Research on Autonomous Obstacle Avoidance for Quadrotor UAVs Based on Deep Learning

Qiyan Song
Engineering&Technical College, Chengdu University of Technology, Hebei, China

Abstract: To address single-sensor limitations and real-time constraints in quadrotor UAV obstacle avoidance, this paper proposes a closed-loop framework integrating cross-modal RGB-D/radar fusion, PPO-based dual-stream decision-making with online incremental learning, and differential-flatness trajectory generation. Under ideal conditions, detection accuracy reaches 95.4%, avoidance success rate 91.8%, and end-to-end latency 109 ms; under dark and fully occluded conditions, accuracy is 89.2% and avoidance success rate drops to 82.0%, indicating a performance boundary. These results validate the framework's potential in simulated unstructured dynamic environments while revealing areas for improvement under extreme perception degradation.

Keywords: UAV Autonomous Obstacle Avoidance; Deep Learning; Multi-Sensor Fusion; Trajectory Planning

1. Introduction

In recent years, with the advancement of science and technology, UAV technology has developed rapidly and has been widely applied in both military and civilian fields[1]. In this wave of rapid UAV development, various types of UAVs have emerged worldwide, including fixed-wing, single-rotor, and multi-rotor UAVs [2]. Currently, traditional target recognition and obstacle avoidance technologies for quadrotor UAVs mostly rely on GPS navigation, LiDAR, and similar methods. Although these approaches can achieve satisfactory obstacle avoidance in specific scenarios, they reveal obvious shortcomings and deficiencies under harsh environmental conditions[3].

LiDAR is expensive and affected by adverse weather, ultrasonic sensors have short range and multipath interference, and vision-based depth estimation suffers from feature matching issues and processing delays. Therefore, developing efficient, intelligent obstacle avoidance technology that is not reliant on a single sensor and can adapt to complex environments is essential for advancing autonomous drone navigation. Recent advances in deep learning, particularly convolutional neural networks, offer a data-driven approach to robust obstacle detection, classification, tracking, and trajectory prediction. This paper focuses on combining deep learning with multi-source information to address autonomous obstacle avoidance challenges, including multi-source heterogeneous sensing fusion, anti-interference real-time communication in complex electromagnetic environments, and reinforcement learning-based emergency obstacle avoidance decision trees, aiming to achieve an end-to-end millisecond-level response with over 93% obstacle recognition accuracy in simulation, which holds significant scientific and practical value for applications such as emergency rescue and urban logistics.

2. Related Technical and Theoretical Foundations

2.1 Quadrotor UAV Dynamics Model

In this paper,the Newton-Euler rigid body dynamics modeling method is adopted to establish the fundamental mathematical relationship between the control inputs and the motion state of the unmanned aerial vehicle.In the inertial coordinate system and the body coordinate system, the system state variables are defined as the position vector $P=[x,y,z]^T$,the Euler angle vector $\Theta=[\phi,\theta,\psi]^T$,the linear velocity vector $V=[\dot{x},\dot{y},\dot{z}]^T$,and the angular velocity vector $\omega=[\omega_x,\omega_y,\omega_z]^T$ .The control inputs are the total thrust U_1 and the three-axis torques U_2, U_3, U_4 in the body coordinate system.Under the rigid body assumption and neglecting high-order aerodynamic effects such as blade aerodynamic coupling and vortex interference, the standard dynamic model of the quadrotor UAV can be expressed as follows[4]:

$$\begin{cases} \ddot{x}=(\cos\phi\sin\theta\cos\psi+\sin\phi\sin\psi)\frac{U_1}{m} \\ \ddot{y}=(\cos\phi\sin\theta\sin\psi+\sin\phi\cos\psi)\frac{U_1}{m} \\ \ddot{z}=(\cos\phi\cos\theta)\frac{U_1}{m}-g \\ \ddot{\phi}=\frac{\dot{\theta}\dot{\psi}(I_y-I_z)}{I_x}+\frac{lU_2}{I_x} \\ \ddot{\theta}=\frac{\dot{\phi}\dot{\psi}(I_z-I_x)}{I_y}+\frac{lU_3}{I_y} \\ \ddot{\psi}=\frac{\dot{\phi}\dot{\theta}(I_x-I_y)}{I_z}+\frac{U_4}{I_z} \end{cases} \quad (1)$$

2.2 Theoretical Foundation of Deep Learning for Object Detection

Traditional drone control relies on precise modeling and manual parameter tuning, making it vulnerable to environmental uncertainty and nonlinear coupling, whereas deep reinforcement learning learns optimal policies directly from raw sensor data without requiring an environment model, offering a new approach for autonomous obstacle avoidance in high-dimensional, unstructured scenarios.

2.2.1 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) is currently the most widely used feature extraction method in the field of object detection. Its advantage lies in its ability to extract discriminative spatial features directly from raw images[5].

A CNN typically consists of convolutional layers, pooling layers, activation layers, and fully connected layers[5].

Convolutional Layer: The convolutional layer is the core module of a CNN. Its parameters consist of a set of learnable kernels. Each kernel performs convolution operations within the local region of the input feature map. This approach of local connectivity and weight sharing greatly reduces the number of parameters.

Pooling Layer: The pooling layer performs downsampling operations along the spatial dimensions of the input feature map, thereby reducing the number of parameters and the computational load of the network.

Activation Layer: The activation layer applies a nonlinear transformation to the output values using activation functions, introducing nonlinearity into the network. ReLU has become the most widely used activation function due to its ability to mitigate the vanishing gradient problem in the positive region, as well as its fast computation and rapid convergence.

Fully Connected Layer: After multiple convolutional and pooling operations, the final classification task is accomplished by a fully connected layer. The output feature dimension of the last fully connected layer equals the number of target classes, and the final prediction is obtained through a softmax function followed by an argmax operation[5].

2.2.2 YOLOv10

YOLOv10 is the latest model in the YOLO series specifically designed for real-time object detection tasks. It ensures high detection accuracy while improving inference efficiency and deployment complexity. Unlike traditional object detection methods, the YOLO series adopts a single-stage detection approach, integrating the object's location and classification into a regression problem, significantly reducing model inference latency and making it suitable for applications with high real-time requirements.

YOLOv10 introduces end-to-end detection, eliminating non-maximum suppression post-processing for improved inference efficiency.

2.3 Differential Flatness

Differential flatness is a concept in nonlinear control theory first introduced by French scholars Fliess, Lévine, Martin, and Rouchon in the 1990s. For a control system, if there exists a set of outputs such that all state variables and control variables can be expressed as functions of these outputs and their finite-order derivatives, the system is said to be differentially flat, and this set of outputs is called flat outputs[6].

Consider a nonlinear system:

$$x=f(x,u),x\in\Re^n,u\in\Re^m \quad (2)$$

If there exists an output:

$$z=\mathrm{h}\left(x,u,\dot{u},\ldots,u^{(r)}\right),z=\Re^m \quad (3)$$

such that:

$$\begin{aligned} x&=\phi_0\left(z,\dot{z},\ldots,z^{(r)}\right), \\ u&=\phi_1\left(z,\dot{z},\ldots,z^{(r+1)}\right) \end{aligned} \quad (4)$$

then the system is differentially flat, and z is a flat output[6].

Within the differential flatness framework, path planning is performed directly in the flat output space, where a continuous and smooth trajectory from the start to the target position is generated in 3D space. This trajectory must satisfy position, velocity, and acceleration continuity constraints to ensure flight smoothness and controllability.

During the actual implementation, the path is parameterized using polynomial trajectories. For

a single segment trajectory, it is expressed in the form of a fifth-order polynomial as

$$p(t)=a_0+a_1t^1+a_2t^2+a_3^3t+a_4t^4+a_5t^5 \quad (5)$$

When generating the trajectory, the dynamic constraints and safety constraints of the unmanned aerial vehicle should be taken into account, including the maximum speed, maximum acceleration, and maintaining the minimum safe distance from obstacles, etc. These constraints can be indirectly satisfied by limiting the flat output and its time derivative, thereby ensuring the feasibility and safety of the planned trajectory at the execution level.

3. Autonomous Obstacle Avoidance Method and Algorithm Design

3.1 System Architecture

The system output is composed of the raw data from various sensors. The RGB-D camera provides dark and color images, the millimeter-wave radar gives target detection results, and the IMU provides acceleration and angular velocity. After multi-modal fusion, a unified environmental representation is obtained. The obstacle avoidance decision-making module, upon receiving the environmental representation and target position information, will output the desired speed command. The motion control module, based on the desired speed and current state, uses path planning and trajectory tracking to finally obtain the motor control signal.

3.2 Environmental Sensing Module

To achieve robust perception in both normal and degraded visual environments, the proposed sensing module fuses data from an RGB-D camera and a millimeter-wave radar. These two sensors exhibit fundamentally heterogeneous characteristics: the camera provides dense, high-resolution point clouds but is susceptible to failure under low light, smoke, or dust; the radar, conversely, maintains reliable ranging in adverse conditions but outputs sparser measurements with inherent angular uncertainty. Directly combining these data sources at the point-cloud level risks discarding sensor-specific noise models and inflating computational overhead. To address this, we adopt a direct gridding method for multi-source heterogeneous fusion[7]. Each sensor's raw output is projected and mapped directly onto a unified occupancy grid space without generating an intermediate enhanced point cloud. Critically, an adaptive scale transformation is applied to the radar data: based on the grid resolution d_{res} and the radar's angular resolution, each radar point is symmetrically expanded by a dynamically computed factor n_{ext} to faithfully represent the underlying measurement uncertainty. This ensures that at close range the radar retains its precision, while at longer distances its sparse, uncertain points are appropriately dilated to achieve accurate alignment with the dense camera point cloud within the same grid framework.

Within this unified grid space, a probabilistic topographic map is constructed and updated using a ring buffer sliding strategy[7]. A fixed-size local map centered on the drone is maintained, with new data inserted along the flight direction and obsolete grids behind the drone discarded, stabilizing memory consumption within a constrained bound regardless of mission scale. Sensor fusion occurs directly at the probability update level: each grid's log-odds occupancy value is recursively updated using sensor-specific inverse models weighted by a confidence coefficient $\omega(n)$. When one sensor is degraded (e.g., camera in smoke), its weight can be reduced, allowing the other sensor's measurements to dominate the probability accumulation. Furthermore, a hysteresis mechanism with an occupancy coefficient J is introduced, requiring transient noisy points to persistently hit a grid multiple times before it is classified as occupied. This strategy effectively suppresses random interference—such as dust- or smoke-induced ghost points—while enabling true obstacles consistently detected by both sensors to quickly reach the occupancy threshold.

To handle the behavioral uncertainty of dynamic obstacles, a parallel dual-stream spatio-temporal modeling branch is introduced atop the static probabilistic topographic map. Inspired by the two-stream graph convolution network that decomposes traffic flow into stable and dynamic components[9], the spatial stream employs 3D sparse convolution to extract static geometric features constrained by physical structures (the stable component) from the current grid map, while the motion stream captures the dynamic evolution patterns of obstacle velocity and trajectory (the dynamic component) via temporal convolutional networks[8]; the two streams are fused through a deformable cross-modal attention mechanism that dynamically focuses

on high-risk regions and generates a spatio-temporal threat potential field, endowing the perception module with predictive foresight of future occupancy.

3.3 PPO Decision System

The risk-aware environmental tensor generated by the sensing module is projected into a local observation state space and serves as input to the PPO policy network. At time t ,the system state is defined as $S_t=\{E_t,p_t,v_t,p_{goal}\}$. where p_t and v_t denote the UAV's current position and velocity, and p_{goal} is the target position. The policy network outputs a continuous action distribution representing the desired velocity commands along the three axes. To alleviate the training difficulty caused by sparse positive samples in the continuous three-dimensional space, a heuristic velocity component V_h directed toward the target is superimposed onto the action output, with its magnitude adaptively adjusted based on the UAV's distance to the target[10][11]:

$$a_t = \pi_\theta(S_t) + V_h \tag{6}$$

This heuristic component addresses sparse reward challenges during initial training, accelerating convergence rather than altering the final learned policy structure.The reward function adopts a multi-objective weighted form:

$$\begin{aligned} R_t &= \lambda_1 R_{safe} + \lambda_2 R_{eff} + \lambda_3 R_{smooth} + \lambda_4 R_{goal} \\ &\begin{cases} R_{eff} = v \cdot \frac{p_{goal}-p}{\|p_{goal}-p\|} \\ R_{safe} = -\sum_i \max(0, d_{min} - d_i) \\ R_{smooth} = -\|\Delta v\|^2 \end{cases} \end{aligned} \tag{7}$$

R_{safe} behavior of punishing and approaching obstacles too closely, R_{eff} encourage movement in the desired direction, R_{smooth} prevent drastic changes in speed., R_{goal} reward for successfully reaching the target area.

According to the above framework, the policy parameters are optimized using the proximal policy optimization algorithm. The objective function is:

$$L_{PPO}(\theta)=\hat{E}_t\left[\min\left(r_t(\theta)\hat{A}_t, clip(r_t(\theta),1-\varepsilon,1+\varepsilon)\hat{A}_t\right)\right] \tag{8}$$

Where $r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{old}(a_t|s_t)}$ is the probability ratio of the new to old policy, and $\hat{A}_t$ s the advantage function. Training employs an asynchronous online incremental learning approach, where an independent experience replay pool stores high-priority failure samples, and a background thread fine-tunes the policy network parameters without blocking the main control loop. This design enables the UAV to reach the target via the shortest feasible path while avoiding obstacles.

4. Algorithm Design

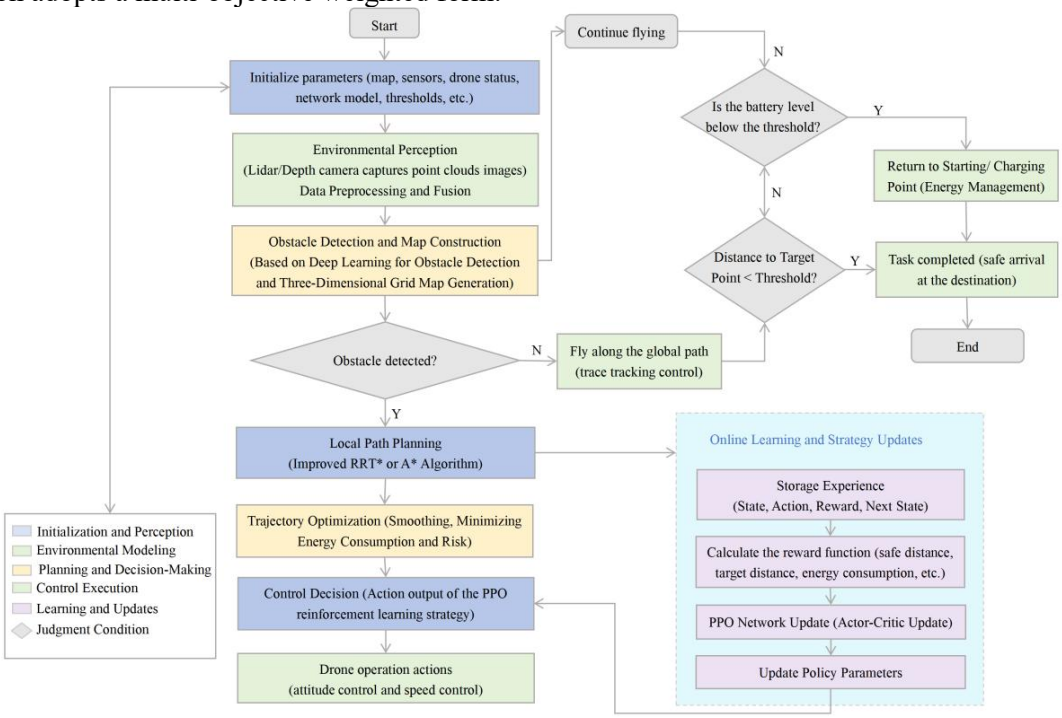

Figure 1. Algorithm Design Diagram

This paper proposes a closed-loop autonomous obstacle avoidance control framework that integrates multi-modal perception, deep reinforcement learning decision-making, and trajectory planning. Designed and simulated in a MATLAB/Simulink environment at a 100 Hz

control frequency, the algorithm loops through six sequential stages and is intended for resource-constrained quadrotor platforms. The design rationale follows three key technical principles suggested by recent literature: combining active ranging with rule-based strategies for simple low-cost collision prevention[1]; incorporating deep reinforcement learning to enable end-to-end decision-making and avoid reliance on prior maps[12]; And fusing complementary sensors to enhance environmental perception robustness under degraded visual conditions[2]. The overall workflow is described below.

(1) Multi-source alignment & motion compensation

The system first reads raw IMU, RGB-D and millimetre-wave radar streams. High-frequency IMU measurements are used in a pre-integration step to remove motion blur, after which the depth point cloud is registered with the previous frame via Iterative Closest Point to obtain a high-precision geometric representation.

(2) Cross-modal attention fusion

RGB-D features serve as queries while radar data provide keys and values in a cross-attention mechanism. This allows the vision stream to actively retrieve stable ranging information from the radar under poor lighting or smoke[2]. The fused result is encoded as a spatio-temporal environment tensor, which updates a local occupancy grid map maintained around the drone by a ring-buffer sliding strategy[7].

(3) PPO decision with safety constraint

The environment tensor and drone state (position, velocity, target location) form a local observation that is fed into a pre-trained Proximal Policy Optimization policy network. The network outputs a desired three-axis velocity command, which is then adjusted by a safety filter built on the risk gradient of the occupancy grid. The filter projects the action into a safe feasible domain, preventing exploration-induced collisions.

(4) Trajectory generation via differential flatness

Using the inherent differential flatness of quadrotor dynamics, a fifth-order polynomial parameterises the desired motion command into continuous feed-forward references for position, velocity, and acceleration. Kinematic constraints such as maximum velocity and minimum safe distance are satisfied by limiting the flat output coefficients.

(5)Asynchronous online incremental learning

When near-miss or collision-risk events are detected, the corresponding state–action–reward tuples are stored in a high-priority replay buffer. A background thread fine-tunes the PPO network parameters without blocking the main control loop, enabling adaptation to incrementally introduced scenarios. Dedicated sequential testing to verify resistance to catastrophic forgetting remains as future work.

5. Experimental Analysis

A MATLAB/Simulink simulation platform is established to evaluate the proposed method. The converged policy is deployed for online testing. The analysis covers experimental setup, training with ablation study, and comprehensive performance evaluation.

5.1 Experimental Setup

The simulation platform simulates the six-degree-of-freedom UAV dynamics model, RGB-D camera, and millimeter-wave radar heterogeneous sensor outputs in a data-driven manner, and constructs a comprehensive test scenario containing both static and dynamic obstacles. The experiment adopts a five-dimensional evaluation system: obstacle detection accuracy and obstacle avoidance success rate measure the environmental perception performance; safety distance and trajectory tracking error evaluate the motion control accuracy; and system real-time performance serves as the main constraint indicator. The test scenarios comprehensively cover various complex conditions, including sudden changes in light intensity, variations in obstacle movement speed, and sensor noise interference.

In the design of object detection and obstacle avoidance systems, multi-rotor UAVs are widely adopted due to their advantages of small size, flight flexibility, ease of control, strong flight stability, and low cost. However, this also means their payload capacity is limited and they are sensitive to power consumption. Consequently, the volume, weight, and power consumption of the image processing module deployed on a multi-rotor UAV are all constrained[13]. To accomplish accurate and fast object detection tasks within limited memory and computational capacity, in addition to algorithm optimization, network acceleration via model compression techniques is also required[13]. The main physical parameters of the quadrotor UAV used in the

simulation are shown in Table 1.

Table 1. Parameters of Quadcopter Unmanned Aerial Vehicle

Parameter symbol	Name	Value	Unit
m	Body mass	1.2	kg
l	Arm length	0.25	m
g	Gravity acceleration	9.81	m/s^2
I_{xx}	X-axis moment of inertia	0.014	$kg{\cdot}m^2$
I_{yy}	Y-axis moment of inertia	0.014	$kg{\cdot}m^2$
I_{zz}	Z-axis moment of inertia	0.024	$kg{\cdot}m^2$
J_m	Motor rotor mass	2.08e-5	$kg{\cdot}m^2$

5.2 Training Process and Strategy Convergence Analysis

The reward curve stabilizes after approximately 800 iterations, indicating convergence of the Proximal Policy Optimization (PPO) policy. Incremental learning triggered at later episodes (1200, 1800, 2400) causes brief fluctuations followed by rapid recovery, demonstrating the method's adaptability to environmental changes.

To quantify the contribution of each module, an ablation study was conducted. As shown in Figure 2, removing radar fusion, the dynamic obstacle stream, or the online learning component results in significant degradation of both detection accuracy and avoidance success rate. Having confirmed module effectiveness, the complete system is compared against an external baseline.

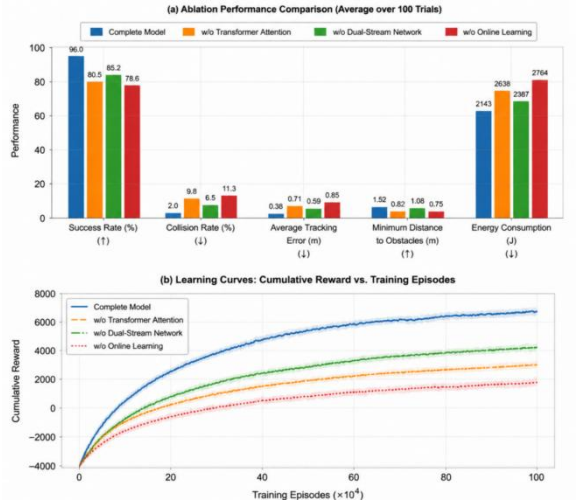

Figure 2. Ablation Study Showing the Contribution of Each Module

5.3 Experimental Results and Performance Analysis

5.3.1 Comparison test

The YOLOv10 baseline serves as an external reference but cannot isolate the effect of each proposed module. A separate ablation study (Section 5.2) decomposes internal contributions. As presented in Fig. 3, under bright conditions at 5 m/s, the proposed method achieves 95.4% detection accuracy and 91.8% avoidance success rate, outperforming the baseline, particularly under occlusion and varying illumination.

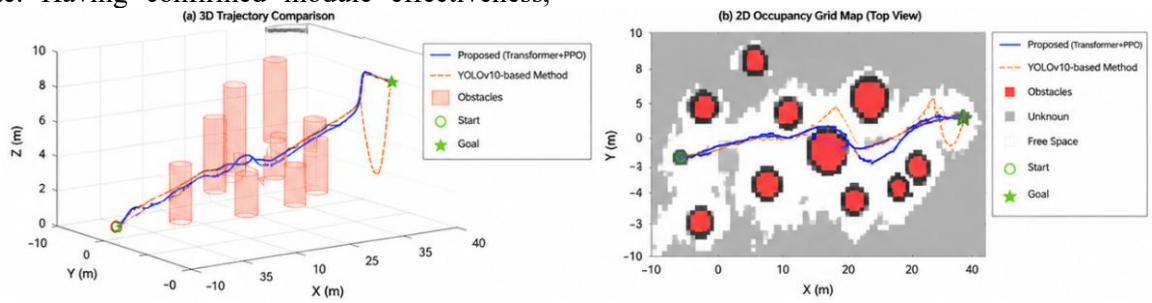

Figure 3. Comparison Experimental Results Chart with YOLOv10

5.3.2 Real-time performance, environmental robustness and failure analysis

In terms of dynamic response, once an obstacle is confirmed, a new control command is generated within 20ms(control stage). Speed tracking from 8m/s to 3m/s shows an overshoot of 2.1% and a steady-state error below 0.15 m/s. The end-to-end processing latency is 109ms, decomposed into perception (65ms), decision (24ms), and control (20ms) stages, satisfying the 10 Hz real-time requirement.To assess robustness, the system was tested under controlled variations in lighting, occlusion, and flight speed, as summarized in Table 2. Under ideal conditions (bright,no occlusion), detection accuracy reaches 95.4%, avoidance success rate 91.8%, and safe distance 3.13 m. As conditions degrade, performance declines gradually: in dark and fully occluded scenes, detection accuracy remains at 89.2% and avoidance success rate at 82.0%, confirming the compensating role of millimeter-wave radar. At a higher speed of 10 m/s, the safe distance adaptively increases to 3.36–3.58 m. Although detection accuracy

decreases from 93.4% to 90.8% in dark conditions at high speed, the avoidance success rate remains largely stable because the conservative safety strategy compensates for the perception degradation.

Table 2. Test Results of Algorithm Performance under Different Environmental Conditions

Number of Trials	Lighting Conditions	Obstruction Degree	Flight Speed (m/s)	Accuracy Rate	Obstacle Avoidance Success Rate	Safe Obstacle Avoidance Distance
1	Bright	No	5	95.4	91.8	3.13
2	Dark	No	5	92.3	88.7	3.02
3	Bright	Partially	5	92.8	90.0	2.92
4	Dark	Partially	5	90.9	84.0	2.98
5	Bright	Blocked	5	89.7	85.6	2.91
6	Dark	Blocked	5	89.2	82.0	3.08
7	Bright	No	10	93.4	90.6	3.36
8	Dark	No	10	90.8	87.1	3.58

Failure analysis reveals that most failures occur under severe perception degradation or when sudden obstacles appear at close range. Despite a 20 ms decision latency, if an obstacle emerges within 2 m at a relative speed exceeding 5 m/s, the platform's physical acceleration limit of 1.8 m/s² prevents successful avoidance within the available time. The failures explicitly delineate the simulation performance boundary. Future work should integrate actuator dynamics constraints into trajectory planning and enhance perception robustness under extreme conditions.

5.4 Summary

The proposed system achieves 95.4% detection accuracy and 91.8% avoidance success rate in ideal conditions, with an end-to-end latency of 109 ms meeting real-time requirements. Performance degrades gradually under adverse conditions, demonstrating the compensating capability of multimodal fusion and PPO-based decision-making, while revealing boundaries for further improvement in extreme perception loss and high-speed close-range obstacle scenarios.

6. Conclusions and Prospects

6.1 Conclusions

This paper achieves three main contributions:

1) In simulation, a Transformer-based cross-modal attention fusion mechanism combining RGB-D camera and millimeter-wave radar data achieved 95.4% detection accuracy under ideal conditions and 89.2% under dark and fully occluded conditions.

2) In simulation, a PPO-based decision system with dual-stream spatio-temporal encoding and online incremental learning achieved 91.8% avoidance success rate with an end-to-end latency of 109 ms.

3) In simulation, a trajectory generation method based on differential flatness produced smooth maneuvers with 2.1% overshoot and steady-state error below 0.15 m/s.

6.2 Future Work Outlook:

Future work will focus on bridging the sim-to-real gap through real-world flight tests, incorporating actuator dynamics constraints into planning to address close-range high-speed obstacle scenarios, and integrating battery state models to optimize the energy-time trade-off for extended endurance.

References

[1] B. Fan and G. Zeng, "Design of autonomous obstacle avoidance scheme for quadrotor UAV in cruise flight mode," Electronic Design Engineering, no. 2, pp. 180–184, 2020.

[2] P. Gao, "Binocular vision obstacle avoidance algorithm for multi-rotor UAV based on deep learning," D. thesis, National University of Defense Technology, 2017.

[3] P. Shang, "Object detection and obstacle avoidance for quadrotor UAV," D. thesis, Xi'an Shiyou University, 2025.

[4] Y. Yan, "Research on trajectory tracking control algorithm for quadrotor UAV," D. thesis, Shenyang Ligong University, 2024.

[5] Y. Liang, "Research on visual obstacle avoidance technology for quadrotor UAV based on convolutional neural network," D. thesis, Nanjing University of Aeronautics and Astronautics, 2023.

[6] X. Ma, "Trajectory planning for wheeled mobile robot on mobile platform based on differential flatness," D. thesis, Nanjing University of Information Science and Technology, 2018.

[7] X. Wang, K. Xia, and D. Song, “Multi-source heterogeneous sensor fusion mapping method for UAV,” Robot, 2024.
[8] J. Hu and Q. Chen, “Path planning optimization method for unmanned vehicle based on dual-stream spatio-temporal fusion network,” Intelligent IoT Technology, no. 4, pp. 73–77, 2025.
[9] Z. Li, L. Li, and X. Tao, “Dual-stream graph convolutional network for dynamic traffic flow prediction,” Journal of Computer Science and Exploration, no. 2, pp. 384–394, 2022.
[10] H. Wang, J. Huang, W. Wang, K. Zeng, N. Wang, and H. Hong, “Multi-UAV formation obstacle avoidance control method based on PPO algorithm,” Ordnance Industry Automation, no. 2, pp. 108–112, 2026.
[11] W. Zhang, W. Zhang, F. Song, and L. Long, “Monocular vision obstacle avoidance method for quadrotor UAV based on deep learning,” Journal of Computer Applications, vol. 39, no. 4, pp. 1001–1005, 2019.
[12] C. Zhu, “UAV obstacle avoidance algorithm based on deep reinforcement learning,” D. thesis, Dalian University of Technology, 2025.
[13] J. Li, “Research on object detection and autonomous obstacle avoidance for multi-rotor UAV,” D. thesis, Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, 2020.

