Deep Learning-Based Unmanned Aerial Vehicle Control with Hand Gesture and Computer Vision

Conference Paper · May 2022

DOI: 10.23919/ASCC56756.2022.9828347

6 authors, including:

Farhat Naseer
National University of Sciences and Technology

Muhammad Aleem Siddiqui
Prince Sattam bin Abdulaziz University

Muhammad Jawad
University of Lahore

Keum-Shik Hong
Pusan National University

Deep Learning-Based Unmanned Aerial Vehicle Control with Hand Gesture and Computer Vision

Farhat Naseer
School of Mechanical and Manufacturing Engineering, National University of Science and Technology, Islamabad, Pakistan
farhaty16@gmail.com

Ghufran Ullah
School of Mechanical and Manufacturing Engineering, National University of Science and Technology, Islamabad, Pakistan
rasst.ghufran@nust.edu.pk

Muhammad Aleem Siddiqui
School of Mechanical and Manufacturing Engineering, National University of Science and Technology, Islamabad, Pakistan
rasst.aleem@ncai.nust.edu.pk

Muhammad Jawad Khan
School of Mechanical and Manufacturing Engineering, National University of Science and Technology, Islamabad, Pakistan
jawad.khan@smme.nust.edu.pk

Keum-Shik Hong
School of Mechanical Engineering, Pusan National University, Busan 46241, Republic of Korea
kshong@pusan.ac.kr

Noman Naseer
Department of Mechatronics Engineering, Air University, Islamabad, Pakistan
noman.naseer@mail.au.edu.pk

***Abstract*— Human drone Interaction is the subfield of Human-robot interaction (HRI), which deals with human interaction with drones; Drones are conventionally controlled by joysticks, onboard computers, mobile applications, and remote controllers. Drones controlled by conventional methods are affected by electromagnetic wave interference due to an unreliable connection between the drone and the controller. Special care is needed when working with drones in close vicinity to humans. In this proposed work, we have developed an agentless and non-wearable communication system between the drone and the user. The proposed system consists of two modules, i.e., the hand recognition module and drone controlling module. The hand recognition model is further subdivided into two modules, i.e., hand detection and gesture recognition. After the landmarks (palm and hand detection) are detected, they are given to the TensorFlow-based Deep learning model for classification among different hand gestures. The classified hand movement is further converted into actionable drone movements such as take-off, hovering, and landing. Experimentation was done on Parrot Mambo Drone, and it was concluded that there should be no more than 121.92cm distance between user and webcam for working system effectively.**

Keywords*— Human Drone Interaction (HDI), Computer Vision, Hand gesture, Deep Learning

I. Introduction

Human-Robot Interaction (HRI) is a field of study which focuses on understanding, designing, and evaluating robotic systems for use by or with humans. The applications of HRI include everything from assisting the elderly, to assisting severely disabled patients, to entertainment in amusement parks [1]. Where the definition of interaction means communication between humans and robots. There are different forms of communication between humans and robots. These forms depend upon the closeness of proximity or vicinity between the human and the robots. Hence interaction and communication can be divided into two categories [2].

- Proximate interaction
- Remote interaction

This work was supported in part by the Korea Institute of Energy Technology Evaluation and Planning under the Ministry of Trade, Industry and Energy, Korea (grant no. 20213030020160) and in part by the National Research Foundation of Korea under the Ministry of Science and ICT, Korea (grant no. 2020R1A2B5B03096000).

1. In the first category, the human is physically separated from the robot at a minimum distance of one or two meters.

2. In the second case, the robot and humans are physically in two locations [3].

Here in this paper, our primary focus will be on remote interaction and its application for controlling the drone. The application of remote interaction of drones with humans is so vast that it has led to the development of a new field known as Human drone interaction (HDI); HDI is the sub-field of HRI that focuses on designing, understanding, and evaluating drones' interaction with Humans, with distinct features of drones to locomote independently in 3D space [4]. The current research in HDI consists of assessing and flourishing new modalities for controlling drones, creating robust human drone communication, estimating the interaction distance between the drone and the human, and creating new applications and use cases of drones. These four significant applications are shown in Fig. 1. Often HDI is achieved by using hand gestures or body posture.

An external module such as a wearable device or an unintrusive system is usually required to achieve HDI. For flight control based on hand movement, the pose of the hand is directly mapped to the angular (roll, pitch, and yaw) and linear motions (x, y, z). Research on Glove based modalities has been carried out since the 1990s to collect hand pose information. Besides glove-based devices, Electromyography (EMG) sensors are also used for hand pose detection. In [5] and [6], the authors have made Natural User Interface (NUI) by utilizing the signals of EMG from the forearm. In [7] and [8], the authors investigated the haptic feedback response to the pilot. Another study shows that Flyjacket from [9] and [10] has been used to collect the body gestures by implementing the inertial measurement unit on the pilot's bodies. All the wearable systems provide natural piloting, but they have to be worn and are intrusive and hence cause physical constraints to the user.

In [11], authors have developed the Brain-computer interface (BCI) system to control the drone, which uses electroencephalography–functional near-infrared spectroscopy (EEG–fNIRS) scheme for decoding eight active commands generated from the frontal brain region to control the drone, where the system shows an average accuracy of 75.6% with fNIRS for decoding of four commands and 86%

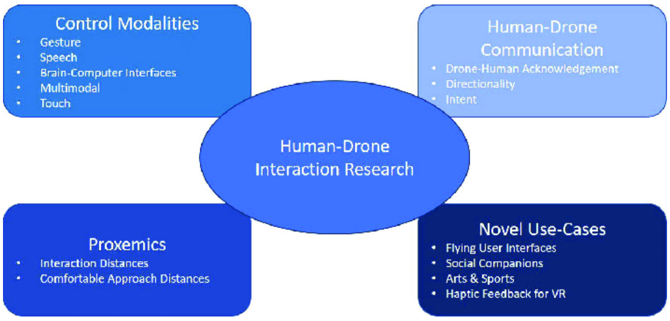

Fig. 1. The four major fields of human drone interaction research

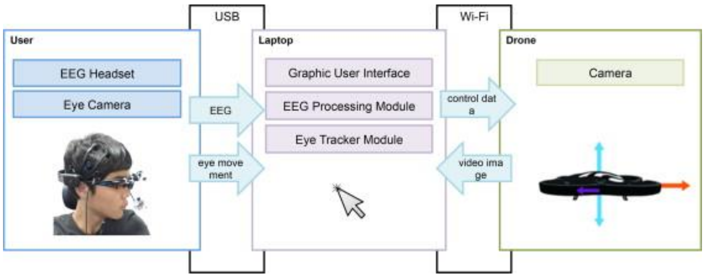

Fig. 2. EEG based hybrid drone controller system

accuracy for EEG for the rest of the four commands. Similar work has been done in [12] using multimodal BCI the accuracy obtained for complex flight tasks was reported as 86.5%.

In [13], authors have developed a gaze control system for unmanned aerial vehicles (or drones) in which ten participants have volunteered to perform the flying task of an unmanned aerial vehicle. During experimentation of the system, four control modes (M1-Altitude and translation by gaze; speed and rotation by keyboard, M2- Altitude and rotation by gaze; speed and translation by keyboard, M3- Speed and translation by gaze; Altitude and rotation by keyboard, M4- Speed and rotation by gaze; altitude and translation by keyboard) which were a combination of x-y gaze movement and manual input from the keyboard for controlling roll, pitch and yaw movements for the drone control were tested. It was concluded that the participants with prior gaming experience could control the drone without much training. In [14], the authors have developed a drone control system for a wheelchair-bound person, where the system is controlled via EEG signal in combination with eye blinking and off-shelf components. The system described by the authors was called the "Flying buddy" system and was intended to increase the visual view of the motor disabled person who cannot see far objects.

In [15], another EEG-based eye-tracking-based controller was designed; the controller could control the drone's movement in 8 different directions. The system setup described in [12] is shown in Fig. 2. The controller was tested on five healthy subjects, and the average BCI classification accuracy for the controller was 91.67%.

However, the systems described in [11] require high attention from the user, which will cause strain on the eyes and head. The maximum concentration level of an average human is no more than 10 to 15 minutes [16], and it will not be possible for the user to control the drone remotely during long flights.

Non-wearable and unintrusive systems which overcome these physical constraints include vision systems. Successful research has been done to make the single-camera embedded drone. In [17] and [18], the authors have done the projects using hand gesture piloting. However, the results from this solution were not very accurate and efficient on hand gestures. The same problem of inaccurate and inefficient detection was for body piloting, so it was concluded that gesture recognition depends upon the quality of the onboard video camera [19]. This also infers that the drone should fly close to the vicinity of the hand to detect it correctly; this puts constraints of use on the drone [19]. So, to increase the accuracy of hand detection, the research focus was shifted to depth cameras. So, in 2012 a technology named Leap motion controller (LMC) was specially developed for hand recognition [20]. The LMC gives high precision with low latency, and in [21], the authors have used it for developing a NUI system for drone piloting, where LMC has certain constraints, such as the system should have to be located under the hands and in the closed vicinity as detection range is limited to 80cm. Microsoft developed the Kinect sensor in 2012, and it is widely used for this purpose. It is precise and can perform whole-body movement recognition in real-time [22], and its Software Development Kit (SDK) supports easy development. In [23] and [24], both authors have used Kinect-based NUI drone control in their work, wherein [23] the authors have shown in their work that Kinect-based NUI has higher precision as compared to the controller. Still, the average mission completion time of Kinect is twice that of the controller. Both [23] and [24] have not covered the other functionalities (instruction menu to guide the drone to take a photo or video or flight speed control) to control the drone altogether.

Different approaches have been taken for mapping the body/ hand gesture to drone control commands during NUI design. In [25], the authors have performed the study, and its focus was to see 3D spatial interaction techniques for drones. Many analogous methods were used, such as using arms instead of the controller sticks or asking them to control the drone as though they were holding it in their hands. A study was conducted with 14 participants with different backgrounds and experiences, where the participants were asked to complete the flight mission and score each flight control technique. The scoring criteria were: Comfortable, Natural, Fun, Confusion, Easy, Like, Expectation, and Frustration. This experiment's results highlight each technique's strengths and weaknesses, which helped determine the preferred technique to imitate an aircraft. The motion of the upper body is lined with the drone's motion, which seems easy to understand and natural. However, there have not been many studies in the field where the gestures have been investigated to the same level of detail, and gesture vocabulary that assists all functions and features needed to control the drone successfully would line up the research efforts across the field. In [26], the authors developed an algorithm that tracks and saves the coordinates of the initial trajectory along with altitudes for a fail-safe drone mode, whereas the approach used by [26] is based on thresholding.

The current research focuses on the piloting aspect of the drone. The novelty of this research comes from the framework of drone control on a dataset to detect hand gestures. More functionalities like fail-safe mode using hand gesture detection would significantly contribute to the current research.

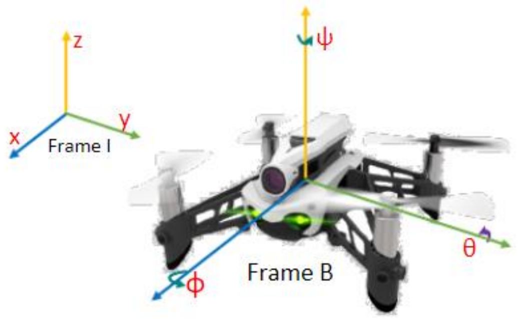

Fig. 3. Parrot Mambo drone with coordinate frames.

In this paper, we have developed a non-invasive hand gesture-based drone controller, which can be used as a fail-safe mode that enables the drone control, i.e., Take-off and land when a sudden loss of communication happens between the drone and the controller. We have used the Parrot Mambo drone for our experiment.

The remainder of the paper is ordered as follows; In section 2, we will talk about the System Description, section 3 about Methodology, section 4 about Architecture, sections 5 and 6 on results and discussion, section 7 on the conclusion, and section 8 on future work.

II. System Description

For experiments Parrot Mambo drone is used; the Parrot was initially founded in 1994 in France, and now it is famous for its wireless products [27]; the first drone produced by the company was in 2016. The Mambo Drone used for experimentation weighs 2.22g, and its dimensions are 5.2 x 5.2 x 1.6 inches. The flight time of the drone is 9 minutes. The drone has an accelerometer, ultrasonic sensor, pressure sensor, gyroscope, and a downward-facing camera [28]. The Mambo drone, along with its global and body frame, is shown in Fig. 3.

All gesture detection and drone control processing has been done on the Lenovo Idea pad 320, core i5 7th generation. A webcam on the laptop is used, having a resolution of 720p. The range for accurate hand detection is 121.92cm.

III. Methodology

The system consists of two modules, i.e., hand gesture recognition and drone control.

A. Hand recognition

The module uses the MediaPipe framework, Tensor flow, and OpenCV in python for hand detection and gesture recognition.

MediaPipe is an open-source, lightweight, and customizable machine learning framework created by Google. The MediaPipe framework has an extensive collection of hand detection and tracking models, which are trained on a massive and diverse dataset of Google. The MediaPipe has pre-trained models for pose estimation and hand detection. The node's edges, skeleton, and landmarks detect the different critical points of the hand. The hand tracking Algorithm of MediaPipe consists of two models that work dependently, i.e., the Palm

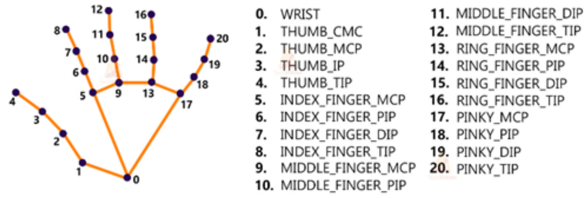

Fig. 4. MediaPipe landmarks for detection of hand

detection model and the Land Mark model. The palm detection model gives a perfectly cropped image of the palm and then gives this image to the landmark model. This process diminishes data augmentation, which is required in deep learning-based landmark Localization.

In the MediaPipe model, first, the video stream is taken using a webcam then the MediaPipe framework is used to recognize 21 critical points on the hand [29]. The key points are labeled in Fig.4.

These key points are then used to detect the hand. Further, the detected hand, along with the key points, are given to the tensor flow-based gesture recognition model trained on our dataset for detecting hand pose. The parameters are set to detect a single hand in each frame to avoid confusion in commands sent for drone control. The trained model can recognize two hand gestures, i.e., Take off and land.

B. Drone control

The drone controller consists of two parts, i.e., server and client. A connection must be established between the system and the drone to control the drone using the detected hand gesture. The connection is established via the system, Bluetooth.

Socket programming is used to build the connection between the server and the client, where the server acts as a master while the client act as a slave. The server is responsible for controlling the video stream, the hand recognition part, and the client is responsible for sending the commands to the drone controller. The hand gesture predictor gives the instructions to the drone controller. The hand gesture and drone controller are two separate nodes communicating via sockets. There is a delay of 3 seconds between the processing and execution of instructions sent by the hand gesture predictor and its execution on the drone controller. The generalized setup is shown in Fig. 5.

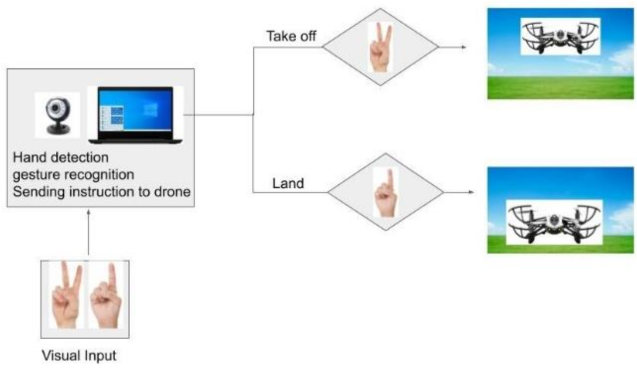

Fig. 5. Generalized diagram of the system

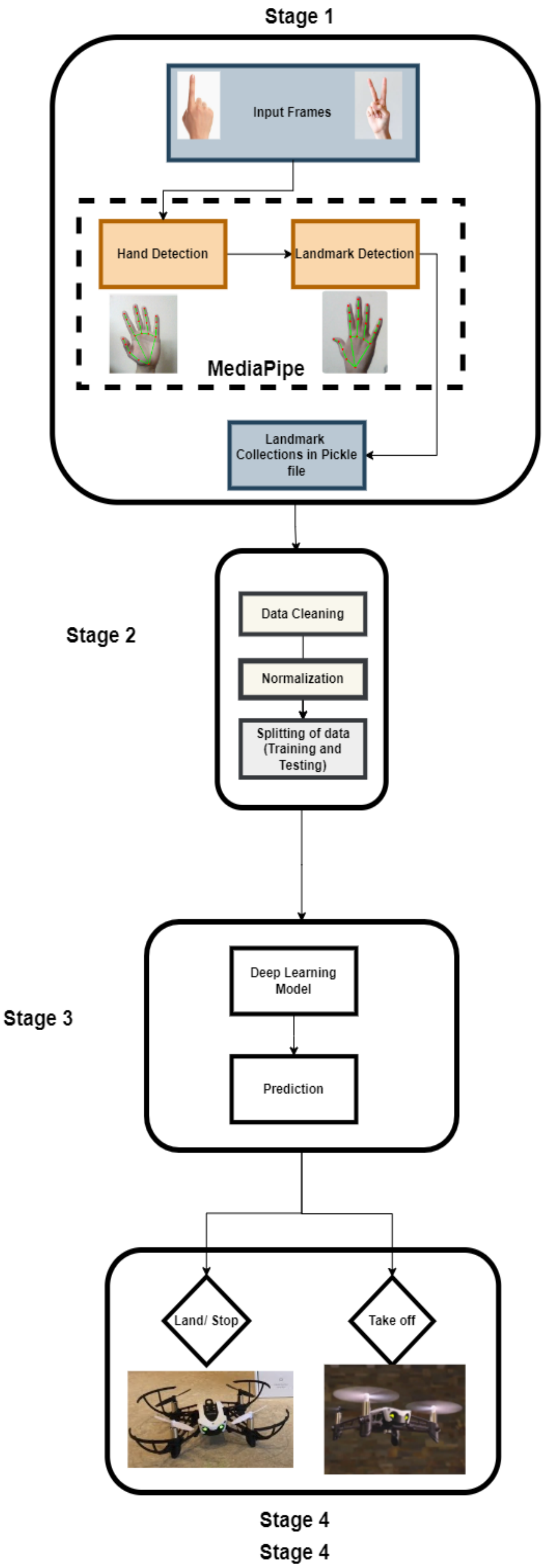

Fig. 6. Architecture of drone controller.

TABLE I. COMMANDS AND THEIR DESCRIPTION

Gestures	Commands	Action detail
Victory sign	Take off	Drone takeoff
Open index finger with the rest of the hand close	Stop	Drone landing

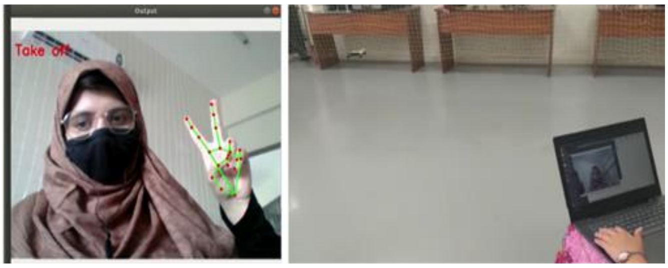

Fig. 7. Take off

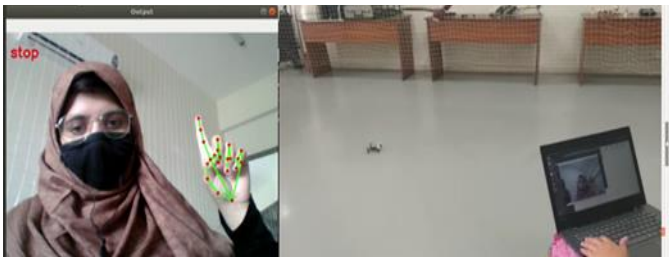

Fig. 8. Land

C. *Quantitative Analysis*

To measure the accuracy of the system, the ratio is taken between the correctly classified gestures to the total number of the same gestures.

IV. Architecture

The Architecture for the drone controller is demonstrated in Fig. 6. The code for drone control and hand detection algorithm with the trained data model has been released on GitHub [15] for other researchers.

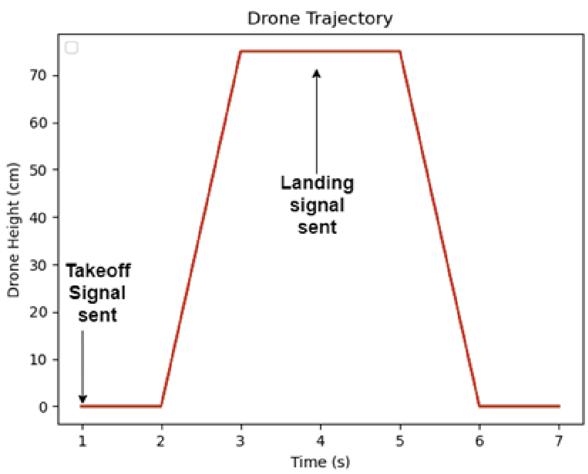

Fig. 9. Drone trajectory

TABLE II. EXPERIMENTAL RESULTS OF GESTURE DETECTED BY SYSTEM

No. of trials	Hand gesture	Correct Detection	Incorrect Detection	Accuracy
40	Takeoff	36	4	90%
40	Stop	38	4	95%

TABLE III. EXPERIMENTAL RESULTS OF THE COMMANDS EXECUTED BY DRONE

No. of trials	Hand gesture	Correct Detection	Incorrect Detection	Accuracy
40	Takeoff	34	8	85%
40	Stop	36	4	90%

V. RESULTS

We have successfully implemented the two commands for controlling the drone using hand gestures. The results are shown below. Fig. 7 shows a Drone takeoff with the Victory sign, while Fig. 8 shows a drone landing with the index finger.

Fig. 6 demonstrates the drone trajectory; as the command is sent to the drone from the master, the client takes 3 seconds to process the command; the increasing curve from 2 to 3 seconds indicates the change in altitude and velocity, whereas the flat curve from 3 to 4 seconds indicates no change in Fig. 6. Drone stop (landing) velocity, i.e., Hovering state, the decreasing curve from 4 to 6 indicates the drone's landing. On average, the 3 seconds were taken by the client to process each command, i.e., take off or land.

VI. DISCUSSION

The user has to sit at a minimum distance of 20 cm away from the webcam for proper detection of hands and the maximum allowable range to sit away from the camera for proper hand detection is 121.92cm. For testing the model's accuracy, 40 trials were done in simulation for gesture detection. The model predicts 32 times correct results for taking and 36 times correct detection for stop (landing). After hand detection, the same experiment was done for 40 trials with the drone, i.e., take off and stop (landing) commands were sent to the client from the master. The client performs 31 operations correctly for landing, while for landing, the correctly performed operations are 36. The reason behind the detection of inaccurate Take-off operations is a delay in establishing the connection between the drone and the controller. The client misses some of the Take-off instructions.

The other reason for the inaccurate detection of hand gestures and command execution is varying light conditions. The tensor flow model was only trained for indoor light conditions, and hence it gives false-positive results when tested in outdoor environments or low light conditions.

VII. CONCLUSION

We have successfully developed a hand gesture-based drone controller for Mambo. The model is currently trained on the indoor data set. The controller can perform Takeoff, hovering, and stop (landing) operations on the drone. The delay between the execution of each command is 3s, the system's accuracy for hand gesture detection is 95%, and for command, execution is 90%.

VIII. FUTURE WORK

In the future, the model can be trained on different data sets with varying light conditions, and work can be done to decrease the delay between executing each command. Another problem that can be addressed in the future is that the camera-based solution performs poorly in a Foggy environment. This problem can be solved by adding a non-visual sensor to the drone. Currently, there is a delay with the commands sent to the drones and drone flight. This delay could be mitigated with further optimizations in the drone software development kit.

The system can be further extended by modifying the controller to control the drone so that it can move and perform multiple commands using hand gestures.

ACKNOWLEDGMENT

The research was funded by the Aerial Robotics Lab, Sines, and the National Institute of Science and Technology, SMME.

REFERENCES

[1] D. J. Bruemmer and M. S. Swinson, “Humanoid Robots,” Encycl. Phys. Sci. Technol., pp. 401–425, 2003, doi: 10.1016/B0-12-227410-5/00317-3.

[2] “1: Introduction | Human-Robot Interaction.” https://humanrobotinteraction.org/1-introduction/ (accessed May 27, 2022).

[3] C. L. Gittens, “Remote HRI: a Methodology for Maintaining COVID-19 Physical Distancing and Human Interaction Requirements in HRI Studies,” Inf. Syst. Front., pp. 1–16, Aug. 2021, doi: 10.1007/S10796-021-10162-4/TABLES/5.

[4] D. Tezza and M. Andujar, “The State-of-the-Art of Human-Drone Interaction: A Survey,” IEEE Access, vol. 7, pp. 167438–167454, 2019, doi: 10.1109/ACCESS.2019.2953900.

[5] A. Stoica, F. Salvioli, and C. Flowers, “Remote Control of Quadrotor Teams, Using Hand Gestures,” doi: 10.1145/2559636.2559853.

[6] Y. Doshi and D. Nath, “Designing a Drone Controller using Electromyography Signals,” Proc. - Int. Conf. Commun. Inf. Comput. Technol. ICCICT 2021, 2021, doi: 10.1109/ICCICT50803.2021.9510045.

[7] “Humphrey II – Ars Electronica Futurelab.” https://ars.electronica.art/futurelab/en/projects-humphrey-2/ (accessed Feb. 14, 2022).

[8] A. Cherpillod, D. Floreano, and S. Mintchev, “Embodied Flight with a Drone,” Proc. - 3rd IEEE Int. Conf. Robot. Comput. IRC 2019, pp. 386–390, Mar. 2019, doi: 10.1109/IRC.2019.00070.

[9] C. Rognon, S. Mintchev, F. Dellagnola, A. Cherpillod, D. Atienza, and D. Floreano, “FlyJacket: An Upper Body Soft Exoskeleton for Immersive Drone Control,” IEEE Robot. Autom. Lett., vol. 3, no. 3, pp. 2362–2369, Jul. 2018, doi: 10.1109/LRA.2018.2810955.

[10] “FlyJacket Lets You Control a Drone With Your Body - IEEE Spectrum.” https://spectrum.ieee.org/epfl-flyjacket-exosuit-lets-you-control-a-drone-with-your-body (accessed Jun. 03, 2022).

[11] M. J. Khan and K. S. Hong, “Hybrid EEG-FNIRS-based eight-command decoding for BCI: Application to quadcopter control,” Front. Neurorobot., vol. 11, no. FEB, p. 6, Feb. 2017, doi: 10.3389/FNBOT.2017.00006/BIBTEX.

[12] X. Duan, S. Xie, X. Xie, Y. Meng, and Z. Xu, “Quadcopter Flight Control Using a Non-invasive Multi-Modal Brain Computer Interface,” Front. Neurorobot., vol. 13, p. 23, 2019, doi: 10.3389/FNBOT.2019.00023/BIBTEX.

[13] J. P. Hansen, A. Alapetite, I. S. MacKenzie, and E. Møllenbach, “The use of gaze to control drones,” Eye Track. Res. Appl. Symp., pp. 27–34, 2014, doi: 10.1145/2578153.2578156.

[14] Y. Yu et al., “FlyingBuddy2: A brain-controlled assistant for the handicapped,” UbiComp’12 - Proc. 2012 ACM Conf. Ubiquitous Comput., pp. 669–670, 2012, doi: 10.1145/2370216.2370359.

[15] B. H. Kim, M. Kim, and S. Jo, “Quadcopter flight control using a low-cost hybrid interface with EEG-based classification and eye tracking,”

Comput. Biol. Med., vol. 51, pp. 82–92, Aug. 2014, doi: 10.1016/J.COMPBIOMED.2014.04.020.

[16] N. A. Bradbury, “Attention span during lectures: 8 seconds, 10 minutes, or more?,” https://doi.org/10.1152/advan.00109.2016, vol. 40, no. 4, pp. 509–513, 2016, doi: 10.1152/ADVAN.00109.2016.

[17] M. Hasanuzzaman, V. Ampornaramveth, T. Zhang, M. A. Bhuiyan, Y. Shirai, and H. Ueno, “Real-time vision-based gesture recognition for human robot interaction,” Proc. - 2004 IEEE Int. Conf. Robot. Biomimetics, IEEE ROBIO 2004, pp. 413–418, 2004, doi: 10.1109/ROBIO.2004.1521814.

[18] K. Natarajan, T. H. D. Nguyen, and M. Mete, “Hand Gesture Controlled Drones: An Open Source Library,” Proc. - 2018 1st Int. Conf. Data Intell. Secur. ICDIS 2018, pp. 168–175, Mar. 2018, doi: 10.1109/ICDIS.2018.00035.

[19] T. Sun, S. Nie, D. Y. Yeung, and S. Shen, “Gesture-based piloting of an aerial robot using monocular vision,” Proc. - IEEE Int. Conf. Robot. Autom., pp. 5913–5920, Jul. 2017, doi: 10.1109/ICRA.2017.7989696.

[20] “Leap Motion Controller TM,” Accessed: Feb. 14, 2022. [Online]. Available: https://www.ultraleap.com/product/vr-developer-mount/.

[21] A. Sarkar, K. A. Patel, R. K. G. Ram, and G. K. Capoor, “Gesture control of drone using a motion controller,” 2016 Int. Conf. Ind. Informatics Comput. Syst. CIICS 2016, Apr. 2016, doi: 10.1109/ICCSII.2016.7462401.

[22] J. Shotton et al., “Real-time human pose recognition in parts from single depth images,” Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., pp. 1297–1304, 2011, doi: 10.1109/CVPR.2011.5995316.

[23] A. Sanna, F. Lamberti, G. Paravati, and F. Manuri, “A Kinect-based natural interface for quadrotor control,” Entertain. Comput., vol. 4, no. 3, pp. 179–186, Aug. 2013, doi: 10.1016/J.ENTCOM.2013.01.001.

[24] A. Mashood, H. Noura, I. Jawhar, and N. Mohamed, “A gesture based kinect for quadrotor control,” 2015 Int. Conf. Inf. Commun. Technol. Res. ICTRC 2015, pp. 298–301, Jul. 2015, doi: 10.1109/ICTRC.2015.7156481.

[25] K. P. Pfeil, S. L. Koh, and J. J. LaViola, “Exploring 3D gesture metaphors for interaction with unmanned aerial vehicles,” Int. Conf. Intell. User Interfaces, Proc. IUI, pp. 257–266, Mar. 2013, doi: 10.1145/2449396.2449429.

[26] D. Cho, “A Study on a Flight Safe System in Unmanned Aerial Vehicles,” Int. J. Appl. Eng. Res., vol. 13, no. 9, pp. 7128–7130, 2018, Accessed: Feb. 14, 2022. [Online]. Available: http://www.ripublication.com7128.

[27] “Parrot SA - Wikipedia.” https://en.wikipedia.org/wiki/Parrot_SA (accessed May 27, 2022).

[28] M. R. Kaplan, A. Eraslan, A. Beke, and T. Kumbasar, “Altitude and Position Control of Parrot Mambo Minidrone with PID and Fuzzy PID Controllers,” ELECO 2019 - 11th Int. Conf. Electr. Electron. Eng., pp. 785–789, Nov. 2019, doi: 10.23919/ELECO47770.2019.8990445.

[29] “Hands - mediapipe.” https://google.github.io/mediapipe/solutions/hands (accessed May 27, 2022).