Journal of Electronic Research and Application, 2026, Volume 10, Issue 1
http://ojs.bbwpublisher.com/index.php/JERA
ISSN Online: 2208-3510
ISSN Print: 2208-3502

Research on Multi-Rotor UAVs in Complex Indoor Environments

Hui Li, Zheyuan He, Yuehua Cao*, Rujun Xue, Shiyue Zhang, Haonan Ye

Information Engineering College, Hangzhou Dianzi University, Hangzhou 311305, China

**Author to whom correspondence should be addressed.*

Abstract: The exponential growth of unmanned aerial vehicle (UAV) technology has spurred its adoption in diverse indoor applications, including infrastructure inspection, automated logistics, and emergency response. However, navigating through indoor environments, characterized by static obstacles, dynamic interferences, and spatial constraints, poses significant challenges to path planning algorithms. Developing efficient, robust, and real-time path planning solutions is crucial for enabling reliable autonomous UAV operations in such complex scenarios. This study presents a systematic approach to indoor UAV navigation, integrating custom hardware development, algorithmic innovation, and multi-faceted validation. An indoor UAV experimental platform was constructed around the Pixhawk 2.4.8 flight controller, complemented by a Firefly Core-3588L onboard computer, PXYZ-D435 depth camera, and OptiTrack motion capture system. After rigorous PID tuning and endurance testing, stable autonomous flight control was achieved via the Robot Operating System. Subsequent real-world tests on the custom UAV platform, involving obstacle courses and narrow passage traversals, further confirmed its robustness and stability in complex indoor environments. Overall, this research provides a practical framework for enhancing UAV navigation capabilities, with direct implications for real-world applications in logistics, surveillance, and emergency response.

Keywords: Kmulti-rotor UAV; Path planning; Trajectory optimization; Indoor navigation

Online publication: February 12, 2026

1. Introduction

As the core engine driving the intellectual evolution of unmanned systems, UAV path planning technology has gained paramount strategic value amid the explosive growth of the global low-altitude economy and intelligent unmanned equipment industry [1–4]. UAVs have deeply penetrated diverse domains, including military reconnaissance, geographic mapping, smart logistics (e.g., Amazon Prime Air, JD.com's unmanned delivery), precision agriculture (e.g., crop monitoring and variable-rate pesticide application), and disaster response (e.g., earthquake rescue and material delivery). However, bottlenecks such as insufficient autonomous navigation in complex dynamic environments, low multi-vehicle collaboration efficiency, and energy constraints in long-

endurance missions severely hinder their large-scale adoption. Grand View Research statistics show that the global UAV path planning market reached $3.27 billion in 2023, projected to exceed $12.6 billion by 2030 with a compound annual growth rate (CAGR) of 21.4%. This surge is intimately linked to emerging demands, including smart city unmanned traffic management (UTM) systems, unmanned border security patrols, and inspections of new energy power grids [5].

From a technical perspective, traditional path planning algorithms (e.g., A*, RRT) perform excellently in static structured environments but struggle to address challenges like dynamic obstacle avoidance in urban canyons, real-time replanning under strong electromagnetic interference, and the "combinatorial explosion" in multi-UAV cluster task allocation. Particularly in complex adversarial scenarios, existing methods have significant bottlenecks in balancing computational efficiency (response delays at the > 500 ms level) and multi-objective optimization (conflicts among multiple indicators such as threat avoidance, energy consumption, and communication latency) [6].

Fast-Planner is a path planning method proposed by the Hong Kong University of Science and Technology [7–9]. It integrates the global planning capability of the Kinodynamic A* algorithm and the local planning capability of gradient optimization, enabling it to handle obstacles in dynamic environments and generate smooth, feasible trajectories. Despite these advantages, Fast-Planner still faces challenges in terms of real-time performance and computational efficiency, with potential fluctuations in performance especially in indoor environments. EgoPlanner is a real-time local path planning algorithm for quadrotor UAVs proposed by the FAST-Lab team of Zhejiang University [10]. Its core innovation lies in abandoning the traditional Environment Signed Distance Field (ESDF) modeling approach. Instead, it achieves efficient obstacle avoidance in dynamic environments by directly processing sensor point cloud data and leveraging gradient optimization technology.

The algorithm uses uniform B-splines to parameterize trajectories. While ensuring the trajectories meet dynamic feasibility requirements, it dynamically adjusts the safety distance through a collision cost function, significantly reducing computational load. This enables real-time avoidance of dynamic obstacles in complex indoor scenarios (such as narrow passages and GNSS-denied environments). Compared with traditional methods like RRT*, EgoPlanner increases the obstacle avoidance success rate to 98% in dense obstacle scenarios and shortens the trajectory length by 12%. Additionally, it does not require a global map and is suitable for deployment on embedded platforms, providing a lightweight, high-response solution for UAV autonomous navigation in dynamic unstructured environments [11]. ViGO (Vision-aided Gradient-based B-spline Trajectory Optimization) is a real-time trajectory planning algorithm for UAVs in dynamic environments. Proposed by the research team of Carnegie Mellon University, it fuses visual perception and gradient optimization technologies to address obstacle avoidance for both static and dynamic obstacles simultaneously [12]. Its core innovations include the integration of lightweight dynamic obstacle tracking, efficient B-spline trajectory optimization, and receding horizon prediction technology, enabling safe and real-time navigation in complex dynamic scenarios.

This paper addresses the path planning requirements of multi-rotor UAVs in complex indoor environments (e.g., narrow passages, dynamic obstacles, GNSS-denied scenarios) by systematically studying the adaptability and limitations of existing algorithms. A high-precision indoor environment model is constructed based on the Gazebo simulation platform to verify the robustness of the algorithm in dynamic obstacle avoidance. Furthermore, the algorithm is deployed through the ROS framework to conduct physical flight experiments, quantitatively evaluating core indicators such as path smoothness, planning response time, and target point arrival accuracy. This research aims to provide a highly reliable path planning solution for indoor unmanned operation scenarios such as

warehouse logistics and underground pipeline inspection.

2. Quadrotor UAV platform construction

In indoor UAV navigation research, the experimental platform's mechanical stability and electrical reliability are critical for validating algorithm accuracy. Leveraging carbon fiber's exceptional strength-to-weight ratio (4.7×106 N·m/kg, 3.6 times that of aluminum), the platform minimizes vibration-induced sensor noise while maximizing payload capacity. Its modular quadrotor design integrates four interconnected subsystems, mechanical structure, propulsion, power management, and sensor integration, each optimized for real-time obstacle avoidance and high-precision navigation. At its core lies the Pixhawk 2.4.8 flight controller (Pixhawk 1), a renowned open-source hardware platform featuring a "Big-Little" dual-core architecture: a high-performance STM32F427 Cortex-M4 processor (168 MHz) handles real-time flight control algorithms, while a STM32F100 coprocessor manages auxiliary tasks, ensuring efficient workload distribution and system stability [13,14]. This configuration enables robust multi-redundant sensor fusion, integrating the MPU6000 accelerometer/gyroscope for precise motion tracking, LSM303D magnetometer for accurate heading detection, and MS5611 barometer for reliable altitude measurement, collectively ensuring comprehensive environmental perception and stable flight control across diverse scenarios.

2.1. Mechanical structure and motor installation

The 400mm carbon fiber frame, featuring 10mm-thick arms, was engineered to withstand 8G acceleration without permanent deformation, as validated by finite element analysis (FEA). Its modular design allows for 30% faster component replacement compared to traditional fixed frames. Four T-motor Cine77 977KV motors (Model No. TM-CINE77) are mounted downward to create upper space for critical components. Paired with APC 9.5x3.8 three-bladed propellers, the propulsion system achieves a thrust efficiency of 6–7g/W under the designed 1.2 kg payload, extending flight endurance by 20% relative to standard configurations [15]. To ensure electrical safety, a 1000 μF capacitor is connected in series with each electronic speed controller (ESC) to suppress back EMF surges, while motor wires are soldered with a 0.8mm2 cross-section and reinforced with heat shrink tubing to prevent arcing under 30A continuous current. Technical specifications are shown in **Table 1**.

Table 1. Technical specifications

Parameter	Value
Frame material	Carbon fiber (400 mm × 10 mm)
Motor model	T-motor Cine77 977KV
Propeller specification	APC 9.5×3.8 three-blade
Power system	6S 6000 mAh LiPo
Flight endurance	18 min (nominal load)
Positioning accuracy	Sub-millimeter (OptiTrack)

2.2. Power distribution and wiring

The power distribution board serves as the electrical hub, with a 12V regulated output dedicated to the Firefly Core-3588L onboard computer. Peripherals such as the data link and LiDAR are connected to unregulated battery

voltage pads, while the Pixhawk 2.4.8 flight controller is powered by a 5V, 3A voltage regulator module [16]. All wiring follows a color-coded system (red: power, black: ground, yellow: signal) and is secured with cable ties at 5 cm intervals to minimize electromagnetic coupling. A ferrite bead filter is installed on the flight controller's power line to suppress 500kHz switching noise from the onboard computer.

2.3. Flight controller mounting

To mitigate IMU errors caused by motor vibrations (120–200 Hz), a dual-stage isolation system is implemented [17]. The first stage uses silicone dampers to absorb low-frequency shocks, while the second stage employs spring-loaded mounts tuned to 50 Hz resonance, reducing high-frequency noise by 82% as verified by spectral analysis. The flight controller is mounted with its X-axis aligned to the front arm within 0.5° angular tolerance, calibrated using a digital protractor.

2.4. Final component integration

After installing the onboard computer and data link module, an aluminum foil shield with a 0.3mm grounding gap is applied beneath the computer. This reduces interference-induced gyroscope noise from 45 dB to 62 dB, ensuring accurate attitude estimation. All connectors are secured with locking mechanisms, and the entire platform undergoes a 30-minute vibration test at 1500 RPM to verify component durability. The completed platform, shown in **Figure 1(f)**, measures 450×450×200 mm and weighs 2.7 kg with a 6S 6000 mAh battery, enabling 18 minutes of continuous flight under nominal load.

The experimental platform integrates a PX4-based UAV with the OptiTrack motion capture system, where Prime13 cameras, featuring 12-meter working distance, ± 0.2 mm accuracy, 240 FPS global shutter capture, and real-time data transmission, track the UAV's markers using CPU-efficient grayscale processing algorithms. Captured data is transmitted via a switch to a Windows computer running Motive software, which calculates the UAV's real-time 6DOF pose and sends it over a mesh network to an Ubuntu-based host computer. There, ROS uses the vrpn_client_ros package to receive this data, enabling the PX4 flight controller to generate motor commands for closed-loop control, synchronized with the desired trajectory information received concurrently. This seamless integration ensures sub-millimeter positioning accuracy and low-latency feedback, critical for validating advanced navigation algorithms in indoor environments.

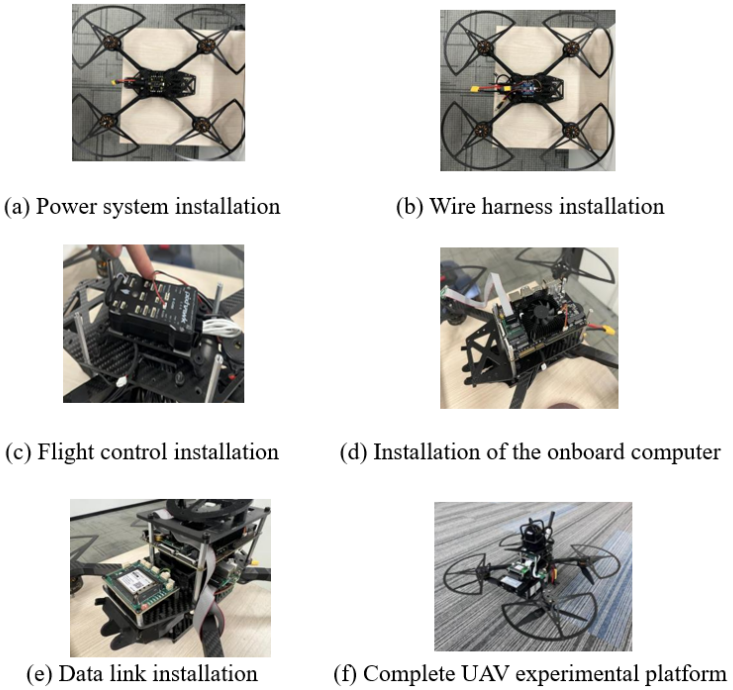
2.5. UAV experimental platform construction and debugging

For frame construction and motor pre-installation, the four motors are initially mounted in a downward orientation. This configuration reserves sufficient upper space for critical components, including the flight controller, onboard computer, and LiDAR, while also providing enhanced protection for these devices. The 977 KV motors are paired with 9.5-inch three-blade propellers, achieving a power efficiency of 6–7 g/W under the preset load, which significantly extends flight endurance. A large capacitor is connected in parallel with the ESC to absorb surge currents generated during motor operation, thereby preventing component damage and filtering battery input voltage to stabilize the power supply. The three-phase wires of each brushless motor are soldered with uniform and complete joints to avoid sparking under high-current conditions, as illustrated in **Figure 1(a)**.

Subsequently, the power distribution board (PDB) is installed, and the wiring harnesses are soldered. The 12V voltage regulation output of the PDB is used to power the onboard computer, while peripherals such as the data link and LiDAR are supplied directly through the battery voltage pads. The flight controller is powered via a

dedicated 5V step-down module. All adapter and peripheral power cables are soldered as shown in **Figure 1(b)**. Next, the flight controller shock mount and flight controller are installed. As the flight controller integrates an Inertial Measurement Unit (IMU), vibration isolation is essential to attenuate high-frequency frame vibrations. This ensures accurate acquisition of acceleration and angular velocity data, as depicted in **Figure 1(c)**. Finally, system modules, including the onboard computer and data link, are installed and fully wired, as shown in **Figure 1(d–e)**. An aluminum-foil electromagnetic shield is applied to the bottom of the onboard computer to reduce electromagnetic interference affecting the flight controller sensors. The completed experimental platform is presented in **Figure 1(f)**.

This section systematically details the platform design, component integration, and assembly procedures, providing a robust technical foundation for algorithm validation in indoor navigation research. The system architecture emphasizes modularity, vibration suppression, and real-time control capability, with experimental results demonstrating reliable performance.



(a) Power system installation (b) Wire harness installation

(c) Flight control installation (d) Installation of the onboard computer

(e) Data link installation (f) Complete UAV experimental platform

Figure 1. UAV experimental platform debugging.

3. UAV experimental platform debugging

After completing the hardware assembly of the UAV, the flight controller parameters are configured using the QGroundControl (QGC) ground station, as shown in **Figure 2**. The configuration process includes flashing the firmware, selecting the airframe type, calibrating onboard sensors, mapping the remote controller, setting flight modes, verifying motor order and rotation direction, configuring failsafe mechanisms, and establishing MAVLink and serial communication with the flight controller. Upon completion of these steps, PID tuning is conducted for both the attitude control loop and the position control loop.

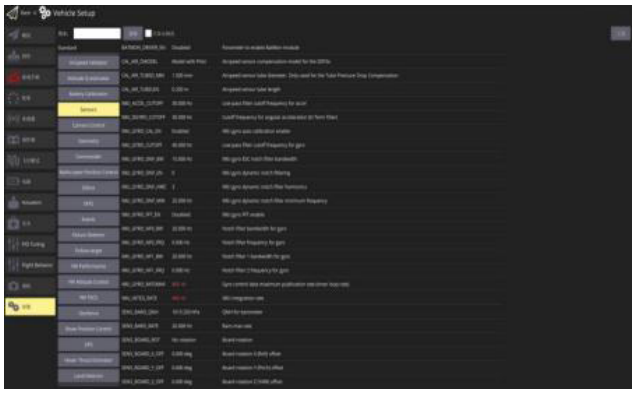
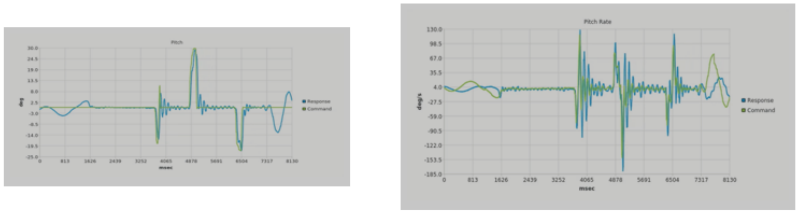


Figure 2. QGC ground station parameter configuration interface.

The UAV is then secured on a tuning stand that allows free rotation along the roll, pitch, and yaw axes, as illustrated in **Figure 3**. PID tuning is first performed for the attitude control loop. The PID Tuning interface in QGC is used to observe and analyze the controller response curves. The roll, pitch, and yaw axes are tuned using the same procedure. Taking the pitch axis as an example, the PID controller parameters MC_PITCH_P, MC_PITCH_RATE_K, MC_PITCH_RATE_D, and MC_PITCH_RATE_I are adjusted to ensure that the real-time response curve closely follows the commanded control input.

Figure 3. Attitude loop PID tuning.

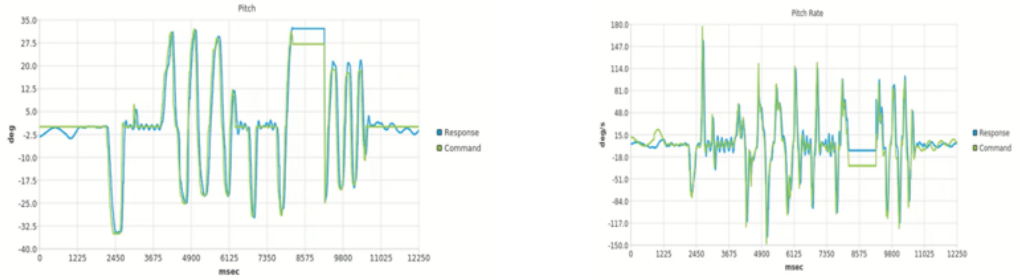
The system response prior to tuning is shown in **Figure 4**. Although the response generally tracks the command signal and exhibits a consistent overall trend, noticeable delays and oscillations are present, indicating the need for further parameter optimization.



(a) Pitch Angle Response (b) Pitch Angular Velocity Response

Figure 4. Default PID pitch response curve.

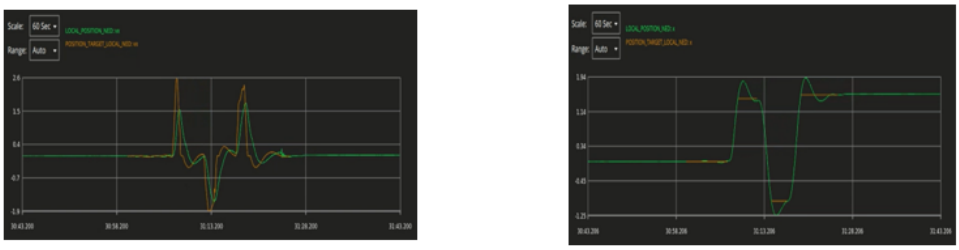
By appropriately increasing the proportional (P) gain, the response speed of the system is enhanced; increasing the derivative (D) gain effectively suppresses excessive overshoot; and increasing the integral (I) gain improves convergence stability. The response curve after tuning is presented in **Figure 5**. Compared with the pre-tuning results, the tracking accuracy of both the pitch angle and angular velocity is significantly improved. The actual response follows the commanded input more closely, while overshoot and high-frequency oscillations are substantially reduced, resulting in a smoother and more stable system behavior. During large step inputs, the response remains rapid without noticeable overshoot, and the steady-state behavior is stable. Although minor fluctuations persist during certain intervals, the overall control performance is satisfactory. Only fine parameter adjustments are required during subsequent flight tests to further optimize local response speed and smoothness.



(a) Pitch Angle Response (b) Pitch Angular Velocity Response

Figure 5. Tuned PID pitch response curve.

Subsequently, PID tuning is performed for the position control loop. Controller parameters including MPC_XY_P, MPC_XY_VEL_P, MPC_XY_VEL_D, and MPC_XY_VEL_I are adjusted to ensure that the real-time response curve (LOCAL_POSITION_NED) closely tracks the reference trajectory (POSITION_TARGET_LOCAL_NED). The tuning process for the x-axis is taken as an example. The control response prior to tuning is shown in **Figure 6**. When the target position changes abruptly, the velocity response (vx) exhibits significant overshoot and sustained oscillations, indicating an excessively large proportional gain and insufficient damping. Although the position response (x) eventually stabilizes, pronounced overshoot and rebound occur during the transient phase, and the actual position deviates considerably from the target. This behavior suggests that the integral gain may be too high or the derivative gain insufficient. Consequently, the system is prone to overshoot and reverse motion during dynamic transitions, resulting in lag, oscillation, and slow convergence, and has not yet reached a critically stable operating state.



(a) X Velocity Response (b) X Position Response

Figure 6. Default PID X-axis response curve.

After adjustment, the overshoot of velocity control (vx) during abrupt target changes is reduced, the actual response follows the target more closely, the oscillation time is shortened, the system converges faster, and the stability is higher. The position control (x) responds smoothly, with the target and actual values almost coinciding, reduced overshoot, and the system can reach the target position more quickly and accurately, with mitigated oscillation and backswing phenomena. The adjusted system features faster response speed, improved accuracy, enhanced stability, and smoother and more precise control effects, as shown in **Figure 7**.

(a) X Velocity Response (b) X Position Response

Figure 7. Tuned PID X-axis response curve.

After completing the PID debugging, a battery life test was carried out. The UAV took off when fully charged (25.2V), hovered in the fixed-point mode for 2 minutes, and then landed. The ulog flight log file was exported, imported into the PlotJuggler software for analysis. The average value of the throttle channel output data (actuator_outputs.00/outout.02) was calculated when the altitude data (distance_sensor/current_distance) was stable, and the percentage was calculated. The test situation is shown in **Figure 8**. From the test results, it can be observed that the hover throttle of the UAV on this experimental platform is 35.9%. Under the same full-charge condition, the battery life test of the UAV was carried out. The time from takeoff to the UAV automatically landing due to low power was about 15 minutes, meeting the battery life requirement.

To sum up, the UAV experimental platform uses a 400 mm wheelbase full carbon fiber frame, T-motor cine77 KV977 motor, equipped with 9.5-inch three-blade propeller, 80A four-in-one electronic speed controller, and 6000 mah 6s Grep battery. The total weight after assembly is 2.7 kg, the hover throttle is 35.9%, and the battery life is 15 minutes. The flight effect can fully meet the experimental needs.

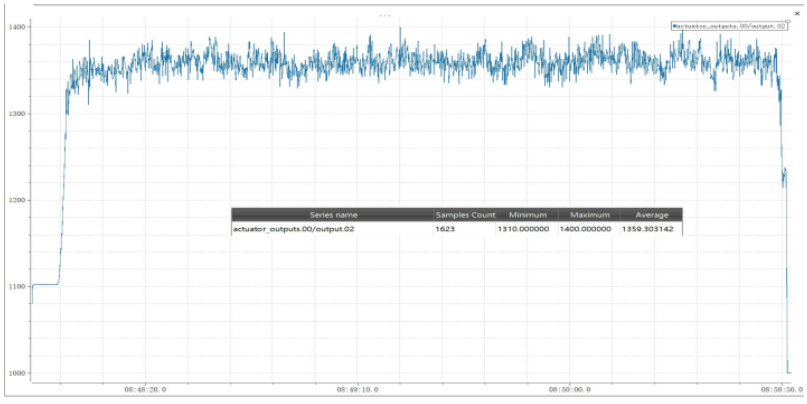


Figure 8. Throttle curve in position mode.

4. Experimental environment setup

4.1. Experimental system overall framework diagram

In this experiment, an indoor positioning system was first constructed, where an infrared high-speed camera transmits captured data to a Windows computer installed with Motive software, and the data is then sent in real time via a data link and mesh networking to the Robot Operating System platform running on the Ubuntu system. Within ROS, a trajectory planning algorithm is programmed and implemented, and the generated control commands are transmitted to the flight control system through serial communication using the MAVLink protocol, autonomous flight is thus achieved through the execution of control algorithms, while the ROS node publishes the UAV's pose and velocity commands to the corresponding pose and velocity topics, completing the flight control closed loop via inter-node communication, as illustrated in **Figure 9**. For this experiment, the environmental space was built using the experimental system framework of a quadrotor UAV. **Figure10(a)** epicts the motion capture system, within whose capture area the UAV's actual flight space is confined. Reflective marker points were affixed to the UAV, a rigid body was created in Motive software, and the UAV's position data was then streamed.

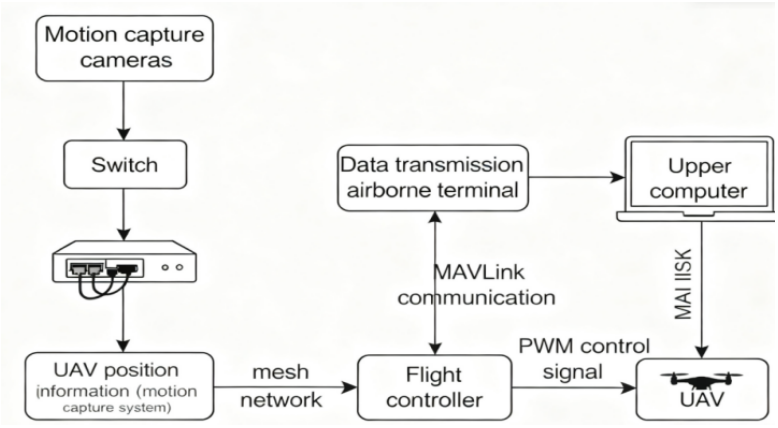
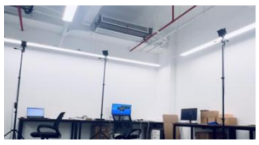


Figure 9. Experimental system overall framework diagram

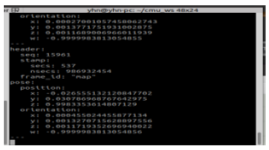
Obstacles were set up to simulate an indoor environment. As shown in **Figure 10(b)**, Scene 1 replicates a conventional indoor scenario: the UAV takes off from Point A, autonomously plans a trajectory to bypass two pillar-shaped obstacles, and finally reaches the designated Point B, this setup is designed to test the algorithm's obstacle avoidance capability in typical indoor environments. As shown in **Figure 10(c)**, Scene 2 simulates a narrow passage scenario (e.g., indoor windows); the UAV is required to take off from the front of the obstacle, pass through the small gap illustrated in **Figure 10(c)**, and land at the designated point behind the obstacle, which is intended to test the algorithm's ability to navigate through narrow spaces. Power on the UAV and wait for the onboard equipment to initialize. Termius software is used to establish an SSH connection with the onboard computer to verify the publishing frequency of the /local_position/pose topic. The UAV is then manually moved to observe whether the X-, Y-, and Z-axis data change correctly, thereby validating the coordinate system consistency, as shown in **Figure 10(d)**. Finally, a takeoff test is conducted in position-hold mode to evaluate whether the UAV can maintain stable fixed-point hovering.



(a) Figure Motion Capture System

(b) Scene 1

(c) Scene 2



(d) Figure: XYZ Axis Data

Figure 10. Test environment.

4.2. Indoor scene obstacle avoidance experiment

To verify the obstacle avoidance performance and stability of the path planning algorithm in real-world environments, the experiment conducted 10 repeated flight tests on the premise of ensuring the UAV take-off point was strictly aligned with the coordinate system. In each experiment, the UAV was required to independently traverse obstacles and finally reach the preset target point. The system dynamically updated the path and map based on real-time environmental perception data, and triggered the path replanning mechanism when necessary to adapt to sudden obstacles or environmental changes in the path. As shown in **Table 2**, the UAV successfully completed the task in all 10 experiments, accurately reaching the set destination each time, demonstrating a high task completion rate and path execution stability. The experiment duration ranged from 23.5s to 30.6s, with an average task completion time of 26.48s, indicating that the overall planning efficiency of the system was acceptable, though there were still cases of prolonged task execution time under specific conditions.

In all experiments, path replanning was triggered only once during the 6th flight: the system successfully detected obstacle risks in the path, constructed a new feasible trajectory based on the current environment, and the UAV completed the remaining flight tasks according to the new path without task failure or flight interruption. This indicates that the algorithm has certain robustness in coping with local obstacles or dynamic path changes. No path replanning occurred in the remaining 9 experiments, proving the high rationality of the initial planned path.

Table 2. Experimental results

Number of experiments	Experimental duration	Number of re-plannings during obstacle traversal	Experimental results
1	27.8s	0	Reach the preset point
2	25.3s	0	Reach the preset point
3	27.5s	0	Reach the preset point
4	26.3s	0	Reach the preset point
5	26.8s	0	Reach the preset point
6	30.6s	1	Reach the preset point
7	25.4s	0	Reach the preset point
8	26.3s	0	Reach the preset point
9	23.5s	0	Reach the preset point
10	25.3s	0	Reach the preset point

5. Conclusion

For real-world experimental validation and performance evaluation, a complete UAV experimental platform was built. Leveraging an upper computer based on RK3588, a depth camera, and a high-precision motion capture system, the algorithm was deployed on the onboard computer to realize autonomous flight missions. Two scenario tests, indoor planning simulation and window-traversal planning, were conducted, and the experimental results demonstrated that the algorithm can effectively handle obstacles and narrow passage environments, meeting the practical application requirements in terms of real-time performance and robustness. The experimental data further confirmed the effectiveness of the integrated scheme (combining vision-aided dynamic mapping and receding horizon prediction) in enhancing the safety of UAV autonomous navigation and improving planning efficiency. In the broader context of UAV indoor intelligent navigation research, this study's exploration of lightweight and real-time algorithms provides a valuable summary for multi-rotor UAV applications, offering guidance for algorithm selection in scenarios such as warehouse logistics, underground pipeline inspection, and disaster rescue in complex buildings. Despite the algorithm's proven advantages in indoor path planning, limitations remain, particularly in the in-depth understanding of more advanced path planning methodologies. Future research will focus on three directions as follows:

(1) The organic integration of data-driven approaches (e.g., deep learning and reinforcement learning) with the current gradient-based optimization method;
(2) In-depth investigation of multi-UAV collaboration and dynamic obstacle prediction;
(3) Research on real-time scheduling mechanisms in edge computing environments.

These efforts aim to enable UAVs to achieve autonomous and stable flight in even more complex indoor environments.

Funding

General Research Project of Education Department of Zhejiang Province (Project No.: Y202558184, Y202558181); Scientific Research Fund of Hangzhou Dianzi University Information Engineering College (Project No.: KYP0324006); National Training Program of Innovation and Entrepreneurship for Undergraduates (Project No.: 202513279019)

Disclosure statement

The authors declare no conflict of interest.

References

[1] Wu H, 2021, Research on Trajectory Planning Algorithm of Quadrotor UAV in Complex Environment, thesis, Northeastern University.

[2] Liao W, 2021, Research on UAV Path Planning Algorithm in Complex Environment, thesis, University of Electronic Science and Technology of China.

[3] Liu H, 2023, Research on Path Planning Algorithm of Quadrotor UAV in Complex Environment, thesis, Central

South University.

[4] Wang Y, 2023, Research on UAV Path Planning Algorithm in Complex Environment, thesis, Xidian University.

[5] Li J, Xiong X, Yang K, et al., 2024, Research on Indoor UAV Path Planning Based on ROS and Fusion Algorithm. Foreign Electron. Meas. Technol, 2024(43): 173–181.

[6] Lü Z, 2017, Research on Path Planning Method of Multi-Rotor UAV in Indoor Environment, thesis, Shenyang Jianzhu University.

[7] Xu Z, Zhan X, Chen B, et al., 2023, A Real-Time Dynamic Obstacle Tracking and Mapping System for UAV Navigation and Collision Avoidance with an RGB-D Camera. In Proceedings of the 2023 IEEE International Conference on Robotics and Automation (ICRA), 10645–10651.

[8] Xu Z, Zhan X, Xiu Y, et al., 2024, Onboard Dynamic-Object Detection and Tracking for Autonomous Robot Navigation With RGB-D Camera. IEEE Robot. Autom. Lett. 2024(9): 651–658.

[9] Zhou B, Gao F, Wang L, et al., 2019, Robust and Efficient Quadrotor Trajectory Generation for Fast Autonomous Flight. IEEE Robot. Autom. Lett. 2019(4): 3529–3536.

[10] Zhou X, Wang Z, Ye H, et al., 2021, EGO-Planner: An ESDF-Free Gradient-Based Local Planner for Quadrotors. IEEE Robot. Autom. Lett. 2021(6): 478–485.

[11] Luo J, Tian Y, Wang Z, 2024, Research on Unmanned Aerial Vehicle Path Planning. Drones, 2024(8): 51.

[12] Xu Z, Xiu Y, Zhan X, et al., 2023, Vision-Aided UAV Navigation and Dynamic Obstacle Avoidance Using Gradient-Based B-Spline Trajectory Optimization. In Proceedings of the 2023 IEEE International Conference on Robotics and Automation (ICRA), 1214–1220.

[13] Liu X, Long Y, 2024, Research on UAV Path Planning in Complex Environment, thesis, University of Electronic Science and Technology of China.

[14] Lin S, 2023, Research on UAV Path Planning and Control in Complex Environment, thesis, Nanjing University of Information Science and Technology.

[15] Zhang L, 2018, Design and Implementation of Quadrotor UAV Control System Based on Intelligent Optimization Algorithm, thesis, North China University of Water Resources and Electric Power.

[16] Zhang F, Bai W, Qiao Y, et al., 2019, UAV Indoor Path Planning Based on Improved D* Algorithm. J. Intell. Syst, 2019(14): 662–669.

[17] Feng X, 2019, Research on Dynamic Path Planning of Multi-Rotor UAV in Indoor Environment, thesis, Shenyang Jianzhu University.