International Journal of Innovative Research in Engineering
Volume 4, Issue 3(May-June 2023), PP: 483-487.
www.theijire.com **ISSN No: 2582-8746**

AI & ML Powered Autonomous Drone System

Ramprasad Kulkarni[1], Omkar Patil[2], Viraj Gavhane[3], Shrikant Mane[4], Nitin Zinzurke[5]
[1,2,3,4,5]Department of Computer Engineering, JSPM's Jayawantrao Sawant College of Engineering Pune, India.

How to cite this paper:
Ramprasad Kulkarni[1], Omkar Patil[2], Viraj Gavhane[3], Shrikant Mane[4], Nitin Zinzurke[5], "AI & ML Powered Autonomous Drone System", IJIRE-V4I03-483-487.

Check for updates

https://www.doi.org/10.59256/ijire.20230403116

Abstract: *The use of drone technology is growing in popularity and has many different industrial uses. Drones are used in agriculture for pesticide and fertilizer application, crop monitoring, and mapping. Drones can be used in the building sector to check job sites and track development. In the movie industry, drones are also utilized for aerial photography and in search and rescue efforts to reach inaccessible places. In general, drone technology has the potential to revolutionize a variety of industries and offer fresh approaches to problems that were previously challenging to solve. Our idea is to develop AI ML based autonomous drone system that can be deployed in many areas for multiple applications such as surveillance, warehouse management etc. The layout includes a PI-CAM V2 camera on a quadcopter using GPS and system powered by Robot Operating System, Open CV and other technologies that gives ability to drone to fly without need of pilot autonomously.*

Key Word: *Multipurpose, Robot Operating System, Simulation, Surveillance, Tracking, Quadcopter, OpenCV, Machine Learning, Object Detection.*

I.INTRODUCTION

Unmanned aerial vehicles (UAVs), also referred to as drones have provided a wide range of applications during the previous ten years, including environmental sensing, sampling, and surveillance. Because they can visit places that humans would otherwise find difficult to get, UAVs are also seen as a crucial piece of technology. Robotics in general, and UAVs in particular, need robot operating system, artificial intelligence (AI) and machine learning algorithms to be able to perceive and interact with the actual world in order to work autonomously and help people with daily tasks. In this paper, we report the design and development of an AI ML based drone for multiple purposes such as surveillance, warehouse management etc.

II.LITERATURE SURVEY

A. Study of Existing System

Manual drones or remote-controlled drones are the drone systems that must be manually operated by pilots in order to carry out any duty. Human operators fly these kinds of drones by using a remote control to guide it through the air. Drones that are operated manually are frequently used for flying as a hobby or for leisure activities like aerial photography etc. Additionally, they are employed for business purposes, such as crop monitoring, land surveying, and infrastructure inspection. Manual drones have the advantage of allowing for fine control over the drone's movement and of being able to operate in conditions where autonomous drones would not be able to. Nevertheless, manual drones may not be as effective as autonomous drones for some tasks and need a skilled operator to operate.

B. Study of Existing System

The main drawback of traditional manual control drone system is that they need continuous monitoring for performing any task. Additionally, the drone pilot needs to be had all required skills and experience of drone flying. Although manually controlled drone are more precise in performing any critical task such as rescue missions etc. But those task which are repetitive in nature such as warehouse management in which products are monitored for managing warehouse or task such as surveillance in which a specific area is monitored repeatedly again and again in this scenarios this autonomous drone system can be implemented that can perform a repetitive task more efficiently and precisely in less time and less human efforts.

III.METHODOLGY

Today Drones are used for multiple purposes such surveillance, but the main drawback of these types of drones is that they manual observation by pilot or operator to control them. Single board computers such as raspberry pi, jetsonnano are capable of running robot operating system, artificial intelligence and machine learning that can give drones abilities like, decision making, flight path detection, obstacle avoidance, image processing and object detection. The system will consist of a drone, a control center, data collecting center and data logging server

A. Objectives

- Use of technologies such as obstacle avoidance, flight path detection, intelligent mission following, machine learning, ROS (Robot Operating System) to provide drones ability to fly autonomously with less interaction of drone pilots.
- Least amount of human/pilot interaction for controlling and commanding the drone system.
- Developing multipurpose autonomous drone system that can be deployed in many areas for multiple applications.
- Making the repetitive tasks more time efficient and precise with the help of autonomous mission-oriented drones.

B. System Overview

The system consists of the primary UAV platform, which is based on the Ardupilot with raspberry-pi SBC as companion computer with PYMAVLINK. It also includes drone kit API for drone with 8 MP camera with SPI interface for real-time object detection with MJPG stream. System is supported with Robot Operating System, Machine learning etc. technologies to implement autonomous ability in the drone. A powerful ground station for computation and command purposes with high speed WLAN network. Following are some tools and technologies that are used to developed the proposed system.

1. Open CV: - A well-known open-source computer vision package called Open CV offers a variety of processing options for images and videos, including object detection, tracking, and identification. Robotics, autonomous systems, and machine learning are just a few of the fields where Open CV is frequently utilized. Open CV can be utilized to give autonomous drone systems vision-based perception capabilities.
2. Robot Operating System: - An open-source framework called Robot Operating System (ROS) is used to create robot software. It offers a set of protocols, libraries, and tools to assist programmers in building powerful, sophisticated robot applications. Autonomous vehicles, drones, industrial automation, and healthcare are just a few of the robotics industries where ROS is extensively employed.
3. Gazebo Simulator: - One of the most popular simulation environments in the robotics world is Gazebo, a physics-based robot simulator. It offers a platform for testing and evaluating robotics algorithms in a virtual environment before implementing them on actual robots. It is integrated with the Robot Operating System (ROS).Compared to physically building and testing robots, simulation in Gazebo can be a major financial savings. In Gazebo tests are conducted in a simulated setting are safer since there is little chance that real robots or the environment would be harmed
4. MAV Link Protocol: - For unmanned aerial vehicles (UAVs) and other robotic systems, MAV Link (Micro Air Vehicle Communication Protocol) is a simple, open- source communication protocol. A common method of communication between various drone system parts, such as the flight controller, ground station, and other onboard electronics, is provided by MAV Link.

MAV Link is intended to work with a variety of hardware and software and is platform-independent. Many well-known open-source drone software platforms, including ArduPilot and PX4, support MAV Link, which is widely used in drone systems.

C. System Building Process

1. Drone Building Process: - Installed 2600KV motors with 920KV 2212 ready to sky motors and Mamba 405 with Pixhawk 2.4.8. Configured it according to the needs with PID tunings. Installed 1000mAh 2S 20C Orange Li- PO with 3000mAh 3s 30C orange Li-PO and Jetson Nano with Raspberry-pi and cooling-fan. Installed Blue- tooth with ESP-8266. Added an extra 18650 battery for Raspberry-PI. Installed OLED-Screen. Done PID tunings and other optimizations for stable flight.
   Result: High flight time, automated flight capabilities, very high flight stability, LOITER mode support.
2. Flight Controller and Telemetry: - Installed Mamba F405 MK2 mini flight controller with Pixhawk 2.4.8 flight controller.
   Result: Open source Ardupilot flight controller with lot more feature than F405 ex. Open-source and fully configurable, MAV Link support and Telemetry 2 support. Installed and configured Node MCU (ESP8266).
   Result: Very High-Speed telemetry and drone Wi-Fi.
3. Video Streaming Process: - To increase the data transfer rates and reduce the latency switched from 2.4 GHz to 5 GHz Wi-Fi. Created a python code for video streaming server used default IP address of the raspberry-pi as video streaming. To optimize the stream used Open CV API to change the resolution, framerate, bitrate, Auto- exposure, orientation, Buffer-size, Autofocus, zoom, camera roll and other required features of the raspberry- Pi camera.
   Result: very low latency 720p 60FPS well optimized video streaming server without lag and stuttering.
4. Object Detection Process: - As the Raspberry-Pi has very low computational power decided to use laptop as a processing station for object detection. Installed second windows-10 OS as a final deployment OS. Installed all the dependencies like CMake, latest Nvidia Drivers, etc. Installed pip3, NumPy, putty, Open CV with Open CV- contribute CUDA CUDNN world, CMake, Anaconda, Visual Studio, Visual Studio Code, Latest NV idia drivers and other required dependencies. Optimized OpenCV python code for the use of CUDA and DNN drivers.
   Result: 60 FPS object detection with CUDA acceleration with latest Nv idia Drivers.

Fig 1. Drone Prototype

D. System Architecture Diagram

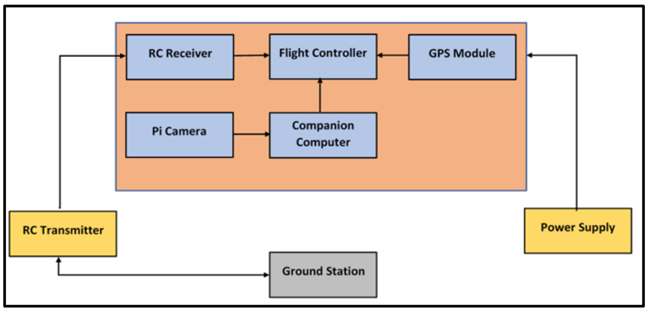

Fig 2. System Architecture Diagram

IV.IMPLEMENTATION RESULTS

A. Automated Drone Arming

A drone must be "armed" before it can take off by turning on its motors. However, it is crucial to confirm that all safety measures have been taken before arming the drone in order to avoid any mishaps or drone damage. The general procedures for arming a drone is as follows:

1. Prior to arming the drone, it is crucial to inspect the flight environment for any potential risks, such as power wires or trees, and to make sure that no people or animals are present.
2. Before arming the drone, conduct a pre-flight check to make sure it is in good condition, the battery is fully charged, and all sensors are operating as they should.
3. Turn on the remote controller: Make sure the drone is linked to the remote controller and turn it on.
4. Arm the drone: The technique for arming the drone can vary depending on the type of drone and flight controller. In most cases, arming the drone requires pressing a few switches or buttons on the remote controller, such as pushing both sticks to the bottom-right corner.
5. Check to see if the drone is armed: When the drone is armed, the motors will begin to spin. Check the LED lights or listen to the motors to make sure the drone is properly armed.
6. Launch: After being armed, the drone is prepared for take-off. Turn up the throttle gradually to raise the drone off the ground.

The above-mentioned arming process is automated by us with the help of python programming language. We created one python script that automates the whole arming process in very efficient and easier way.

B. Object Detection Process

In this module we have used Open CV a machine learning based python library for object detection. With the use of Caffe (Convolutional Architecture for Fast Feature Embed- ding) model which is deep learning framework specially used for image classification and segmentation implemented for task of object detection i.e. human face detection. Detection of human faces will be useful for implementing tasks such as surveillance etc. This module was implemented at first stage of this project development to showcase one of the applications of this drone system. Following images included to show the actual working of this module.

Fig 3. Object Detection using Open CV

C. Simple Obstacle Avoidance with Range Finder

Autonomous drones must be able to avoid obstacles in order to safely travel through challenging situations, making it a critical competency. In this module we have used a Lidar range finder sensor, which utilizes lasers to detect distances to objects, is a common sensor for obstacle avoidance. This sensor is mounted at front side of the drone and it detects object in about 40 degree of angle and find a distance between that object and drone. Depending on the distance the drone will perform further action. Maze Following with Coordinates System in ROS

D. Maze Following with Coordinates System in ROS

In this module drone maze navigation accomplished using the ROS coordinate system for maze following it is simulated using Gazebo. The following general procedures describe how to perform maze following for a drone in ROS using a coordinate system:

1. Create a map of the maze by using a mapping application, such gmapping or Cartographer. In order to navigate the maze, a route will be planned using this map.

2. Create a coordinate system: Create a system of coor- dinates that converts the drone's location in the maze into a set of (x, y, z) coordinates. The data from the drone's onboard sensors, including a GPS or a LiDAR range finder, as well as the data from the odometry and odometry sensors can be used for this.
3. Create a path: Using the maze's map and the specified coordinate system, create a path from the drone's current location to the destination. An algorithm for path planning, such as A* or Dijkstra's algorithm, can be used to do this.
4. Control the drone's movement by following the predeter- mined route through the maze. A control algorithm, like a state machine or a PID controller, can be used to do this.
5. Continually track the drone's location using odometry data and information from its onboard sensors, and up- date the drone's position in the coordinate system as it navigates the maze.
6. Obstacle detection: To identify obstacles in the drone's path, use sensors like a camera or a LiDAR range finder. Then, change the planned course as necessary.

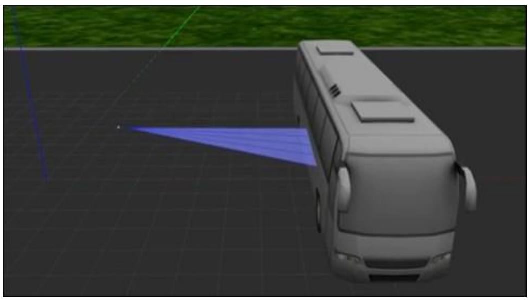

Fig 4. Simple Obstacle Avoidance with Range Finder Simulation

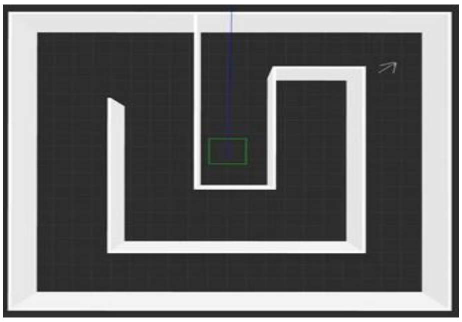

Fig 5. Maze Path Following

E. Obstacle Detection & Path Finder

An autonomous drone must be able to recognize obstacles and determine a route. Both path finding and obstacle detection can be done with a Lidar (Light Detection and Ranging) sensor. A LiDAR sensor uses laser pulses to create a 3D map of the drone's surroundings by measuring the time it takes for the pulses to reflect back. The path that is made using this data avoids obstacles and directs the drone to its intended location. Compared to conventional manual drones, autonomous drones can navigate complicated situations with more accuracy and safety by using a Lidar sensor for obstacle recognition and path finding. A drone with a LiDAR sensor, for instance, might safely fly in locations that would be challenging or impossible to traverse with manual control since it could avoid obstacleslike trees, buildings, and electrical lines.

In this module we have used a Lidar sensor mounted it on drone, this lidar sensor have capability to detect object in 40 degree of range. To find a way when drone is a surrounded with obstacles, first the drone will rotate around 360 degree of angle at the same time the lidar sensor detects the obstacles around the drone and find the empty way to move the drone in specific direction. This task is tested with the help of Gazebo simulator as follows.

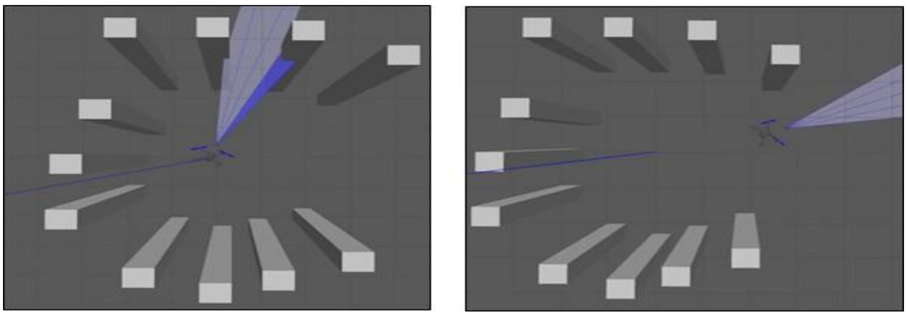

Fig 6. Obstacle Avoidance and Path Finding Simulation

F. Precision Landing with Aruco Marker & Open CV

For an autonomous drone system, it is important that drone should land precisely. In this module we have made us of Open CV and Aruco marker to land drone precisely. Here Aruco maker is similar to QR code but the difference between them is that Aruco marker takes less bits as compared to QR code to form Aruco marker code, because of this it's become easier to detect Aruco marker as compared to QR code. We have made use of Open CV prebuilt libraries for Aruco marker detection. Following steps need to be taken to perform precision landing using Aruco marker.

1. Attach an Aruco marker to the landing pad: The landing pad's centre should be marked with an Aruco marker, a sort of fiducial marker used in computer vision applica- tions. The marking needs to be big enough to be seen from afar.
2. Identify the marker: To identify the Aruco marker in the visual stream, use Open CV and the drone's onboard camera. The Aruco module of Open CV can be used for this because it has capabilities for both detecting and decoding Aruco markers.
3. After the marker has been located, use the Aruco module to determine the marker's pose, or its position, distance and orientation in relation to the camera. The position of the drone with respect to the landing pad can be calculated using this information.

4. Implement safety precautions: Take precautions to avoid collisions or other mishaps while landing. In order to detect obstructions in the environment, additional sensors, such as a LiDAR sensor or an ultrasonic sensor, may be used, such as creating a safety perimeter around the landing pad.
5. In this way precision landing can be done using Lidar sensor. Following are some snapshots of Precision Landing with Aruco Marker Open CV performed and tested using ROS based Gazebo Simulator:

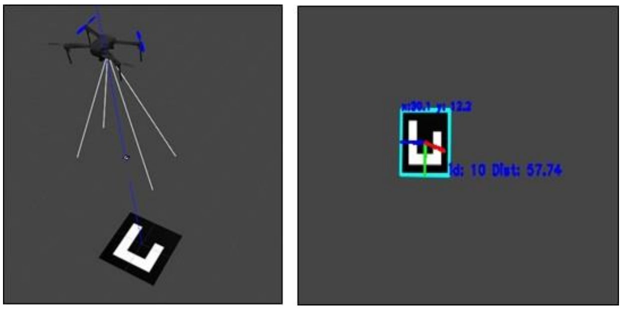

Fig 7.Detecting Aruco Marker and Distance Between Marker and Drone

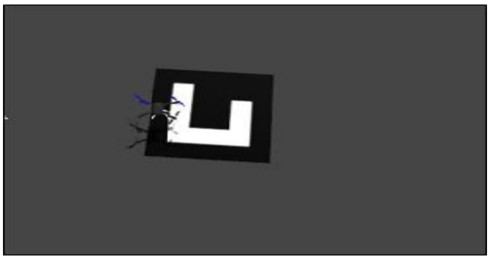

Fig 8. Precision Landing with Aruco Marker and Open CV Simulation

V.CONCLUSION

In this research paper we conclude that, this is very strong, reliable and versatile autonomous drone system which can deployed in many areas for multiple applications that can reduce human efforts and increase work productivity in efficient way.

Acknowledgment

We appreciate our supervisor’s and project guides ongoing advice and assistance throughout this study effort. Their helpful criticism and insightful comments helped us refine our study and reach our goals. We also acknowledge the consistent assistance and direction we received from everyone at the JSPM Jayawantrao Sawant College of Engineering at Pune throughout the course of our research. We also want to express our sincere gratitude to all the scientists and researchers who gave us access to their invaluable materials and enabled us to complete our task. Their contributions have greatly influenced our research and allowed us to explore new directions in this area.

References

[1]. OpenCV: Cascade Classifier.
[2]. Chandan G, Ayush Jain, Harsh Jain, Mohana “Real Time Object De- tection and Tracking Using Deep Learning and OpenCV” Proceedings of the International Conference on Inventive Research in Computing Applications (ICIRCA 2018).
[3]. Aakash Sehrawat, Tanupriya Choudhury, Gaurav Raj “Surveillance Drone for Disaster Management and Military Security” International Conference on Computing, Communication and Automation (IC- CCA2017).
[4]. Reem Alshanbari, Sherjeel Khan, Nazek El-Atab “AI Powered Un- manned Aerial Vehicle for Payload Transport Application”.
[5]. Leo Pauly, Deepa Sankar, “A Novel Online Product Recommendation System Based on Face Recognition and Emotion Detection”, (ICCICCT, 2015).