

Autonomous Drone-Based Food Delivery System Using GPS Navigation and IOT Monitoring

Sakshi Sharad Nagare¹, Rutika Rajesh Nerkar²

^{1,2} Department of Electronics & Telecommunication Engineering,
Matoshri College of Engineering and Research Center, Eklahare, Nashik, Maharashtra, India
(Affiliated to Savitribai Phule Pune University) —

Project Guide: **Dr. Ravindra G. Dabhade**

Corresponding author: firstauthor@example.com

Abstract – Conventional road-based food delivery depends heavily on human couriers and is slowed by traffic congestion, rising fuel and labour costs, and poor reach to remote or crowded locations. This paper presents the design and prototype implementation of an autonomous quadcopter for contactless food delivery. The drone is built on a 450 mm frame with four 1000 KV brushless motors and 30 A electronic speed controllers, and is governed by an APM 2.8 flight controller running ArduPilot firmware. A u-blox NEO-6M GPS module and an MPU6050 inertial measurement unit provide position and attitude data, while an HC-SR04 ultrasonic sensor assists altitude hold and safe landing. An ESP32 module handles wireless telemetry and payload commands, and a Raspberry Pi camera streams live video for monitoring and delivery verification. Flight missions are defined as GPS waypoints in Mission Planner; the drone performs automated take-off, autonomous navigation, servo-actuated package release, and return-to-home without manual control. The prototype demonstrates stable flight, accurate waypoint navigation, mass-light payload release, and reliable return, confirming that UAV and IoT technologies can be combined into a fast, eco-friendly, and scalable delivery solution suited to urban, campus, remote, and smart-city logistics.

Key Words: *Unmanned aerial vehicle, drone delivery, GPS waypoint navigation, APM 2.8 flight controller, ESP32, IoT telemetry, autonomous flight, servo payload release.*

I. INTRODUCTION

The food-delivery industry has expanded rapidly with the growth of online ordering platforms and the demand for quick, on-demand service. Most deliveries, however, still rely on human couriers and road transport, which makes them vulnerable to traffic congestion, high fuel and manpower costs, and long delays during peak hours. Reaching remote, hilly, or densely crowded areas is especially difficult and time-consuming by road.

Unmanned aerial vehicles (UAVs), or drones, offer a modern and contactless alternative. By using aerial routes, a drone-based system avoids road traffic, reduces dependence on delivery personnel, and shortens delivery time. When equipped with satellite navigation and embedded control electronics, such a drone can fly autonomously between a source and a destination with minimal human intervention.

This work presents the design and prototype of a quadcopter that delivers lightweight food packages to predefined GPS coordinates. The complete flight path is programmed in Mission Planner software, and the drone performs automated

take-off, waypoint navigation, package release, and return-to-home. An onboard camera and wireless telemetry give the operator a live view of the mission. The objectives of the project are: (i) to develop a drone-based autonomous food-delivery platform; (ii) to use GPS navigation for accurate, automated route control; (iii) to provide real-time monitoring through an onboard camera and IoT telemetry; (iv) to ensure flight stability and safety using inertial and ultrasonic sensing; and (v) to reduce delivery time, human effort, and fuel consumption.

The remainder of the paper follows the IMRaD structure. Section II reviews related delivery and UAV systems. Section III describes the system architecture, hardware and software subsystems, and the operational sequence. Section IV reports the observed prototype operation, and Section V discusses the results, advantages, and limitations. Section VI concludes and outlines future scope.

II. LITERATURE REVIEW

Traditional human-based delivery, performed entirely over road networks, remains the most common model, but it is intrinsically limited by traffic delays, fuel usage, and the need for a courier per order. These limitations motivate automated and aerial alternatives that can bypass ground congestion.

Ground delivery robots automate short-range, last-mile delivery on pavements and within campuses, yet they cannot negotiate uneven terrain, stairs, or large obstacles, which restricts them to controlled urban environments. Aerial platforms overcome this by removing the dependence on ground surfaces altogether.

Research on drone parcel delivery has demonstrated GPS-based route navigation with a high degree of automation. Anderson [1] surveys the broader drone economy and its delivery applications, while Austin [2] and McGrath [3] provide the design foundations for UAV systems and embedded robotic control. These works confirm the feasibility of waypoint-driven autonomous flight but note that flight time and payload capacity remain the chief constraints.

Singh and Patel [4] proposed an autonomous drone-delivery scheme that integrates GPS with IoT communication, and Sharma and Joshi [5] reported a dedicated drone-based food-delivery design; both highlight the value of real-time tracking but report limited range owing to battery power. Lee and Chen [6] focused on optimising last-mile delivery through UAV-based systems, and Rahman [7] examined real-time monitoring with adaptive route optimisation, pointing to energy management as a key area for improvement.

Taken together, the literature establishes that autonomous, GPS-guided aerial delivery is technically viable, while battery endurance, payload limits, and live monitoring remain open issues. The present work addresses this gap with a low-cost quadcopter that combines GPS waypoint navigation, ESP32-based IoT telemetry, and onboard camera monitoring within a single, repeatable delivery cycle.

III. METHODOLOGY

A. System Architecture

The system is organised around the APM 2.8 flight controller, which acts as the central processing and control unit. It reads attitude data from the onboard MPU6050 inertial measurement unit and position data from the u-blox NEO-6M GPS module, and drives the four electronic speed controllers (ESCs) that regulate the brushless motors. A 3S 11.1 V Li-Po battery feeds a power-distribution board (PDB), which supplies high current to the ESCs and, through a voltage regulator, a stable rail to the controller, GPS, sensors, and communication modules. An ESP32 module provides the wireless telemetry and payload-command link, while a Raspberry Pi with a camera module performs live video streaming and delivery verification. The overall signal and power flow is shown in Fig. 1, and the principal components are summarised in Table I.

[Insert original high-resolution figure (min 300 DPI)]

Fig. 1. Block diagram of the autonomous drone-based food-delivery system.

TABLE I. PRINCIPAL SYSTEM COMPONENTS

Component	Specification	Role in the System
APM 2.8 flight controller	ATmega2560, ArduPilot	Stabilization, waypoint execution, RTH, fail-safe
BLDC motors (×4)	1000 KV, A2212-class	Generate lift and thrust
ESC (×4)	30 A, PWM input	Regulate motor speed for each rotor
Airframe	450 mm quadcopter	Structural mounting and balance
Propellers (×4)	10×4.5 in, 2 CW + 2 CCW	Convert rotation into lift and yaw control
GPS module	u-blox NEO-6M	Real-time position for navigation and RTH
IMU	MPU6050 (6-axis)	Orientation/attitude sensing for stability
Rangefinder	HC-SR04 ultrasonic	Altitude hold, obstacle detection, soft landing
Telemetry unit	ESP32 (Wi-Fi)	Wireless telemetry and payload command link
Onboard vision	Raspberry Pi + camera	Live video feed and delivery verification
Payload mechanism	Servo-actuated drop box	Secures and releases the food package
Power	3S Li-Po 11.1 V 5500 mAh + PDB	Powers motors, controller, and electronics

B. Hardware Subsystem

The propulsion subsystem uses four 1000 KV brushless DC motors driven by 30 A ESCs and fitted with 10×4.5 in propellers in a two-clockwise, two-counter-clockwise arrangement, mounted on a 450 mm quadcopter frame. This combination provides sufficient thrust and yaw control for a medium payload while keeping the airframe lightweight. The

stabilisation subsystem comprises the MPU6050, which continuously measures roll, pitch, and yaw so that the flight controller can apply PID corrections, and the HC-SR04 ultrasonic sensor, which measures ground distance for altitude hold and soft landing. Navigation is provided by the NEO-6M GPS module, which supplies latitude, longitude, and altitude for waypoint following and return-to-home.

The payload subsystem is a servo-actuated drop box: the servo holds the package closed in flight and rotates to release it at the destination. Status indication is provided by a buzzer and LED indicators that signal arming, GPS lock, low battery, and successful delivery. Key design values are listed in Table II.

TABLE II. SYSTEM DESIGN PARAMETERS

Parameter	Design value
Airframe	450 mm quadcopter (X-configuration)
Flight controller	APM 2.8 (24 ch, ArduPilot)
Propulsion	4 × 1000 KV BLDC, 4 × 30 A ESC
Propellers	10×4.5 in (2 CW + 2 CCW)
Battery	3S Li-Po, 11.1 V, 5500 mAh
Navigation	GPS waypoint (u-blox NEO-6M)
Stabilization	MPU6050 IMU + PID (ArduCopter)
Telemetry / control	ESP32, Wi-Fi 802.11 b/g/n
Payload release	Servo-actuated drop box
Flight modes	Auto take-off, waypoint, RTH, land
Ground software	Mission Planner (MAVLink)

C. Software and Communication

Mission Planner is used as the ground-control station to configure the APM controller, define GPS waypoints, and monitor live telemetry over the MAVLink protocol. The Arduino IDE is used to program the ESP32 and to interface the auxiliary sensors and payload servo. The Raspberry Pi runs a Linux-based operating system with Python and OpenCV for camera handling and live video streaming. Telemetry such as drone location, altitude, speed, and battery status is transmitted by the ESP32 to a browser or mobile dashboard, and commands such as payload release can be sent back to the drone over the same link.

D. Operational Sequence

The delivery cycle proceeds in five stages, as shown in the operational flow chart of Fig. 2. In the mission-setup stage, the operator enters the delivery coordinates in Mission Planner, which generates a set of GPS waypoints that are uploaded to the controller; the package is loaded into the servo box and the battery, motor calibration, and GPS lock are verified. In the take-off stage, the controller stabilises the drone using IMU and GPS data while the ESCs lift it smoothly. During autonomous navigation, the drone follows the uploaded waypoints without manual control, the ESP32 streams telemetry, the camera streams live video, and the ultrasonic sensor maintains safe altitude. In the delivery stage, the drone descends to the target, the ultrasonic sensor confirms a stable hover or landing, the servo opens the box to release the package, and a buzzer and LED confirm delivery. Finally, in the return-to-home stage, the controller navigates the drone back to its launch point using GPS and performs a controlled landing.

[Insert original high-resolution figure (min 300 DPI)]

Fig. 2. Operational flow chart of the delivery cycle.

IV. RESULTS

The assembled prototype executed the full delivery cycle under autonomous control. On start-up, the controller verified GPS lock and sensor health before arming, and proceeded to take-off only when these checks passed. Using the APM 2.8 controller, the drone achieved stable take-off and controlled landing, and held a level attitude in hover, with the MPU6050 and PID loop correcting for small disturbances.

During the mission, the drone navigated automatically along the GPS waypoints uploaded through Mission Planner, and the ESP32 transmitted live telemetry — location, altitude, speed, and battery status — to the monitoring dashboard. The Raspberry Pi camera streamed a live video feed throughout the flight, allowing the operator to observe the route and confirm arrival. On reaching the destination, the servo mechanism released the package as commanded, and the buzzer and LED confirmed delivery; the drone then switched to return-to-home and landed at the launch point without manual intervention.

These observations provide a qualitative validation of the navigation, monitoring, and payload-release functions. Quantitative characterisation of delivery accuracy, payload limits, and flight endurance across repeated missions is identified as future work.

V. DISCUSSION

Governing the delivery by GPS waypoints and automated stage sequencing removes the principal sources of delay and inconsistency in road-based delivery — traffic and courier availability — and is consistent with the autonomy demonstrated in prior UAV-delivery studies [4]–[6]. Compared with ground delivery robots, the aerial platform reaches locations that are inaccessible by road, and compared with manual road delivery it offers a faster, contactless, and more hygienic service that is particularly valuable during health emergencies. The use of a rechargeable Li-Po battery rather than fuel also lowers operating emissions.

Several limitations remain. Payload capacity is restricted, and increasing it reduces flight time and stability. Battery endurance limits the practical delivery radius, and the system is sensitive to strong wind, rain, and fog, which can affect both stability and GPS accuracy. Flight is also subject to regulatory restrictions near airports and crowded zones, and any loss of GPS or wireless signal degrades navigation. Finally, the present prototype has not yet been characterised quantitatively for accuracy and repeatability. These constraints define the priorities for the future work described below.

VI. CONCLUSION

This paper presented an autonomous quadcopter that automates the take-off, GPS-waypoint navigation, payload release, and return-to-home cycle for contactless food delivery. The prototype demonstrated stable autonomous flight, accurate waypoint following, live camera and telemetry monitoring through ESP32 and a Raspberry Pi, and reliable servo-based package release, using low-cost components suited to academic and small-scale deployment. The main limitations are restricted payload and flight endurance, weather and regulatory sensitivity, and the absence of quantitative accuracy testing.

Future work will extend the platform with higher-thrust motors and stronger frames for larger payloads, higher-capacity or hybrid power for longer range, AI-based route optimisation, and LIDAR or vision-based obstacle avoidance. Automatic charging or battery-swap stations, a temperature-controlled delivery box, integration with food-ordering applications, multi-drone swarm coordination, and geo-fenced flight authorisation would advance the system toward a smarter and more sustainable delivery solution.

ACKNOWLEDGEMENT

The authors thank their project guide, Dr. Ravindra G. Dabhade, for his valuable guidance, and Dr. Devidas D. Dighe (Head, Department of E&TC) and Dr. Gajanan K. Kharate (Principal) for their support. They are also grateful to the management of Matoshri Education Society, Nashik, and to the faculty of the Department of Electronics and Telecommunication Engineering for providing the facilities used in this work.

REFERENCES

- [1] C. Anderson, *The Drone Economy: The Future of Flight and Delivery Systems*. New York, NY, USA: HarperCollins, 2021.
- [2] R. Austin, *Unmanned Aircraft Systems: UAV Design, Development and Deployment*, 2nd ed. Chichester, UK: Wiley, 2016.
- [3] J. E. McGrath, *Introduction to Robotics and Drone Systems*. Pearson Education, 2020.
- [4] A. Singh and R. Patel, "Autonomous drone delivery system using GPS and IoT integration," *Int. J. Emerg. Technol. Eng. Res. (IJETER)*, vol. 8, no. 12, 2022.
- [5] K. Sharma and N. Joshi, "Design and implementation of a drone-based food delivery system," *J. Robot. Autom. Res.*, 2021.
- [6] J. Lee and M. Chen, "Optimizing last-mile delivery through UAV-based systems," *IEEE Access*, 2020.
- [7] A. Rahman, "Smart delivery drones: real-time monitoring and route optimization," *Sensors and Systems Journal*, 2023.
- [8] *Mission Planner Documentation*, ArduPilot Development Team, 2023. [Online]. Available: <https://ardupilot.org/planner>
- [9] *Raspberry Pi Camera Setup Guide*, Raspberry Pi Foundation, 2023. [Online]. Available: <https://www.raspberrypi.org/documentation>
- [10] *ESP32 Developer Documentation*, Espressif Systems, 2023. [Online]. Available: <https://docs.espressif.com>
- [11] "Drone delivery: advantages and challenges in urban areas," *DJI Enterprise Blog*, 2023. [Online]. Available: <https://enterprise.dji.com>
- [12] "Integration of GPS and sensors for autonomous drones," *Arduino Project Hub*, 2023. [Online]. Available: <https://projecthub.arduino.cc>