within the domain-based dictionary. The use of phoneme matching demonstrated an improvement in speech-to-action recognition for both languages, decreasing the difference of accuracy between them. (**a**) Raw input, no phoneme matching; (**b**) Noise 5%, no phoneme matching; (**c**) Noise 15%, no phoneme matching; (**d**) Raw input, with phoneme matching; (**e**) Noise 5%, with phoneme matching; (**f**) Noise 15%, with phoneme matching.

In terms of noisy inputs, as mentioned, we performed experiments using the raw input, 5% noisy input, and 15% noisy input. Figure 4a,d illustrate the results obtained without and with the phoneme matching technique when using the raw audio input. When no phoneme matching was applied, the algorithm recognized 232 out of 270 instructions using both languages, achieving 85.93% accuracy in voice-to-action recognition. In particular, the use of Spanish language achieved 97.04% accuracy, while the use of English reached 74.81% accuracy. However, when phoneme matching was used, the algorithm considerably improved the recognition accuracy for both languages achieving 96.67% accuracy. While the use of Spanish achieved 100.00% accuracy, the recognition of

English commands significantly improved in comparison to the non-phoneme-matching approach, reaching 93.33% accuracy.

To test the robustness of the proposed method, we applied 5% of noise to the audio input. The results obtained without and with phoneme matching are portrayed in Figure 4b,e respectively. On average without applying phoneme matching, the algorithm recognized 214 out of 270 instructions considering both languages, achieving an accuracy of 82.96% in voice-to-action recognition. In particular, Spanish instructions achieved 96.30% accuracy, while English instructions were 69.63% accurate. Using phoneme matching, the algorithm accomplished 95.93% accuracy, i.e., 100.00% accuracy for Spanish commands and 91.85% accuracy for English commands. When comparing the recognition accuracy with a 5% noisy input to the raw input, the results obtained were just slightly worse, especially when phoneme matching was used. This demonstrated the robustness of the proposed approach in the presence of noisy audio inputs.

Finally, we used an audio input signal with 15% noise. The results are shown in Figure 4c,f without and with phoneme matching, respectively. Without applying phoneme matching, the algorithm recognized 198 out of 270 instructions, achieving 77.41% accuracy in speech-to-action recognition on average for both languages. The use of Spanish instructions accomplished 95.56% accuracy, while English instruction was 59.26% accurate. When phoneme matching was introduced into this setup, the algorithm accomplished 88.89% accuracy, i.e., 99.26% accuracy for Spanish instructions and 78.52% accuracy for English instructions. Although similar to the previous case, the introduction of noise affected the recognition accuracy obtained. This was expected due to the input signal distortion. The use of phoneme matching mitigated this issue considerably. The mitigation of the recognition accuracy fall was especially important considering that the use of English was a foreign language for the participants for the experiments. This resulted in defective utterances or mispronounced instructions. Table 3 summarizes the aforementioned results for all the setups with both approaches.

Table 3. Audio recognition accuracy obtained with and without phoneme matching using both Spanish and English languages.

Approach	Language	Raw Input	Noise 5%	Noise 15%
No phoneme matching	Spanish	97.04%	96.30%	95.56%
	English	74.81%	69.63%	59.26%
	Both	85.93%	82.96%	77.41%
With phoneme matching	Spanish	100.00%	100.00%	99.26%
	English	93.33%	91.85%	78.52%
	Both	96.67%	95.93%	88.89%

Figure 5a,b illustrate the system performance as boxplots for English and Spanish instructions respectively. The boxes are grouped considering six sets, i.e., raw inputs with no phoneme matching (NPM), raw inputs with phoneme matching (WPM), 5% noisy inputs with no phoneme matching (NPM ~5%), 5% noisy inputs with phoneme matching (WPM ~5%), 15% noisy inputs with no phoneme matching (NPM ~15%), and 15% noisy inputs with phoneme matching (WPM ~15%). The use of English instructions resulted in greater variability among the participants during the experiments due to the participants' native language, as previously pointed out. Although using Spanish commands obtained better results overall, the phoneme matching technique improved the automatic speech recognition for the proposed scenario using either English or Spanish instructions.

Figure 6 depicts the confusion matrices for the recognition of class actions using English instructions in all the experimental setups. When no phoneme matching was used, the label "no class" referred to no coincidence between the hypothesis obtained from GCS and the instructions within the domain-based language. Results obtained demonstrated many instances in which the hypothesis did not match any sentence in the dictionary, leading to a misclassification of the instruction. The implementation of phoneme matching, i.e., the algorithm computing the distance between the

hypothesis received from GCS and each instruction in the domain-based dictionary, led to a better action class recognition. The improvement was achieved for all commands that the proposed approach could use independently of the user's language ability. Moreover, Figure 7 illustrates the confusion matrices for the recognition of class actions for Spanish instructions in all the experimental setups. In this regard, when the individual's native language was used, fewer misclassification occurred in comparison to English instructions were articulated. This remained true even when noisier audio signal was emitted.

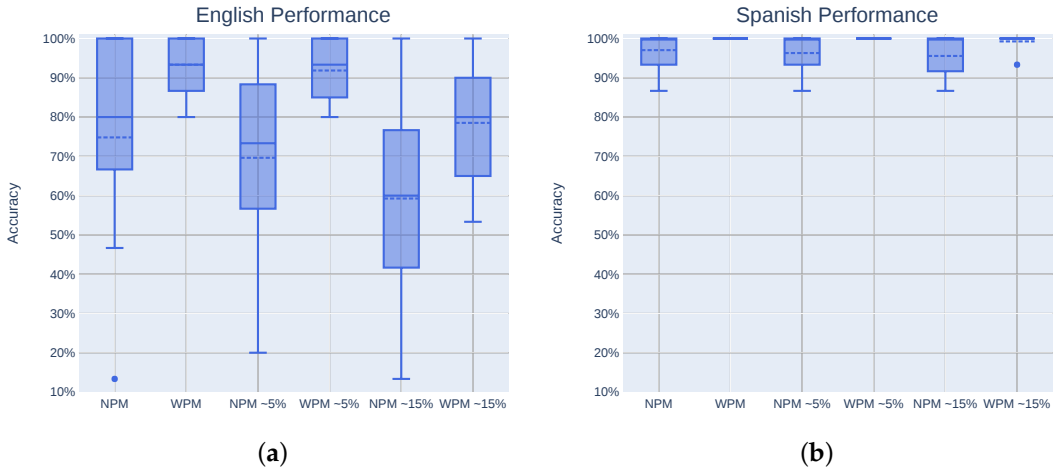


Figure 5. Audio recognition accuracy for all the experimental setups using both languages. NPM and WPM stand for no phoneme matching and with phoneme matching, respectively, and the percentage beside the approach in x-axis represents the noise value of each setup. Continuous and segmented lines represent the median and mean values in each box. The use of phoneme matching significantly improved the speech-to-action recognition even in the presence of noisy inputs. (**a**) Recognition accuracy using English instructions; (**b**) Recognition accuracy using Spanish instructions.

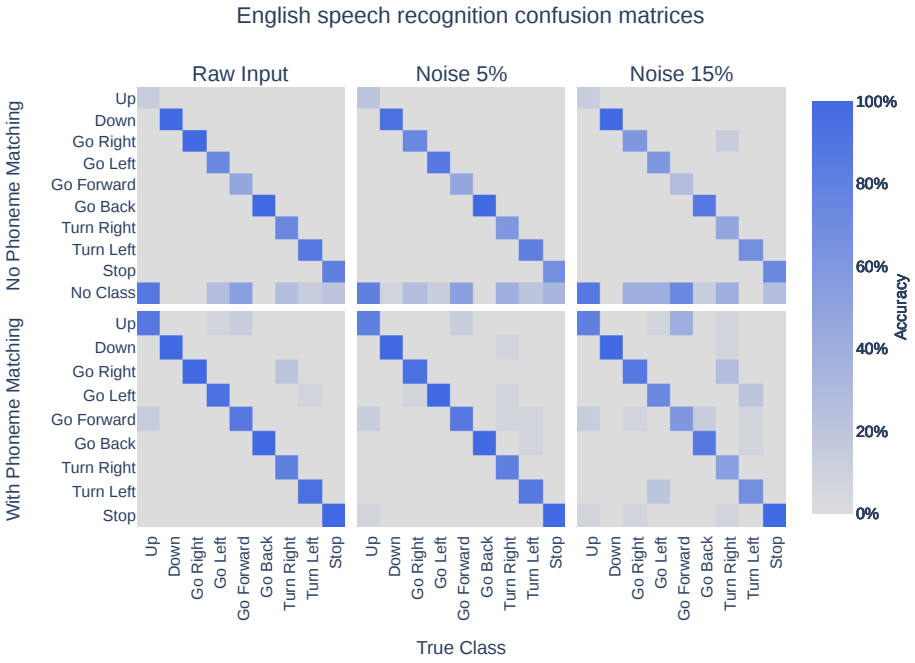


Figure 6. Predicted and true class distribution for each experimental setup using English instructions. Phoneme matching improved the overall results obtaining fewer misclassifications for all the action classes. Although a noisy input impoverished the action class classification in both approaches, the use of phoneme matching allowed for better recognition accuracy for all levels of noise.

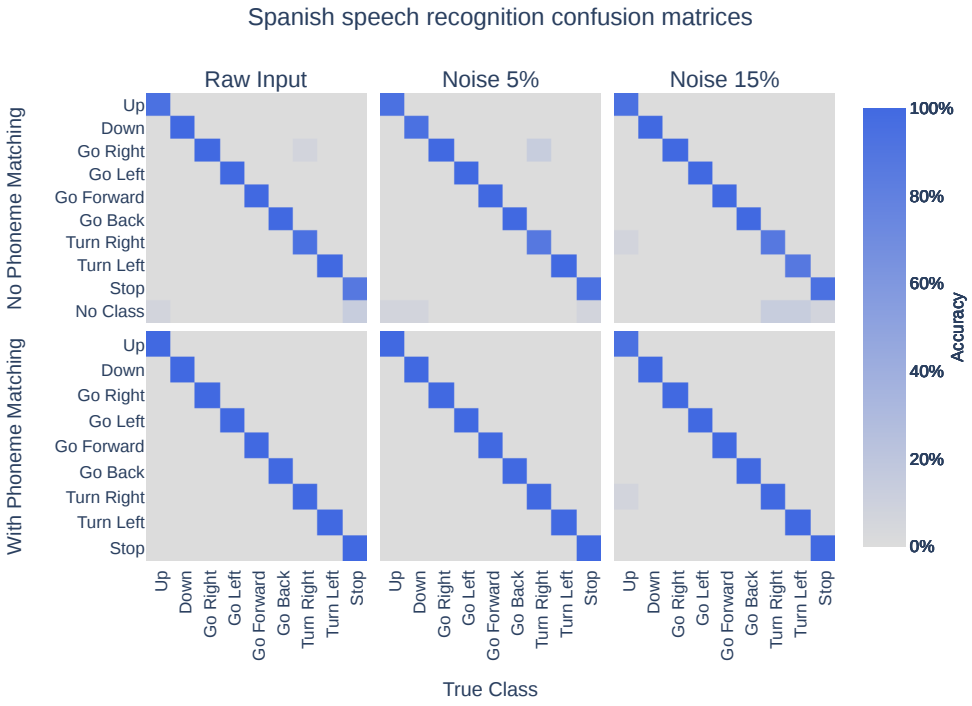


Figure 7. Predicted and true class distribution for each experimental setup using Spanish instructions. When Spanish language was used, fewer errors in action classification occurred in comparison to English instructions. Nevertheless, the phoneme matching still allowed better recognition accuracy for all levels of noise in comparison to the approach not using it.

6. Conclusions

In this work, we presented an architecture to control a simulated drone through voice commands interpreted via a cloud-based automatic speech recognition system and a domain-based language dictionary. The use of phoneme matching improved the level of accuracy in instruction recognition. Raw inputs without phoneme matching resulted in 97.04% and 74.81% accuracy in action recognition for Spanish and English, respectively. On average, voice command recognition without phoneme matching achieved 85.93% accuracy. After testing the speech recognition method complemented by a domain-based language to operate the UAV in a domestic environment, better results were obtained. Overall, performance in instruction recognition improved with phoneme matching, obtaining 93.33% and 100.00% accuracy for English and Spanish, respectively. On average, we obtained 96.67% accuracy when the instructions were interpreted using phoneme matching. Moreover, we tested our approach with 5% and 15% noise in the input. In general, by using phoneme matching, our method achieved good results showing the robustness of the proposed algorithm against noise.

In conclusion, the algorithm obtained high accuracy when interpreting instructions given by an end-user through speech. The interpretation in Spanish provided better results. The main reason Spanish interpretation results were significant was that the people involved in the experiments are all native Spanish speakers. However, we found that phoneme matching improved voice-to-action recognition, reducing the gap between languages and obtaining similar results for native Spanish language users.

Although at this stage, our approach presented some limitations, such as issues with network interruptions or collisions with obstacles, as well as conducting the experiments in a simulated environment and thus controlling noise variables, we obtained results that indicate the need to extend this research into several other areas. For instance, a more extensive dictionary of instructions needs to be developed as well as adding recognition in more languages. Moreover, an important next step is to transfer the proposed approach to a real-world scenario where some variables may not be easily

controlled. In this regard, this study was the initial stage of a larger project, where we are currently developing deep reinforcement learning algorithms using interactive feedback to teach an agent how to operate a drone. Future research would also take into account multi-modal sensory inputs as well as a combination of policy and reward shaping for the interactive feedback approach.

Author Contributions: Conceptualization, R.C. and F.C.; Funding acquisition, R.C. and F.C.; Investigation, R.C.; Methodology, R.C. and A.A.; Supervision, F.C.; Writing—original draft, R.C.; Writing—review & editing, A.A. and F.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was financed in part by Universidad Central de Chile under the research project CIP2018009, the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior—Brasil (CAPES)—Finance Code 001, Fundação de Amparo a Ciência e Tecnologia do Estado de Pernambuco (FACEPE), and Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq)—Brazilian research agencies.

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

References

1. Kardasz, P.; Doskocz, J.; Hejduk, M.; Wiejkut, P.; Zarzycki, H. Drones and possibilities of their using. *J. Civ. Environ. Eng.* **2016**, *6*, 1–7. [CrossRef]
2. Seymour, A.; Dale, J.; Hammill, M.; Halpin, P.; Johnston, D. Automated detection and enumeration of marine wildlife using unmanned aircraft systems (UAS) and thermal imagery. *Sci. Rep.* **2017**, *7*, 1–10. [CrossRef] [PubMed]
3. Géza Károly, K.L.; Tokody, D. Radiofrequency Identification by using Drones in Railway Accidents and Disaster Situations. *Interdiscip. Descr. Complex Syst.* **2017**, *15*, 114–132.
4. Fernandez, R.A.S.; Sanchez-Lopez, J.L.; Sampedro, C.; Bavle, H.; Molina, M.; Campoy, P. Natural user interfaces for human-drone multi-modal interaction. In Proceedings of the 2016 International Conference on Unmanned Aircraft Systems (ICUAS), Arlington, VA, USA, 7–10 June 2016; IEEE: Arlington, VA, USA, 2016; pp. 1013–1022.
5. Schalkwyk, J.; Beeferman, D.; Beaufays, F.; Byrne, B.; Chelba, C.; Cohen, M.; Kamvar, M.; Strope, B. "Your word is my command": Google search by voice: A case study. In *Advances in Speech Recognition. Mobile Environments, Call Centers and Clinics*; Springer Science: New York, NY, USA, 2010; pp. 61–90.
6. Adorf, J. *Web Speech API*; Technical Report; KTH Royal Institute of Technology: Stockholm, Sweden, 2013.
7. Twiefel, J.; Baumann, T.; Heinrich, S.; Wermter, S. Improving domain-independent cloud-based Speech recognition with domain-dependent phonetic post-processing. In Proceedings of the Association for the Advancement of Artificial Intelligence Conference AAAI, Quebec City, QC, Canada, 27–31 July 2014; AAAI Press: Quebec City, QC, Canada, 2014; pp. 1529–1535.
8. Cruz, F.; Parisi, G.I.; Wermter, S. Learning contextual affordances with an associative neural architecture. In Proceedings of the European Symposium on Artificial Neural Network, Computational Intelligence and Machine Learning ESANN, UCLouvain, Bruges, Belgium, 27–29 April 2016; pp. 665–670.
9. Cruz, F.; Wüppen, P.; Magg, S.; Fazrie, A.; Wermter, S. Agent-advising approaches in an interactive reinforcement learning scenario. In Proceedings of the Joint IEEE International Conference on Development and Learning and Epigenetic Robotics ICDL-EpiRob, Lisboa, Portugal, 18–21 September 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 209–214.
10. Cruz, F.; Wüppen, P.; Fazrie, A.; Weber, C.; Wermter, S. Action Selection Methods in a Robotic Reinforcement Learning Scenario. In Proceedings of the 2018 IEEE Latin American Conference on Computational Intelligence (LA-CCI), Gudalajara, Mexico, 7–9 November 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 13–18.
11. Moreira, I.; Rivas, J.; Cruz, F.; Dazeley, R.; Ayala, A.; Fernandes, B. Deep Reinforcement Learning with Interactive Feedback in a Human–Robot Environment. *Appl. Sci.* **2020**, *10*, 5574. [CrossRef]
12. Cruz, F.; Dazeley, R.; Vamplew, P. Explainable robotic systems: Understanding goal-driven actions in a reinforcement learning scenario. *arXiv* **2020**, arXiv:2006.13615.
13. Rohmer, E.; Singh, S.P.; Freese, M. V-REP: A versatile and scalable robot simulation framework. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, Tokyo, Japan, 3–7 November 2013; IEEE: Tokyo, Japan, 2013; pp. 1321–1326.
14. Boyle, M.J. The race for drones. *Orbis* **2015**, *59*, 76–94. [CrossRef]

15. Marshall, D.M.; Barnhart, R.K.; Hottman, S.B.; Shappee, E.; Most, M.T. *Introduction to Unmanned Aircraft Systems*; CRC Press: Boca Raton, FL, USA, 2016.
16. Muchiri, N.; Kimathi, S. A review of applications and potential applications of UAV. In Proceedings of the Sustainable Research and Innovation Conference, Nairobi, Kenya, 4–6 May 2016; Open Journal Systems: Nairobi, Kenya, 2016; pp. 280–283.
17. Amin, R.; Aijun, L.; Shamshirband, S. A review of quadrotor UAV: Control methodologies and performance evaluation. *Int. J. Autom. Control* **2016**, *10*, 87–103. [CrossRef]
18. Clough, B. Metrics, Schmetrics! How Do You Track a UAV's Autonomy? In Proceedings of the 1st UAV Conference, Portsmouth, VA, USA, 20–23 May 2002; p. 3499.
19. Peng, Z.; Li, B.; Chen, X.; Wu, J. Online route planning for UAV based on model predictive control and particle swarm optimization algorithm. In Proceedings of the 10th World Congress on Intelligent Control and Automation, Beijing, China, 6–8 July 2012; pp. 397–401.
20. Al-Madani, B.; Svirskis, M.; Narvydas, G.; Maskeliūnas, R.; Damaševičius, R. Design of Fully Automatic Drone Parachute System with Temperature Compensation Mechanism for Civilian and Military Applications. *J. Adv. Transp.* **2018**, 1–11. [CrossRef]
21. Ivanovas, A.; Ostreika, A.; Maskeliūnas, R.; Damaševičius, R.; Połap, D.; Woźniak, M. Block matching based obstacle avoidance for unmanned aerial vehicle. In Proceedings of the International Conference on Artificial Intelligence and Soft Computing, Zakopane, Poland, 3–7 June 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 58–69.
22. Pham, H.X.; La, H.M.; Feil-Seifer, D.; Nguyen, L.V. Autonomous UAV navigation using reinforcement learning. *arXiv* **2018**, arXiv:1801.05086.
23. Shiri, H.; Park, J.; Bennis, M. Remote UAV Online Path Planning via Neural Network-Based Opportunistic Control. *IEEE Wirel. Commun. Lett.* **2020**, *9*, 861–865. [CrossRef]
24. Kusyk, J.; Uyar, M.U.; Ma, K.; Samoylov, E.; Valdez, R.; Plishka, J.; Hoque, S.E.; Bertoli, G.; Boksiner, J. Artificial intelligence and game theory controlled autonomous UAV swarms. *Evol. Intell.* **2020**, 1–18. [CrossRef]
25. Chen, H.; Wang, X.; Li, Y. A Survey of Autonomous Control for UAV. In Proceedings of the 2009 International Conference on Artificial Intelligence and Computational Intelligence, Shanghai, China, 7–8 November 2009; Volume 2, pp. 267–271.
26. Quigley, M.; Goodrich, M.A.; Beard, R.W. Semi-autonomous human-UAV interfaces for fixed-wing mini-UAVs. In Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Sendai, Japan, 28 September–2 October 2004; IEEE: Chicago, IL, USA, 2004; Volume 3, pp. 2457–2462.
27. Wopereis, H.W.; Fumagalli, M.; Stramigioli, S.; Carloni, R. Bilateral human-robot control for semi-autonomous UAV navigation. In Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, 28 September–3 October 2015; pp. 5234–5240.
28. Perez-Grau, F.J.; Ragel, R.; Caballero, F.; Viguria, A.; Ollero, A. Semi-autonomous teleoperation of UAVs in search and rescue scenarios. In Proceedings of the 2017 International Conference on Unmanned Aircraft Systems (ICUAS), Miami, FL, USA, 13–16 June 2017; pp. 1066–1074.
29. Imdoukh, A.; Shaker, A.; Al-Toukhy, A.; Kablaoui, D.; El-Abd, M. Semi-autonomous indoor firefighting UAV. In Proceedings of the 2017 18th International Conference on Advanced Robotics (ICAR), Hong Kong, China, 10–12 July 2017; pp. 310–315.
30. Sanders, B.; Shen, Y.; Vincenzi, D. Design and Validation of a Unity-Based Simulation to Investigate Gesture Based Control of Semi-autonomous Vehicles. In Proceedings of the International Conference on Human-Computer Interaction, Copenhagen, Denmark, 19–24 July 2020; Springer: Berlin/Heidelberg, Germany, 2020, pp. 325–345.
31. Wuth, J.; Correa, P.; Núñez, T.; Saavedra, M.; Yoma, N.B. The Role of Speech Technology in User Perception and Context Acquisition in HRI. *Int. J. Soc. Robot.* **2020**, 1–20. [CrossRef]
32. Lavrynenko, O.; Konakhovych, G.; Bakhtiiarov, D. Method of voice control functions of the UAV. In Proceedings of the 2016 IEEE 4th International Conference on Methods and Systems of Navigation and Motion Control (MSNMC), Kyiv, Ukraine, 18–20 October 2016; IEEE: Kyiv, Ukraine, 2016; pp. 47–50.
33. Fayjie, A.R.; Ramezani, A.; Oualid, D.; Lee, D.J. Voice enabled smart drone control. In Proceedings of the 2017 Ninth International Conference on Ubiquitous and Future Networks (ICUFN), Milan, Italy, 4–7 July 2017; IEEE: Milan, Italy, 2017; pp. 119–121.

34. Landau, M.; van Delden, S. A System Architecture for Hands-Free UAV Drone Control Using Intuitive Voice Commands. In Proceedings of the Companion of the 2017 ACM/IEEE International Conference on Human-Robot Interaction, Vienna, Austria, 6–9 March 2017; Association for Computing Machinery: New York, NY, USA, 2017; HRI '17, p. 181–182. [CrossRef]
35. Chandarana, M.; Meszaros, E.L.; Trujillo, A.; Allen, B.D. 'Fly Like This': Natural Language Interface for UAV Mission Planning. In Proceedings of the 10th International Conference on Advances in Computer-Human Interactions (ACHI 2017), Nice, France, 19–23 March 2017; IARIA XPS Press: Nice, France, 2017; pp. 40–46.
36. Jones, G.; Berthouze, N.; Bielski, R.; Julier, S. Towards a situated, multimodal interface for multiple UAV control. In Proceedings of the 2010 IEEE International Conference on Robotics and Automation, Anchorage, AK, USA, 3–8 May 2010; IEEE: Anchorage, AK, USA, 2010; pp. 1739–1744.
37. Lavrynenko, O.; Taranenko, A.; Machalin, I.; Gabrousenko, Y.; Terentyeva, I.; Bakhtiiarov, D. Protected Voice Control System of UAV. In Proceedings of the 2019 IEEE 5th International Conference Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD), Kyiv, Ukraine, 22–24 October 2019; IEEE: Kyiv, Ukraine, 2019; pp. 295–298.
38. López, G.; Quesada, L.; Guerrero, L.A. Alexa vs. Siri vs. Cortana vs. Google Assistant: A comparison of speech-based natural user interfaces. In Proceedings of the International Conference on Applied Human Factors and Ergonomics, Los Angeles, CA, USA, 17–21 July 2017; Springer: Los Angeles, CA, USA, 2017; pp. 241–250.
39. Glonek, G.; Pietruszka, M. Natural user interfaces (NUI). *J. Appl. Comput. Sci.* **2012**, *20*, 27–45.
40. Cruz, F.; Twiefel, J.; Magg, S.; Weber, C.; Wermter, S. Interactive reinforcement learning through speech guidance in a domestic scenario. In Proceedings of the 2015 International Joint Conference on Neural Networks (IJCNN), Killarney, Ireland, 12–16 July 2015; IEEE: Killarney, Ireland, 2015; pp. 1341–1348.
41. Levenshtein, V.I. Binary codes capable of correcting deletions, insertions, and reversals. *Sov. Phys. Dokl.* **1966**, *10*, 707–710.
42. Ayala, A.; Cruz, F.; Campos, D.; Rubio, R.; Fernandes, B.; Dazeley, R. A Comparison of Humanoid Robot Simulators: A Quantitative Approach. In Proceedings of the IEEE International Joint Conference on Development and Learning and Epigenetic Robotics ICDL-EpiRob, Valparaiso, Chile, 26–30 October 2020; p. 6.