PUSHPAK : A SMART UAV

Capstone Project Report

End Semester Evaluation

Submitted by:

101903619 AYUSH KAUSHIK

102083028 AKSHAT THAKUR

102083033 RAHUL MISHRA

102083025 RIDHAM BEHL

BE Third Year- COE

CPG No. 284

Under the Mentorship of

Dr. Vinay Arora

Associate Professor

Computer Science & Engineering Department

Computer Science and Engineering Department

Thapar Institute of Engineering and Technology,

Patiala August 2022

ABSTRACT

Through our study, we want to develop a drone that can be precisely controlled by human hand gestures. Our group wanted to develop a solution that used all of our computer engineering knowledge. Keeping this in mind, we were able to collaborate on the notion of merging deep learning, drone programming, and machine learning to develop a practical and trustworthy solution. Because we as a team understood that piloting a drone may be difficult at times, we sought to develop a more intuitive interface. This allows us to build actions for the drone that account for stabilization automatically and move as expected, without the need for remote control to maintain the drone level. Our strategy for creating a widely marketable product requires us to examine the product's usability, price, and the potential to employ consumer-familiar devices to make interaction with the drone easier. Every aim specified in our project strategy was accomplished. We were able to optimize usability by inventing hand gestures to control the drone that are generally universal, in the sense that people from all over the globe can intuitively understand and apply many of the basic movements to control the drone. In order to keep the cost of the drone low, we designed our prototype with the cost of each component in mind, and after the building method and components list are finalized, we will be able to further lower the cost. Since our product requires the use of an external device, we decided to enable people to pilot the drone by using the required set of programmes along with the interface on their local laptops or PCs. In this article, we explain why it would be desirable to develop this drone, discuss our creative process, and describe how we aim to build and test the device. The drone will be controlled by a user interface, flight controls, power supply, and communication network that all operate in concert. With the fundamental hand actions discussed in the next sections, the drone will be able to do all flight-related tasks.

DECLARATION

We hereby declare that the design principles and working prototype model of the project entitled Pushpak: A Smart UAV is an authentic record of our own work carried out in the Computer Science and Engineering Department, TIET, Patiala, under the guidance of Dr Vinay Arora during 6th & 7th semester (2022).

Date:

Roll No.	Name	Signature
101903619	Ayush Kaushik	Ayush
102083028	Akshat Thakur	Akshat
102083033	Rahul Mishra	Rahul Mishra
102083025	Ridham Behl	Ridham Behl

Counter Signed By:

Faculty Mentor: Dr Vinay Arora

Designation: Associate Professor

Computer Science & Engineering Department

Thapar Institute of Engineering & Technology, Patiala

ACKNOWLEDGEMENT

No project is ever complete without the guidance of those experts who have already traded this past before and hence become master of it and as a result, our leader. So we would like to take this opportunity to take all those individuals who have helped us in visualizing this project.

We express our deep gratitude to our project mentor Dr. Vinay Arora for providing timely assistance to our query and guidance that he gave owing to his experience in this field for the past many years. He had indeed been a lighthouse for us in this journey.

We would also take this opportunity to thank Dr. P.S. Rana, Dr. Maninder Singh for their guidance in selecting this project and also for providing us all these details on proper presentation of this project.

We extend our sincerity appreciation to all the professors from Thapar Institute Of Engineering & Technology for their valuable insight and tips during the designing of the project. Their contributions have been valuable in so many ways that we find it difficult to acknowledge them individually.

We are also grateful to Dr. Shalini Batra, Head, Computer Science and Engineering Department For extending her help directly and indirectly through various channels in our project work. Lastly, we would also like to thank our families for their unyielding love and encouragement. They always wanted the best for us and we admire their determination and sacrifice.

Date:

Roll No.	**Name**	**Signature**
101903619	Ayush Kaushik	Ayush
102083028	Akshat Thakur	Akshat
102083033	Rahul Mishra	RahulMishra
102083025	Ridham Behl	Ridham Behl

ABSTRACT

Drones have become increasingly popular over the last decade. Every year their abilities are rapidly increasing and we want to do our part to add to this continuously growing field. With the knowledge we have obtained throughout our studies we want to challenge ourselves and develop a drone that is strictly controlled by human hand motions. Our team wanted to build a product that combined every aspect of our computer engineering coursework. With this in mind, we were able to collaborate on the idea to utilize PCB construction, embedded programming, and machine learning to build a useful and sound product. Because we, as a team, believed that drones can sometimes be difficult and so wanted to offer the ability to control a drone with a much simpler interaction. This allows us to build in actions for the drone that will automatically account for stabilization and move as expected without having to try to keep the drone flat and level via remote control. Our plan to create a widely marketable product forced us to take into account the usability of the product, the cost, and the ability to use devices that customers were already familiar with to allow for an easier interaction with the drone. We were able to fulfil all of those goals in our project plan. We were able to maximize usability by creating hand gestures to control the drone that are generally universal, meaning that people around the globe can understand many of the basic gestures and would immediately think to use those gestures to operate the drone. In order to make the cost of the drone low, we were taking into account the cost of each and every component when planning our prototype, and once the build process and components list is finalized, we will be able to improve upon that even further. Since our product requires the use of an external device, we decided to make it so that other people can simply install the software required to operate the drone on their local laptops or PCs. Throughout this document we explore why creating this drone would be beneficial, our creative process and map out how we plan to build and test the device. Consisting of four main components, we will have a user interface, the drone flight controls, power system, and the communication network all working in harmony to control the drone. With the simple hand motions explored in the following sections, the drone will be able to perform all the necessary actions needed for flight.

DECLARATION

We hereby declare that the design principles and working prototype model of the project entitled Pushpak: A Smart UAV is an authentic record of our own work carried out in the Computer Science and Engineering Department, TIET, Patiala, under the guidance of Dr Vinay Arora during 6th & 7th semester (2022).

Date:

Roll No.	Name	Signature
101903619	Ayush Kaushik	Ayush
102083028	Akshat Thakur	Akshat
102083033	Rahul Mishra	Rahul Mishra
102083025	Ridham Behl	Ridham Behl

Counter Signed By:

Faculty Mentor: Dr Vinay Arora

Designation: Associate Professor

Computer Science & Engineering Department

Thapar Institute of Engineering & Technology, Patiala

ACKNOWLEDGEMENT

No project is ever complete without the guidance of those experts who have already traded this past before and hence become master of it and as a result, our leader. So we would like to take this opportunity to take all those individuals who have helped us in visualizing this project.

We express our deep gratitude to our project mentor Dr. Vinay Arora for providing timely assistance to our query and guidance that he gave owing to his experience in this field for the past many years. He had indeed been a lighthouse for us in this journey.

We would also take this opportunity to thank Dr. P.S. Rana, Dr. Maninder Singh for their guidance in selecting this project and also for providing us all these details on proper presentation of this project.

We extend our sincerity appreciation to all the professors from Thapar Institute Of Engineering & Technology for their valuable insight and tips during the designing of the project. Their contributions have been valuable in so many ways that we find it difficult to acknowledge them individually.

We are also grateful to Dr. Shalini Batra, Head, Computer Science and Engineering Department For extending her help directly and indirectly through various channels in our project work. Lastly, we would also like to thank our families for their unyielding love and encouragement. They always wanted the best for us and we admire their determination and sacrifice.

Date:

Roll No.	Name	Signature
101903619	Ayush Kaushik	Ayush
102083028	Akshat Thakur	Akshat
102083033	Rahul Mishra	Rahul Mishra
102083025	Ridham Behl	Ridham Behl

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

UAV	Unmanned Aerial Vehicle
PCB	Printed Circuit Board
PC	Personal Computer
TIET	Thapar Institute of Engineering and Technology
FBGA	Field Programmable Gate Array
IMPOz	Innovative Multidisciplinary Product Optimization
NFFP7	National Aeronautical Research Program
GPU	Graphics processing unit
SOS	Save Our Souls
SSD	Solid state drives
ACM	Association for Computing Machinery
IEEE	Institute of Electrical and Electronics Engineers
GPS	Global Positioning System
CNNs	Convolutional Neural Network
CPU	Central Processing Unit
GB	GigaByte
DFD	Data Flow Diagram
AR	Augmented Reality
HSV	Hue Saturation Value

1. INTRODUCTION

1.1 Project Overview

In a world wherein video surveillance has come to be of important significance to civil and navy safety scenarios, unmanned aerial vehicles (UAVs), or in reality, drones, are of strategic significance. These motors are actually within everyone's reach, both in terms of cost and simplicity of piloting. They have the excellent gain of being nearly invisible to radar and might deliver a payload like a high resolution digital camera with optical zoom. Among the packages of aerial surveillance, facial reputation is one of the favored modalities. Thank you also to the deep study that has given an extraordinary boost to this area of research. To ensure the protection of the inhabitants, drones, however, are frequently no longer allowed to function over humans and ought to maintain a safe distance. For this reason, to carry out facial reputation, the pix taken with the aid of using the drone must be zoomed in, as a result exposing the pix to great degradation because of any actions of the drone. Therefore, given the popularity of humans, the use of faces can pose an undertaking in many aerial surveillance packages. In this work, we analyzed the overall performance of deep neural networks for face verification in an aerial surveillance scenario, and in particular, we compared cutting-edge neural networks to facial reputation systems trained on low-decision photos. To put in force our neural network for the responsibilities of very low and cross-resolutions face recognition (FR), we fine-tuned the network structure primarily based on ResNet-50 with Squeeze-and Excitation blocks. For the education process, we used the VGGFace2 dataset, after which we examined the overall performance of the very last version on the IJB-B dataset for the 1:1 verification task. We have analyzed the overall performance of the community for that reason, creating the Drone SURF dataset, seeking to reproduce as intently as feasible the experiments proposed within the unique article.

Our plan to create a broadly marketable product forced us to consider ease of use, cost, and the ability to use devices that customers were already familiar with to allow for easier interaction with the drone. We were able to meet all of these goals in our project plan. We were able to maximize usability by creating generally universal hand gestures to control the drone, meaning that people around the world would understand many of the basic gestures and would immediately think of using those gestures to operate the drone. In order to reduce the cost of the drone, we consider the cost of each individual component when planning our prototype, and once the build process and component list are complete, we can continue to improve it. Because our product requires the use of an external device, we decided to make it so that other people could easily install the software needed to operate the drone on their local laptops or PCs. This allows users to interact with something they are familiar with and makes the drone user experience more seamless.

Drones, also known as unmanned aerial vehicles, are used in recreational and a variety of industrial applications, including security, defense, agriculture, energy, insurance, and hydrology. Drones are essentially specialized flying robots that perform functions such as taking pictures, recording videos, and capturing objects. multimodal data in their local area. There are two types of drones based on their shape and size, fixed wing and multirotor. Additionally, by using a protective helmet, multirotor drones are highly resistant to collisions, making them even more valuable for exploring uncharted territory. Nowadays, flying robots are used in various companies, such as parcel delivery systems. For example, companies like Amazon Prime and UPS that use multi-rotor They use drones to deliver packages. The New York Police Department uses quadcopters for crime prevention. For example, for agricultural surveillance purposes, the use of multiple sensors such as video cameras and thermal infrared is advantageous. Drones are especially useful on risky missions. For the sake of clarity in the rest of this work.

A visual camera is an essential sensor for today's drones. The low cost, low power consumption, and small size of image capture and transmission devices. There are numerous drones on the market. The output of a drone camera can be used in a variety of ways depending on the application. In a common scenario, the camera output is routed to the drone operator, who can order a new one. Instruction based on the current visual environment via a remote control that acts as an intermediary between the drone and its operator. In this work, we investigate an alternative method to control multirotor drones by hand gestures as the main communication channel. We propose a framework that maps separate images from a video stream as one of five commands or gestures. The camera can capture visual instructions from the operator, eliminating the need for a control device and paving the way for agentless communication. Haar features serve as the basic masks for capturing gradient changes in images. Each mask block can be scaled or rotated to capture given targets. These advantages allow us to recognize different gestures of many sizes. Therefore, a machine learning algorithm based on Haar functions is used as an action planner. Safety aspects are also taken into account as the drones automatically carry out commands initiated by operator gestures. This project also presents a case study for autonomous drones powered by image recognition. Our main contributions to this project include

i) A novel hand-based drone control framework Gestures.

ii) A comparison of state-of-the-art computer vision hand gesture recognition applied to our hand gesture dataset.

iii) A discussion of key challenges and lessons learned since building the hand gesture recognition component of the framework.

1.2 Need Analysis

Both drones and computer vision are very popular, and in some cases, both have been combined for tracking purposes. That means there are still no commercial products strictly controlled by hand gestures. Being the first to do so would be an extremely rewarding achievement. Flying a drone for the first time can be quite difficult and cause a lot of trouble for the user. With your own hand gestures, you can add a sense of lightness and fluidity not found on a typical smartphone or handheld controller. In addition, our solution includes controlling the drone with one hand, unlike traditional drones where you use both hands to operate a physical remote control. controller. Our solution allows the user to control the drone with just one hand, as long as that hand stays in the correct field of view. This allows your alternate hand to move freely, which is something that is often overlooked when it comes to drones. Drones are often used to capture something in motion, whether it's outdoor action sports or photographers and videographers trying to get a bird's-eye view that isn't easily achievable without them. A free hand will immediately benefit drone users.

i) Motivation

Compared to stationary cameras unmanned surveillance has a greater mobility and viewing area. Disadvantages include unstable background, low resolution and changing lighting. Intelligent drones are in high demand in practical applications. On the other hand, human and human face detection in drone images or videos is different from conventional object detection in aerial photography. Human faces vary in size; this is due to the difference in size of human faces of the same type as well as the spatial resolution of the sensors. Another reason to automate aerial photo analysis is the need to predict changes in areas of interest. Traditionally humans are responsible for analyzing images transmitted directly from security cameras to the command center. As expected this approach suffers from a number of problems including human lag or errors in event detection and lack of sufficient imagery from standard fixed security cameras. In this regard the purpose of this project is to discuss advanced approaches to object detection in drone images and explore their applications in real time.

1.3 Research Gap

Despite many achievements, deep learning methods, drone technologies and the combination of both present many challenges like battery life, dynamic object recognition, etc. that still need to be solved. The literature review shows that researchers use deep learning or traditional image processing methods for drone object detection. However, the drone has weight, size and power consumption limitations that make onboard processing a difficult task. Deep learning algorithms are good for both feature extraction and sorting, making them a better choice.More difficult when large image data needs to be transferred but less bandwidth is available. There is a need for dedicated security measures for UAV applications. In real-time applications, using the internet in UAVs poses both security and privacy risks. These challenges motivate researchers to develop more efficient accuracy, speed and less overhead deep learning architectures.

1.4 Problem Definition and Scope

Despite many advances, deep learning methods, drone technology, and a combination of the two present many challenges. A literature review shows that researchers are using deep learning or traditional image processing techniques for object detection by drones. However, drones have limitations in weight, size, and power consumption, making it difficult to process data in-flight. Deep learning algorithms are the best choice as they are suitable for both feature extraction and sorting. It is more difficult if you need to transfer large amounts of image data, but the available bandwidth is less. UAVs require special security measures. These challenges force researchers to develop more efficient deep learning architectures.

1.5 Assumptions and Constraints

Along with our planning comes a lot of different constraints that govern the choices we make and the path we take with our project. Those range in various types such as economic, environmental, ethical, health, manufacturability, safety, social, and sustainability. The following are the constraints and the assumptions which are made to ensure the proper working of the system:

TABLE 1: Assumptions and Constraints

Sr. No.	**Assumptions and Constraints**
1.	This project is fully funded by our group. We came up with the idea of this project as a group and did not involve any third parties. As a result, we do not have any sponsors for our project. It is nice to plan the project ourselves however the assets from a sponsor would alleviate some of the economic stress. We understand that we will have to allocate a lot of money to fund the project. As students we have limited funds and want to do our best to make our project as affordable as possible. There are a lot of steps we can take to make that more plausible. A lot of the components we are buying are fairly sensitive and need to be taken care of properly. If we can avoid breaking pieces unnecessarily, we can save a lot of money in the long run. Also, if we do more research beforehand, we cannot waste money on the wrong parts.
2.	Another benefit of scanning the market thoroughly is finding the best balance of

	price and quality where we can obtain the best option possible. Having to buy replacements is inevitable but limiting the number of mishaps will lessen the economic constraints. We do not have unlimited funds and we keep that in our minds when we choose our parts.
3.	It is assumed that the user is situated in a well-lit room so that the video stream captured by the webcam can be of good quality.
4.	The accuracy of the result depends upon how well the gesture is made by the user. Thus making the quality of prediction dependent upon the user.
5.	Drones sometimes have a negative connotation and are often banned, as they can be a disturbance. Drones are typically associated with having cameras and even though our drone does not have a camera, people may feel as if we are spying on them. It would also be unethical of us to fly a drone in certain places. If we fly our drone in the wrong places not only can it be illegal but also offensive. Because of this we are going to limit the places we fly our drone. We are going to ensure that we do not cross any ethical or social borders when testing and flying our drone.
6.	Environmentally our drone will be constrained by its ability to only be flown indoors. It is not easy to find indoor spaces where flying the drone is allowed without permission. It is important that our drone is capable of performing well in tight situations. Having the constraints of four walls around the drone can complicate some of the testing. To work around this, we received permission from our hostel that is going to work with us. They have extra indoor courts that can be used throughout the day, and they have given us the time to fly a drone around.
7.	Environmentally we are also legally constrained. To fly heavier drones outdoors in India a license is required. To save money and time we decided to avoid flying the drone outdoors completely and to focus on only flying indoors. The benefit of this is the controlled environment that we have indoors. There are no factors

	like wind and rain to worry about. Having an indoor drone will lessen the constraints of the more unpredictable conditions of the outdoors. There are also socially acceptable and ethical places to fly the drone. Before flying a drone in any location, it is important to have permission, whether this is a public gym or our personal apartments, it is essential we let everyone know that we will be flying a drone. It is not socially acceptable to fly a drone over people.

1.6 Standards

When working on a project standards are essential as they create a level of quality and expectation across the board it also helps make the project adaptable and easy to incorporate if another company or team were to incorporate our project they would easily be able to adapt to our industry standard protocols both I2C and UART are very common communication peripherals and most third-party sensors and devices we are using are compatible most flight controllers utilize UART while all of the sensors we have looked into our i2c using the i2c bus can significantly simplify and make our design more efficient 10 drones all must follow the ISO/TC 20/SC 16 standard for unmanned aircraft systems a drone is an unmanned aircraft system and these standards map out what is allowed and what is not allowed in regard to locations to fly your drone this allows for a more responsible and better educated population of drone operators which is especially important as drones gain popularity defines Bluetooth as a standard for wireless personal area network Wireless Personal Area Network we decided to use this standard for our wireless communication because it is heavily supported and continually updated this allows for us to implement a technology that is familiar to the common user and is a respected engineering standard ipc-a-610 is the acceptability of electronic assemblies this standard will verify that our product has a highly reliable printed wiring assembly this is a crucial criterion for our project to meet because it will verify our product even further to allow it to be more marketable this will also allow us to proceed to manufacture the product faster because it already meets the industry standard and does not need to be verified in that regard again.

1.7 Approved Objectives

Our objective is to create a low-cost hand gesture controlled drone. Once we have a working product, our goal is to add as many hand signals as possible. We are starting with eight essential hand signals and we strive to get that number up to around 15 different hand signals. Once we get the working product, we can limit our design to the bare minimum of what needs to be completed and make the project as affordable as possible.Another objective of ours is to make the build process as simple as possible and as repeatable as possible.

Following objectives were kept in mind while designing the project:

i) To programme a drone to create a more personalized approach to control it using hand gestures.

ii) To programme a drone so that it will be capable of face tracking. It will be able to follow the person keeping a specific distance from him/her whose face the drone would have detected.

iv) To programme a drone with a functionality which will enable it to follow a specified color path using line path reflection and corresponding clockwise and anticlockwise direction turns.

iv) To programme a drone to move it in a particular trajectory in case of emergency as SOS Signals.

1.8 Methodology

i) Face Tracking (Facial Identification and then following that person)

Real-world recognition of face recognition systems is a long way from human face recognition. There are several factors that affect the performance of facial recognition and recognition techniques, especially on drones. Current In drones, facial recognition technology is capable of recognizing faces, but with some limitations in terms of angle and distance, especially when the drone captures High-altitude images and facial images are captured at a distance and with a large depression angle. For face recognition, you must first collect photos of a person's face. The photos with faces are stored in different folders depending on the person's name. After collecting the photos, run the face training process using OpenCV in Python, which implements the Haar Cascade algorithm and produces a face dataset. OpenCV is an open source machine learning and computer vision software library. The result of face training is a collection of vectors according to the weight built up during the qualifying phase. After that, run face detection with the FPV video stream. The submitted video will be processed in real time. If at least one photo of the face is found that corresponds to the sorting weight, the facial condition is known. The face recognition results are displayed on the screen.

ii) Hand Gesture Control

We use haar features to represent each image in the dataset. Although Haar's features were introduced in 1910, they only became popular for image recognition problems after extensive analysis. A haar based feature uses rectangular areas at different locations in the detection window when pixels are merged in intensity at each detected window location and calculates the difference between these sum values. These differences are then used to categorize the image. In our scenario, the feature extraction engine uses the pattern. Hand gestures are generated by many local Haar functions. Later, the classifier maps the function vector of the gestures to either one of the existing gesture labels or empty. The reasons for choosing Haar classifiers over other algorithms are as follows: The haar cascade has a better detection rate than other functions. It's also easy to implement with descriptors like "Preserve fewer clear images," achieves higher accuracy with training images, and consumes less memory than GPU-enabled image classification systems like Convolutional Neural networks.

iii) Keyboard Control

i) Programming the UAV to move corresponding to the keyboard inputs.

ii) The user inputs keyboard keys corresponding to which the drone is moved in the corresponding direction with respect to the programmed sensitivity in the direction.

iv) Line follower

The reference for the UAV controller is generated by the line detection algorithm. The general idea is to smooth the image, binarize it, and identify its main lines. The main orientation is then calculated as the mean of the orientations of the main lines, also taking into account the previous image frames. A more detailed description is provided below. First, a new frame of video transmitted from the camera downstream of the UAV is obtained. It is then converted to grayscale and cropped (e.g., by cutting off its black edges). The resulting image is correlation-filtered with a rotationally symmetric Gaussian low-pass filter of size 84 x 84 pixels with a standard deviation of 28. The Canny method is then applied to the filtered image to convert it to binary intensity. Additional convolution is performed with the horizontal and vertical 3x3 Sobel filters to emphasize the edges. Then, due to its robustness to gaps and noise, the standard Hough transform is applied to the binary image, and lines corresponding to the 10 strongest peaks in the Hough transform matrix are drawn. The orientation of these lines is calculated and classified as belonging to one of four intervals ranging from 0 to 180 degrees. Only the rows belonging to the interval that has the most rows are considered for calculating the average position and origin in that frame. Finally, a moving average filter is used to include the peak line information from the previous 5 frames in the final average. This sequence is repeated for each frame of the image. As a result, after each frame, you get the position and orientation of its main lines.

v) Neural Network

Consists of the input layer, the hidden layers, and the output layer. The input layer provides the output data for the neural network. The hidden layers are those between the input and output layers. And this is where all the arithmetic and learning takes place. More hidden before building a neural network application, an understanding of neural network components must have the following characteristics and limitations: Neural networks are a subset of machine learning in terms of hierarchy. Artificial intelligence is a broad term that encompasses machine learning. Machine learning allows a system to learn and make progress from data entered in the past without being explicitly programmed. Neural networks are a subset of machine learning. Certain components and properties of neural networks enable this learning. Essentially, a neural network is a set of algorithms designed to recognize and learn from patterns, allowing these patterns to perform some tasks without being explicitly programmed to do so. Some popular neural network applications include speech recognition, object recognition, image processing, and text recognition. Neural networks are modeled by ours. The brain and how our brain processes information They are made up of interlocking nodes or neurons that pass input and output to different neurons. All nodes are connected by weighted edges. A weight represents the strength of a connection between nodes and determines the influence one has. The node has switched to miscellaneous. The greater the weight between two nodes, the greater the influence that node has on the other. Neural networks are usually trained on some set of data. As this workout takes place, the weights are updated to give optimal results. Neural networks are also divided into 3 generalized layers: The input layer layers there are, the deeper the neural network, the greater the number of hidden layers. It all depends on the machine learning application itself. The output The last layer is the last layer in the network and produces the final result. The idea of having a machine train itself to process data and learn from it without explicitly teaching the machine is called deep learning. The hidden layers of the neural network enable this learning.

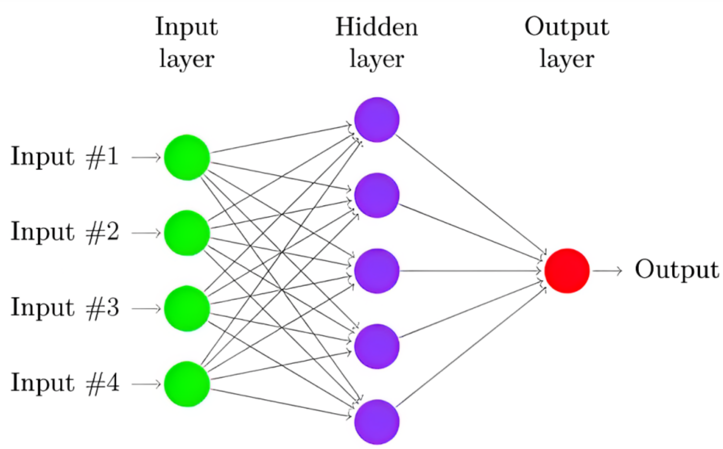

FIGURE 1: Neural network architecture

Currently, there are many different types of neural networks. Some examples include: recurrent neural networks; long-term and short-term memory, convolutional neural networks, etc. For our project, the neural network that we will implement is the Fully connected and Convolutional Neural Network. This particular type of neural network helps bridge the gap between computer vision and deep learning. Convolutional neural networks have proven effective in areas such as image recognition and classification and have had great success. tasks related to object recognition. Based on these facts, we decided to implement a convolutional neural network in our project. The challenge of accurately detecting and classifying hand gestures in real time can be solved. Training a convolutional neural network input CNN takes an image as input. In our project, this will be an image of a hand. The image is then sent through hidden layers where the image is refracted. Different properties of the hand The image of the gesture is extracted and learned through the net. For example, some features that can be extracted are edges and corners. A closed hand gesture image has different looking edges than an open hand gesture image. As features are extracted and learned, the weights assigned to each node in the network are modified to account for the learning features. This is called the "feature." learning phase. In the prediction phase, the network makes a prediction about what it thinks the input image is, or classifies the image based on features extracted from the network. In our project, an input hand gesture image can only be one of eight. Therefore, the network must classify the entry point. Gesture image as one of eight different classes The specific components that go into learning and classifying features are called the building blocks of CNN.

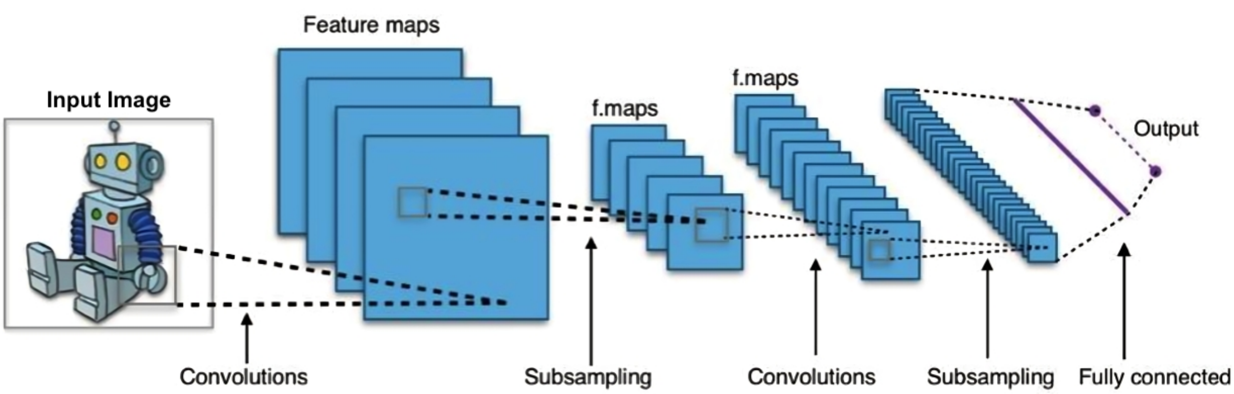

FIGURE 2: Basic CNN architecture

1.9 Project Outcomes and Deliverable

The applications could be very large and could be used for noble causes and in helping challenged individuals this could also be a next big thing in the security and surveillance systems likewise when used with care and security these systems could be having a wide application although these systems are highly risked towards hacking and unauthorized controls once evolved these systems would be highly relied on at that stage once hacked the hacker can have control over a major part of the victims routine avoiding such complications would help us make the most effective usage of the gesture recognition recent times machine learning and artificial intelligence have been popular and we could see the application in every other application researches being conducted which would include the gesture recognition with machine learning and artificial intelligence in which the habits and mannerisms could be captured by the system and could be reproduced by the artificial intelligence and could recreate them projects like the current one would be stepping stones for the future more stabilized in which the outcomes could be used and recreated to achieve a similar or more complex task [1].

1.10 Novelty of Work

Implementing a face recognition and face tracking system using the Haar cascading classifier. A Haar classifier, or a Haar cascade classifier, is a machine learning object detection program that identifies objects in an image and video. By using this Haar cascading, we will detect the face and recognise it. The advantage of this system is that we can use it in the army or in the police force to identify and track the terrorist/criminal. We also use one feature that is SOS, which is helpful for a third person as an alert so that we can save him by SOS pattern.

2.1 Literature Survey

2.1.1 Theory Associated with Problem Area

The most frequent problem of the video part, when running the drone detector system outside, is the autofocus feature of the video camera. Unlike the Fcam and IRcam, clear skies are not the ideal weather, but rather a scenery with some objects that can help the camera to set the focus correctly.The scope of this thesis is twofold: First, to explore the possibilities and limitations of designing and constructing an automatic multisensor drone detection system building on state-of-the-art methods and techniques. In some cases, these methods will also be extended from conclusions and recommendations found in the related work. This extension is achieved by the use of a different sensor combination, exploring the performance based on the sensor-to-target increasing the number of target classes, and incorporating a novel sensor fusion method for the drone detection task.

2.1.2 Existing Systems and Solutions

Computer vision-based methods rely on a drone's camera's ability to capture images of its surroundings and use pattern recognition to translate the images into meaningful and/or actionable information.These are: separating images from the video sequences to create a robust and reliable image recognition system based on separated images, and finally converting classified gestures into actionable drone movements, such as: starting, landing, etc. In this work, a set of five gestures is examined. Haar's feature-based AdaBoost classifier is used for gesture recognition. We also consider operator safety and drone actions when calculating distance based on computer vision for this task. A series of experiments are performed to measure the accuracy of gesture recognition, taking into account the main variables of the scene: lighting, background,and distance. Classification accuracies show that well-lit gestures clear within 3 feet are correctly recognized [2]. More than 90%. Limitations of the current framework and feasibility solutions for better gesture recognition are also discussed. The software library we developed and the hand gesture datasets are available as open source on the project website.

Disadvantages of Existing System:

i) Facial recognition on drone video footage.

ii) One problem we have found is the performance, especially at low-resolution performance.

2.1.3 Research Findings for Existing Literature

TABLE 2: Research findings

Sr. No.	Roll Number	Name	Paper Title	Tools / Technology	Findings	Citations
1	101903619	Ayush Kaushik	A Comparison of Face Verification with Facial Landmarks and Deep Features	Dlib, Histogram of oriented gradients, VGG Facenet,	The results show that the accuracy of depth features in checking whether a face belongs to a specific person is much higher than that of the face feature-based approach.	Amato, Giuseppe & Falchi, Fabrizio & Gennaro, Claudio & Vairo, Claudio. (2018). A Comparison of Face Verification with Facial Landmarks and Deep Features.
2	101903619	Ayush Kaushik	Okutama-Action: An Aerial View Video Dataset for Concurrent Human ActionDetection	Single Shot MultiBox Detector (SSD), VGGnet	Models are good at localizing objects, but bad at distinguishing classes. The gap between map pedestrian detection and action detection confirms this.	Barekatain, Mohammadamin & Martí, Miquel & Shih, Hsueh-Fu & Murray, Samuel & Nakayama, Kotaro & Matsuo, Yutaka & Prendinger, Helmut. (2017). Okutama-Action: An Aerial View Video Dataset for Concurrent Human Action

						Detection. 10.1109/CVPRW.2017.267.
3	102083028	Akshat Thakur	VGGFace2: A dataset for recognising faces across pose and age	Resnet50. VGGFace2	In the evaluation of pose and age protocols,models trained on VGGFace2 always achieve the highest similarity scores,	Cao, Qiong & Shen, Li & Xie, Weidi & Parkhi, Omkar & Zisserman, Andrew. (2018). VGGFace2: A Dataset for Recognising Faces across Pose and Age. 67-74. 10.1109/FG.2018.00020.
4	102083028	Akshat Thakur	Face Recognition on Drones: Issues and Limitations	Face++, Rekognition	They also investigated how altitudes, distances, and the angles of depression are influential in the performance of face recognition on drones and also concluded that present face recognition technologies are able to perform adequately on drones.	Hwai-Jung Hsu and Kuan-Ta Chen. 2015. Face recognition on drones: Issues and limitations. In *Proceedings of the First Workshop on Micro Aerial Vehicle Networks, Systems, and Applications for Civilian Use*. 39–44
5	102083033	Rahul Mishra	Human Control of UAVs using Face Pose Estimates and	Support vector regression (SVR), Haar	The UAV moves based on the user's hand position and	Nagi, Jawad et al. “Human Control of UAVs using Face Pose

			Hand Gestures	cascade classifier, Computer vision	also recognizes the person.	Estimates and Hand Gestures." 2014 9th ACM/IEEE International Conference on Human-Robot Interaction (HRI) (2014): 1-2.
6	102083025	Ridham Behl	A Vision-based Line Following Strategy for an Autonomous UAV	Computer vision, Line detection algorithm	The proposed algorithm is able to recognize the main orientation of the image pattern and can thus generate a correct reference for line tracking control.	Brandão, Alexandre & Martins, Felipe & Soneguetti, Higor. (2015). A Vision-based Line Following Strategy for an Autonomous UAV. 10.13140/RG.2.1.1132.2729.
7	102083025	Ridham Behl	Vision-based Line Tracking for an Indoor Micro Quadrotor	Computer vision, fuzzy logic, Ground control station	line tracking achieved through visual feedback is an efficient methodology in GPS-denied indoor environments	: Han Sol Kim, Andrew Hong & Jin Bae Park (2013) Vision-based Line Tracking for an Indoor Micro Quadrotor, Journal of International Council on Electrical Engineering, 3:3, 245-249, DOI: 10.5370/JICEE.2013.3.3.245

2.1.4 Problem Identified

India is a land of festivals, including local festivals such as Ganesh Utsav and Kumbh Mela, the largest gatherings in the world. Kumbh Mela told BBC News that there were 560 missing cases per day and 510 people were reunited with their families. But finding one missing person out of millions is a huge task. Drones with facial recognition systems can cope with this task much more easily. Continuously face recognition system. Watch out for the flying faces. The system compares the recognized face to the database and identifies a match. Ground server equipment connected to the camera performs facial recognition and analysis functions. Data sent to local servers may include video, images, and aircraft locations. When you capture a frame from a live video stream, an algorithm finds all faces in the frame and crops all detected faces for identification. Modern algorithms can recognize multiple images per second.

2.1.5 Tools and Technologies Used

i) Python

Python could likewise be a taken item organized basic level language with dynamic derivation its straightforward level in-created information structures got together with unique organization and dynamic restricting sort it outrageously interesting for speedy application advancement what's more on be utilized as a pre piece or glue language to relate existing components on pythons clear direct to be told accentuation highlights quality by then decreases the cost of program fixes python maintains modules and packs that moves program quality and code utilize the python go-between and what's more the escalated standard library are offered in give or combined sort to nothing of charge for each and every fundamental stage and wish to be uninhibitedly spread oft programmers fall stricken with python because of the misrepresented strength it gives since there is no aggregation step the special stepped area test-investigate cycle is unfathomably expedient work python programs is basic a bug or unfortunate information won't ever cause a division deformity taking everything into account once the interpreter discovers a blunder it raises an extraordinary case once the program doesn't get the exception the go-between prints a stack follow a stock level program licenses assessment of local and world elements examination of self-emphatic enunciations setting breakpoints wandering through the code a line at a rapidly on the program is written in python itself vouching for pythons smart power barring generally the quick in view of right a program is to incorporate a few print clarifications to the accessibility the quick modify test-explore cycle makes this simple philosophy dreadfully amazing.

ii) FLASK

A Flask is a Web Application Framework that is built with Flexibility and Speed In the Mind. Flask is Built in Python , which many data Scientists are familiar with . Flask takes care of the Environment and Project setup involved in web Applications Allowing the Developer to focus on their application rather than thinking about HTTP , routing , dataset etc. Flask allows Data Scientists to create simple Single page Applications and one should Help or look into if they want to create Products for Consumers Flask is a micro web framework written in Python. It is classified as a microframework because it doesn't require particular tools or libraries. it's no database abstraction layer, form validation, or the other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions which will add application features as if they were implemented in Flask itself [3]. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and a number of other common framework related tools Flask was created by Armin Conacher of Pocono, a world group of Python enthusiasts formed in 2004.According to Conacher, the thought was originally an April Fool's joke that was popular enough to form into significant application. When Conacher and Georg Brandi created a bulletin board system written in Python, the Pocono projects Werke and Jinja were developed. Flask has become popular among Python enthusiasts. As of October 2020, its second most stars on GitHub among Python web-development frameworks, only slightly behind Django, and was voted the foremost popular web framework within the Python Developers Survey 2018.

These are some Important features of the Flask:

a. It is a Development Server

b. Debugger

c. RESTful request dispatching

d. Unicode Based

e. Flask have google app engine Compatibility

iii) MySQL

MySQL is prestigious as world's most by and large utilized ascii archive data back-end its most guarantee data for pup as MySQL is most habitually utilized ascii record pre arranging data attempt workers offer for MySQL is ideal and diminishes our work to an outsized degree.

iv) Drone hardware design

Our drone design is a classic quadcopter with four arms, four brushless motors, and four two-blade propellers. Each motor is accompanied by its own ESC (Electronic speed Controller). They are all powered by rechargeable lithium batteries. The ESC is connected to the flight controller, which communicates with the user via Bluetooth. Each command is received by the Bluetooth module and interpreted by the microcontroller on our circuit board. Figure 3 shows an overview of the drone layout.

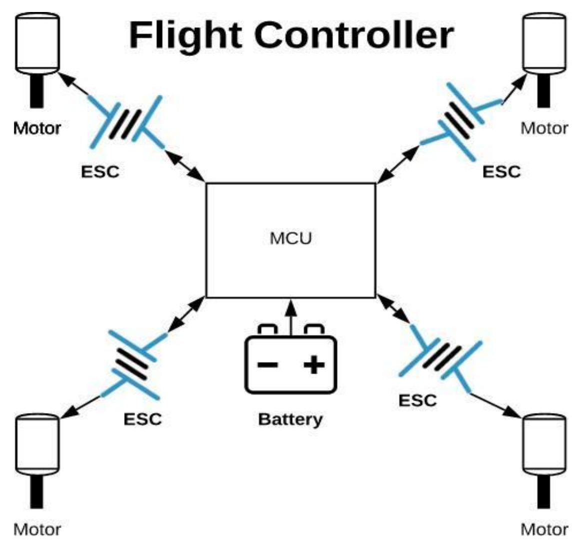

FIGURE 3: Flight Controller System

v) Electronic Speed Controller

The motors rely on electronic speed controls to function properly. Essentially, the Electronic Speed Controller, or ESC for short, communicates between the motors and the flight controller. Regulates the speed at which the springs can beIt is programmed to work as intended. The ESC for motors is easy to use. They feature three motor wires as opposed to the two motor wires on a brushed ESC. These wires carry the signals from the flight controller to theEngine. A stronger signal spins the motors faster. All of this is determined by the air traffic controller, who receives instructions from the user.

vi) Sensors

The drone has some sensors that serve their own purpose. single purpose. The three most important sensors are the gyroscope, the accelerometer, and the altitude sensor. They are all connected to our flight controller and their data is used to balance the drone and move it to the position specified by the user.

a. **Gyroscopes:** Gyroscopes come in different types. Space shuttles use laser gyroscopes, while something more common, like your car, uses a vibrating gyroscope. The gyroscope used by our drone is also a vibrating gyroscope and is an integral part of our design. It is especially important for the PID control loops that keep the drone's flight stable. The MPU-6050 is a popular choice, and we'll use it for our design. The MPU-6050 has more than just a gyroscope. One of the main reasons why it is widely used is that it has very useful auto-leveling support that makes the balancing process easier. The sensor vibrates in a way when the device rotates. The gyroscope provides this information. flight control and appropriate action is taken [4]. The sensor measures how fast the drone turns. There are three coordinates that the gyroscope measures: x, y, and z. Both x and y can be determined by how the gyroscope is rotated. That means it must be laid perfectly flat for the vertical measurement, z, to be accurate.

b. **Accelerometer:** The accelerometer we use shares the same chip, MPU-6050, as the gyroscope. Accelerometers are attached to most electronic devices that move. They are used on most aircraft and measure both orientation and devices. The acceleration sensor constantly measures all the acting forces. In a drone, some are constant, like gravity, while others are user-induced. Newton's second law defines acceleration as net force divided by the mass of the object. The accelerometer works the same way. A mass is attached that measures the change in force and determines the value of the acceleration. This is given to the air traffic controller and used to move as requested.

c. **Altitude Sensor:** Attitude measurement can refer to many different things. Simply put, it is a measure of how high the drone flies. However, everything is based on the reference point. This can be absolute or relative. Different ways of measuring altitude include measuring atmospheric pressure, height above sea level, or height above ground. To control the altitude of the drone, we would need a drone that measures air pressure. We need to make sure that the altitude from the starting point does not exceed a certain value. When the drone is launched, the current altitude must be saved. The drone can be given a limit on how much space it must have in order to fly safely. This value can be changed and is assigned before takeoff.

vii) Voltage Regulator

For each component that receives power, mainly the motors and the Arduino, we need to regulate the voltage so that we don't overload them and in turn damage the components. While the power supplied to the motors does not need to be very tightly regulated, the Arduino's power needs to be very tightly regulated.

2.2 Software Requirement Specification

2.2.1 Introduction

2.2.1.1 Purpose

In this work, we studied the overall performance of deep neural networks for face re-identity particularly designed for low-nice snap shots, and implemented them on a drone. We also added some other functionality that is line following, Applications for this functionality include things like product delivery, warehouse management, library management, etc. To execute this functionality, we will use open-cv with an algorithm that can detect the line and execute the commands based on the line pattern. In the end, if there is any emergency and the user wants to alert someone, the user can send an SOS message to the drone, which follows one start flying in that pattern to grab the attention of a third person and alert them.63

Keeping in mind the problems mentioned above, we aim to provide a well-equipped, fully functioning, and interactive drone for public safety and surveillance. Our project is designed to skip this extreme learning curve by making predefined actions for the drone, such as elevating, de-elevate, and moving left, right, forward, and backward. The operator only needs to get familiar with hand gestures. We provide functionalities for a wide range of use-cases. We have enabled facial and object recognition with which the drone will follow the target. To overcome the problem of jamming, we have also configured gesture-based controls. Another feature that we have added is path following, using which a predefined path can be traced and a repetitive process can be thus automated.

i) Advantages of the Proposed System

It takes extensive training and time to learn how to operate the drone safely with the common remote that comes with most drones, and drones can be very dangerous if flown incorrectly or if it goes off-course due to the operator not knowing how to use the remote control. Our project is designed to skip this extreme learning curve by making predefined actions for the drone such as elevate de-elevate move left, right, forward and backward so that the user can simply pick up the drone and start using it without the worry of accidentally thrusting the drone into a tree, damaging the several hundred dollar drone that they just bought minutes after using it. On top of that, the user does not even need to operate any extraneous hardware to perform those actions; they simply need to use their hands. Our gesture schema has been set up to be accessible to any and all that have full motion of all of their fingers. Thus, our system tackles the problem of extensive training requirements to operate the drones with our easy-to-use hand gesture control mode of operating anyone with the basic understanding and one-time training can operate the drone 9 also the functionality of automating the repeated process of tracing a predefined path is extremely advantageous for surveillance purposes, such as in agricultural farms to prevent crop damage by animals or track infiltrators etc moreover in cases of military counter-insurgency operations 10 Our proposed system could prove to be a great asset as it can locate the positions of infiltrators without risking the lives of our soldiers and overcome the hindrances in operating the traditionally available remote-controlled drones caused by low-cost jamming equipment, which is readily available and used by infiltrators to block the radio signals from the remote control. To conclude, our project is going to be a new way to interact with drones, making it user-friendly .

2.2.1.2 Project Scope

Facial recognition answers are anticipated to be found in 1.33 billion gadgets by 2024. Powered by AI, facial recognition software in cell phones is already being utilized by corporations like iProov and Mastercard to authenticate bills and different high-give-up authentication tasks.

2.2.2 Overall Description

We are proposing a small indoor drone that is entirely driven with hand gestures. The project will have a user-friendly webcam-based GUI that will communicate with the drone. The GUI's main component will be the webcam, along with other indicators showing the drone's current status and useful live information. From the user's end, the user will perform the desired hand gesture and the drone will react accordingly. For example, the user will signal a thumbs up and the drone will respond, within a reasonable response time, and increase its flying altitude. We have a set number of hand movements we plan to incorporate [5]. As the project develops, we will add functionality and push ourselves to create as many movements as possible. As a group of computer engineers, we have a fundamental understanding of the programming and electrical skills necessary for this project. We plan to develop the flight controller ourselves and incorporate and apply the corrective features of a closed loop system. Furthermore, this project will allow us to work with and learn about popular technologies of the time, including Computer Vision and Machine Learning. Both are emerging industries and are growing rapidly. As this is a self-funded project, our goal is to make this project as low cost as possible. This will ensure that the product can be supremely accessible to the public and can be improved upon moving past our first prototype.

2.2.2.1 Product Perspective

Our team wanted to build a product that combined every aspect of our computer engineering coursework with this in mind we were able to collaborate on the idea to utilize deep learning and machine learning to build a useful and sound product because we as a team believed that drones can sometimes be difficult and so wanted to offer the ability to control a drone with a much simpler interaction this allows for us to build in actions for the drone that will automatically account for stabilization and move as expected without having to try to keep the drone flat and level via remote control this product has not yet found its way on the market and we want to be the first to make this a reality our plan to create a widely marketable product forced us to take into account the usability of the product the cost and the ability to use devices that customers were already familiar with to allow for an easier interaction with the drone we were able to fulfill all of those goals in our project plan we were able to maximize usability by creating hand gestures to control the drone that are generally universal meaning that people around the globe can understand many of the basic gestures and would immediately think to use those gestures to operate the drone in order to make the cost of the drone low we were taking into account the cost of each and every component when planning our prototype and once the build process and components list is finalized we will be able to improve upon that even further since our product requires the use of an external device we decided to make it so that other people can simply install the software required to operate the drone on their local laptops or pcs this allows for users to interact with something they are familiar with and make the drone user experience more seamless.

2.2.2.2 Product Features

It takes extensive training and time to learn how to operate the drone safely with the common remote that comes with most drones, and drones can be very dangerous if flown incorrectly or if it goes off-course due to the operator not knowing how to use the remote control. Our project is designed to skip this extreme learning curve by making predefined actions for the drone such as elevate, de-elevate move left, right, forward, and backward so that the user can simply pick up the drone and start using it without the worry of accidentally thrusting the drone into a tree, damaging the several hundred dollar drone that they just bought minutes after using it. On top of that, the user does not even need to operate any extraneous hardware to perform those actions; they simply need to use their hands. our gesture schema has been set up to be accessible to any and all that have full motion of all of their fingers thus our system tackles the problem of extensive training requirements to operate the drones with our easy-to-use hand gesture control mode of operating anyone with the basic understanding and one-time training can operate the drone 9 Furthermore, the functionality of automating the repeated process of tracing a predefined path is extremely advantageous in surveillance purposes, such as in agricultural farms to prevent crop damage by animals, or tracking infiltrators etc., in cases of military counter-insurgency operations. Our proposed system could prove to be a great asset as it can locate the positions of infiltrators without risking the lives of our soldiers and overcome the hindrances in operating the traditionally available remote-controlled drones caused by low-cost jamming equipment, which is readily available and used by infiltrators to block the radio signals from the remote control. To conclude, our project is going to be a new way to interact with drones, making it user-friendly.

2.2.3 External Interface Requirements

2.2.3.1 User Interfaces

TABLE 3: User Interfaces

Function 1	Login/Signup
Input	Name, email address, and password of the user
Processing	Validate the given details and record the information into the database.
Output	Redirect to user's home page

2.2.3.2 Hardware Interfaces

Processor: Intel Core i3 or more.

RAM: 4GB or more.

Hard disk: 250 GB or more.

Drone

2.2.3.3 Software Interfaces

Operating System: Windows 10, 7, 8, MAC

OS. Python.

Anaconda.

Spyder, Jupyter notebook, Flask.

MYSQL.

i) Functional Requirements:

a. Fundamental skill and Knowledge in Technical Field

b. System should be able to match required configurations.

c. data structures to learn about the complex nature of neural networks.

2.2.4 Other Non-functional Requirements

a. The reliability of the product will be dependent on the accuracy of the data.

b. Website is hands on or friendly so that customers can view / use it easily.

c. CPU usage and memory occupation must be less.

2.2.4.1 Safety Requirements

After a gesture is recognized and regenerated to a command, like move to the left, the action planner on the drone kicks in to figure the foremost acceptable course of action that satisfies the recent command. During this method, it is imperative for the drone to hold out the action while ensuring safety to itself, encompassing objects, and therefore the environment. Collision to any of those entities probably causes serious injury to the parties concerned, that is extremely undesirable. In our framework, the action of coming up with a module requires the drone to utilize its sensors (e.g., camera and proximity)to estimate the world wherever it will safely fly or hover. Collision dodging may be a topic addressed in AI. Drones square measure way more liable to external factors that cause their movements to be unstable, like wind or air flows. Collision dodging in drones needs additional considerations for such factors. whereas some approaches rely on the on-board camera for this task others propose the utilization of additional advanced sensors, like ultra sonic optical masers vary [6]. One limitation of camera-based solutions is that they'll perform poorly once there are optical noises, like in low lighting or foggy environments. Victimization of additional non-vision primarily based sensors helps alleviate this downside, however adds additional load to the general weight of the drone, which can not forever be possible.

2.3 Cost Estimation

Cost estimation is done using cocomo model

TABLE 4: Cost estimation

Cost Drivers	**Ratings**					
	Very Low	**Low**	**Nominal**	**High**	**Very High**	**Extra High**
Product attributes						
Required software reliability	0.75	0.88	1.00	1.15	1.40	
Size of application database		0.94	1.00	1.08	1.16	
Complexity of the product	0.70	0.85	1.00	1.15	1.30	1.65
Hardware attributes						
Run-time performance constraints			1.00	1.11	1.30	1.66
Memory constraints			1.00	1.06	1.21	1.56
Volatility of the virtual machine environment		0.87	1.00	1.15	1.30	

Required turnabout time		0.87	1.00	1.07	1.15	
Personal attributes						
Analyst capability	1.46	1.19	1.00	0.86	0.71	
Applications experience	1.29	1.13	1.00	0.91	0.82	
Software engineer capability	1.42	1.17	1.00	0.86	0.70	
Virtual machine experience	1.21	1.10	1.00	0.90		
Programming language experience	1.14	1.07	1.00	0.95		
Project attributes						
Use of software tools	1.24	1.10	1.00	0.91	0.82	
Application of software engineering methods	1.24	1.10	1.00	0.91	0.83	
Required development schedule	1.23	1.08	1.00	1.04	1.10	

The Intermediate Cocomo formula now takes the form:

$$\mathbf{E = a_i (kloc)^{(b_i)} . EAF}$$

Using the above calculation we found that the total time period of the project is around 6 months, the per month cost comes out to be Rs.12,000/- so the total comes to be Rs.72,000/-.

i) FEASIBILITY STUDY

This system is feasible for all healthcare departments like lab, hospital and clinic, bank etc. and this system will be used without experts in that field who have knowledge about using online services which will help to use this system. Any generation of people will use this system on PCs.

ii) TECHNICAL FEASIBILITY

The system must be evaluated from the technical point of view first. The assessment of this feasibility must be based on an outline design of the system requirement in the terms of input, output, programs and procedures. Having identified an outline system, the investigation must go on to suggest the type of equipment, required method developing the system, or running the system once it has been designed.

- Technical issues raised during the investigation are:
- Is the existing technology sufficient for the suggested one?
- Can the system expand if developed?

The project should be developed such that the necessary functions and performance are achieved within the constraints. The project is developed within the latest technology. Though the technology may become obsolete after some period of time, due to the fact that newer versions of the same software support older versions, the system may still be used. So there are minimal constraints involved with this project. The system has been developed using Python. The project is technically feasible for development.

iii) ECONOMIC FEASIBILITY

The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on a project, which will give best return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require.

The following are some of the important financial questions asked during preliminary investigation:

- The costs conduct a full system investigation.
- The cost of the hardware and software.
- The benefits in the form of reduced costs or fewer costly errors.

Since the system is developed as part of project work, there is no manual cost to spend for the proposed system. Also all the resources are already available, it gives an indication that the system is economically possible for development.

iv) BEHAVIORAL FEASIBILITY

This includes the following questions:

- Is there sufficient support for the users?
- Will the proposed system cause harm?

The project would be beneficial because it satisfies the objectives when developed and installed. All behavioral aspects are considered carefully and conclude that the project is behaviorally feasible.

2.4 Risk Analysis

Regardless of the prevention techniques employed, possible threats that could arise inside or outside the organization need to be assessed. Although the exact nature of potential disasters or their resulting consequences are difficult to determine, it is beneficial to perform a comprehensive risk assessment of all threats that can realistically occur to the organization. Regardless of the type of threat, the goals of business recovery planning are to ensure the safety of customers, employees and other personnel during and following a disaster.

The relative probability of a disaster occurring should be determined. Items to consider in determining the probability of a specific disaster should include, but not be limited to: geographic location, topography of the area, proximity to major sources of power, bodies of water and airports, degree of accessibility to facilities within the organization, history of local utility companies in providing uninterrupted services, history of the area's susceptibility to natural threats, proximity to major highways which transport hazardous waste and combustible products. Potential exposures may be classified as natural, technical, or human threats. Examples include:

Natural Threats: internal flooding, external flooding, internal fire, external fire, seismic activity, high winds, snow and ice storms, volcanic eruption, tornado, hurricane, epidemic, tidal wave, typhoon.

Technical Threats: power failure/fluctuation, heating, ventilation or air conditioning failure, malfunction or failure of CPU, failure of system software, failure of application software, telecommunications failure, gas leaks, communications failure, nuclear fallout.

Human Threats: robbery, bomb threats, embezzlement, extortion, burglary, vandalism, terrorism, civil disorder, chemical spill, sabotage, explosion, war, biological contamination, radiation contamination, hazardous waste, vehicle crash, airport proximity, work stoppage (Internal/External), computer crime.

All locations and facilities should be included in the risk analysis. Rather than attempting to determine exact probabilities of each disaster, a general relational rating system of high, medium and low can be used initially to identify the probability of the threat occurring. The risk analysis also should determine the impact of each type of potential threat on various functions or departments within the organization [7]. A Risk Analysis Form, found here(PDF Format), can facilitate the process. The functions or departments will vary by type of organization. The planning process should identify and measure the likelihood of all potential risks and the impact on the organization if that threat occurred.

To do this, each department should be analyzed separately. Although the main computer system may be the single greatest risk, it is not the only important concern. Even in the most automated organizations, some departments may not be computerized or automated at all. In fully automated departments, important records remain outside the system, such as legal files, PC data, software stored on diskettes, or supporting documentation for data entry. The impact can be rated as: 0= No impact or interruption in operations, 1= Noticeable impact, interruption in operations for up to 8 hours, 2= Damage to equipment and/or facilities, interruption in operations for 8 - 48 hours, 3= Major damage to the equipment and/or facilities, interruption in operations for more than 48 hours. All main office and/or computer center functions must be relocated. Certain assumptions may be necessary to uniformly apply ratings to each potential threat.

Following are typical assumptions that can be used during the risk assessment process:

i) Although impact ratings could range between 1 and 3 for any facility given a specific set of circumstances, ratings applied should reflect anticipated, likely or expected impact on each area.

ii) Each potential threat should be assumed to be "localized" to the facility being rated.

iii) Although one potential threat could lead to another potential threat (e.g., a hurricane could spawn tornados), no domino effect should be assumed.

iv) If the result of the threat would not warrant movement to an alternate site(s), the impact should be rated no higher than a "2."

3. METHODOLOGY ADOPTED

3.1 Investigative Techniques

i) Haar Cascading

The waterfall classifier has a group of stages, each stage is full of sensitive pupils. Weak starters that are often simple classifiers and classify as option stubs. A technique called momentum is used to plan each level. Impulse-ing implies a weighted measure of preference exercised by frail students. The most reliable classifier can be planned. The domain is identified by each stage of the classifier as the current sliding window condition. Maybe it's optimistic or bad. If it is positive, it was found in the sharp object. Coincidentally, the negative article was not found at the time. Zone grouping ends at the stage where the label is negative. At this point, the handler moves the windows to the mesh position. The classifier generally shifts the range to the net level. At the first opportunity, the phases have harmful examples open. In order to plan the course classifier, a lot of positive and negative images are needed. One of Viola and Jones' responsibilities was to use region summary tables that accounted for integral images. Integral images can be defined as two-dimensional question tables as a matrix with an identical scale to the first image.

Face Recognition For face recognition, the preferred method is the Viola-Jones algorithm. In general, this algorithm is not limited to face recognition, but can also be used for many esoteric object recognition problems. The Viola-Jones algorithm consists of three main concepts that enable the development of real-time face detectors: haar-like features, image integration, adaboost training, and cascading classifiers. These features allow the system to determine whether a human face is present or not. Haar-like features Haar-like features are used by the Haar cascade classifier to recognize human faces. There are three forms with characteristics similar to haar. In Figure 4, the first type is an edge object, the second type is a line object, and the last type is a four rectangular object. Using image integration, Haar's principle provides a fast computation. It is said to have the same function as haar.

ii) Media Pipe

Gesture recognition can be an engineering topic alongside speech technology for the purpose of translating single bits using mathematical algorithms. A subfield of computer vision. Gestures usually return from any movement or body position, regardless of how they appear on the face or hand. Approaches in this area include emotional recognition of faces and hands [8]. The user can use Easy Bit to control or interact with the device without even touching the device. Various sacrificial strategies, cameras, and computer vision algorithms have been developed to translate sign language. It uses a media pipeline algorithm for gesture recognition. MediaPipe Hands can be a reliable solution for hand and finger tracking devices. Capture 21 original 3D hand signals in one frame using machine learning (ML). While current detection methods are primarily based on powerful desktop computers, our approach benefits from the real-time performance of mobile phones and multi-hand scaling. We hope you can incorporate this practical plan into in-depth analysis and development. Society can end up in cases of misuse to push new applications and new forms of analysis. MediaPipe Hands uses a dedicated built-in CC pipeline of the different models working together: The palm detection model that works on the whole image returns the controlled manual bind box directly. Hand gesture model applicable to the image cutting area delineated by a palm detector once it returns 3D hand key points with high reliability. This strategy is analogous to that used in our MediaPipe face mesh resolution, using a face detector and a face detector as the reference model.

3.2 Proposed System

In this work, we studied the overall performance of deep neural networks for face re-identity particularly designed for low-nice snap shots, and implemented them on a drone. We also added some other functionality that is line following, Applications for this functionality include things like product delivery, warehouse management, library management, etc. Keeping in mind the problems mentioned above, we aim to provide a well-equipped, fully functioning, and interactive drone for public safety and surveillance. Our project is designed to skip this extreme learning curve by making predefined actions for the drone, such as elevating, de-elevate, and moving left, right, forward, and backward. The operator only needs to get familiar with hand gestures. We provide functionalities for a wide range of use-cases. We have enabled facial and object recognition with which the drone will follow the target. To overcome the problem of jamming, we have also configured gesture-based controls. Another feature that we have added is path following, using which a predefined path can be traced and a repetitive process can be thus automated.

3.3 Work Breakdown Structure

A. Identification, Formulation and Planning of the Project
   i. Capstone Team formation
   ii. Mentor Selection
   iii. Brainstorming and Idea Selection
   iv. Research and Literature Survey

B. Generating Dataset of Various Hand Gestures
   i. Generating dataset for various hand gestures
   ii. Generating dataset for face tracking system

C. Training Deep Learning Model
   i. Train model on the generated dataset
   ii. Improving accuracy of the model
   iii. Deploying the model

D. Drone Programming and Incorporating Various Functionalities
   i. Acquiring drone
   ii. Checking the working of the drone by performing all maneuvers
   iii. Programming drone and creating main.py which incorporates all the functionalities

E. Prototype Testing
   i. Connecting the drone through wifi to the laptop
   ii. Running the programme and testing the prototype

F. Performing Modifications
   i. Checking for errors
   ii. Performing all required modifications

G. Results Evaluation

i. Documenting the final working of the project

ii. Noting down various observations and conclusions

H. Final Report

i. Compiling all reports, diagrams and final observations to generate the final report

3.4 Tools and Technology

1. Processor: Intel Core i3 or more.
2. RAM: 4GB or more.
3. Hard disk: 250 GB or more.
4. Operating System: Windows 10, 7, 8.
5. Python.
6. Anaconda.
7. Spyder, Jupyter notebook, Flask.
8. MYSQL

4.1 System Architecture

i) System flowchart: A stream diagram could even be a spread of characteristics that address a standard or strategy showing the proposes that as boxes of shifted types and their solicitation by interfacing them with bolts this depict outline shows a response for a given recoil procedure exercises territory unit depict in these compartments and bolts rather there calm by the sequencing of undertakings flowcharts region unit used in taking apart emerging with documenting or managing the lone way or program in a few fields.

ii) Arrows: Showing "stream of the board" partner bolt returning from one picture and finishing at another picture addresses that control passes to the picture the bolt focuses to. the street for the bolt is strong or broken. The significance of the bolt with broken line could differ from one stream diagram to an uncommon and ought to be laid out inside the legend.

iii) Generic processing steps: Addressed as square shapes Examples: "Add one to X"; "supplant known part"; "save changes" or comparable.

iv) Subroutines: Tended to as square shapes with twofold influenced vertical edges these are adjusted show tangled connection steps which may be included during an exceptionally particular language model live records one bundle may require various clear area centers or leave streams see co day by day follow forward therefore these are showed up as checked wells inside the quadrangle and subsequently the leaders bolt interface with these wells.

v) Input/output: Tended to as a quadrilateral Examples: Get X from the customer; show X Prepare prohibitively drawn as a two-dimensional figure Shows exercises that don't have any outcome close to fitting a cost for a later unforeseen or elective development (see under).

vi) Conditional or decision: Tended to as a gem rhombus showing wherever a decision is fundamental typically a certifiable request or genuine bogus check the prohibitive picture is whimsical during this its two bolts start of it to a great extent from total base explanation and right explanation one adore affirmed or valid and one love no or bogus the bolts had the opportunity to be named more than two bolts may moreover be utilized in any case this is regularly frequently regularly frequently generally a simple marker that a tangled choice is being taken inside that case its having the possibility to should be isolated any or replaced with the pre-portrayed live picture [9].

vii) Junction symbol: For the most part portrayed with a dark mass showing any place different administration streams meet during one leave stream an intersection picture can have very one bolt returning into it anyway only one going out in direct cases one may simply have partner bolt reason to an exceptional bolt all things being equal these are supportive to address partner monotonous strategy what in designing is named a circle a circle may for instance incorporate a connective any place the board first enters measure steps a contingent with one bolt leaving the circle and one returning to the connective for added clearness where 2 lines incidentally cross inside the drawing one through and through them could even be drawn with minimal plane figure over the other showing that no intersection is assumed.

viii) Labeled connectors: Addressed by A distinctive mark inside a circle. marked connectors are used in cutting edge or multi-sheet charts to fill in for bolts. for each name, the "surge" connective ought to be unmistakable, anyway there's additionally such a "inflow" connectors. During this case, an intersection up to the hustle stream is known.

ix) Concurrency symbol: Addressed by a twofold cross-over line with any scope of section and leave bolts These images are utilized at whatever point 2 or extra administration streams ought to work at a comparable time. The leave streams are enacted at an identical time once the entirety of the passage streams have arrived at the simultaneousness picture. A simultaneous picture with one section stream could likewise be a fork; one with one live stream could likewise be a piece of. it is fundamental to make sure to keep these associations consistent so as. All cycles need to move from prime to base and left to right.

4.2 Design Level Diagrams

i) ER DIAGRAM

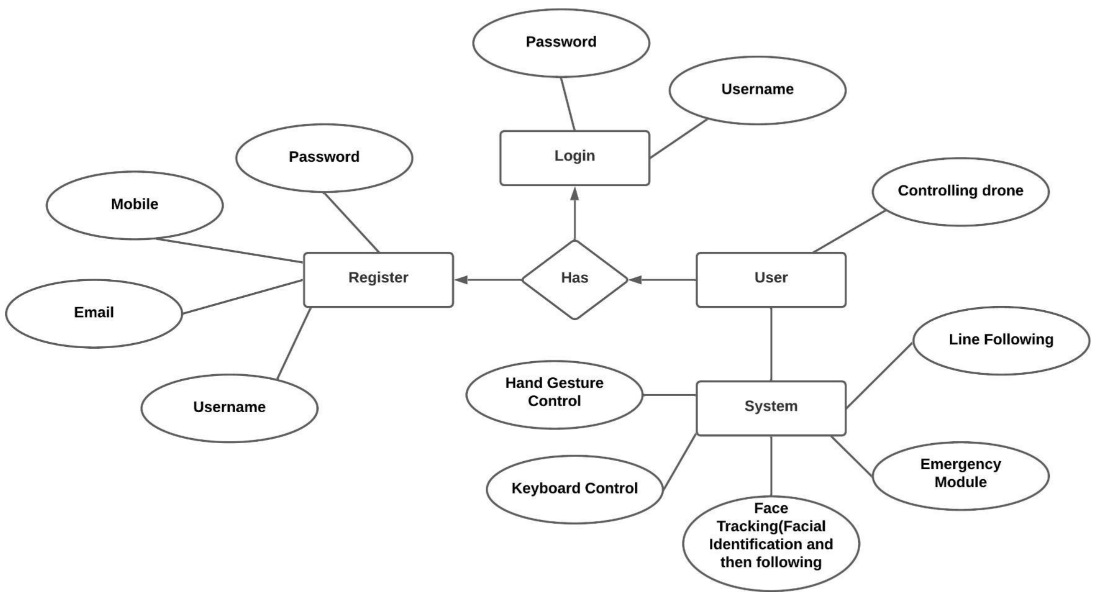

FIGURE 5: ER diagram

The above diagram is an E-R diagram, which has a user and a system. Users have 2 options like Register and Login. If a user wants to use this technology, then they have to register themselves first and, after validating the login credentials, he/she can access the features which we developed. Under the system, there are some points like the system can validate the user credentials and can also handle the features that we build (Hand gesture control, Keyboard control, Face detection and tracking, Line following and Emergency module).

ii) Data Flow Diagram (DFD)

An information programming language did may be a graphical outline of the progression of information through associate in nursing framework an information programming language can likewise be utilized for the visual picture of information measure organized plan its ordinary apply for an architect to draw a setting level did first that shows the collaboration between the framework and out of entryways substances this setting level did is then detonated to implies extra detail of the framework being sculpturesque.

a) DFD level 0

DFD 0 defines the structure of the complete project. Like there is one website where from that user can register,login and can access multiple features which we have. We also connect a database to the system so that we can store all user data in it.

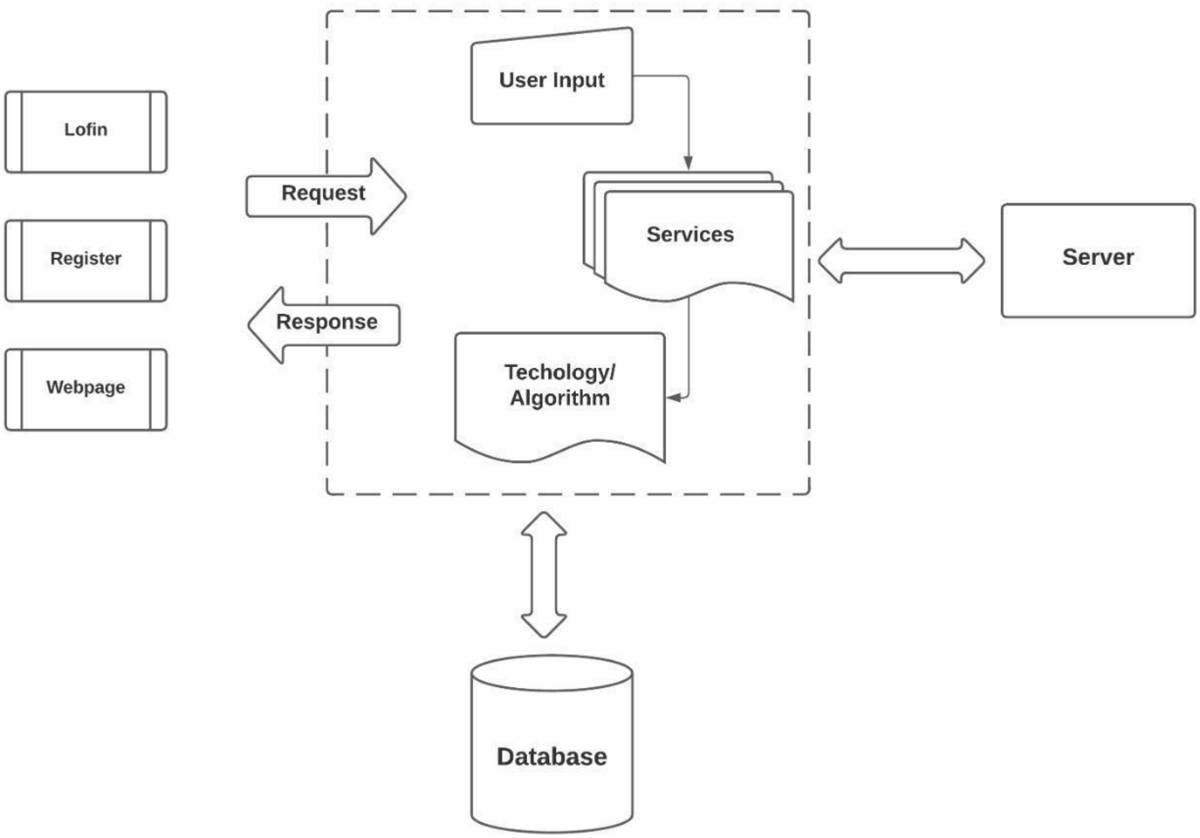

FIGURE 6: DFD architecture

b) DFD of Hand gesture

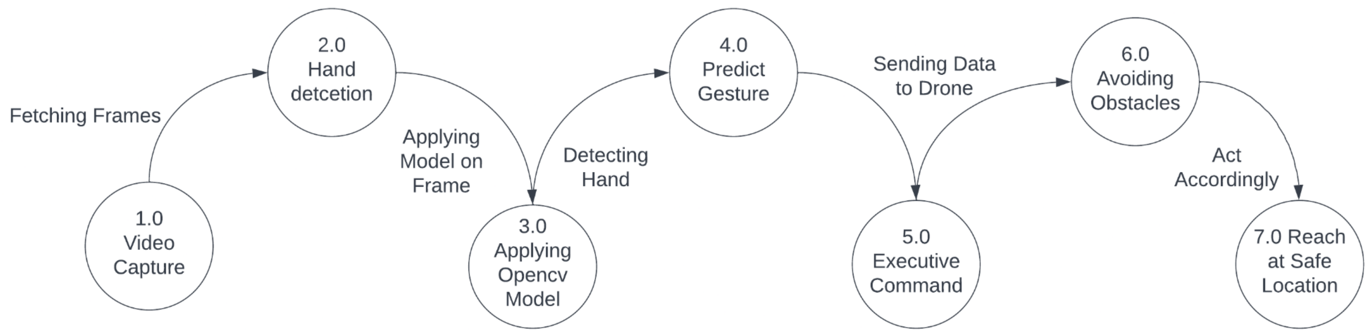

FIGURE 7: DFD architecture

The above diagram is a data flow diagram, which defines how the system can work based on particular data. If a user selects a hand gesture feature, the system will initially open the camera and start capturing the real time video by using open-CV. Then the system will read each frame from that video and apply the media pipeline model to it. which recognizes the gesture and transmits the data to the drone so that the drone can act according to the gesture.

c) DFD of Line follower

Below, DFD is for line followers. After user login, the user selects the line follower feature, and the function starts running in the system. That function has some functionality, like a user can handle drones by using a keyboard. For that, we assign some hot keys to form each drone movement. If the user uses the up arrow key then the drone will take off, if the user uses the left arrow key then the drone will reach the left position like this. So whenever a user presses any hotkey, the system will transmit the data to the drone and the drone will act like that.

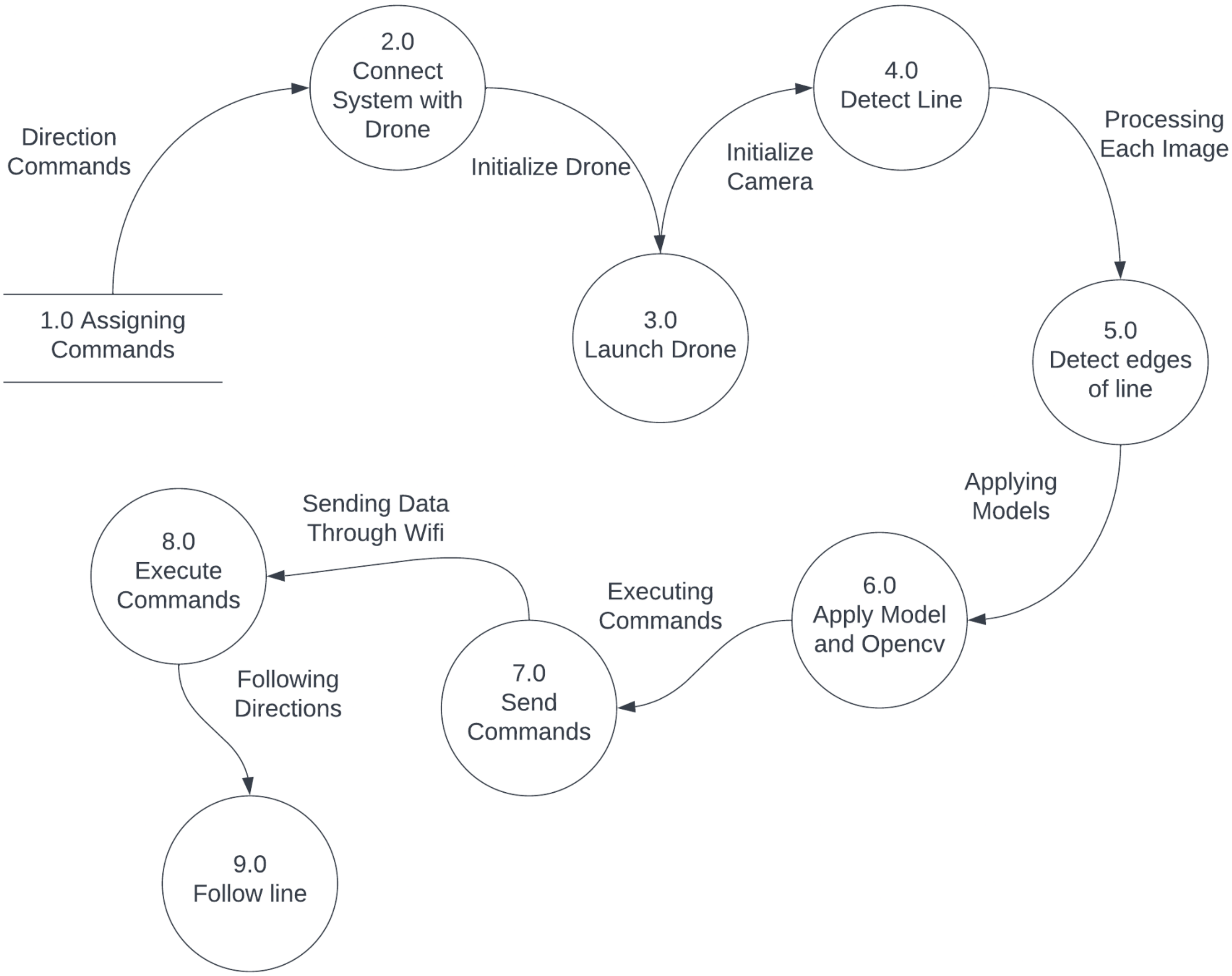

FIGURE 8: DFD-1 architecture

d) DFD of face recognition

Below DFD defines face recognition. When the user selects this function, the system will initialize the camera and start capturing the video by using open-CV. After that, the system will read each frame and apply a haar cascaded model on that which detects the face and will be recognised to the user.

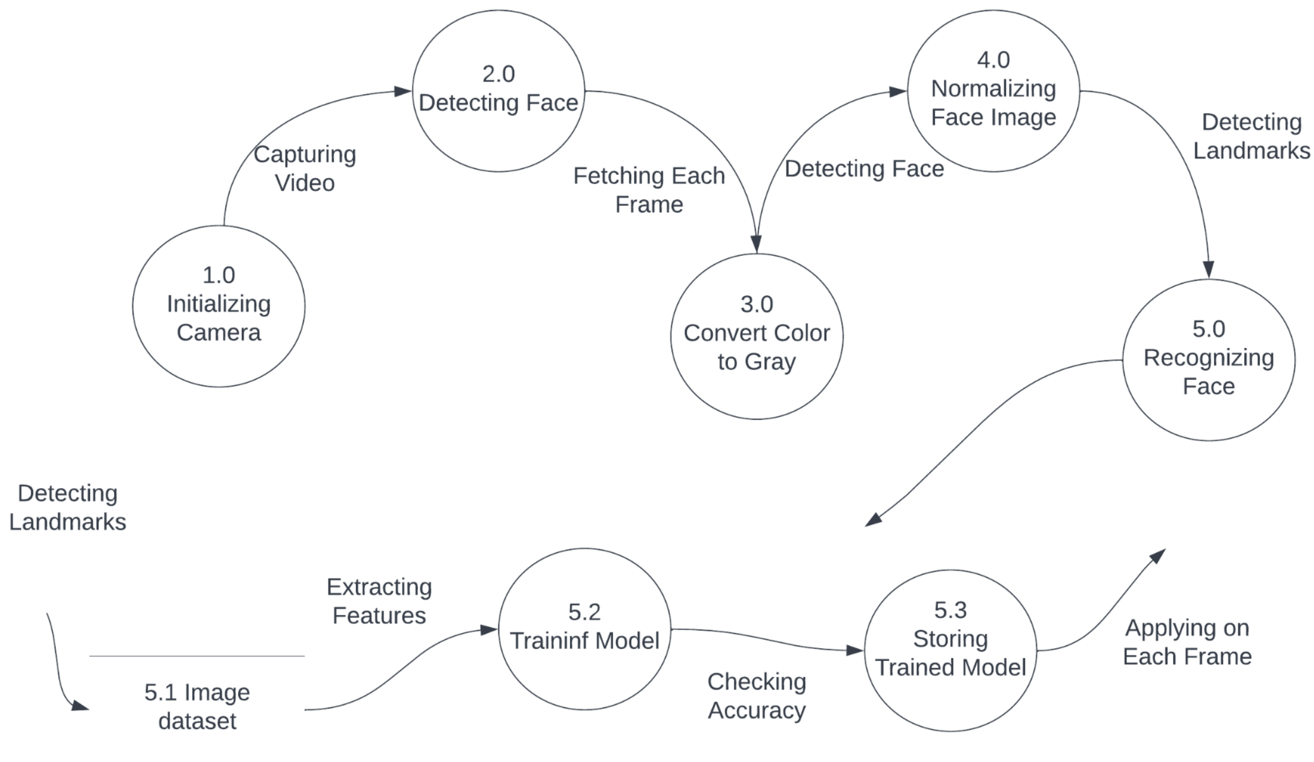

FIGURE 9: DFD-2 architecture

4.3 User Interface Diagrams

i) Flow chart

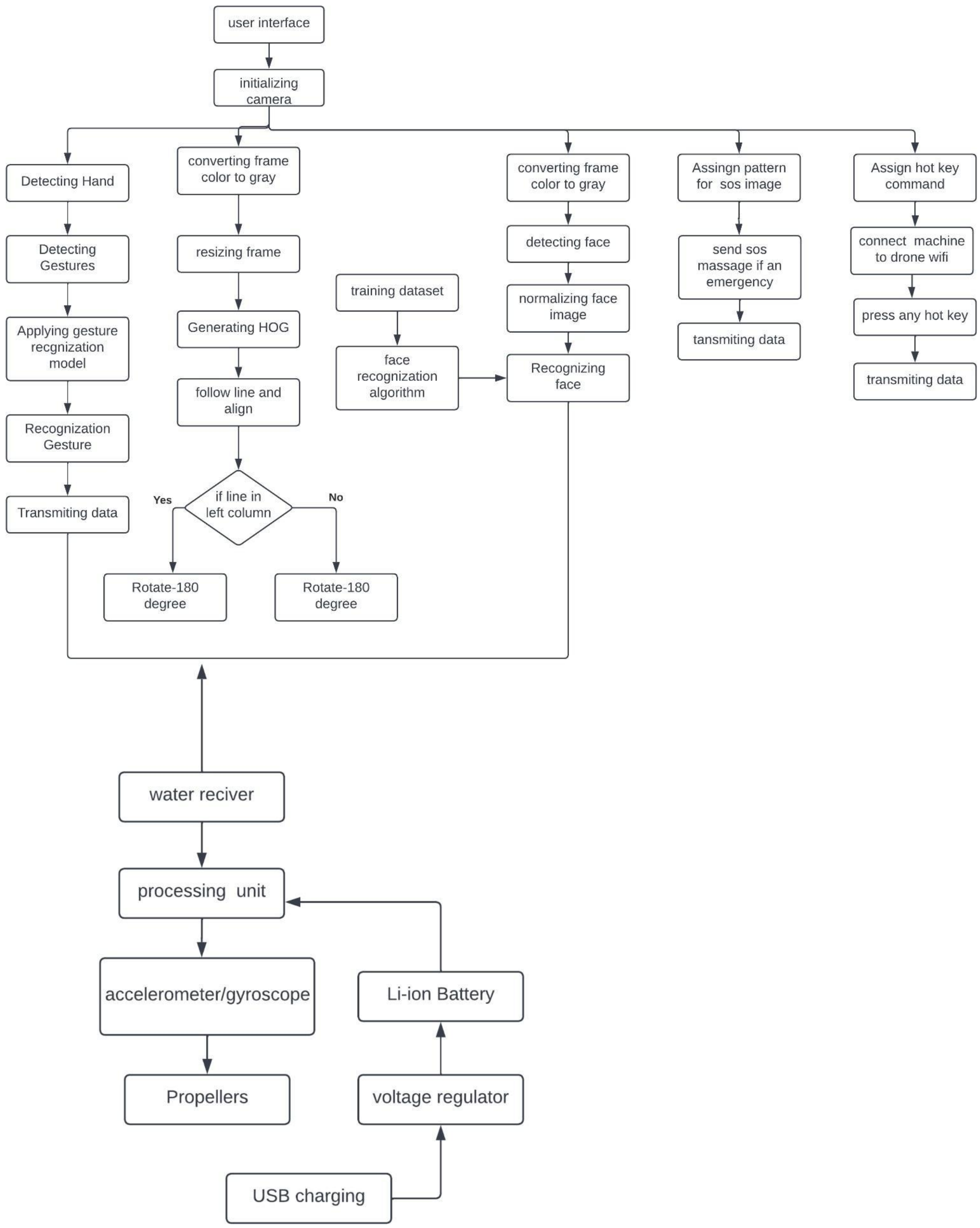

FIGURE 10: Basic UI architecture

The above diagram defines the flow of the system. First the user has to register himself and then the user can login. After the login, there are multiple modules like Hand gesture control, Keyboard control, Face detection and tracking, Emergency module.

ii) Architecture Diagram

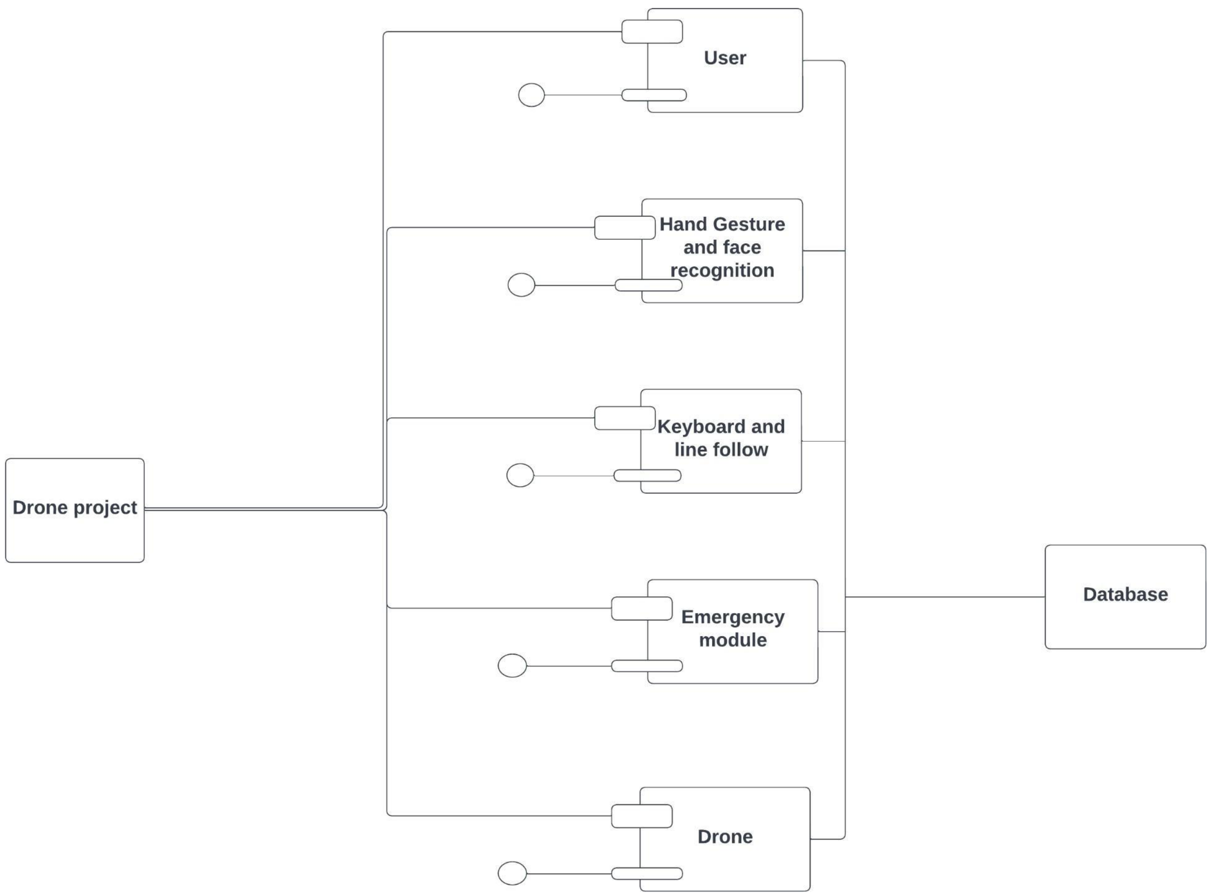

FIGURE 11: System architecture

iii) Architecture Diagram 2

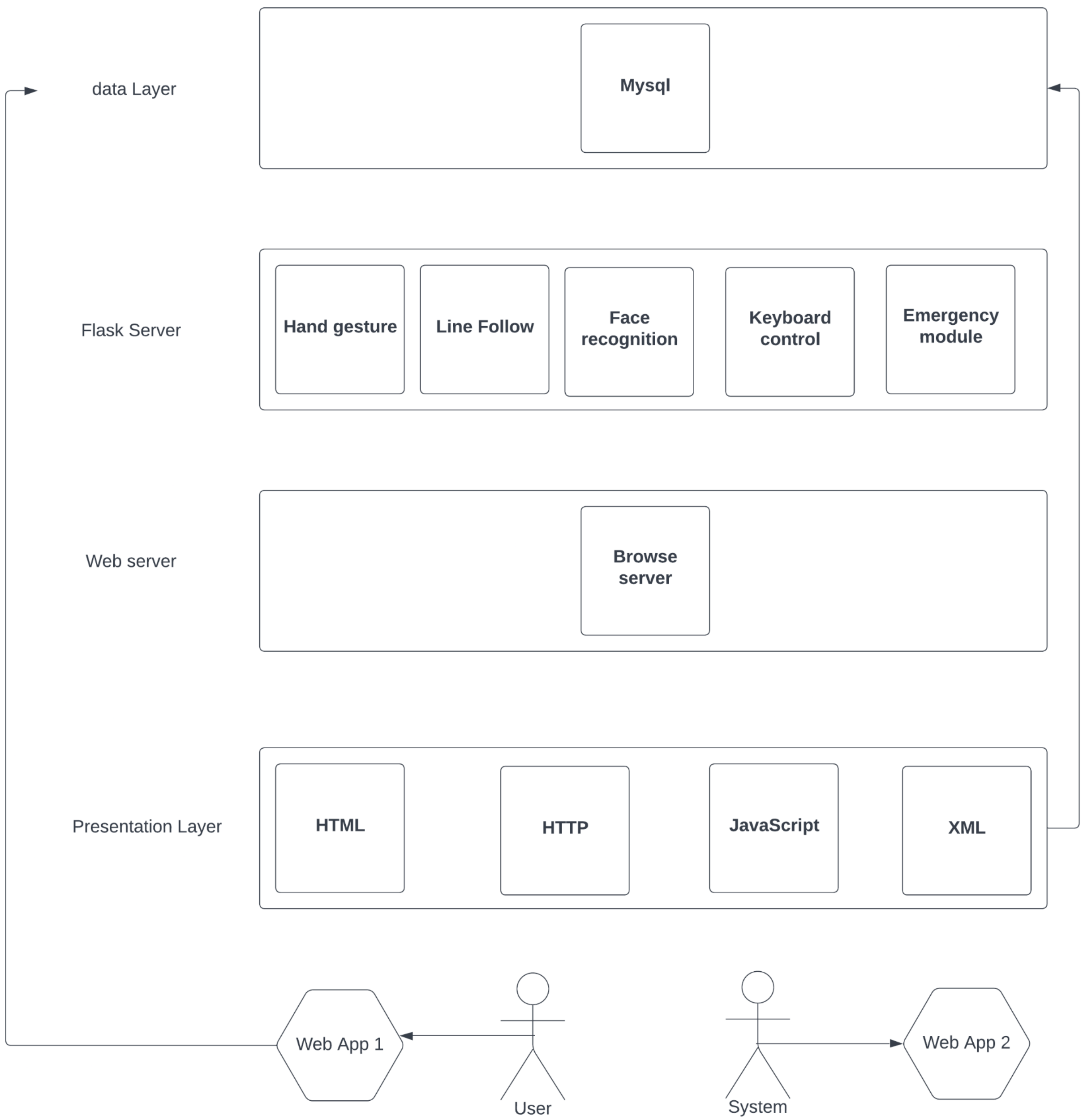

FIGURE 12: System architecture 2

The above diagram is a brief description of the Architecture which was used in the project. We totally have 4 architecture levels. Each architecture has its own functions.

iv) Class Diagram

The class diagram defines the sections used and the systems they have. user can register/login ny using email,password,username,mobile no. The system can validate them and can also recognise faces and gestures. Then drones have some functionalities like following a path, taking off, landing safely, and maintaining altitude under system commands.

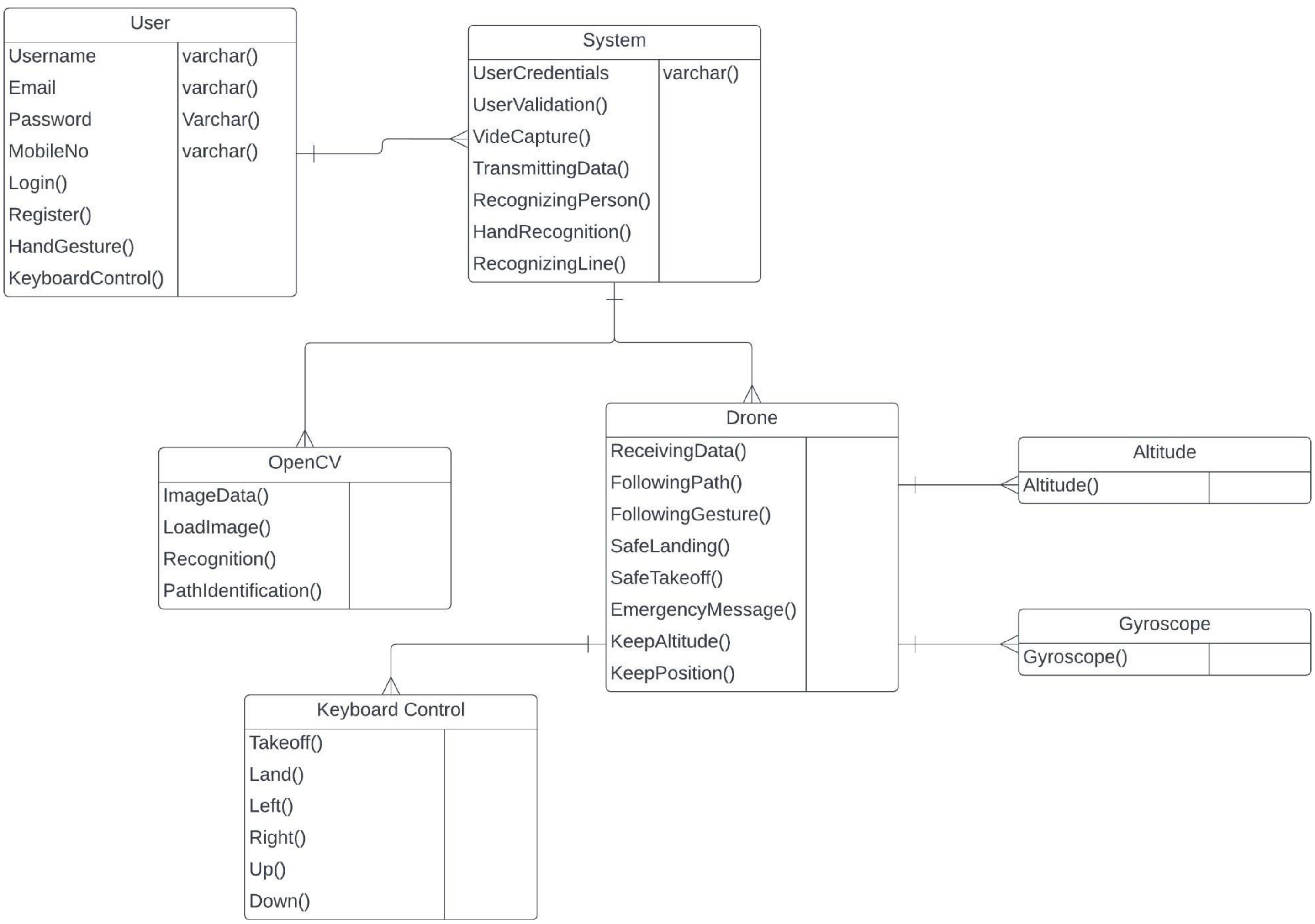

FIGURE 13: Class diagram

v) Activity Diagram

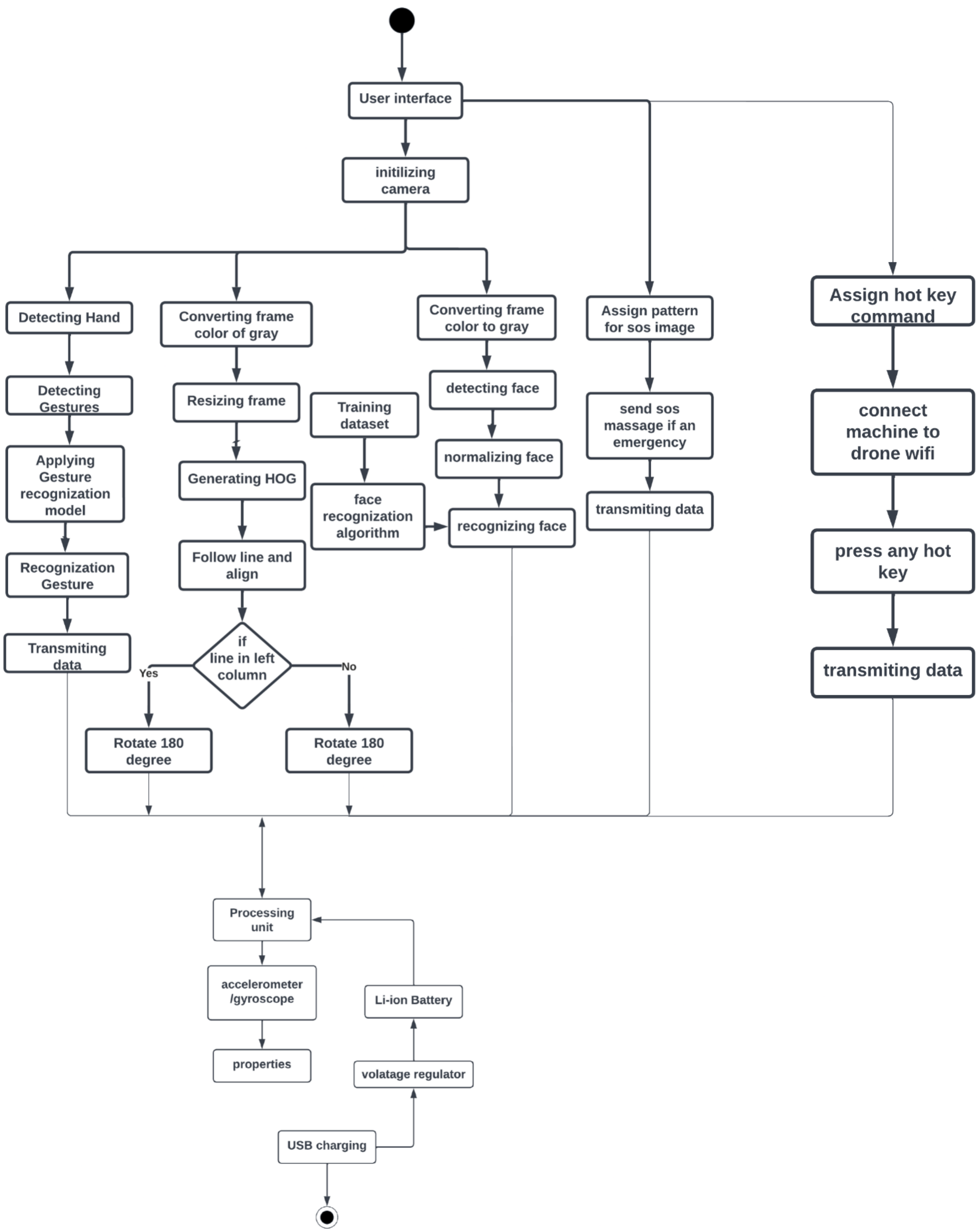

FIGURE 14: Activity diagram

vi) Use Case Diagram

FIGURE 15: Use-case diagram

The above diagram is nothing but a use case diagram, which defines which user has which rights and which system has which rights. SO users can register, login themself. where the system can validate the data and start capturing the video.
After that, the system can be recognized to the user and can recognize the gestures. Also, users have keyboard control rights. By using these, users can control the drone.

vii) State chart diagram

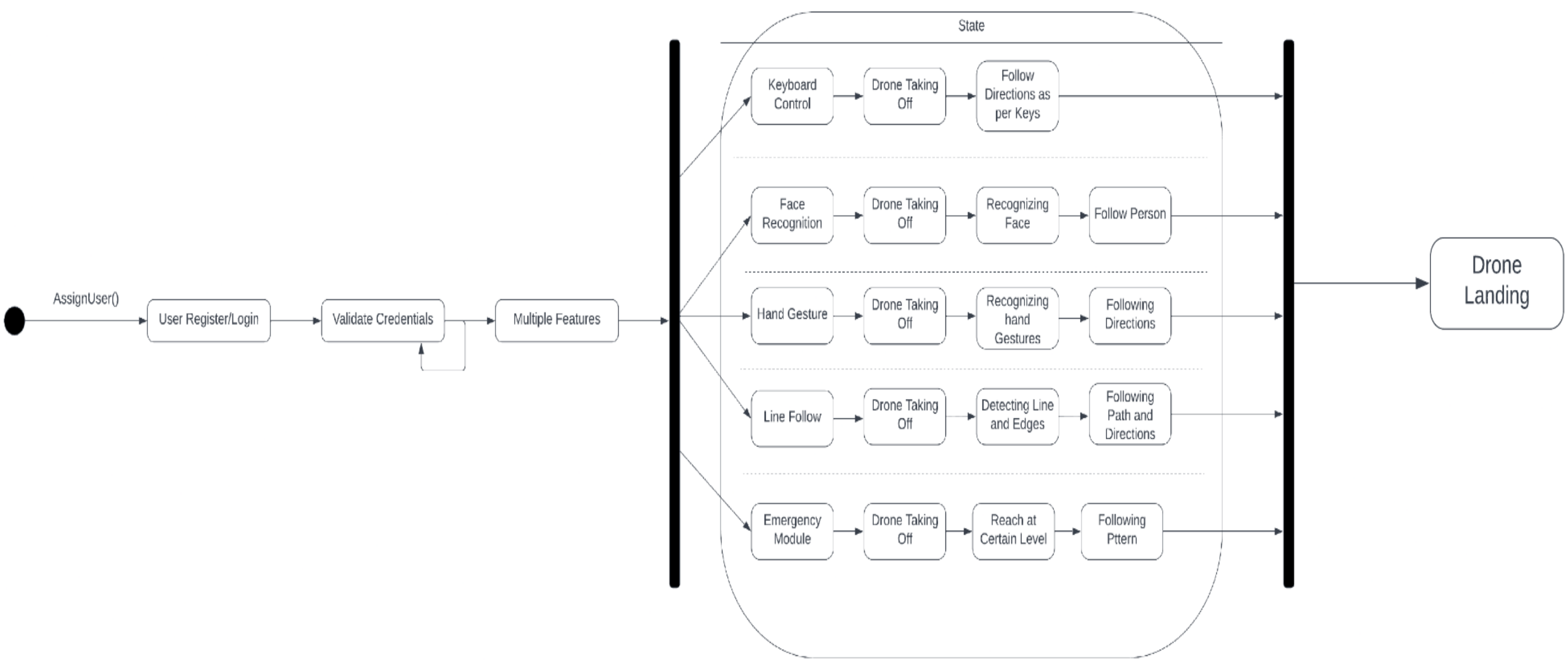

FIGURE 16: State chart diagram

The above diagram is a state chart diagram. There are some states which the system can follow. If a user starts to login, the system will validate the data. If the data is incorrect, the system can then route the user on login. If any of the features is selected by the user, then the first drone will take off, then some functionalities will start happening, and after that, the drone will land. If some functionalities are not working, the drone will automatically land.

viii) Sequence diagram

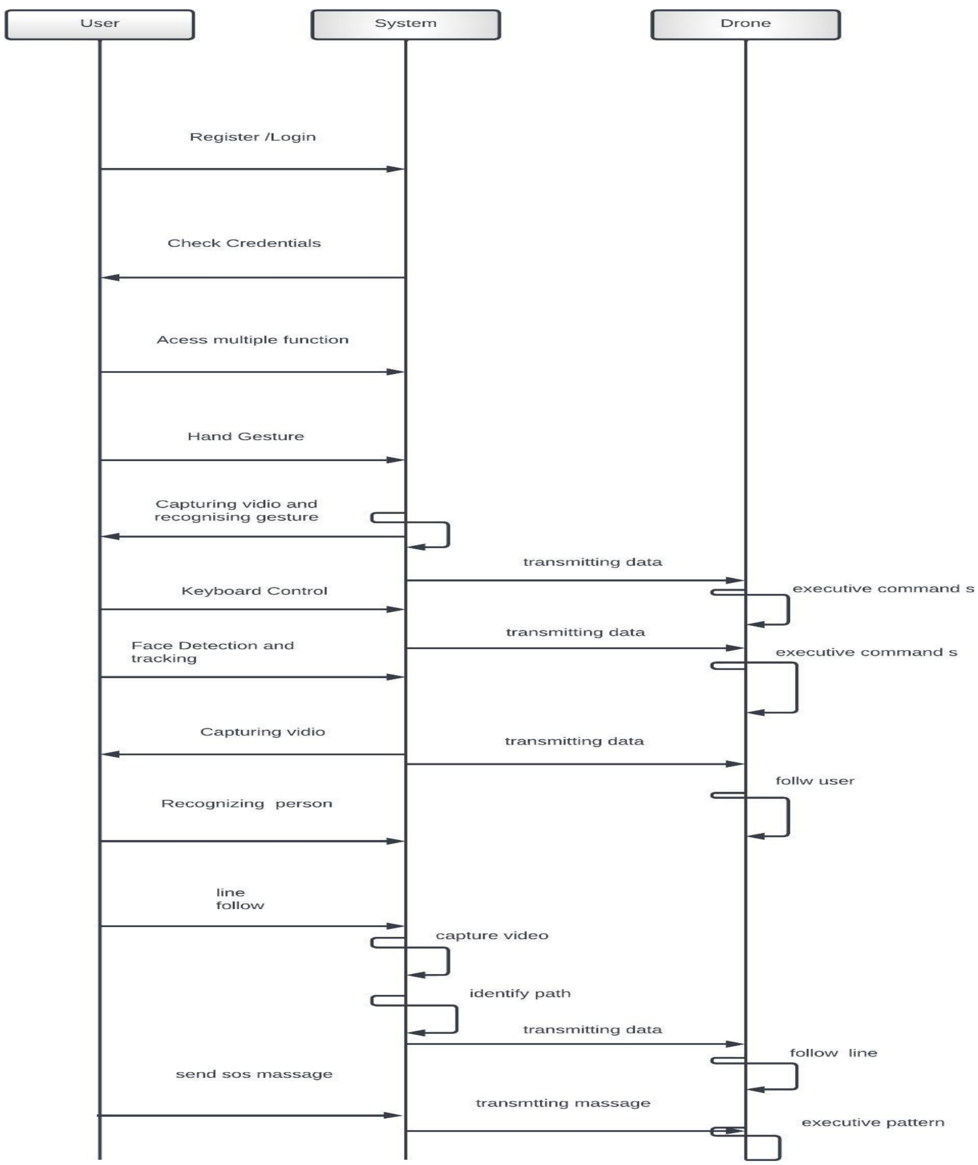

FIGURE 17: Sequence diagram

The above figure is nothing but a sequence diagram This diagram defines the working sequence. So the first user can register and login so things get acknowledged by the system whether credentials are valid or not. Then there are some functionalities, so whenever a user uses any one of the functionalities, the system will accept that request and then acknowledge the user. After acknowledgment, the system will transmit some data to the drone, and the drone will acknowledge the system and the user.

5. IMPLEMENTATION AND EXPERIMENTATION RESULTS

5.1 Experimental Setup

We have built a Smart Drone Control System that can take input from the webcam as the user gives a hand gesture in front of the screen and output the result by controlling the drone for respective motion. Our experimental setup includes capturing images using the webcam of the user's system.

(i) CONTROLLING UAV THROUGH HAND GESTURE

(a) Generation of Dataset for Hand Gesture Recognition

Uses mediapipe and opencv to generate x and y axis coordinates for 21 feature points spread across the hand. It takes a gesture ID as input and logs the x and y coordinates for the 21 feature points. We take 8 gesture ID's and take correspondingly sufficient data for each ID to perform gesture recognition.

(b) Training of a deep learning model to recognise Hand gestures

We use a fully connected 4 dense layer sequential model with 'relu' as activation function in the first two layers and 'softmax' as activation function in the last dense layer. The model inputs 42 features corresponding to the dataset generated as x and y coordinates of feature points. The first two dense layers have 32 nodes each, the third node has 16 nodes and the output layer with softmax as activation function has 8 nodes corresponding to each one of the gesture ID's [10]. The model has a hyperparameter tuning script which changes the model parameter if and when new gestures are to be added so that the model performs accurately even when new data is added.

(c) Programming the UAV to move corresponding to the recognised gesture

The deep learning model returns the gesture ID of the recognised gesture corresponding to which the drone is moved in the corresponding direction with respect to the programmed sensitivity in the direction. In case no gesture is recognised a buffer is maintained and the most common previous gesture is used for the said purpose.

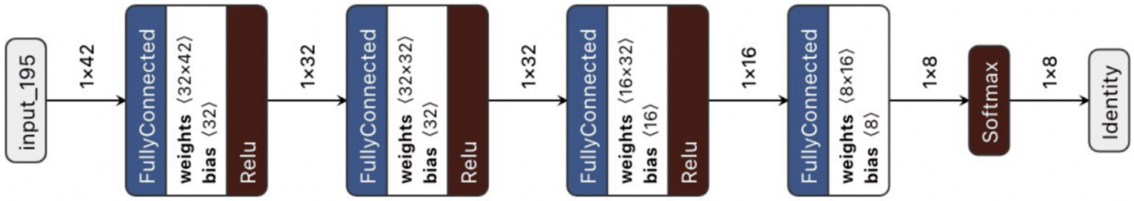

Fig 18. Structure of classification neural network for gesture recognition

a. **ModelCheckpoint:** ModelCheckpoint callback is used in conjunction with training using model.fit() to save a model or weights (in a checkpoint file) at some interval, so the model or weights can be loaded later to continue the training from the state saved.

b. **Batch Normalization:** It is a technique for training very deep neural networks that standardizes the inputs to a layer for each mini-batch. This has the effect of stabilizing the learning process and thus reducing the number of training epochs required to train deep networks.

c. **Dense Layer:** In any neural network, a dense layer is a layer that is deeply connected with its preceding layer which means the neurons of the layer are connected to every neuron of its preceding layer. This layer is the most commonly used layer in artificial neural network networks [11]. The dense layer's neuron in a model receives output from every neuron of its preceding layer, where neurons of the dense layer perform matrix-vector multiplication.

d. **RELU:** The Rectified Linear Unit is the most commonly used activation function in deep learning models. The function returns 0 if it receives any negative input, but for any positive value x, it returns that value back. So it can be written as f(x)=max(0, x).

e. **Dropout Layer:** we have chosen the value of dropout as 0. This implies that 0% of the nodes in the hidden layer have dropped out. This is done to avoid the problem of overfitting.

f. **Early Stopping:** Assuming the goal of a training is to minimize the loss. With this, the metric to be monitored would be 'loss', and mode would be 'min'. A model.fit() training loop will check at the end of every epoch whether the loss is no longer decreasing, considering the min_delta and patience if applicable. Once it's found no longer decreasing, model.stop_training is marked True and the training terminates. In our model, we have used early stopping with a patience of 50 i.e. 50 epochs with no improvement after which training was stopped.

The final output is a result of the Softmax activation layer which gives the probabilities for each class and then we apply the argmax() function to get the maximum probable class for the same.

(ii) CONTROLLING UAV THROUGH KEYBOARD INPUTS

(a) Programming the UAV to move corresponding to the keyboard inputs

The user inputs keyboard keys corresponding to which the drone is moved in the corresponding direction with respect to the programmed sensitivity in the direction.

(b) Pygame is a set of Python modules designed for writing video games. Pygame adds functionality on top of the excellent SDL library. This allows you to create fully featured games and multimedia programs in the python language.

Whenever a key is pressed or released, pygame.event() queue methods pygame. KEYDOWN and pygame.KEYUP events respectively.

(iii) FACE TRACKING BY THE UAV

(a) Tracking the face using haarcascade classifier

The drone is connected and then facial features are tracked and mapped using the frontal haarcascade classifier.s

(b) Programming the UAV to move with the face

The facial features are mapped by the area in the frame to facilitate the movement of the UAV. When the area of the face falls below a threshold in the frame the drone makes corrections in its flight corresponding to the programmed sensitivities to make up for it such that a minimum threshold area remains centered in the drone's frame. Hence, the drone tracks the face.

(c) Tracking the face with the help of face_recognition api

The tracking of the face is done by using face_recognition api. The person and the drone both are in motion so we are using the face_recogition api. First of all the api is capturing the image of the person from the drone feed and after that the image is taken relative to the bottom-right corner of feed to specify the location of the person with respect to the drone. In this process, a specific threshold is defined according to which the drone is made to move such that the person always remains within that threshold and that the person is always tracked [12]. The api has an accuracy of 99.38%. The api gives us an option to use a variety of models, default model is (small) which is the one we used, it only involves 5 key points and is relatively faster as compared to other options. We required a model that runs very fast on the CPU just because the demand of our project is determination of a person is done as fast as possible because the drone also needs to follow the person in the meantime.

(iv) LINE FOLLOWING BY THE UAV

(a) Making external modifications to the drone:

Using a plane mirror and fixing it to the camera head of the drone in such a way that it produces the image of objects directly in the bottom of the drone.

(b) Programming a script to find and tune the (hue saturation values) hsv to follow a line.

To follow a line we use BGR mode where we find the values for maximum and minimum hue, saturation and value such that we can differentiate a black and white line.

(c) Programming UAV to follow a line.

To follow the line we map the contours of the path such that we can calculate the deviation of the drone from the centerline.We divide the image of the path into three parts which correspond to the bit 1 or 0 depending on the area path in the frame. Now we have a three bit output using which we can define eight operations so as to adjust the flight such that the drone remains centered with respect to the path.

5.2 Experimental Analysis

This section includes the details of the data being used to train the models along with their performance parameters.

5.2.1 Data

(i) Hand Gesture Recognition Dataset

Hand recognition detector can add and change training data to retrain the model on their own gestures. But before this, there are technical details of the detector to understand how it works and how it can be improved.

MediaPipe has a python implementation for their Hand Keypoints Detector. It is returning 3D coordinates of **21 hand landmarks**.

In this project, only **2D** coordinates will be used. Here you can see all of 21 key points.

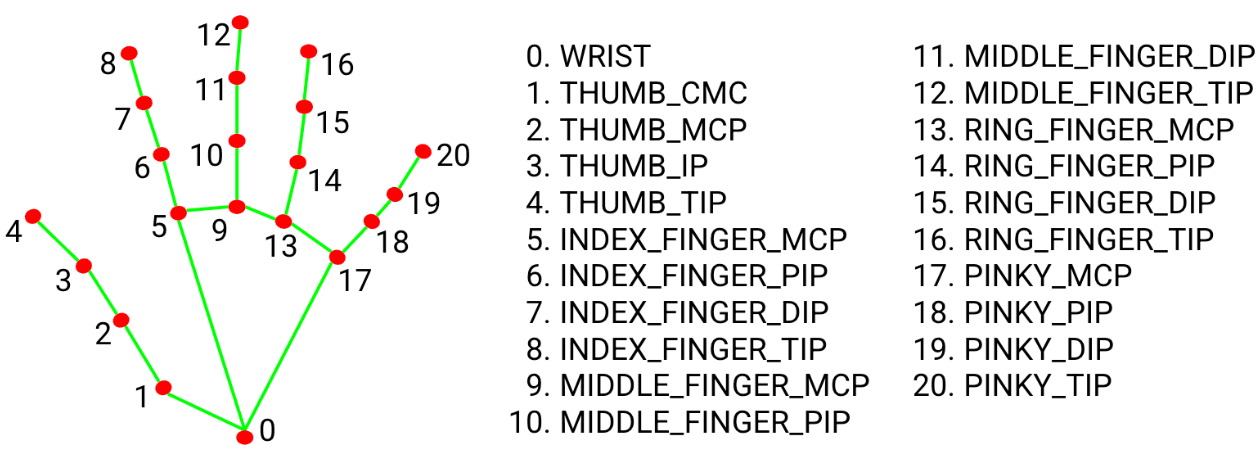

Fig 19. 21 key points of hand using mediapipe

Then, these points are preprocessed for training the model in the following way.

(Landmark coordinates)

ID : 0	ID : 1	ID : 2	ID : 3	……	ID : 17	ID : 18	ID : 19	ID : 20
[551, 465]	[485, 428]	[439, 362]	[408, 307]	……	[633, 315]	[668, 261]	[687, 225]	[702, 188]

(Convert to relative coordinates from ID:0)

ID : 0	ID : 1	ID : 2	ID : 3	……	ID : 17	ID : 18	ID : 19	ID : 20
[0, 0]	[-66, -37]	[-112, -103]	[-143, -158]	……	[82, -150]	[117, -204]	[136, -240]	[151, -277]

(Flatten to a one-dimensional array)

ID : 0		ID : 1		ID : 2		ID : 3		……	ID : 17		ID : 18		ID : 19		ID : 20	
0	0	-66	-37	-112	-103	-143	-158	……	82	-150	117	-204	136	-240	151	-277

(Normalized to the maximum value(absolute value))

ID : 0		ID : 1		ID : 2		ID : 3		……	ID : 17		ID : 18		ID : 19		ID : 20	
0	0	-0.24	-0.13	-0.4	-0.37	-0.52	-0.57	……	0.296	-0.54	0.422	-0.74	0.491	-0.87	0.545	-1

Fig 20. Preprocessing for training the model

The above datasets were used to create different activities. For each of these datasets, 20% of the images were used for testing and 20% of the training dataset was used for validation and the rest was used for training the deep learning model. For achieving better results, data augmentation was applied by cropping, rotating, flipping, and zooming the image of the gestures.

(ii)Face Tracking Dataset

To generate the dataset for face tracking, we have implemented a module to record the images of the user to be tracked. The user enters his details (name) and his/her 10 images are recorded. These images are used to recognise the user and further track the movements of the user.

5.2.1 Performance Parameter

The performance of a neural network depends on the hyper parameters used for its training. These hyperparameters are tuned over multiple training sessions to achieve better results.

i. Learning Rate: Defines the step size at each iteration while moving toward a minimum of a loss function. It is a configurable parameter that is assigned a low value that could take longer to train the model and assigning a high value could result in converging the suboptimal solution. We use a learning parameter of 0.01 while training a model.

ii. Loss Function: The aim of the training is to reduce the value of the loss function by updating the model's weight vector. Most classification problems use a negative log-likelihood function to calculate the loss which is defined as a cross-entropy loss function.

$$J(\mathbf{w}) = -\frac{1}{N}\sum_{i=1}^{N}[y_i\log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)]$$

iii. Batch size: The number of data samples that propagate to the network in a given iteration is defined by the batch size. This cell produces the memory used while training the model and also trains the network faster as weights are updated regularly. We use a batch size of 64 for our final training session.

iv. Optimizers: We use Optimizer to minimize the loss function and find optimal weight values over the course of the training session we employ the Adam Optimizer in the training of a model as it performs a parameter update for each training example and is comparatively faster providing the advantages of both gradient descent with momentum and RMSprop.

The following parameters are used to analyze the performance of our model-

a. Accuracy
b. F1-Score
c. Precision
d. Recall
e. Support

5.3 Working of The Project

The detailed working of the project which includes the procedural workflow and algorithms used is explained in this section.

5.3.1 Procedure Workflow

The workflow of the usage is described below:

i. The new user goes to the github and clones the given repository.
ii. The user now installs the setup on their system..
iii. The user has to have a DJI-Tello drone.
iv. The user connects the system with the drone Wi-Fi.
v. The user runs the app.py site and signs up on the interface.
vi. The user selects the module he wants to use and runs it.
vii. The user commands the drone to take-off and runs it simultaneously with the software.
viii. The user output is logged as per the flow-chart given below.

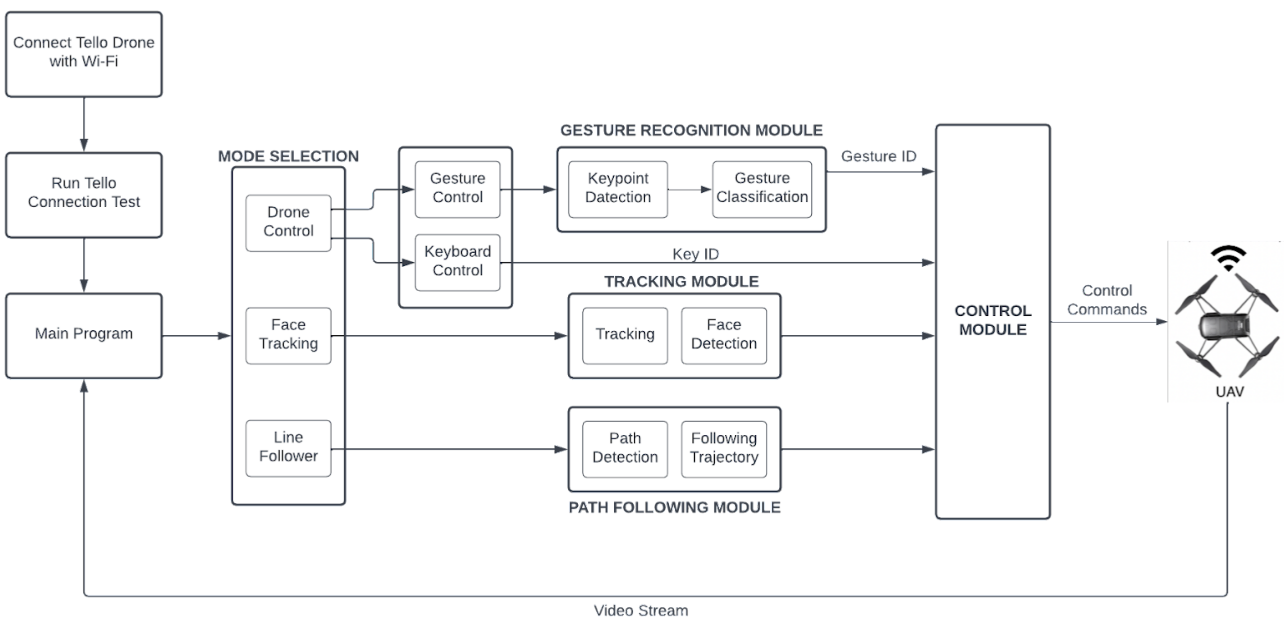

Fig 21. Workflow of the project

5.3.1 Algorithmic Approaches Used

The neural network architecture used to train the sketch recognition model employs many deep learning algorithms to achieve high accuracy in the prediction of the hand gestures. They are as follows:

i. Forward and Backward Propagation in Neural Networks:

Neural networks are interconnected computing systems that function similarly to neurons in the human brain for recognizing hidden patterns and correlations in raw data continuously learning and improving over time.

Forward propagation refers to the process of propagating weights and biases from inputs to outputs. To arrive at a predicted output, randomly initialized weights and bias values are propagated forward through the model. The activation function is applied to the linear combination of inputs at each neuron. Following that, the loss for the training example is computed.

Backward Propagation is a technique for calculating the partial derivatives from the loss function back to the inputs with respect to various parameters, thereby updating the values of weights and biases that lead us to the minimum of the cost function.

ii. Batch Normalisation and Callbacks:

Batch normalization applies a transformation that maintains the mean output close to 0 and the output standard deviation close to 1. Importantly, batch normalization works differently during training and during inference.

Simply put, callbacks are the special utilities or functions that are executed during training at given stages of the training procedure. Callbacks can help you prevent overfitting, visualize training progress, debug your code, save checkpoints, generate logs, create a TensorBoard, etc. There are many callbacks readily available in TensorFlow, and you can use multiple.

iii. Adam Optimization:

Adam combines the advantages of both gradient descent with momentum and RMSprop. It stands for Adaptive Moments estimation. It computes both the weighted moving average of gradients and the RMS average of gradients in each iteration and uses both to update weights and bias.

$$W_L = W_L - \alpha \times \frac{(VdW_L)}{\sqrt{SdW_L} + \epsilon} \qquad b_L = b_L - \alpha \times \frac{(Vdb_L)}{\sqrt{Sdb_L} + \epsilon}$$

5.3.3 Project Deployment

The project was hosted successfully on a local machine.

5.3.4 System Screenshots

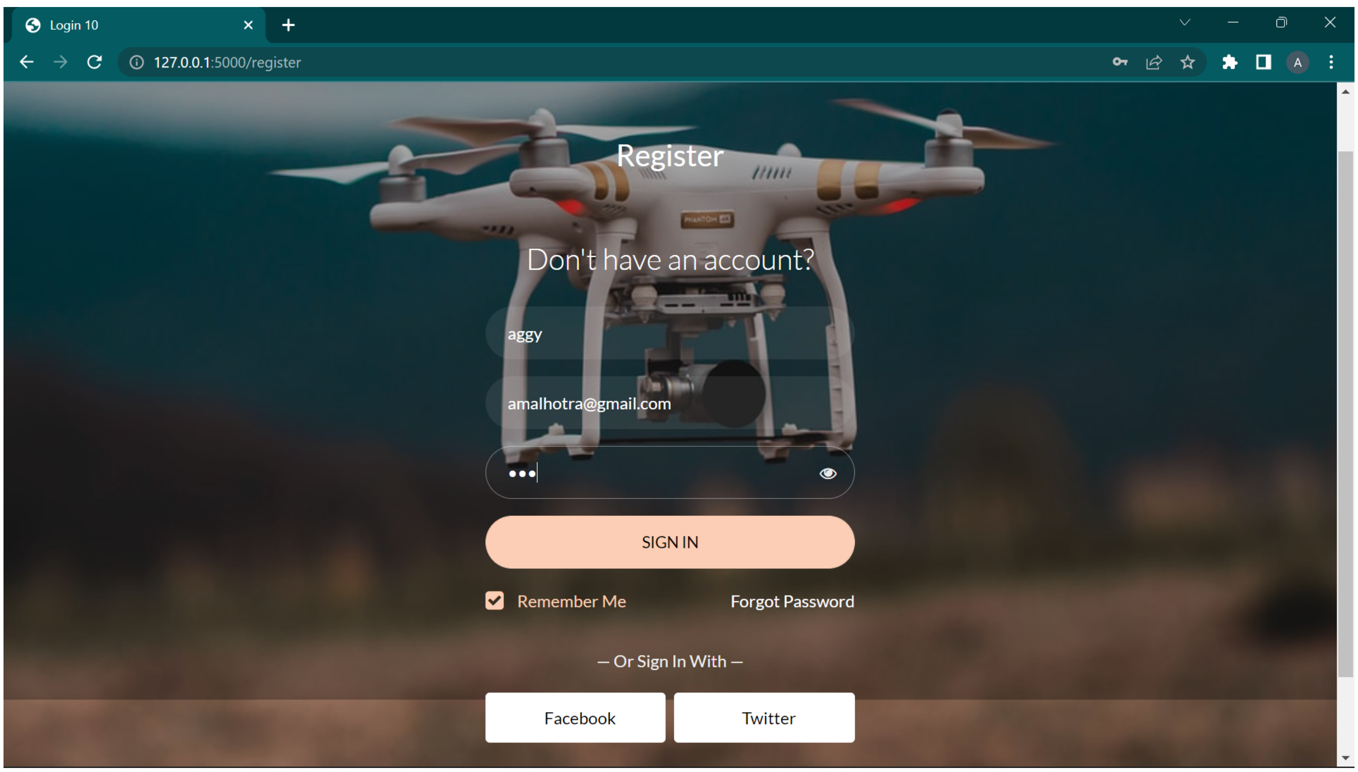

Fig 22. Registration Window

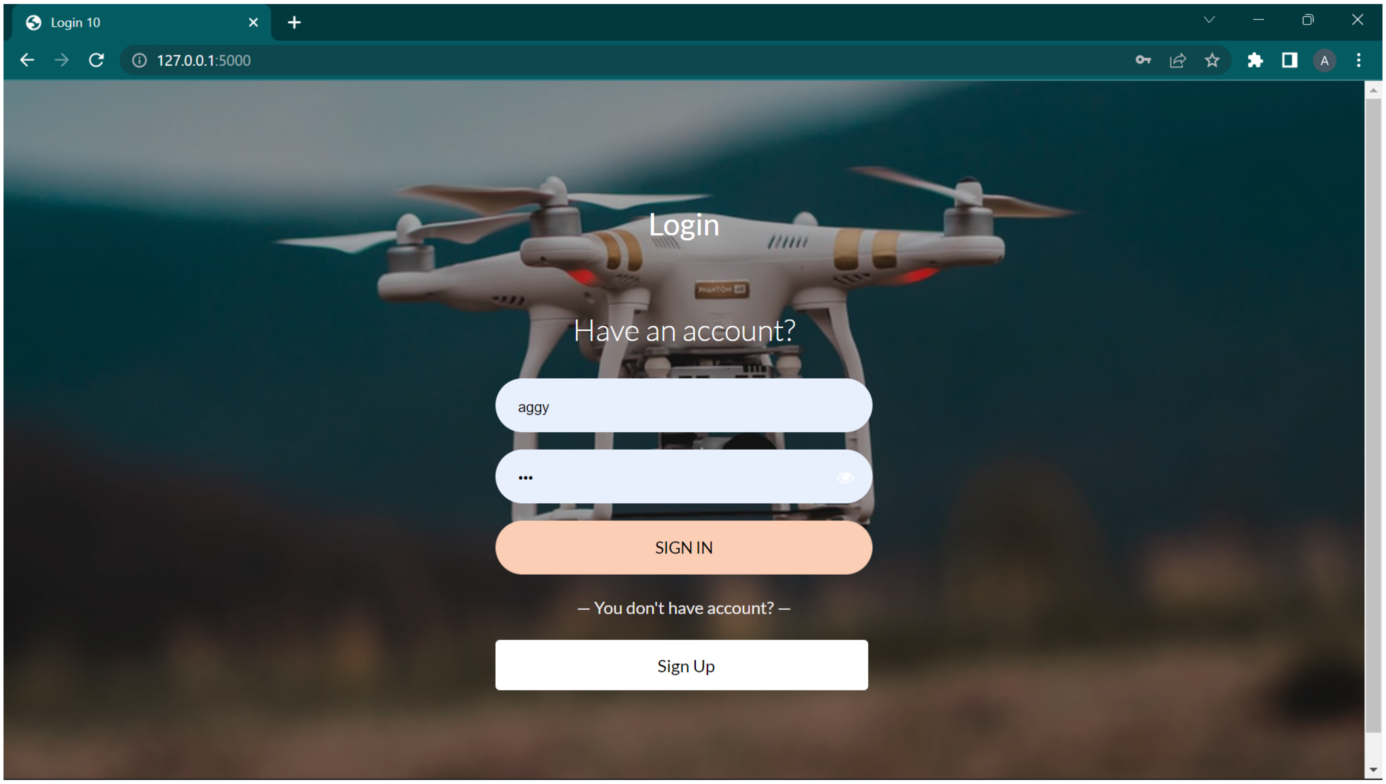

Fig 23. Login Window

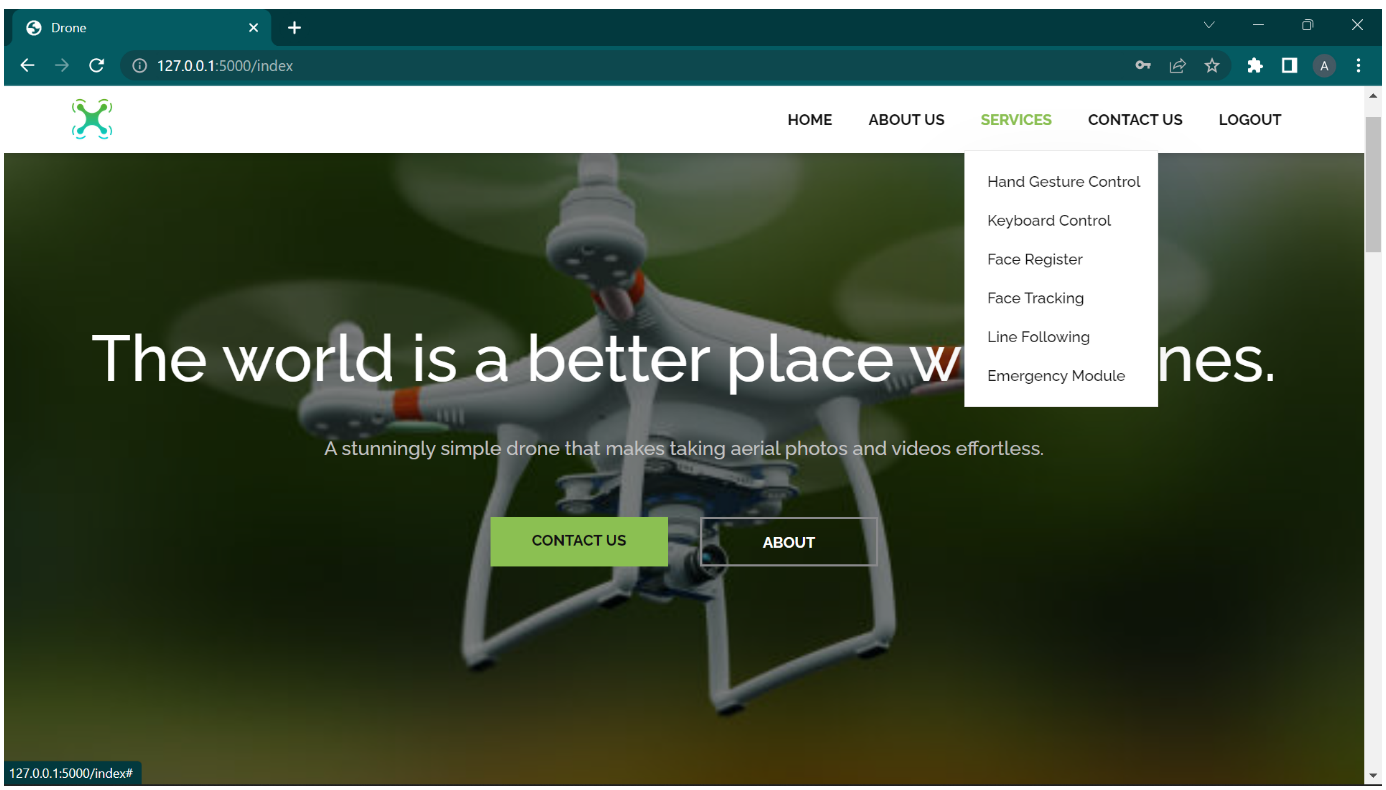

Fig 24. Home Page

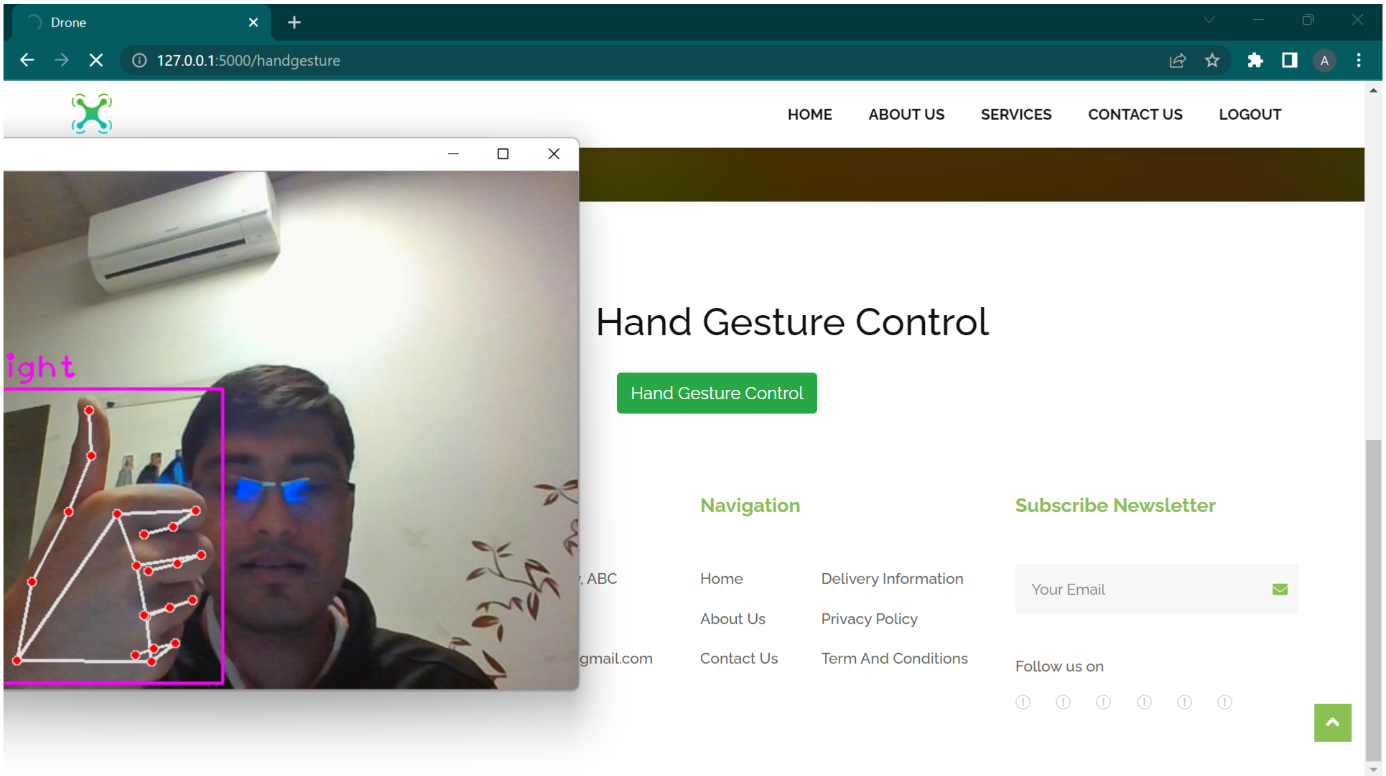

Fig 25. Hand Gesture Control Module

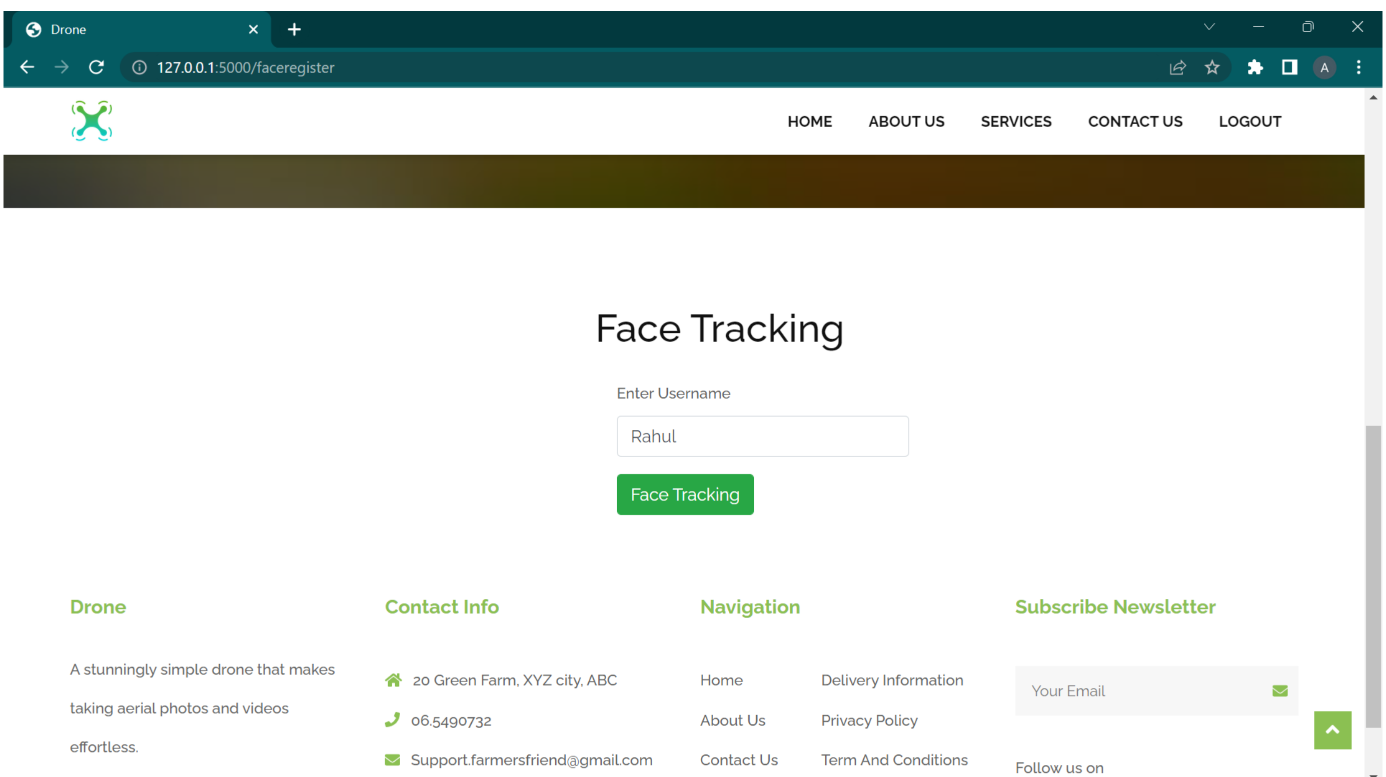

Fig 26. Face Tracking Module (Registration)

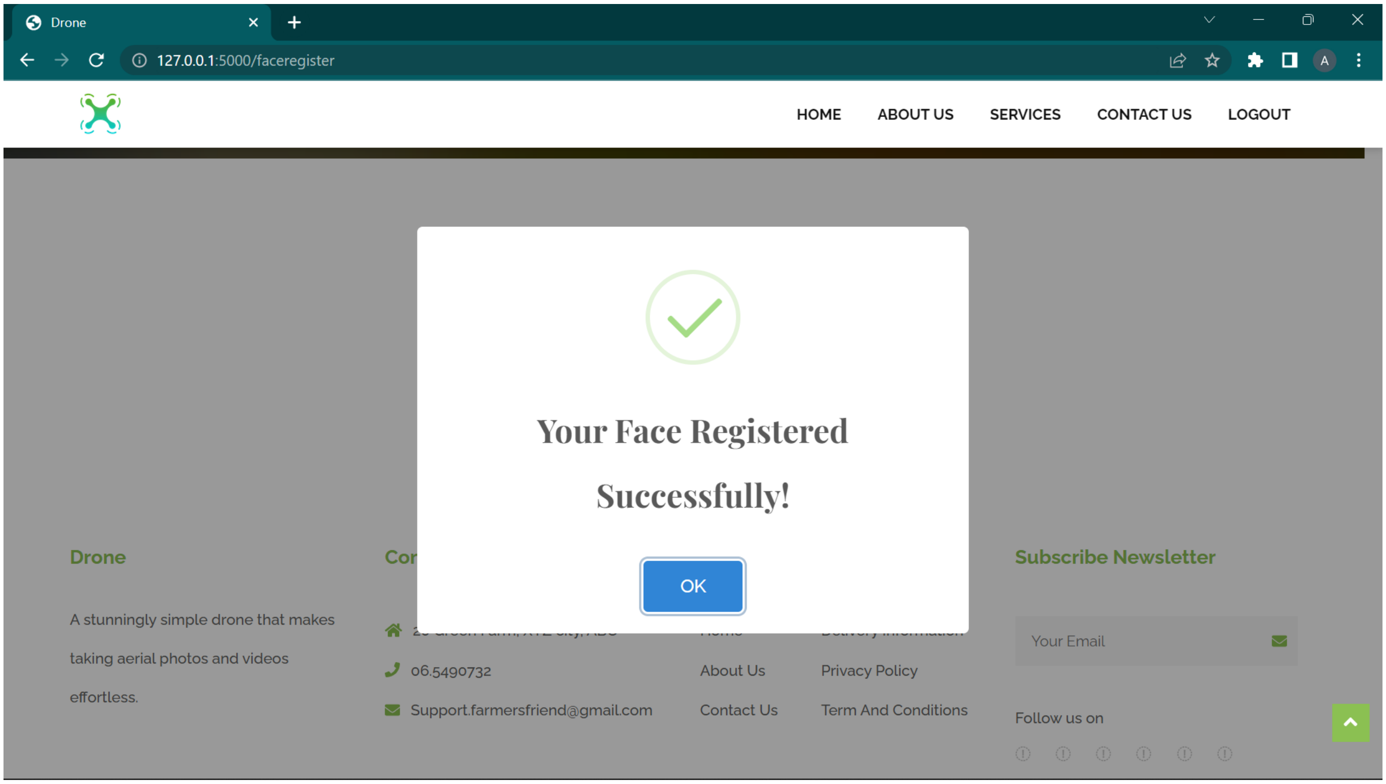

Fig 27. Face Tracking Module (Registration Successful)

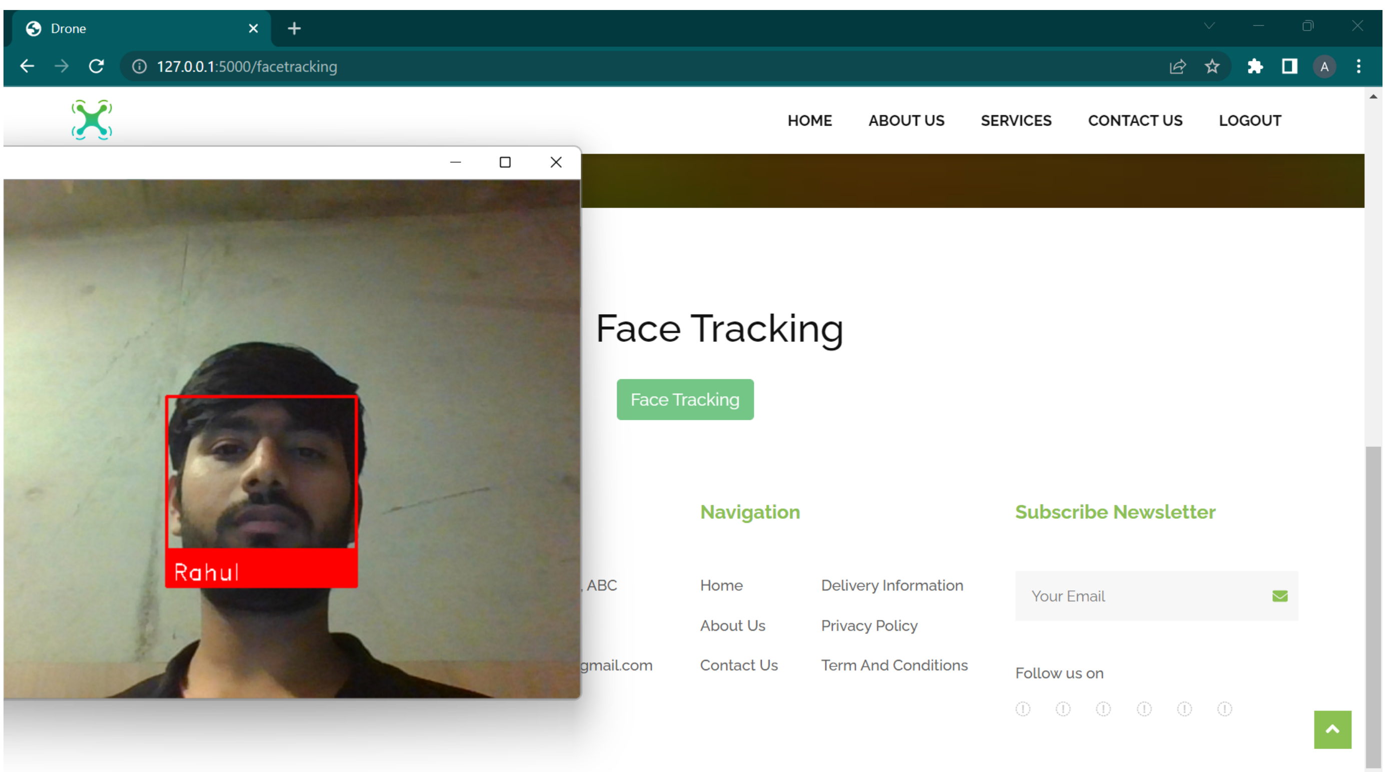

Fig 28. Face Tracking Module (Face Recognition)

5.4 Testing Process

The testing process involves evaluating the designed neural network model with the goal of finding whether or not it satisfies the specified requirements i.e. does it correctly predict the output produced by drawing the alphabets/numbers etc. on a webcam interface which is used as an input for the trained model.

5.4.1 Test Plan

After the model is fully trained and ready for further process, it has to be validated and tested.

5.4.2 Features To Be Tested

i. Checking the Authorization of User

The very first step after opening the software is to verify that the user is authorized and it has our subscription to run the software. Also, we check if any user is using multiple devices to run the software with the same login credentials.

ii. Checking the Internet Connectivity

As our software requires an active internet connection to operate as we are storing the children's progress at our cloud server and also validating the children's credentials.

iii. Application Integration

In this module, we will test whether the system is able to receive the video input from the webcam as it is used to detect the hand gestures used for drawing and dragging the objects. Here we also test whether the video streaming is fluent and does not involve any distracting objects and background.

a. Webcam initialization
b. Fluency of hand movements
c. Proper drone connection

iv. Model Characteristics

In this module, we test the Model's characteristics by evaluating its loss and accuracy metrics on the training set.

5.4.3 Test Strategy

Our project is software-based and it is also linked to the Web site. We first test whether our website is running correctly and is able to handle the traffic and the other is where we test the application in real-time and get the outputs for the given inputs.

We will be using the following strategies-

i. Unit Testing

Involves testing the smallest components of an application that can be tested individually. As we are having multiple modules in our software, we will run each and every module and perform extensive testing on it and after getting satisfactory accuracy, we will move to the next module and will repeat the same procedure.

ii. Integration Testing

Involves integrating individual modules of an application and testing how well they function together. The aim here is to search for the faults in the interaction between the units. As we have a few modules, we will be using the Non Incremental approach where we combine the unit and test the interaction all in one go.

5.4.4 Test Cases

i. Checking user authentication

As our software is subscription-based, the user must register with us prior to utilizing the product; only after confirming the login credentials may the user use our software. When an unregistered user attempts to log in, the notice 'User Not Registered' is shown on the screen.

ii.Checking the Drone Connectivity

As our software requires an active connection with the drone Wi-Fi to operate as we are flying the drone and also identifying faces and gestures. Whenever our software does not get an active connection then the drone hovers and lands automatically.

iii.Application Integration

a. Webcam Initialization

The system must have the ability to initialize the webcam. This is verified everytime a new module is started; after 5 seconds, the camera interface is visible to the user, at which point they may begin drawing alphabets/numbers or utilize any other function.

b. Fluency of the hand movements while showing to the drone

While operating using gestures, the hand movements must be precise else the model will not be able to predict correctly also no extra faces/hands should be present in the frame as it may cause hindrance.

iv. Model Characteristics

The performance of any neural networks model depends on the hyper parameters used to train it and the data that is used as input. We trained our model on the NVIDIA GE-FORCE GTX 1650-TI GPU and turned the hyper parameters till we discovered a configuration that gave us a really good value of accuracy with minimal loss. The following values of hyper-parameters were used in the training of the model:

Note: The FCN model for hand gesture recognition was tested on a dataset of 8 classes, as described in the Procedural Workflow section.

a. Model: FCN – Quick Testing

Learning Rate: 0.001
Epochs: 100
Batch size: 50
Number of Gesture Classes: 8

b. Model: FCN

Learning Rate: 0.01
Epochs: 1000
Batch size: 32
Number of Gesture Classes: 8

c. Model: FCN

Learning Rate:0.001
Epochs: 1000

Batch size: 64
Number of Gesture Classes: 8

5.4.5 Test Results

TABLE 5: Test Results

Test Case ID	Scenario	Test Step	Expected Outcomes	Actual Outcomes	Type of Testing
T001	To check whether the prediction model is working accurately and in time.	Give the image to the model and note down the result and time.	Correct hand gesture prediction.	Correct hand gesture prediction successfully returned.	Blackbox testing
T002	To check whether the model is working after integrating it with the module.	The image was sent to the model by running a module on the UI.	The model should run in less time to give output.	The model ran at an appropriate time and gave the accurate results.	Blackbox testing
T003	To check whether the user is registered with us and has the subscription to run the module.	The login details were asked from all the users and are checked from the database.	Existing users should be logged in and non-existing users should get an error message.	The output was correctly displayed on the screen.	Whitebox testing
T004	To check the user is having an active connection to the drone.	The drone connection of the user is checked on the opening of the module	Absence of a Drone connection should display error messages on the screen.	The output we received was as per expected results.	Blackbox testing
T005	To check whether the webcam is initialized while running the software.	The webcam interface is in running state or not on the opening module of software.	The webcam interface should appear on opening any module of the software.	The webcam was working fine on opening any module.	Blackbox testing.
T006	To check the fluency of the hand movements while controlling the Drone.	The hand movements must be fluent while performing any of the activities..	The webcam should capture the right details or gestures made by the user.	The output was accurate as per the expected outcome.	Blackbox testing.

5.5 Results and Discussion

The module is checked on various parameters and as mentioned above to check its efficacy. The majority of results were above satisfactory and the prediction model performed as expected and thus leading to an overall positive outcome on evaluation. The module is easy to use and it is easily scalable and requires fewer props. Also, the system is cost-efficient as no other hardware is required except for the webcam mounted on the laptop and a Drone.

However, there are a few limitations such as while using the software presence of any extra objects causes hindrance in the working of the software, and also the background used be plane or uniform while operating software else it may cause disturbance while detecting the hand gestures.

5.6 Inferences Drawn

The inferences drawn from the testing process and evaluation are:

i. The prediction models run in appropriate time in the software and are near real-time.
ii. There should be appropriate lightning conditions while capturing the image from the webcam device in order to correctly trace the hand movements.
iii. The person’s background should be simple or plain for accurate predictions to be made for the face tracking module.

5.7 Validation Objectives

We were successfully able to achieve the objectives which were specified at the beginning of this project.

TABLE 6: Validation Objectives

Sr. No.	Objective	Status
1.	To programme a drone to create a more personalized approach to control it using hand gestures.	Successful
2.	To programme a drone so that it will be capable of face tracking. It will be able to follow the person keeping a specific distance from him/her whose face the drone would have recognized.	Successful
3.	To programme a drone with a functionality which will enable it to follow a specified color path using line path reflection and corresponding clockwise and anticlockwise direction turns.	Partially Successful
4.	To programme a drone to control it using keyboard controls.	Successful
5.	To programme a drone for an emergency module to make it hover in a defined path.	Successful

6. CONCLUSION AND FUTURE SCOPE

6.1 Work Accomplished

Our goal of this study is to enable the hand gesture based control mechanism with maximum possible accuracy even in mediocre drones which can be easily outperformed by the state-of-art drones due to their inbuilt high camera resolutions like 4k 8k 16k and 64k etc in this empirical study we investigated more on software development for the AR drones we presented an image recognition-based communication framework to control drones with hand gestures the framework is successfully tested using a basic drone we found that the accuracy of the framework is highest once the operator poses within 3 ft well lit and clear background this controlling distance can be further improved by utilizing better cameras such as those supporting 4k or 16k resolution in the drones which allows capturing of images with good resolution at longer distance or implementing the same framework on state-of-art drones with better imaging capabilities with the available hd camera in basic drones the hand gesture recognition in the distance between 3 and 5 ft is highly accurate and this controlling distance can be improved by enabling high resolution cameras in drones with the current hand gesture-based control mechanism we envision that drones can be sent to any feasible distances and perform operations before flying back to the controller for further close-ranged interactions to explore the effects of different hand poses and deviation an in-depth statistical analysis on the applicability of the framework in different environment settings is planned for future work.

6.2 Conclusions

The UAV is smaller in size, which assists in flying in compact construction blocks; it has good balance; it might fly in specific weather regions; and it is able to be operated on the spontaneous movement of people. It'll assist in the border regions of India for the surveillance and detection of humans. The video has been recorded in an entry tool which assists in investigating it and doing many photo processes. The manufacturing value is much less and the replacement is smooth. To conclude, our project is going to be a new way to interact with drones and can pave the way for a new way to interact with other machines as well. The extensibility of gesture-controlled devices is rapidly growing, and it is also extremely beneficial to those with disabilities regarding sound. Because those with disabilities regarding sound tend to communicate through sign language or the like because they are unable to talk, this will allow them an easy way to communicate with devices via gestures that they are already very familiar with. Our project, a gesture-operated drone, is simply an implementation of a gesture controlled device. The Gesture Operated Drone allows for an extremely simple way to operate a drone in comparison to the unwieldy RC remotes that commonly come with drones to operate them.

6.3 Environmental Benefits

Environmentally our drone will be constrained by its ability to only be flown indoors. It is not easy to find indoor spaces where flying the drone is allowed without permission. It is important that our drone is capable of performing well in tight situations. Having the constraints of four walls around the drone can complicate some of the testing. We therefore tested the drone in outdoor areas as well as in the hostels. These have far less room to work with but have enough space to practice basic maneuvers and test what needs to be looked at. Environmentally we are also legally constrained. To fly the drone in public areas, permission is required. To save money and time we decided to avoid flying the drone outdoors completely and to focus on only flying indoors. The benefit of this is the controlled environment that we have indoors. There are no factors like wind and rain to worry about. Having an indoor drone will lessen the constraints of the more unpredictable conditions of the outdoors. Will promote use of lesser plastics as the need to make remote controllers for drones can be easily eliminated by the use of the demonstrated principles.

6.4 Future Work Plan

i) Adding More Gestures for More Features

A Drone may have many features. Some might have a camera access while some may have a GPS enabled on them. We can extend the algorithm in such a way that a gesture made by the user can control these abilities. Like a wave may make the camera turn on.

We can also make the Drones to form a certain shape with the gestures. For example, consider 5 Drones have been taken and four of them have been arranged as a square and the fifth one is placed at the center. When the user makes a gesture, say, two consecutive upward movements quickly, the Drone moves one step upward and the Drone 5 moves two steps upward. This forms a pyramid structure of Drones. Complex applications of this can be used in many ways.

ii) Implementing Drones to Pick Weights

The given model, when implemented with Drones of higher capability which can carry larger weights, can be used by people with challenges and difficulty in lifting weights to easily relocate objects around them. When studying the sensitivity of the sensors with people having different health conditions with difficulty in movement, and the sensors are tuned to it, the algorithm can be used to enable them in an easy relocation of objects. This can be implemented in large scale by implementing the algorithm over micro-integrated chips to the wheels of trolley bags or trolleys carrying goods 35 so that human effort could be reduced in shifting them as one may be able to do it with a simple gesture. Another application of this technique would be to achieve a complex task. For example, an object has to be placed in a high-altitude place, we can use Drones to do that.

7. PROJECT METRICS

7.1 Challenges Faced

The team faced the following problems during the course of the development and testing of the project:

i. Acquiring a suitable drone.
ii. Finding open spaces with sufficient ambient light to conduct tests.
iii. Improving the accuracy of the ML model for correct predictions.
iv. Correct prediction of hand gestures and face from the input given through webcam and drone camera.
v. Finding the appropriate datasets to train the ML model for hand gestures.

7.2 Relevant Subjects

TABLE 7: Relevant Objectives

Subject Code	**Subject Name**	**Description**
UCS503	Software Engineering	Used the concepts of Software Engg. like the incremental and iterative model in the development process. Also, the documentation taught in the course helped in the report making.
UML501	Machine Learning	Used the concepts of ML in the core functionality of the project.
UCS663	AI Applications	The Convolutional Neural Network model used is based on deep learning concepts. Datasets are an integral part of the project and the concepts of data analytics are used as the core functionality.
UCS542	Full Stack Development	Applied the knowledge gained from the subject to create the website's UI/UX and MYSQL DB linkages.
UCS310	Database Management System	Understanding entities and relationships for designing the database.

7.3 Interdisciplinary Knowledge Sharing

Our project titled PUSHPAK is a smart drone control system used to control a drone using hand gestures, keyboard control,face tracking etc. Different branches of computer science have been utilized which include computer vision using deep learning, full-stack web development along with applying the concepts of software engineering.

In order to transform the datasets according to our needs, data augmentation and the use of probability and statistics have helped us to analyze the data and how inferences are derived from it. The classification was the primary task which was achieved with the usage of Convolutional Neural Networks with appropriate activation layers associated. Data structures using python and various python libraries were put to our use in this project.

For the future purpose of distributing our product to the masses across the globe, web development came to our rescue. The fundamentals of full-stack development helped us to create an attractive website along with providing separate accounts for the users to access the software.

7.4 Peer Assessment Matrix

TABLE 8: Peer Assessment Matrix

		Evaluation of			
		Ayush Kaushik	**Rahul Mishra**	**Akshat Thakur**	**Ridham Behl**
Evaluation By	**Ayush Kaushik**	5	5	5	5
	Rahul Mishra	5	5	5	5
	Akshat Thakur	5	5	5	5
	Ridham Behl	5	5	5	5

7.5 Role Playing and Work Schedule

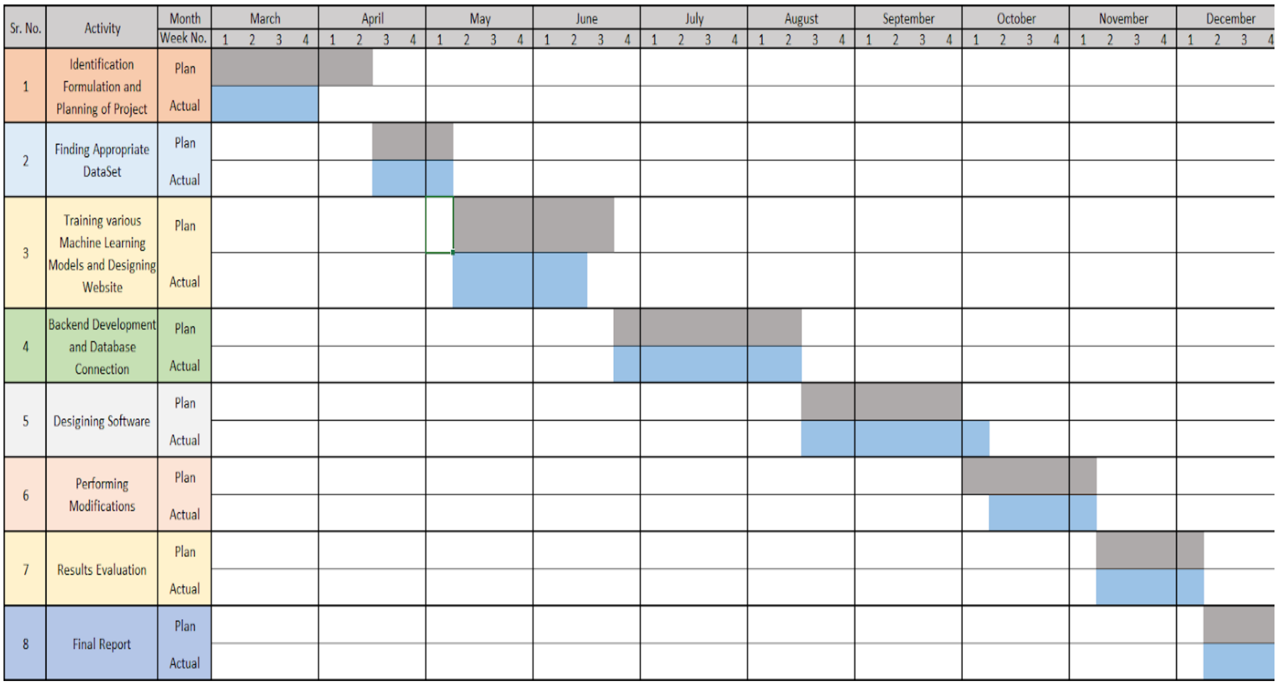

Fig 29. Gantt Chart describing work schedule

Roles:

Ayush Kaushik	Creating dataset, Data preprocessing, Keyboard Control Module Programming, Line Following Module Programming, Testing, Final Documentation
Akshat Thakur	Creating dataset, Data preprocessing, Training DL Model, Face Tracking Module Programming, SOS Module Programming, Testing, Final Documentation
Rahul Mishra	Creating dataset, Data preprocessing, Training DL Model, Hand Gesture Module Programming, SOS Module Programming, Final Documentation
Ridham Behl	Creating dataset, Data preprocessing, Hand Gesture Module Programming, Keyboard Control Module Programming, Modifications in prototype, Final Documentation

7.6 Student Outcomes Description and Performance Indicators (A-K Mapping)

TABLE 9: A-K Mapping

Sr.	Description	Outcome
A1	Applying mathematical concepts to obtain analytical and numerical solutions.	Statistical mathematics is the foundation of machine learning. While reading about neural networks and the many algorithms that drive them, we came across several related ideas.
A2	Basic principles of science towards solving engineering problems.	Our neural network was developed using Tensor Flow, a platform for scientific computing.
B1	Identify the constraints, assumptions, and models for the problems.	We thoroughly reviewed the CNN research and were able to pinpoint certain potential limitations in their operation. When delivering the project, we made assumptions regarding the user's environment.
B2	Appropriate methods, tools and techniques for data collection.	To extract the required gesture classes, we got pre-made datasets and processed them.
C1	Can understand scope and constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability and sustainability.	We understand the availability of low cost long battery drones is meager but it is a fast developing field of research.
D1	And play different roles as a team player.	We divided the job among the four of us, and when necessary, we worked accordingly.

E1	Identify engineering problems.	We sought to address the issues we found throughout the project.
E2	Use analytical and computational methods to obtain solutions.	After analyzing the dataset, we chose to just extract different types of gestures, which required building programming to facilitate and speed up the process.
F1	Produce a variety of documents such as laboratory or project reports using appropriate formats.	We created a project report that included all the necessary details, as well as a software needs survey to help the reader understand how the project operates.
G1	Deliver well organized and effective oral presentations.	At various stages over the semester, we gave multiple presentations outlining the subject of our research and the advancement we had made.
H1	Aware of the environment and societal impact of engineering solutions.	Our proposal would simplify controlling drones and utilizing its various functions to help us in various use cases.
I1	Able to explore and utilize resources to enhance self-learning.	We used the online MOOCs that were available to us to learn about subjects that were brand-new to us.
I2	Recognize the importance of life- long learning.	We must constantly learn to stay current because the world of computing is constantly evolving.
J1	Apply different data structures and algorithmic techniques.	Machine learning techniques were used in our research, along with their implementation and testing.
K1	Use software tools necessary for the computer engineering domain.	We made use of software tools available to us, like : 1. Used lucidchart for designing all necessary diagrams. 2. Microsoft word to formulate our

		report. 3. Tensor flow and python to write code for our project.

7.7 Brief Analytical Assessment

(i) What sources of information did your team explore in order to arrive at the problem statement?

In order to analyze the subject and find potential for growth, we compiled a list of research papers and publications that examine advancement in the field of this recognition. We focused on thoroughly capturing the various uses of a drone.

(ii) Did the project demand demonstration of knowledge of fundamental scientific or engineering principles? If yes, how did you apply them?

Yes, in order to develop a neural networking model for the project, we have investigated the ideas in both deep learning and machine learning. In order to outline our strategy for completing the project and the supplementary report, we also had to apply elements from the software development cycle.

(iii) How did your team share responsibility and communicate the information of schedule with others in the team to coordinate design and manufacture dependencies?

After the allotted times, we had a team meeting where we attempted to clear up any questions or misunderstandings. Despite the fact that our project has been separated, each team member is aware of their own task. Together, the team members strive for excellence in everything they do. When it comes to activities involving the administration and storage of data, WhatsApp Group and Google Drive have been a lifesaver.

(iv) What resources did you use to learn new material not taught in the class for the course of the project?

We learned the basics of Python GUI programming through online classes because it was new to our team and wasn't covered in our curriculum. Learning about the OpenCV library and its essential elements for image processing was done on the YouTube and Udemy platforms. Through online websites, some basic Tensorflow programming and mediapipe dependency were widely learned. Every time we ran into trouble in a particular domain, Stack Overflow came to our aid.

(v) What analytical computational or experimental method did your project team use to obtain a solution to the problem in the project?

Since our goal is to train a convolutional neural network, it was obvious that we would need to conduct a lot of testing to achieve an acceptable level of accuracy. Initially, we thought about performing the computation using the CPU, but we eventually had to switch to GPU for significant computational resources. Then, using our dataset as a baseline, we tried a variety of parameters to fine-tune the performance of our model. Finally, considering our program and whether recognizing a dynamic gesture required activating a key function.

(vi) Does the project make you appreciate the need to solve the problem in the real time using engineering and did it help you gain proficiency in software development tools and environments?

With the development of our software, we were able to devise many functions using drones. We were able to create new and innovative solutions to many areas where drones can be used.. We also gained proficiency in different technology tools such as CNN, MYSQL DB, OpenCv, Mediapipe and more. These new technologies made us proficient in various skills for software development in different environments. We thoroughly enjoyed working with these new software technologies and gaining insights about these development tools.

APPENDIX A: References

[1] International conference on applied robotics for the power industry (CARPI), Montreal, QC, Canada, 5–7 October 2010, IEEE. ISBN: 9781424466351, pp. 1–6, DOI: 10.1109/CARPI.2010.5624431

[2] Lu M, Sheng G, Liu Y, et al. Inspection Based On Unmanned Aerial Vehicle. In: Power and energy engineering conference (APPEEC), Shanghai, China, 27–29 March 2012, pp. 1–5. ISBN: 9781457705472, DOI:10.1177/1729881419855495

[3] Li Z, Walker R, Hayward R, et al. Advances in vegetation management for power line corridor monitoring using aerial remote sensing techniques. In: Proceedings of the first internat, DOI: 10.1109/ACCESS.2021.3106550

[4] Wang B, Chen X, Wang Q, et al. Power line inspection with a flying robot. In: 2010 first international conference on applied robotics for the power industry (CARPI 2010), Montreal, QC, Canada, 5–7 October 2010, ISBN 978-1-4244-6633-7, pp. 1–6. DOI: 10.1109/CARPI.2010.5624430

[5] Liu Y, Mejia's L, and Li Z. Fast power line detection and localization using steerable filters for active UAV guidance. In: 12th international society for photogrammetry & remote sensing (ISPRS2012), Melbourne Convention and Exhibition Centre, Melbourne, 25 August–1 September 2012, vol. 39, pp. 491–496, DOI: 10.5194/isprsarchives-XXXIX-B3-491-2012.

[6] Liu Y and Mejia's L. Real-time power line extraction from unmanned aerial system video images. In: Second international conference on applied robotics for the power industry (CARPI), Zurich, Switzerland, 11–13 September 2012, pp. 52–57. DOI:10.1109/CARPI.2012.6473348.

[7] Yao X, Guo L, and Zhao T. Power line detection based on region growing and ridge-based line detector. In: Sun Z and Deng Z (eds) Chinese intelligent automation conference, Lecture notes in electrical engineering, vol. 255. Berlin, Heidelberg: Springer, pp. 431–437. DOI: 10.1007/978-3- 642-38460-8. ISBN: 978-3-642-38459-2.

[8] Song B and Li X. Power line detection from optical images. Neurocomputing 2014; 129: 350–361. DOI: 10.1016/j.neucom.2013.09.023.

[9] Molchanov P., Gupta S., Kim K., & Kautz J... "Hand gesture recognition with 3D convolutional neural networks." In Proceedings of the IEEE conference on computer vision and pattern recognition workshops pp. 1-7. (2015), DOI: 10.1109/CVPRW.2015.73013.

[10] Vecriah V., Swaminathan P. L.. "Robust hand gesture recognition algorithm for simple mouse control". International Journal of Computer and Communication Engineering, vol. 2(2), pp. 219-222.(2013). DOI:10.7763/IJCCE.2013.V2.175.

[11] Gupta S. D., Kundu S., Pandey R., Ghosh R., Bag R., Mallik, A... "Hand gesture recognition and classification by discriminant and principal component analysis using machine learning techniques." International Journal of Advanced Research in Artificial Intelligence, vol.1(9),pp 32-37. ,(2012). DOI:10.1109/IWW-BCI.2018.8311527.

[12] Camgoz N. C., Hadfield S., Koller O., Bowden R.. "Using convolutional 3d neural networks for user-independent continuous gesture recognition." In proceedings of 23rd International Conference on Pattern Recognition (ICPR), pp. 49-54. IEEE. (2016). DOI: 10.2478/acss-2020-0007.

APPENDIX B: Plagiarism Report