JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Drone navigation through Gesture Recognition using Deep learning

Tanmay Mamdapurkar[1], Shweta Nakate[2], Anand Tajne[3] , S. P. Dhanure[4], P.K.Suryawanshi[5] , P.V.Igave[6]

Dept. of E&TC, SKNCOE, Pune-411041, India

[1]mamdapurkar. tanmay 7@gmail.com
[2]shwetanakate3171@gmail.com
[3]anandtajne500@gmail.com
[4]sudhir.dhanure_skncoe@sinhgad.com
preetisuryawanshi10@gmail.com
pradnya.renke@gmail.com

***Abstract* — Drones are rapidly becoming such a component of everyday life. Drone technology is gradually displacing the bulk among the most challenging and high-paying occupations in several areas, including transportation, agriculture, and architecture. Parallel to this, a worldwide economy for commercial services utilizing drone technology is steadily developing. Drone utilization is fast increasing, and the same is true of human contact with drones. As a consequence, developing realistic human-drone engagement strategies has now become a significant problem. This survey paper has been effectively useful in outlining the current methodologies for the purpose of understanding the image processing based implementations for gesture recognition. These approaches have been valuable in determining the usefulness and drawbacks for operating a drone using computer vision based gesture recognition. This has led to the realization of a gesture based drone navigation system through the use of Convolutional Neural Networks, Neural network estimates the gestures to feed the control to the Drone to control based on the maximized identified gesture process.**

Keywords: *Drone Navigation, Gesture recognition, image processing and Convolutional Neural Networks, DJ Tello Drone.*

I INTRODUCTION

Unmanned aerial vehicles are currently widely employed for a multitude of applications across the world, involving monitoring, cinematography, and aerial filmmaking. In many circumstances, a competent operator is required to execute these jobs using drone that tends to be prohibitively expensive. A basic gesture controller may greatly simplify the work of navigating. The gesture control of an aerial drone is the subject of this article. Anybody action or condition, specifically any arm or facial movement, is referred to as a gesture. Since this Unmanned Aerial Vehicle, or drone, is now commonplace in our everyday lives and has a wide range of uses, practicality has now become a major concern, merely because piloting a drone is not a simple task for a rookie.

Even though there have been remotely operated cars for a long time, remote control seems to have been a primary user interface connecting the user and the drone since about the beginning. Nevertheless, as is widely known, becoming an accomplished drone pilot necessitates a significant amount of time. For example, a notable racing drone tournament has attracted an increasing viewership, and agility in piloting a drone utilizing a remote control or joystick has shown to be a useful attribute.

Robots have now become common in both residential and professional settings. As a result, novel human-robot interfaces for controlling and interacting with robots are in high desire. Current HRIs, which include touch screens, keyboards, and joysticks, are seldom straightforward or comfortable to operate since they need learning and focus while in use. This restricts the usage of robots to highly qualified specialists in many circumstances. Drones are undoubtedly the extremely fast form of device in both professional and personal situations due to potential capacity to augment human awareness and capacity of motion in new ways.

Delivering, transportation, military applications, agricultural, aerial reconnaissance, and search-and-rescue are among of the uses. Drones cooperate with users in many of these domains by increasing human space awareness and offering information that would not have been attainable from a terrestrial viewpoint. Using joysticks or remote controls for drone teleoperation, on the other hand, is a non-intuitive and difficult process it also becomes mentally taxing over lengthy engagements. The creation of more intuitive control systems might minimize mistakes and enhance flying efficiency while also permitting users to move their focus away from controlling the drone and toward evaluating the data it provides.

However, even though many people were expecting a much more straightforward method to engage with the drone, researchers in the Human-Computer Interaction and Computer Vision fields have developed various natural methods for interacting with the UAV in a human-friendly manner. Smart sensors are frequently used in such situations. Kinect seems to have been a popular sensor amongst the many because of its versatility and low cost, as well as the open source software that is accessible. In particular, for pretty much the same reasons, the Leap Motion sensor is beneficial in controlling the UAV. This type of system, however, has several limitations. Because such a device necessitates a PC to identify the user's hand movements and the operational range between the individual and the sensor is very small, the drone operator must be near the ground terminal and cannot operate the drone whilst distant from all of this.

When contrasted with the conventional ways of piloting and commanding drones via remote controls, joysticks, and other equipment, body-machine interfaces have several benefits. BMIs are more straightforward, and they required less preparation on aggregate to attain the very same degree of drone management by the pilot. Human and Robot Interface has piqued the attention of many scholars in a variety of fields, including engineering, social studies, and perhaps even psychology, making it an interdisciplinary idea. Drones, which have been recognized to have a difficult control challenge, are one sort of robotic device that may be categorized into several types. Drones have recently gained popularity in a variety of industries, including delivering mail boxes, thanks to advancements in both hardware and software of just such a robotic device.

Amazon is the forerunner of the aforementioned programme for transporting mail boxes to neighboring locations, which is currently under work. Controlling these kind of robots is a difficult task. Individuals often operate them using a joystick or a smartphone. However, controlling them at a professional rate is incredibly challenging. The required mobility of the drone is conveyed by use of a Joystick, which may be considered the initial alternative transferred through the head when it was originally designed and from a functional aspect. However, with the advancement of vision technology, an option for operating drones has been offered, namely, controlling a drone through hand movement and gesture. HRI can be viewed as a stereotypical depiction of the latter.

This Research paper segregates the section 2 for the evaluation of the past work in the configuration of a literature survey. Section 3 concentrating on narrating the proposed work, on the other hand proposed work is evaluated in section 4 as results and Discussions. Finally, section 5 provides the conclusion and the future work.

1. II RELATED WORKS

L. L. Cheng et al. [1] utilized MATLAB to determine all optimum parameters required for ArDrone location mapping. Therefore, all real-time signals are sent into the ROS platform, and the output commands from ROS are sent directly to the ARDrone over the Wi-Fi channel. This enables this low-cost drone to locate and navigate in previously unknown environments or outdoors in the absence of a GPS connection. The authors have combined this drone with leap motion, which permits them to control the flight using the ROS platform in Ubuntu. To accomplish this purpose, the SLAM system is utilized to construct the map, and the PTAM (Parallel Tracking and Mapping) method is utilized to locate the robot. The extended Kalman filter is utilized for posture estimation, and the PID (Proportional-integral-differential) controller is employed to regulate the drone's pose, allowing the robot to approach the target location swiftly and correctly.

A. Sarkar et al. were able to move the Parrot AR DRONE using hand motion with the aid of the LEAP Motion Controller. Any hand motion elicits a response from the drone, which moves appropriately. The authors were able to turn the drone using a hand motion that was identified by the two stereo cameras and three infrared LEDs. Therefore, they can infer that with the assistance of the Leap Motion Controller, they can utilize the AR DRONE to execute a variety of jobs such as aerial filming and acrobatic feats, to mention a few. By modifying the Python scripts and adding more functionality, the Leap may be trained to recognize new hand motions and movements [2].

Y. Yu et al. provide a unique way for operating an unmanned aerial vehicle (UAV), by utilizing hand gestures. The authors addressed the hand recognition problem using hand position information and obtained positive experimental results. They created an algorithm to identify several operators to pave the way for commanding UAV swarms in the following stage. Finally, they completed the entire system under the ROS system and obtained the required performance [3]. The experiment results show that utilizing hand gestures to control a UAV is a simple and successful way. Hand gesture recognition, multi-operator recognition, and UAV control are all possible.

A. A. Bandala et al. [4] provide a UAV control technique that depends on hand gestures that do not need the use of any wearable gadget. Control motions were related to certain tasks. The control system's efficacy and usefulness are built on a camera-based hand gesture capture system that continually broadcasts the user's hand profile. This, together with the UAV's altitude and location data, is sent into a PID controller. The system's implementation is a two-player drone game akin to laser tag. The obvious success of controlling a drone with a hand gesture may be stretched further by expanding the number of drones that a single person can manage.

Deep learning algorithms are being utilized by B. Hu [5] in this research to recognize dynamic hand gestures. The study's engineering goal is to improve UAV control. To discover the highest performing neural network, he created two fully connected neural networks and one convolutional neural network. For neural network training and testing, he constructed two data models. He developed a deep learning neural network-based software system. He believes this is the first study to employ Leap Motion Controllers as input devices in a deep learning network-depended hand gesture detection system.

K. Haratiannejadi et al. [6] proposed a multirotor aerial vehicle control system that depends on gestures. The introduced system includes three independent validation modules: (1) gesture classifier validation, which may retrain the model as needed depending on new data from a new user, and (2) flight control validation, which causes the flight controller to be adjusted. As a third validation, they incorporate human-in-the-loop. Furthermore, the system uses two human-robot interface systems: a glove that generates control instructions and an image processing framework that generates discrete commands based on hand tracking and gesture identification.

S. Shin et al. offer a novel human-drone interface based on wearables for commanding the drone [7]. It features two modes: one for controlling the drone with directional commands and another for generating varied figural trajectories with a series of hand posture motions. In contrast to prior studies on natural interfaces for operating drones, which used Kinect, Leap, or Myo, the authors employ a convenient wearable device linked to a palm to do the same duties. One of the benefits of the current technology is that it works well without any connection to a PC, allowing the user to roam freely while operating the drone. Furthermore, because the hand posture motions are built in a highly straightforward manner, the user can simply understand and manoeuvre the drone following his goal.

A. Menshchikov et al. demonstrated an AI-based system for intuitive drone control that ensures the recognition of speech and gesture commands. The system depends on a low-power embedded device that incorporates artificial intelligence. The neural networks were trained on a desktop computer, which was not utilized to operate the drone. The Convolutional Neural Network for gesture capture and the Recurrent Neural Network for speech capture have both been shown to work on a low-power embedded device with AI [8]. Their findings indicate that on a battery-powered embedded system, a wide range of real-time control approaches, such as control via a variety of gestures and voice commands, may be done without the use of a video camera.

A novel drone control framework depending on the SSD Deep Neural Network and stereo vision has been suggested by H. Ghasemi et al.. The Ego Hands dataset was used to train the deep learning algorithm to recognize the hand, and the key originality of the suggested strategy was employing this algorithm instead of basic image processing or leap motion approaches, which are susceptible to point of view and background clutter [9]. Another distinguishing feature of this technique was the use of a stereo camera setup rather than a Kinect. The proposed technique was tested and the results were utilized to mimic the drone in the ROS and Gazebo environments.

The FlyJacket exosuit demonstrated by C. Rognon et al. solves the problem of piloting a fixed-wing drone in a natural and immersive manner while staying portable and adaptable to different morphologies. The complete system fits into a backpack and may be carried in the field. Arm fatigue is reduced with the use of simple but effective arm support that does not interfere with the user's performance [10]. The findings of the experiment demonstrate that participants found the jacket to be comfortable, natural, and intuitive and that they were able to easily fly a virtual drone while wearing it. The discomfort was mild, and participants said in the free comments section of the survey that they only felt dizzy due to the virtual reality goggles, a well-known symptom that is unrelated to the exosuit.

R. Ibrahimov et al. suggested a unique DroneLight system that enables real-world human-drone interaction and drone-mediated communication. Depending on motions produced by a user using the designed glove, a micro-quadcopter is outfitted with an RGB LED array of light-paints letters or designs in midair. A data set of five letters from a single user was gathered for gesture recognition [11]. For gesture recognition, an ML-based algorithm was created. The created algorithm's recognition rate yielded high scores. The DroneLight system's real-world implementation was also tested and confirmed.

FlyCam is a multi-touch camera view manipulation framework for gimbal-equipped drones, according to H. Kang et al. The introduction framework replaces typical RC for drone controls by reducing low-level aircraft controls and gimbal operations to six easy and intuitive multi-touch actions. A single finger drag rotates the drone and the camera, a double finger drag propels the drone forward or backward along the camera's optical axis, and a single/double touch hold propels the drone forward or backward along the camera's optical axis [12]. The dragging distance on the screen or the tapping pressure affect the speed of the drone's activities. When using a drone camera, direct manipulation of the camera view, rather than drone and gimbal operations, significantly minimizes the difficulty of the photo composition process.

Drone.io is a revolutionary input-output interface system for human-drone interaction, as described by J. R. Cauchard et al. [13]. Drone.io is a drone-based, totally transportable system. It can identify gestures and adjust projected output in real-time to match the user's actions. The authors show that the system is simple to use, pleasurable to use, and extremely dependable when used outside in near-to real-world settings in three user trials with 27 individuals. Users may now go up to a helper drone and see for themselves what the drone can do for them, as well as make requests for assistance, using drone.io. Drone.io has an easy-to-use UI that doesn't require any prior knowledge. They have the chance to develop a new generation of ambient and semi-public displays with drone.io that are not constrained by the infrastructure in which they are created.

III PROPOSED WORK

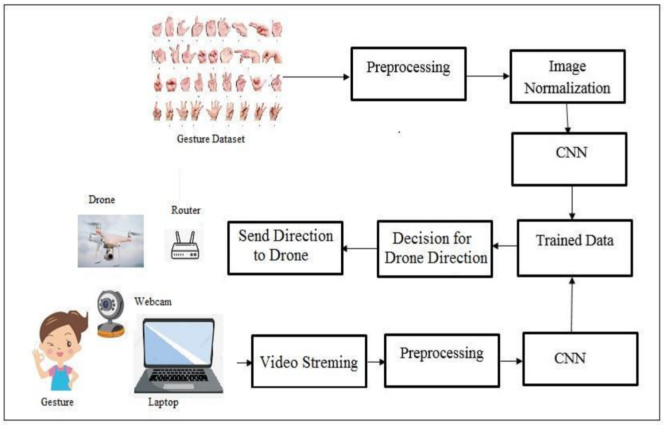

Figure 1: Proposed Work Overview Diagram

The proposed system designed for the purpose of handling Drone through the hand gestures is depicted in the figure 1. The deployed model's steps are described in detail in the below mentioned points.

Step 1: Dataset preparation- This is the basic step of the proposed work, where a synthetic dataset is being prepared for the different gestures through which a Drone can be controlled. The gestures are Back, blank, down, front, Land, left, right and Up. To capture these gestures opencv object CV2 is used using python programming language. Around 800 images for each gestures of training and 500 images for each gestures of testing is collected to form the dataset to train the system. While collecting the images these gesture images are resized to the dimension of 96 × 96. Then these images are converted into BGR-YCBCR model to capture only skin pixels. Once the skin is identified, then the whole gesture image is converted into black and white. Then, only gestures are retained in black color and rest is in white color. The captured images are stored in the respective locations with the incremental file name orders for the specific mentioned gestures.

Step 2: Pre-Processing – This is the initial step of the pre-training the images using the Convolution neural network. In this step the number of the train images and test images is set as 6400 Train images which form 8 Gestures and 4000 images for testing respectively. A batch size of 64 is set for 250 Epochs. After this process an imageData Generator object is created using the Keras and tensorflow libraries of Python Programming language for the rescaling factor 1:255. Then a dimension scale is set for the both train and test images as 96 X 96 along with their respective paths and finally color mode is set as 'grayscale' with class mode as 'categorical' for the 8 mentioned gestures. After this pre-process process now the test and train images are segregated well enough to train using the convolution neural network model as mentioned in the next step.

Step 3: Training through Convolution neural network: This is the core step of the proposed work, where a neural network object called "model" is created for the sequential type using the keras and tensorflow libraries in python. Now this neural network object is added with the first layer of convolution with 32 number of kernels each of the size 3X3. This first layer is enhanced with the Activation function "Relu" with input shape 96 X96. This neural network object is set with color channel number 1, as it is dealing with the Grayscale images.

In the second layer of the convolution a set of 64 kernels with shape 3X3 is added with "Relu" Activation function. After the second layer the neural network model is added with a Max pooling layer with kernel shape 2X2 along with a Dropout layer of 25%.

Third and Fourth layers are added with a set of 128 Kernels with Size 3X3 and activation function " Relu" and a Max pooling layer with kernel shape 2X2 respectively. The fourth layer is ended with a Dropout layer with 25% ratio.

After finishing the fourth layer a flatten layer is added to strop the training with a dense layer of size 1024 with activation function "Relu" with dropout ratio of 50%. Finally the tensors are collected with a dense layer of 8. As 8 indicates 8 different gestures with an activation function "Softmax". The " Relu" activation function is shown in the equation 1.

Relu=max (0,x) _____(1)

Where x is the input attributes values

An "adam" optimizer is used to optimize the tensor values, finally this leads to the fitting of the generation with all the attributes created till now. After completing all the set epochs, the trained data is stored in a file with a format of .h5. The architecture of the Convolution neural network is depicted in the below figure 2.

Layer	Activation
CONV 2D 32 X 3 X 3	Relu
CONV 2D 64 X 3 X 3	Relu
MaxPooling2D 2 X 2	
Dropout 0.25	
CONV 2D 128 X 3 X 3	Relu
MaxPooling2D 2 X 2	
CONV 2D 128 X 3 X 3	Relu
MaxPooling2D 2 X 2	
Dropout 0.25	
Flatten	
Dense 1024	Relu
Dropout 0.25	
Dense 8	Softmax
Adam Optimizer	

Figure 2: Architecture Convolution Neural Network

Step 4: Testing through CNN- This is the step, where gesture is streamed through the camera to get a single frame. This frame is normalized and modeled using the convolution neural network along with the trained data stored in .h5 file format. Now a prediction is done to get an integer value that represents the labels of the classes in the dictionary to identify the proper gestures.

Step 5: Controlling Drone through Decision making- Here in this step a DJI Tello Drone is used to Controlling it through the obtained gestures. The DJI Tello Drone is purchased online through amazon.in [14], which is having a rich set of library for the python programming language. By using the tello library the Drone is controlled by using the IF-ELSE conditions efficiently. The Picture of DJI Tello Drone is depicted in figure 3.

Figure 3: DJI Tello Drone

IV RESULTS AND DISCUSSIONS

The proposed methodology for establishing gesture-based interactive drone navigation employing hand gesture recognition has been implemented utilizing the Spyder IDE and the Python programming language. To obtain the intended results, the technique makes use of the OpenCV, TensorFlow, and Keras packages. The proposed approach was tested on a PC with 8 GB RAM and 1 TB of storage, as well as an Intel Core i5 processor.

The suggested model is trained for 500 epochs on 5 gestures for assessment requirements, and the results must therefore be reviewed in order to achieve appropriate efficiency of the technique.

To establish the effectiveness of the suggested technique, the reliability of the hand gesture recognition component must be evaluated. The precision of the technique may be measured using an error meter; the smaller the error, the higher the precision. The RMSE statistic may be used efficiently for error assessment.

Among the most appropriate performance indicators for determining the error obtained between a group of continuous and associated characteristics is RMSE, or Root Mean Square Error. The attributes being selected for the assessment of the suggested technique are, hand gesture recognized correctly and hand gesture recognized incorrectly. The following equation 1 is used to determine the RMSE.

$$\mathrm{RMSE}_{fo} = [\sum_{i=1}^{N} (z_{f_i} - z_{o_i})^2 / N]^{1/2}$$

Where,

$\sum$ - Summation

$(Zfi - Zoi)^2$ - Differences Squared for the hand gesture recognized correctly and hand gesture recognized incorrectly

N - Number of conducted Experiments.

The RMSE values for a variety of repetitions of hand gesture recognition using this suggested technique are obtained. Every one of the five hand motions is evaluated 10 times for recognition. The identification outcome of the suggested technique is documented each time. The results are then used to calculate the RMSE. These RMSE values were meticulously determined using the results listed in table 1 below.

Table 1: RMSE outcomes for 5 hand gesture recognition.

Gesture	Number of Iterations	Correctly recognized hand Gesture	Incorrectly recognized hand Gesture	MSE
Back	10	9	1	1
Down	10	8	2	4
Front	10	9	1	1
Land	10	7	3	9
Left	10	10	0	0
Right	10	7	3	9
up	10	10	0	0

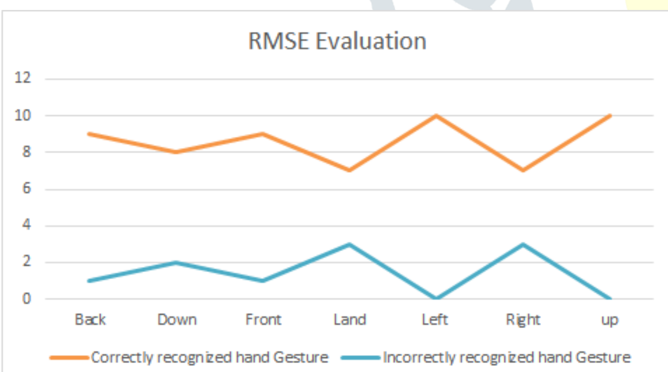

Figure 2: Line Graph for RMSE outcomes for 5 hand gesture recognition.

The line graph in picture 1 is created using the findings obtained for recognition performance and RMSE values in table 1. The following graph and table show the intended approach for hand gesture recognition, which has a very low error probability. The recommended approach's use of deep learning through CNN, which significantly improves detection performance, may account for the increased recognition accuracy. The RMSE of the gesture recognition error is 1.851, indicating is a reasonably good result for the suggested work's initial attempt.

V CONCLUSION AND FUTURE SCOPE

Drones of all sizes are becoming more common in our lifestyles, with applications such as search-and-rescue surveillance, transportation, and cinematography. These airborne drones are mostly used outside and can function in one of two mechanisms: manually (in which a pilot operates the drone in live time using only a controller or a smartphone) or autonomously (in which the drone is completely automated and maintains a pre-determined course or adjusts its trajectory through sensing). These two different ways of interface are restrictive since they do not permit people to communicate on the fly or swap control modes without existing experience. This survey report was quite helpful in laying out the existing approaches for comprehending image processing-based solutions for gesture recognition. Several methods have proven beneficial in identifying the benefits and downsides of employing computer vision-based gesture recognition to operate an UAV. This has resulted in the development of a gesture-based drone navigation system using Convolutional Neural Networks. The convolutional Neural network is implemented for the synthetic dataset created through the use of live streaming of the Gestures.

The obtained results through the evolution of the model for the RMSE indicate that the model is working absolutely fine for the designed gestures efficiently.

In the future this system can be enhanced to work for Controlling the Drone through gestures obtained through the sensors.

REFERENCES

[1] L. L. Cheng and H. B. Liu, "Examples of quadrocopter control on ROS," 2015 IEEE 9th International Conference on Anti-counterfeiting, Security, and Identification (ASID), 2015, pp. 92-96, DOI: 10.1109/ICASID.2015.7405668.

[2] A. Sarkar, K. A. Patel, R. K. Ganesh Ram, and G. K. Capoor, "Gesture control of drone using a motion controller," 2016 International Conference on Industrial Informatics and Computer Systems (CIICS), 2016, pp. 1-5, DOI: 10.1109/ICCSII.2016.7462401.

[3] Y. Yu, X. Wang, Z. Zhong, and Y. Zhang, "ROS-based UAV control using hand gesture recognition," 2017 29th Chinese Control And Decision Conference (CCDC), 2017, pp. 6795-6799, DOI: 10.1109/CCDC.2017.7978402.

[4] A. A. Bandala et al., "Development of Leap Motion Capture Based - Hand Gesture Controlled Interactive Quadrotor Drone Game," 2019 7th International Conference on Robot Intelligence Technology and Applications (RiTA), 2019, pp. 174-179, DOI: 10.1109/RITAPP.2019.8932800.

[5] B. Hu and J. Wang, "Deep Learning-Based Hand Gesture Recognition and UAV Flight Controls," 2018 24th International Conference on Automation and Computing (ICAC), 2018, pp. 1-6, DOI: 10.23919/IConAC.2018.8748953.

[6] K. Haratiannejadi, N. E. Fard and R. R. Selmic, "Smart Glove and Hand Gesture-based Control Interface For Multi-rotor Aerial Vehicles," 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC), 2019, pp. 1956-1962, DOI: 10.1109/SMC.2019.8914464.

[7] S. Shin, Y. Kang and Y. Kim, "Hand Gesture-based Wearable Human-Drone Interface for Intuitive Movement Control," 2019 IEEE International Conference on Consumer Electronics (ICCE), 2019, pp. 1-6, DOI: 10.1109/ICCE.2019.8662106.

[8] A. Menshchikov et al., "Data-Driven Body-Machine Interface for Drone Intuitive Control through Voice and Gestures," IECON 2019 - 45th Annual Conference of the IEEE Industrial Electronics Society, 2019, pp. 5602-5609, DOI: 10.1109/IECON.2019.8926635.

[9] H. Ghasemi, A. Mirfakhar, M. T. Masouleh, and A. Kalhor, "Control a Drone Using Hand Movement in ROS Based on Single Shot Detector Approach," 2020 28th Iranian Conference on Electrical Engineering (ICEE), 2020, pp. 1-5, DOI: 10.1109/ICEE50131.2020.9260864.

[10] C. Rognon, S. Mintchev, F. Dell'Agnola, A. Cherpillod, D. Atienza and D. Floreano, "FlyJacket: An Upper-Body Soft Exoskeleton for Immersive Drone Control," in IEEE Robotics and Automation Letters, vol. 3, no. 3, pp. 2362-2369, July 2018, DOI: 10.1109/LRA.2018.2810955.

[11] R. Ibrahimov, N. Zherdev and D. Tsetserukou, "DroneLight: Drone Draws in the Air using Long Exposure Light Painting and ML," 2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN), 2020, pp. 446-450, DOI: 10.1109/RO-MAN47096.2020.9223601.

[12] H. Kang, H. Li, J. Zhang, X. Lu, and B. Benes, "FlyCam: Multitouch Gesture Controlled Drone Gimbal Photography," in IEEE Robotics and Automation Letters, vol. 3, no. 4, pp. 3717-3724, Oct. 2018, DOI: 10.1109/LRA.2018.2856271.

[13] J. R. Cauchard et al., "Drone.io: A Gestural and Visual Interface for Human-Drone Interaction," 2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI), 2019, pp. 153-162, DOI: 10.1109/HRI.2019.8673011.

[14] https://www.amazon.in/gp/product/B09D82WBBK/ref=ppx_yo_dt_b_asin_title_o05_s00?ie=UTF8&psc=1