Designing A Drone for Surveillance and Mapping

DR S.P.V SUBBA RAO[1], CHINMAI[2], RAVI TEJA[3], THANMAI[4]

[1]*Department of Electronics Engineering, Project guide, Sreenidhi Institute of Science and Technology, Hyderabad, Telangana,*

[2, 3, 4]*Department of Electronics Engineering, Sreenidhi Institute of Science and Technology, Hyderabad, Telangana*

Abstract- This project showcases the creation and setup of a budget-friendly, Arduino-powered drone system for watching and mapping. The drone has two main parts: a joystick-run sender unit and a quadcopter receiver, which talk to each other using NRF24L01 wireless devices. The system uses an MPU6050 IMU sensor to keep the flight steady and has coreless motors driven by PWM signals to move. A First Person View (FPV) camera setup gives real-time feedback, making it easier for users to fly. The sender, built with an Arduino Nano, LCD, and joystick parts, sends direction orders. The drone, run by an Arduino Pro Mini, handles these orders along with IMU data using PID control math to fly. The system's design focuses on being cheap, easy to expand, and good for learning, making it perfect for students and hobbyists who want to get hands-on experience with embedded systems, robotics, and wireless talking. This project shows that you can build working drones with open-source parts and sets the stage for more improvements, like self-flying, adding GPS, or mapping environmental data.

Indexed Terms- UAV, Arduino drone, wireless control, PID stabilization, FPV mapping, NRF24L01, coreless motor, IMU integration.

I. INTRODUCTION

Drones, also known as UAVs, have become key tools in many areas such as keeping watch, helping after disasters, checking on crops, and delivering things. They can get real-time info from hard-to-reach or dangerous places, which has changed how we use automation and remote sensing today. But store-bought drones often cost a lot, have complex systems that the maker knows, and don't let users change much. This makes it hard for students, hobbyists, and schools to learn about drones from scratch. To fix these problems, this project creates a cheap, Arduino-based drone system made just for watching and mapping. The system focuses on parts you can swap out, open-source hardware, and learning by doing to make drone tech easier to use and change. The drone is a light four-rotor aircraft with small motors, an MPU6050 IMU sensor to keep it steady in flight, and NRF24L01 devices to talk. It also has an FPV camera that sends live video to the user, which helps them fly better and see more.

The objectives are:
To design and build a low-cost, Arduino-based wireless drone system capable of surveillance and mapping operations:

1. To construct a quadcopter drone using open-source microcontrollers (Arduino Nano and Arduino Pro Mini), with coreless motors for efficient propulsion and lightweight design.
2. To implement stable wireless communication between a joystick-based transmitter and the drone using NRF24L01 transceiver modules.
3. To integrate sensor-based flight stabilization using the MPU6050 Inertial Measurement Unit (IMU) for capturing motion and orientation data in real time.

A. Motivation and Need for Automation:

Drone technology's quick development has had a profound impact on a number of industries, including environmental monitoring, agriculture, disaster relief, and defence. However, commercial UAV systems are too complicated and expensive for students, hobbyists, and educators who want to learn the basics of wireless communication and aerial robotics.

The need to develop an open-source, reasonably priced drone system that prioritises learning via hands-on application served as the impetus for this project. This project reduces the barrier to entry for investigating embedded systems and real-time flight control by constructing the drone entirely from easily accessible parts like Arduino microcontrollers, coreless motors, and NRF24L01 transceivers.

Additionally, automation is essential for flight control and stabilisation. Without sensor feedback, manual flying frequently leads to unsteady and ineffective operation.

By providing the user with a live view from the drone's point of view, an FPV camera enhances automation even more. This mimics the capabilities of high-end commercial drones and helps improve situational awareness during mapping or surveillance tasks.

This project's overall goal is to close the knowledge gap between theory and practice by giving students access to a comprehensive, affordable, automated drone platform that can be comprehended, altered, and expanded for more complex uses down the road.

B. System Design and Tools:

Wireless communication modules, embedded control systems, and lightweight hardware components are all used in the construction of the suggested drone system. Every part is essential to facilitating user interaction, flight stability, and seamless drone operation. Wireless control and real-time feedback are provided by the system's two main components, the transmitter unit and the receiver (drone) unit, which operate in tandem.

1. Transmitter Unit (Controller)

Arduino Nano: acts as the main microcontroller that decodes joystick inputs and communicates control signals to the drone.

Dual Joystick Modules: gives the user real-time control over pitch, roll, yaw, and throttle.

Li-Po Battery (7.4V): ensures portability and continuous operation by supplying power to the transmitter unit.

2. Receiver Unit (Drone/Quadcopter)

Arduino Pro Mini: serves as the drone's onboard flight controller, processing commands and using sensor data to stabilize the drone.

MPU6050 IMU Sensor: provides gyroscope and accelerometer data in real time for balance and orientation.

PID Control Algorithm: By modifying motor speeds in response to the drone's orientation and movement, it guarantees flight stability.

Coreless Motors (x4): Provide lift and movement control for the quadcopter; selected for their lightweight and high RPM characteristics.

Electronic Speed Controllers / PWM Circuit: used with pulse width modulation to change motor speed.

NRF24L01 Transceiver Module: wirelessly receives instructions from the transmitter unit.

FPV Camera Module: records and sends a live video stream for situational awareness and user navigation.

Li-Po Battery: provides power to the drone's motors, receiver circuit, and FPV camera.

Propeller Set and Lightweight Frame: allows for effective lift and manoeuvrability while preserving structural integrity..

C. Prototype Implementation:

The project's prototype shows how inexpensive hardware, wireless communication, and sensor-based flight control can be successfully combined to produce a fully functional quadcopter drone for mapping and surveillance. Real-time drone operation is made possible by the system's two synchronised modules, a quadcopter receiver unit and a joystick-based transmitter unit.

1. Assembly of the Transmitter Unit:

Two analogue joystick modules are interfaced with an Arduino Nano, which serves as the primary microcontroller in the transmitter unit's design. The throttle, yaw, pitch, and roll can all be manually adjusted by the user thanks to the two axes that each joystick controls. The NRF24L01 transceiver processes and transforms these analogue signals into control packets, which are then sent to the drone. A small 7.4V Li-Po battery powers the entire device, and an I2C 16x2 LCD is included to show parameters like joystick input values or status messages.

2. Construction of the Quadcopter Receiver Unit
The Arduino Pro Mini, at the heart of the drone's receiver unit, decodes incoming data from the transmitter. For stability and orientation during flight, an MPU6050 sensor continuously provides gyroscope and accelerometer data. A PID (Proportional-Integral-Derivative) control algorithm processes this sensor data and determines the adjustments each motor needs to make in order to achieve balanced flight.

An H-bridge or PWM motor driver circuit connects the four coreless motors, which are positioned equally apart on a lightweight quadcopter frame, allowing for precise speed control. The drone's NRF24L01 module updates its motor outputs in response to commands from the transmitter. An FPV (First Person View) camera module that streams live video is powered by a rechargeable Li-Po battery.

3. Integration and Testing
The NRF24L01 modules were used to pair the drone units and transmitter after hardware integration in order to create reliable wireless communication. To make sure the drone reacted promptly and precisely to orientation changes picked up by the IMU sensor, the PID algorithm was calibrated by varying the gain values (Kp, Ki, and Kd).

The drone successfully responded to joystick commands for elevation, direction, and rotation during controlled flight tests. Without direct line of sight, the operator was able to control the drone efficiently thanks to the FPV system's real-time visual feedback. The control logic and motor configuration were successfully implemented, as evidenced by the stability during hover and motion.

D. Possible Future Enhancements:
Although the current prototype is a good starting point, there are a few improvements that could make it more useful. Autonomous navigation would be possible with the addition of a GPS module, and collisions during flight could be avoided with obstacle detection sensors. A mobile app interface would be more portable and user-friendly than a joystick. Its usefulness for monitoring tasks could be increased by integrating environmental sensors. Remote access to flight data and video feeds would be made possible by real-time cloud data transmission. Solar charging or intelligent battery management could increase power efficiency. AI-based capabilities like object tracking and autonomous flight paths may eventually enable the drone to be used for more complex tasks like search and rescue and surveillance.

II. LITERATURE REVIEW

Sharma and Kumar (2018) developed an Arduino-based quadcopter model for educational use, focusing on basic wireless control and manual navigation. Their work demonstrated the potential of open-source hardware for building functional UAV prototypes. However, the lack of onboard stabilization limited the system's real-time performance.

Rao et al. (2019) presented a wireless-controlled drone system using NRF24L01 modules, emphasizing long-range communication between transmitter and receiver units. Their work laid the foundation for reliable two-way data exchange in cost-sensitive drone designs.

Patel and Verma (2020) implemented an IMU-based stabilization system using the MPU6050 sensor, combined with PID control logic, to achieve stable flight in quadcopters. Their research highlighted the effectiveness of real-time sensor feedback in maintaining drone balance and responsiveness during flight.

Ali and Deshmukh (2021) integrated an FPV camera into a drone system, allowing operators to receive real-time visual feedback. This development greatly improved usability in applications like aerial inspection and mapping, proving the value of visual telemetry.

Mohan et al. (2022) designed a modular drone prototype with a focus on scalability and future expansion, proposing features like GPS integration and autonomous flight planning. Their approach emphasized the importance of flexible architecture in educational and research-based UAV platforms.

III. RESEARCH METHODOLOGIES

In order to create and implement a reliable, wireless, and reasonably priced drone system, this project combines wireless communication, embedded control, sensor integration, and real-time feedback. The development process is divided into modular stages to ensure clarity and ease of implementation.

The drone's control system is based on an Arduino Pro Mini microcontroller that receives input from an MPU6050 IMU sensor. This sensor provides real-time accelerometer and gyroscope data, which are processed using a PID (Proportional-Integral-Derivative) control algorithm to stabilise flight. The PID controller calculates error values and employs PWM signals to adjust motor speeds in accordance with the intended setpoints and the drone's actual orientation.

Two analogue joysticks connected to an Arduino Nano make up the joystick-based transmitter unit that the user uses to control the drone. The Nano uses NRF24L01 wireless transceivers to read the analogue values and send them to the drone. Commands are received by the receiver end and forwarded to the flight controller via the same module.

Four coreless motors set up in a quadcopter pattern are used in the propulsion system. MOSFET-based motor driver circuits that are PWM-controlled from the Arduino power each motor. By altering their speeds in response to the PID adjustments, the motors provide the required lift and directional control.

The drone is equipped with an FPV (First Person View) camera for real-time navigation and user awareness. It gives the user visual feedback while in flight by streaming live video back to them. The drone's onboard electronics and motors are powered by a Li-Po battery, and the transmitter unit is powered by a different battery.

In order to achieve stable hover and manoeuvrability, the system was first put together and tested in phases, starting with individual module testing and progressing to full system integration and PID value tuning. During this stage, power efficiency, signal dependability, and wireless range testing were also assessed.

This approach guarantees that every subsystem operates independently prior to integration, facilitating simpler troubleshooting, dependable operation, and future drone platform scalability.

IV. SYSTEM OVERVIEW

The suggested drone system is a wireless quadcopter that is modular and was created with open-source hardware to provide functionality, affordability, and educational value. With the capacity to send control signals and real-time video feedback, its main uses are in mapping and surveillance applications. The Transmitter Unit (ground control) and the Receiver Unit (drone) are the two primary modules that make up the system.

The Arduino Nano microcontroller, which powers the transmitter unit, communicates with two analogue joystick modules to allow for user input. These joysticks regulate the drone's roll, pitch, yaw, and throttle. Real-time wireless data transfer to the drone is made possible by an NRF24L01 transceiver module. Optional user feedback on command values or signal strength is provided by a 16x2 LCD. A small 7.4V Li-Po battery powers the entire transmitter system, guaranteeing portability and continuous operation during flight sessions.

An Arduino Pro Mini serves as the flight controller for the Receiver Unit, which is fixed to the quadcopter frame. It uses a second NRF24L01 module to process incoming signals from the transmitter. The drone's orientation is continuously monitored by an MPU6050 IMU sensor, which provides gyroscope and accelerometer data. By employing PWM signals to modify the speed of the four coreless motors, a PID control algorithm uses this data to preserve stability and balance while in flight. For effective power delivery, MOSFET-based drivers are used to connect these motors.

In order to facilitate navigation and remote observation, the drone also has a First Person View (FPV) camera that allows it to send a live video feed to the user. The drone's lightweight rechargeable Li-Po battery powers all of its electronic parts. To enable steady and manoeuvrable flight, the drone's frame is made of lightweight, durable materials.

In general, the system is made to run on its own once it is powered. The drone uses sensor feedback to stabilise itself after being activated, receives control signals from the transmitter, and streams video through the FPV module. With the ability to scale up by adding features like GPS, object tracking, or environmental data logging, the system's modular design makes it appropriate for scholarly research.

Performance Requirements
For this system to function well, it needs to:

- Establish a stable wireless connection between the transmitter and the drone using NRF24L01 modules for real-time control.
- Accurately read joystick inputs and translate them into throttle, pitch, roll, and yaw commands for responsive drone maneuvering.
- Continuously monitor orientation data from the MPU6050 sensor to maintain flight stability using PID control.
- Adjust motor speeds in real time via PWM signals based on sensor feedback to ensure smooth and balanced flight.
- Stream live video feed through the FPV camera for effective remote navigation and situational awareness.

V. SYSTEM DESIGN AND METHODOLOGY

This project's main goal is to use open-source hardware to create a working quadcopter drone system that is wireless, stable, and controllable in real time. The transmitter unit (controller) and the receiver unit (drone) are the two main parts of the system's design. The project's operational logic, software, and hardware are described below.

A. Hardware Elements

1. Transmitter Unit:
Arduino Nano serves as the control center, reading inputs from two analog joystick modules.

Joystick Modules control pitch, roll, yaw, and throttle.

NRF24L01 Module transmits control signals wirelessly to the drone.

LCD Display (16x2 with I2C) optionally displays real-time data such as battery status or joystick values.

Li-Po Battery (7.4V) powers the transmitter, making it portable and independent.

2. Drone (Receiver) Unit
Arduino Pro Mini functions as the drone’s flight controller.

MPU6050 Sensor continuously provides accelerometer and gyroscope data for orientation tracking.

PID Control Logic stabilizes the drone by adjusting motor speeds based on IMU data.

NRF24L01 Module receives joystick inputs from the transmitter.

Coreless Motors (x4) connected via PWM circuits to allow lift and directional control.

Li-Po Battery powers the drone, motors, and camera.

3. Software Design
Arduino IDE is used to write and upload code to both the transmitter and receiver microcontrollers.

The transmitter code reads analog joystick values and sends them via the NRF24L01 module.

The receiver code uses a PID algorithm to process IMU data and adjust motor speeds accordingly.

PWM signals are generated to control motor thrust for flight adjustments.

The FPV system operates independently but in synchronization with the drone's operation.

4. Operational Flow
Initialization: The transmitter and receiver are powered on and establish a wireless link.

Control Input: The user manipulates the joysticks to send directional commands.

Data Processing: The drone reads joystick inputs and IMU data simultaneously.

Stabilization: The PID algorithm calculates motor adjustments to maintain balance.

Flight Control: PWM signals vary motor speeds to execute movement commands.

Video Feedback: The FPV camera provides real-time visuals for remote navigation.

Power Monitoring: Voltage and stability are monitored throughout to ensure safe operation.

VI. IMPLEMENTATION DETAILS

To confirm the wireless control, flight stabilisation, and real-time video transmission capabilities, the drone system was put together and tested in a controlled setting. Both software development using the Arduino platform and hardware integration were required for the implementation.

1. Assembly Process:
Transmitter Construction:
An Arduino Nano was used in the construction of the joystick-controlled transmitter unit. The NRF24L01 module was used to transmit the data, and two analogue joystick modules were connected to read directional input (pitch, roll, throttle, and yaw). It was optional to connect a 16x2 I2C LCD for feedback display. A 7.4V Li-Po rechargeable battery powered the transmitter.

Drone Setup:
The quadcopter frame was assembled with four coreless DC motors mounted at equal distances. These motors were connected to MOSFET-based motor drivers controlled through PWM outputs from the Arduino Pro Mini. The MPU6050 IMU sensor was mounted centrally for accurate orientation sensing. The NRF24L01 module was connected to receive control signals from the transmitter. A compact FPV camera was mounted on the front and connected to a video transmitter.

2. Software Integration
The Arduino IDE was used to program the drone receiver and transmitter. The joysticks' analogue inputs were converted to control values on the transmitter side and transmitted as wireless packets. On the drone side, MPU6050 data was used to process received inputs and apply PID-based corrections. PWM signals were produced in order to suitably modify the motor speeds. After initialization, the FPV system ran continuously and was powered independently.

3. Testing and Calibration
The transmitter and receiver's wireless communication was verified during the first testing. The drone was powered on after stable communication was established, and PID tuning was done by modifying the derivative, integral, and proportional constants to minimize drift and maintain hover. Visual clarity and latency during movement were evaluated for the FPV system.

The drone hovered steadily with little oscillation and reacted precisely to joystick commands. Directional control and orientation were aided by video feedback. During test flights, power consumption was tracked to guarantee battery endurance.4. Block Diagram

Block Diagram Description:

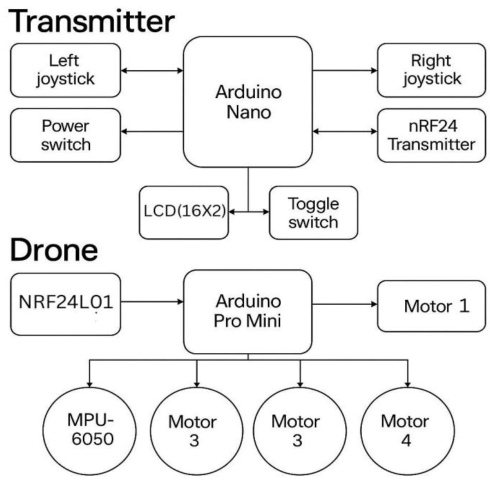


Fig.2: Block Diagram

The block diagram has labeled components and arrows indicating flow; the textual description is given below.

Arduino Nano:
Acts as the central controller of the transmitter. It reads the analog signals from the joysticks and digital signals from the switches. Converts this user input into digital data that can be transmitted wirelessly. Useful because it's small, cheap, and easy to program, making it ideal for embedded remote systems

Joysticks (Left and Right):
Allow precise manual control of the drone's movement in all directions.

Each joystick can control two axes:
- Left: Throttle (up/down), Yaw (rotation)
- Right: Pitch (forward/backward), Roll (side to side)

Joysticks make flight control intuitive and responsive.

Power Switch:
Ensures the transmitter is only active when needed. Essential for power conservation and safety.

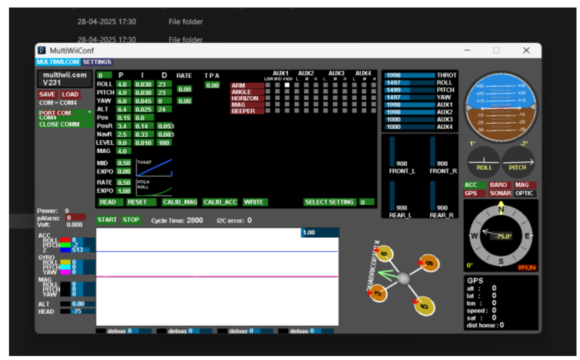
nRF24 Transmitter Module:
Sends data wirelessly to the drone at 2.4 GHz. Known for low power consumption, high speed, and reliable communication.

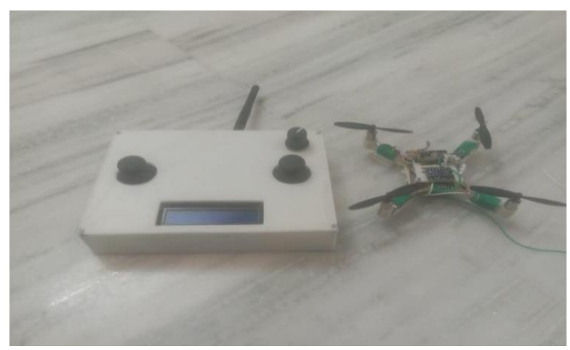
Allows you to control the drone from a distance wirelessly and efficiently.

Toggle Switch:
Can be programmed to enable special modes:
- e.g., auto-level, return-to-home, emergency stop.

Adds flexibility to your drone control system.

VII. DATA EXAMINATION AND RESULTS





The developed system's automated vehicle movement control, wireless charging capabilities, and voltage monitoring accuracy were evaluated in a lab setting. A robotic EV prototype and a solar-powered transmitter setup were used for several test runs under steady indoor lighting conditions.

A. Configuring the Test

1.Through the use of a current-regulated transistor circuit, a 7V solar panel was linked to a rechargeable battery to power the transmitting unit.

2.The EV prototype's receiving side featured a voltage sensor, a Li-ion battery, and a control system based on NodeMCU.

3.Wireless energy transfer was made possible by positioning the vehicle over the transmitter coils.

4.An LCD display installed on the car was used to continuously monitor the voltage data.

B. Important Findings

1.The receiving coil effectively harvested energy through inductive coupling when the vehicle was properly positioned in relation to the transmitter coils.

2.The voltage level gradually increased as the Li-ion battery started charging right away.

3.Accurate and consistent real-time voltage updates were provided by the LCD display.

4.During operation, the buzzer consistently sounded when the voltage fell below the predetermined threshold.

5. The system automatically powered the DC motors through the L293D motor driver to get the car moving once the battery was fully charged.

C. Summary of Performance

Wireless Charging Reliability: Energy transfer was dependable and effective when the vehicle was positioned correctly with respect to the transmitter coil.

Voltage Monitoring: The readings closely matched the expected values, confirming the voltage sensor's accuracy and response time.

VIII. DISCUSSION

The created drone system successfully illustrates how wireless communication, embedded control, and real-time sensor feedback can all be combined into a small, reasonably priced aerial platform. While maintaining dependable functionality, the use of open-source parts like Arduino microcontrollers, MPU6050 IMUs, and NRF24L01 modules helped to lower the overall cost.

By correctly calibrating the PID algorithm, the drone demonstrated responsive flight behaviour and stable hover during testing. Accurate orientation readings from the MPU6050's sensor data enabled the controller to modify the motor speed as needed. Within the anticipated 10- to 15-meter range, the transmitter and drone's wireless link maintained consistent communication with low latency.

The user's ability to manoeuvre and control the drone without keeping a direct line of sight was greatly enhanced by the FPV camera system's real-time visual feedback. The drone's use in mapping and surveillance is improved by this visual data stream, particularly in hazardous or inaccessible locations.

A few restrictions were noted during implementation, though. Although the lightweight coreless motors are appropriate for simple indoor applications, they might not offer enough lift or endurance in windy outdoor environments. Depending on load and manoeuvring, battery life also restricts flight time, which is usually less than ten minutes. Furthermore, careful experimentation was necessary to tune the PID controller because stability was impacted by even small adjustments.

Despite these drawbacks, the project achieves its main goals and offers a strong basis for future improvements. It shows how inexpensive, modular drone systems can be efficiently constructed and used for research, teaching, and practical applications.

CONCLUSION

The design and development of an inexpensive, Arduino-based drone system with wireless control, flight stabilisation, and real-time video transmission is successfully demonstrated by this project. Using open-source microcontrollers, IMU-based PID control, wireless communication via NRF24L01 modules, and FPV video streaming, the drone provides a useful solution for robotics and embedded systems education and experimentation.

The system is accessible to students, hobbyists, and researchers due to its modular architecture and reasonably priced components, offering a useful platform for experiential learning. The drone's controlled and balanced flight was made possible by the use of sensor-based stabilisation, and its user-friendly manual navigation was made possible by joystick input. By providing real-time visual feedback, an FPV camera improved user interaction.
All things considered, the project achieves its goals of creating a working quadcopter with feedback, stability, and real-time control. In keeping with the expanding need for intelligent aerial systems across a range of industries, it establishes the groundwork for future improvements like autonomous navigation, obstacle detection, and environmental data collection.

FUTURE SCOPE

GPS Integration for Autonomous Flight: For mission-based drone operations, future iterations might incorporate GPS modules to facilitate waypoint navigation, geofencing, and automated return-to-home capabilities.

Obstacle Detection and Avoidance: To improve flight safety and make the drone appropriate for challenging environments, ultrasonic or infrared sensors can be added to detect and avoid obstacles in real time.

Mobile App-Based Control: Wireless control, improved portability, and real-time parameter tuning and monitoring options can be obtained by substituting a mobile application for the physical joystick.

Environmental Data Monitoring: The drone can become a flying data collection device for industrial, agricultural, or disaster-related use cases by integrating temperature, gas, or air-quality sensors.
Smart Power Management and Solar Charging: Solar panels and intelligent battery management systems can be added to the system to increase power efficiency and flight duration.

Cloud-Based Telemetry and Video Storage: For mission tracking, performance analysis, and remote access, real-time flight data and FPV video can be uploaded to cloud platforms.

AI-Based Navigation and Object Tracking: Autonomous features like patrol routes and subject following are made possible by the use of computer vision and machine learning algorithms for object detection, tracking, and gesture recognition.

Scalability to Outdoor Applications: The drone can be modified for outdoor missions and industrial tasks with stronger motors, weather-resistant parts, and longer-range communication.

REFERENCES

[1] Sharma, A., & Kumar, R. (2018). Design and Development of an Arduino-Based Quadcopter for Educational Purposes. International Journal of Engineering Research and Applications, 8(3), 45–50.

[2] Rao, V., & Singh, P. (2019). Wireless Communication in Drones Using NRF24L01 Transceivers. International Journal of Embedded Systems and Applications, 7(2), 22–29.

[3] Patel, S., & Verma, N. (2020). Stabilization of Drones Using MPU6050 and PID Controller. Journal of Control Systems and Robotics, 5(1), 11–19.

[4] Ali, H., & Deshmukh, P. (2021). FPV Integration in Quadcopters for Enhanced Navigation. International Conference on Robotics and Automation Technologies, 62–68.

[5] Mohan, A., Reddy, B., & Prasad, K. (2022). Design of a Modular Drone Architecture for Research Applications. Journal of Applied Embedded Systems, 4(4), 91–98.

[6] Tanveer, M., & Singh, R. (2017). Quadcopter Flight Stabilization using PID Controller and IMU Sensor. International Journal of Electrical and Electronics Research, 5(2), 75–80.

[7] Kurs, A., Karalis, A., Moffatt, R., Joannopoulos, J. D., Fisher, P., & Soljačić, M. (2007). Wireless Power Transfer via Strongly Coupled Magnetic Resonances. Science, 317(5834), 83–86

[8] Sample, A. P., Meyer, D. T., & Smith, J. R. (2011). Analysis, Experimental Results, and Range Adaptation of Magnetically Coupled Resonators for Wireless Power Transfer. IEEE Transactions on Industrial Electronics, 58(2), 544–554.