Theory and Practice of Science and Technology
2025, VOL. 6, NO. 7, 89-96
DOI: 10.47297/taposatWSP2633-456918.20250607

Design of the Control System for a Quadrotor UAV Based on Neural Network-PID Control

Jingjie Wang

Chengdu University, Chengdu, Sichuan, 610106, China

ABSTRACT
This study presents a neural network-enhanced flight control system for quadrotor UAVs, developing a BP neural network-based PID (BP-PID) controller. A comprehensive mathematical model of quadrotor aerodynamics is established, enabling the implementation of a complete control architecture in Simulink. Building upon a baseline classical PID controller, the design incorporates a BP neural network algorithm to dynamically adjust PID parameters during operation. Comparative Simulink simulations demonstrate the BP-PID controller's superior performance over classical PID in critical metrics: significantly improved responsiveness, enhanced disturbance rejection, and greater robustness across operational envelopes. These findings establish a foundation for developing advanced control algorithms with heightened sensitivity and stability for autonomous aerial systems.

KEYWORDS
Flight control system; PID control; BP neural network

1 Introduction

When ancient ancestors gazed upon the celestial expanse, they yearned to ride clouds and mist—to soar like eagles through layered heavens, surveying both the star-studded cosmic tapestry and the crisscrossed landscapes of mortal realms. As the spark of modern technology ignited humanity's quest to conquer the skies, transforming mythical aspirations into engineering blueprints, drones—"humanity's flying eyes"—have evolved from sci-fi fantasies to tangible reality. In the modern era, quadrotor UAVs play pivotal roles across civil recreational sectors, commercial cinematography, and military manufacturing industries, redefining humanity's relationship with not only three-dimensional space.

In disaster relief, drones provide key technical support with their agile aerial operations. They quickly reach hard-to-access areas, transmitting real-time data via cameras to show damage and trapped people's locations. During the Sichuan Luding earthquake, this cut rescue planning time to 15 minutes and boosted efficiency by 40%, gaining critical time in the 72-hour rescue window. Their six-axis mobility lets them fly freely over mountains and canyons, accessing places humans can't reach. Drones offer unique aerial cinematography perspectives. Their aerial shots provide stunning visuals, enhancing a film's drama. In The Wandering Earth, they vividly depicted Beijing's apocalyptic scenes. Drones combine mobility with focus and angle adjustments, creating dynamic visual narratives. This gives directors creative freedom to storytelling through movement and perspective.

Figure 1 (a) disaster relief (b) geological exploration (c) film shooting

With the increasing complexity of civil UAV application scenarios, traditional pilot operation models have struggled to meet the demands for safety, efficiency, and precision. Against this backdrop, the Civil Aviation Administration of China (CAAC) issued the industry standard Classification of Operational Levels for Distributed Operations of Civil Unmanned Aerial Vehicle Systems in 2022. For the first time, this standard systematically regulates the operational capabilities of UAV systems through a dual-dimensional framework focusing on level of automation and safety assurance for distributed operations. The flight control automation levels are categorized into six levels from AL-0 to AL-5, with the classification of flight control systems primarily based on automation degree and autonomous decision-making capabilities, covering operational scenarios ranging from fully manual to fully autonomous. This standard establishes a critical framework for the large-scale application and standardized management in the drone industry.

Table 1 The relationship between the level of automation and the degree of human intervention

Degree of automation	Level of automation	Degree of human intervention		
		CNS	AIS	System maintenance
——	AL-0	H	H	H
L	AL-1	M	H	H
M	AL-2	L	M	H
	AL-3	L	L	M
H	AL-4	L	L	L
	AL-5	L	L	L

Figure 2 The relationship between the level of automation and the elements of division

A robust, fast-response flight control system is therefore essential for enhancing UAV maneuverability and responsiveness. The neural network-augmented PID (NN-PID) control studied herein integrates BP neural networks with conventional PID control. Through data-driven training and iterative refinement, this hybrid architecture enables accelerated real-time adjustment of aircraft states. The main contributions of this paper are listed as follows:

1) In this paper, the mathematical formulas and models of the aerodynamics of quadrotor UAVs are applied in Simulink to build a complete UAV control system.

2) Drawing on the mathematical model of the quadrotor UAV, a classic PID control model is constructed, and a real-time BP neural network algorithm is integrated to boost the sensitivity and robustness of flight control.

3) We compared the classical PID control with the proposed BP neural network-based PID control, and the results demonstrate that the BP-PID control achieves better sensitivity and shorter settling time.

2 Quadrotor UAV dynamic model

This section focuses on the coordinate system modeling of a quadrotor UAV and its motion characteristics in three distinct dimensions.

2.1 Quadrotor UAV coordinate system modeling

For more accurate characterization of quadrotor flight dynamics, the body-fixed frame (BFF) is employed in this study for kinematic analysis.

The quadrotor UAV controls its flight attitude by adjusting the rotational speeds of its four rotors, which in turn alters the lift forces generated by each rotor. The aircraft's frame consists of two perpendicularly fixed supports, with the four rotors positioned at the ends of the frame. The quadrotor UAV designed in this paper features a cross-shaped configuration. Rotors 1 and 3 rotate counterclockwise, while Rotors 2 and 4 rotate clockwise.

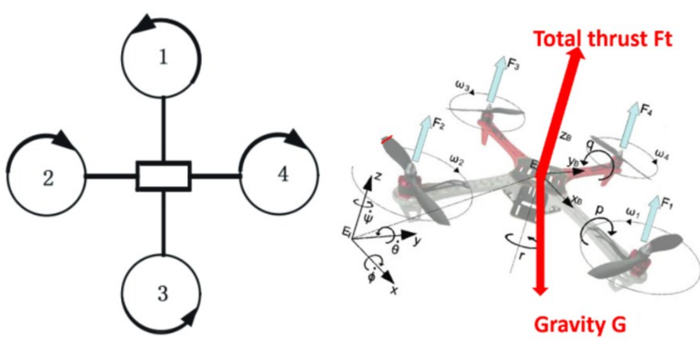

Figure 3 (a) UAV with a cross-shaped frame (b) the body-fixed frame

Figure (b) illustrates the mechanical model and the definition of coordinate systems for a quadcopter. Among them, Φ, θ, and ψ represent the roll angle, pitch angle, and yaw angle, respectively; p, q, and r correspond to their respective angular velocities. The thrusts generated by the four propellers are f_1, f_2, f_3, and f_4, and their rotational angular velocities are ω_1, ω_2, ω_3 and ω_4. The total thrust acting on the quadcopter is F_t, and the gravitational force it experiences is G.

2.2 Coordinate transformation matrix

In the flight dynamics modeling of a quadrotor UAV, the attitude representation between the body-fixed frame and the Earth-fixed (inertial) frame is defined by the Euler angles: roll angle ϕ, pitch angle θ, and yaw angle ψ. This attitude transformation is realized by the orthogonal direction cosine matrix R, which transforms vectors from the body frame to the inertial frame. The corresponding rotation matrices are as follows:

$$
\begin{aligned}
R(x,\phi)&=\begin{pmatrix}1 & 0 & 0\\ 0 & \cos\phi & -\sin\phi\\ 0 & \sin\phi & \cos\phi\end{pmatrix}\\
R(y,\theta)&=\begin{pmatrix}\cos\theta & 0 & \sin\theta\\ 0 & 1 & 0\\ -\sin\theta & 0 & \cos\theta\end{pmatrix}\\
R(z,\varphi)&=\begin{pmatrix}\cos\varphi & -\sin\varphi & 0\\ \sin\varphi & \cos\varphi & 0\\ 0 & 0 & 1\end{pmatrix}
\end{aligned}
\tag{1}
$$

Then the rotation matrix from the body coordinate system to the ground coordinate system is:

$$
R_B^E=\begin{bmatrix}\cos\varphi\cos\theta & \cos\varphi\sin\theta\sin\phi-\sin\varphi\cos\phi & \cos\varphi\sin\theta\cos\phi+\sin\varphi\sin\phi\\ \sin\varphi\cos\theta & \sin\varphi\sin\theta\sin\phi+\cos\varphi\cos\phi & \sin\varphi\sin\theta\cos\phi-\cos\varphi\sin\phi\\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi\end{bmatrix}
\tag{2}
$$

A quadrotor has the characteristic of four inputs and six outputs, where the six outputs include three position parameters and three attitude parameters. We suppose that $\xi=(x \quad y \quad z)^T$ stands for the position vector of the aircraft's center of gravity in the ground coordinate system, and $V=(u \ v \ \omega)^T$ represents the linear velocity vector of the aircraft in the ground coordinate system. Let $\eta=(\phi \ \theta \ \varphi)^T$ denote the attitude angle vector of the aircraft in the ground coordinate system, and $\Omega=(p \ q \ r)^T$ represents the angular velocity vector in the body coordinate system.

For a quadrotor, the angular velocity vector in the body-fixed reference frame and in the Earth-fixed reference frame are linked by a specific transformation, namely:

$$
\begin{bmatrix}\dot\phi\\ \dot\theta\\ \dot\varphi\end{bmatrix}=\begin{bmatrix}1 & \sin\phi\tan\theta & \cos\phi\tan\theta\\ 0 & \cos\phi & -\sin\phi\\ 0 & \dfrac{\sin\phi}{\cos\theta} & \dfrac{\cos\phi}{\cos\theta}\end{bmatrix}\begin{pmatrix}p\\ q\\ r\end{pmatrix}
\tag{3}
$$

2.3 Rigid-body dynamics model

Given the highly coupled dynamics, inherent nonlinearities, and underactuated nature of quadrotor UAV systems, the synthesis of a nonlinear robust control strategy is essential for achieving enhanced flight control performance beyond the limitations of conventional linear approaches. The following will establish the dynamic model of the quadrotor based on the Newton-Euler method:

$$
\begin{cases}\dot\xi=V\\ mV=R_B^E T^B+G^E\\ \dot\eta=W\Omega\\ J\Omega=-\Omega\times J\Omega+M^B+D\end{cases}
\tag{4}
$$

In this context:

$$
\begin{aligned}
&T^B=(0 \quad 0 \quad U_1)^T \\
&G^E=(0 \quad 0 \quad -g)^T \\
&W=\begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \dfrac{\sin\phi}{\cos\theta} & \dfrac{\cos\phi}{\cos\theta} \end{bmatrix} \\
&J=diag(J_{xx} \quad J_{yy} \quad J_{zz}) \\
&D=(d_\phi \quad d_\theta \quad d_\varphi)^T \\
&U_1=\sum_{i=1}^{4}F_i=C_T\sum_{i=1}^{4}\omega_i^2
\end{aligned} \tag{5}
$$

The quadrotor's mass is denoted by m. T^B represents the total aerodynamic thrust. G^E is the gravity vector. The rotation matrix W transforms coordinates from the body frame to the Earth frame. J is the vehicle's inertia matrix. M^B denotes the control torque input, while D represents external disturbance torque. The lift force of the i-th motor is F_i, with rotational speed w_i. The total thrust U_1 relates to motor speeds via the known thrust coefficient C_T. The rotor control torque:

$$
M^B=\begin{pmatrix} M_x^B \\ M_y^B \\ M_z^B \end{pmatrix}=\begin{pmatrix} U_\phi \\ U_\theta \\ U_\varphi \end{pmatrix}=\begin{bmatrix} \dfrac{\sqrt{2}}{2}l(F_4+F_3-F_2-F_1) \\ \dfrac{\sqrt{2}}{2}l(-F_4+F_3+F_2-F_1) \\ k_d(\omega_1^2+\omega_3^2-\omega_2^2-\omega_4^2) \end{bmatrix} \tag{6}
$$

In summary, the mapping relationship between the rotor control inputs and the torques can be derived. In the equation, the distance from the motor axis to the quadrotor's center of mass is denoted by l, and K_d represents the damping torque coefficient:

$$
\begin{bmatrix} U_1 \\ U_\phi \\ U_\theta \\ U_\varphi \end{bmatrix}=\begin{bmatrix} C_T & C_T & C_T & C_T \\ \dfrac{\sqrt{2}}{2}lC_T & \dfrac{\sqrt{2}}{2}lC_T & -\dfrac{\sqrt{2}}{2}lC_T & -\dfrac{\sqrt{2}}{2}lC_T \\ \dfrac{\sqrt{2}}{2}lC_T & -\dfrac{\sqrt{2}}{2}lC_T & -\dfrac{\sqrt{2}}{2}lC_T & \dfrac{\sqrt{2}}{2}lC_T \\ k_d & -k_d & k_d & -k_d \end{bmatrix}\begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} \tag{7}
$$

By summarizing the above equations, the dynamic equations of the quadrotor's position and attitude systems can be derived as follows:

$$
\begin{cases}
\ddot{x}=(\cos\phi\sin\theta\cos\varphi+\sin\phi\sin\varphi)\dfrac{U_1}{m} \\
\ddot{y}=(\cos\phi\sin\theta\sin\varphi-\sin\phi\cos\varphi)\dfrac{U_1}{m} \\
\ddot{z}=(\cos\phi\cos\theta)\dfrac{U_1}{m}-g \\
\ddot{\phi}=\dot{\theta}\dot{\varphi}\dfrac{J_{yy}-J_{zz}}{J_{xx}}+\dfrac{U_\phi}{J_{xx}}+\dfrac{d_\phi}{J_{xx}} \\
\ddot{\theta}=\dot{\phi}\dot{\varphi}\dfrac{J_{zz}-J_{xx}}{J_{yy}}+\dfrac{U_\theta}{J_{yy}}+\dfrac{d_\theta}{J_{yy}} \\
\ddot{\varphi}=\dot{\phi}\dot{\theta}\dfrac{J_{xx}-J_{yy}}{J_{zz}}+\dfrac{U_\varphi}{J_{zz}}+\dfrac{d_\varphi}{J_{zz}}
\end{cases} \tag{8}
$$

3 BP neural network PID control

BP Neural Network PID control is an intelligent control method that integrates traditional PID control with BP neural networks. Traditional PID controllers have a simple structure and strong robustness, but their parameter tuning relies on experience and they are difficult to adapt to complex nonlinear systems. By virtue of the self-learning and adaptive capabilities of neural networks, BP Neural Network PID control adjusts PID parameters online to optimize control performance. In quadrotor control, BP Neural Network PID can cope with strong coupling and nonlinear characteristics, enhance anti-interference ability and trajectory tracking accuracy, and is especially suitable for scenarios with load changes or external disturbances.

3.1 Analysis of PID control algorithm

As shown in Figure 4 below, the overall system control mainly includes two important modules: the classic PID control and the neural network algorithm.

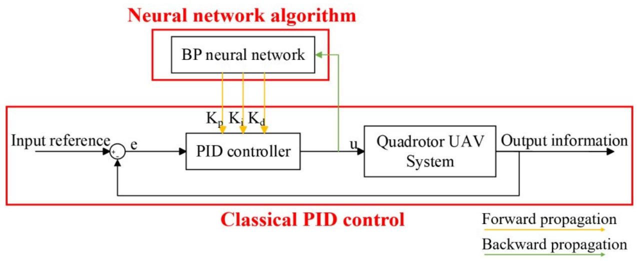

Figure 4 System control flow chart

As a classic feedback-based control method, PID control operates on a core principle that relies on the synergistic interplay of three elements: Proportional (P), Integral (I), and Derivative (D). These components dynamically adjust the control input based on the error (e) – the difference between the system output and the desired setpoint – to stabilize the system at the desired state. The Proportional component (P) is proportional to the current error. It responds rapidly to reduce the error but can cause system oscillation. The Integral component (I) acts on the accumulated value of the error. It eliminates steady-state error, enhancing the system's steady-state accuracy. The Derivative component (D) responds to the rate of change (derivative) of the error. It anticipates the trend of the error, curbs system overshoot, and improves response speed. The control outputs from these three components are combined through a weighted sum to form the final control output, working together to achieve precise system regulation.

The formula of the classic PID control is as follows:

$$u(t)=K_p(e(t)+\frac{1}{T_t}\int_0^t e(t)dt+T_D\frac{de(t)}{dt}) \tag{9}$$

Where e denotes the current error, and u represents the control output calculated by the controller based on the current parameters.

Since digital signals are processed by the computers we use, for the convenience of calculation, we rewrite the original control formula into an incremental digital control formula:

$$\Delta u(k)=K_p[e(k)-e(k-1)]+K_ie(k)+K_d[e(k)-2e(k-1)+e(k-2)]$$
$$K_i=K_p\frac{T}{T_i}K_d=K_p\frac{T_d}{T} \tag{10}$$

T is the sampling period, T_d is the derivative time constant, and T_i is the integral time constant.

3.2 BP neural network PID controller

A neural network performs computations by progressively transmitting and transforming data: raw input data first enters the input layer, then sequentially passes through one or more hidden layers, and finally reaches the output layer. At each layer, every neuron receives output values from all neurons in the previous layer, performs a weighted summation on these inputs—where each input is multiplied by its corresponding weight and then added to a bias term—and feeds this summed result into a nonlinear activation function for transformation (in this case, the Sigmoid function is used).

The transformed result becomes the output of that neuron and serves as input for the next layer. This process begins at the input layer, proceeds forward layer by layer, and ultimately generates the network's prediction at the output layer, thereby gradually converting the raw input into the final output.

3.2.1 BP neural network

Figure 5 illustrates the structure of the BP neural network and the way of data transmission, and its explanation is as follows:

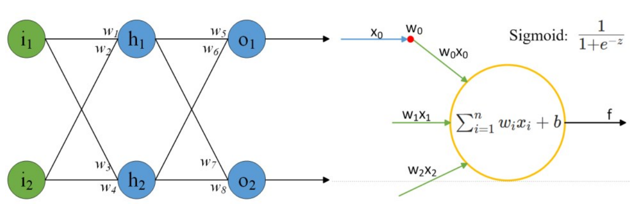

Figure 5 Neural network anatomy diagram

1) Foreword propagation:
Input layer to hidden layer:
Weighted sum of neuron inputs:

$$h_1 = i_1 \times w_1 + i_2 \times w_2 + b_1$$

Output oh_1 of neuron h_1: $oh_1 = \frac{1}{1+e^{-h_1}}$

Hidden layer to output layer:
Weighted sum of neuron inputs:

$$o_1 = oh_1 \times w_5 + oh_2 \times w_6 + b_2$$

Output oo_1 of neuron o_1: $oo_1 = \frac{1}{1+e^{-o_1}}$

2) Backward propagation:

Error calculation: $E_{total} = \sum \frac{1}{2}(t - oo)^2$

Error to output layer:
Backward gradient calculation:

$$\frac{\partial E_{total}}{\partial w_5} = \frac{\partial E_{total}}{\partial oo_1} \times \frac{\partial oo_1}{\partial o_1} \times \frac{\partial o_1}{\partial w_5}$$

Output layer to hidden layer:

Weight update: $w_{5^+} = w_5 - \eta \times \frac{\partial E_{total}}{\partial w_5}$

3.2.2 Principle of BP neural network PID control algorithm

Next, I will explain the overall algorithm principle of BP neural network PID control in sequence:

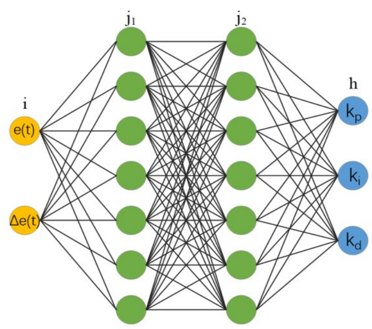

Figure 6 Neural network diagram

1) Random initial weights w_{ij1}, w_{ij2}, and h_{jk}.
2) Preset parameters via the Ziegler-Nichols method
3) Collect the output y(t) and calculate the error:
e(t) = r(t) - y(t)
4) Calculate the error change rate Δe(t) = e(t) - e(t-1)
5) Input [e(t), Δe(t)] into the neural network
6) The neural network performs forward propagation to compute $K_p(t)$, $K_i(t)$, $K_d(t)$
7) The PID controller calculates the control signal u(t) based on current parameters
8) Apply u(t) to the controlled object to obtain new output y(t+1)
9) Calculate the new error e(t+1)
10) Update neural network weights using gradient descent
11) Save current error e(t+1) for the next cycle's calculations

4 UAV simulation system design

The quadrotor flight control system employs a hierarchical architecture to establish a closed loop encompassing 'signal input → dynamic response → control decision'. Progressing from the underlying PWM drivers through noise simulation and control signal conversion, the system implements decoupled modeling via attitude dynamics, attitude kinematics, and translational dynamics. Utilizing full-state feedback of position, velocity, attitude, and angular rates, the controller adopts a cascaded strategy with inner-loop angular rate control for disturbance rejection, mid-loop attitude angle regulation for orientation control, and outer-loop position/velocity tracking. Integrated noise modules validate robustness against simulated disturbances, collectively demonstrating the synergistic interplay between hierarchical structure, cascade control, and engineering design principles throughout the overall design.

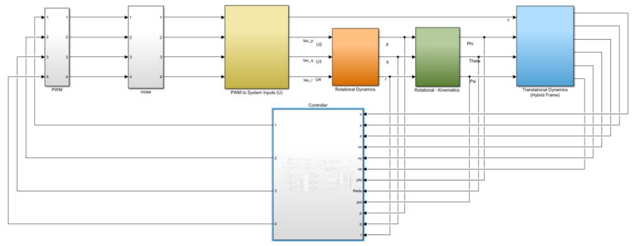

Figure 7 Simulation system design diagram

The quadrotor feedback control module establishes a closed-loop architecture for attitude and motion state regulation. Inputs encompass desired attitude commands along with actual position, velocity, and attitude angle signals. These undergo error comparison between desired and actual states, followed by processing through custom function modules and gain adjustments to output control quantities u_1~u_4. The module integrates switches and fixed values for rate controller testing while leveraging feedback to achieve coordinated attitude and motion control. Parameter optimization is simultaneously enabled via test interfaces.

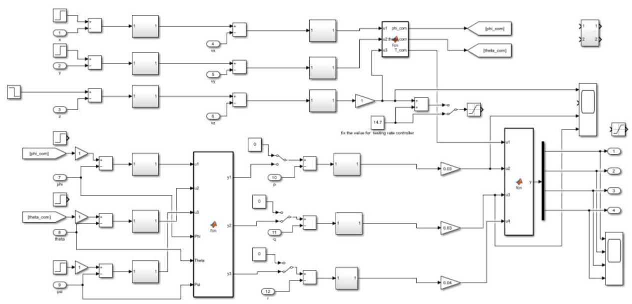

Figure 8 Feedback control module diagram

As shown in Figure 8, the tracking curves of classical PID control and BP neural network PID control for angle regulation are displayed on the oscilloscope. The results demonstrate that while the classical PID control exhibits slightly faster response speed than BPPID control, it suffers from significant overshoot and longer settling time. In contrast, the neural network PID not only achieves the desired value rapidly but also maintains superior dynamic performance without noticeable overshoot or oscillation.

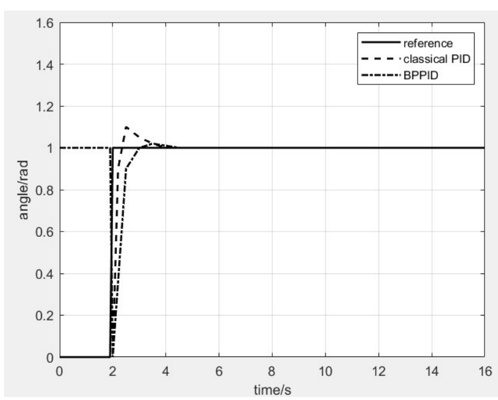

Figure 9 Error curves of different PID control methods

5 Conclusions

This research focuses on the design of flight control systems and proposes and implements a PID controller based on a BP neural network. By establishing a dynamic model of a quadrotor UAV, a classic PID control system framework is constructed, and then a neural network algorithm is introduced into the traditional control structure to adjust control parameters in real time. Simulation experiments based on Simulink verify the design hypothesis: compared with classic PID control, the neural network PID shows significant advantages in terms of dynamic response characteristics and robustness. The research results lay a technical foundation for the subsequent development of practical control algorithms with faster response and stronger robustness.

References

[1] Müller H I , Jr A S H J , Amico C S , et al. Leveraging genetic algorithms and neural networks for stiffness and weight optimisation of an unmanned aerial vehicle (UAV) support arm[J].Materials & Design,2025,256114134-114134.

[2] Wu D , Wei G , Song G , et al. Neural Network-Aided Multiple-Symbol Noncoherent Detection Scheme of LDPC Coded MPSK Receiver for Unmanned Aerial Vehicle Communications[J].International Journal of Pattern Recognition and Artificial Intelligence,2025,39(8).

[3] Abreo R O ,Aviles M ,Reséndiz R J , et al.Neural networks and genetic algorithms-based self-adjustment system for a backstepping controller of an unmanned aerial vehicle[J].Alexandria Engineering Journal,2025,12670-80.

[4] Zhao L ,Um D ,Nowka K , et al.Cotton yield prediction utilizing unmanned aerial vehicles (UAV) and Bayesian neural networks[J].Computers and Electronics in Agriculture,2024,226109415-109415.

[5] P. Lu, T. Sandy and J. Buchli, "Adaptive Unscented Kalman Filter-based Disturbance Rejection With Application to High Precision Hydraulic Robotic Control," 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 2019, pp. 4365-4372.

[6] Zhang Li. Design of the Control System for Quadrotor UAV Based on Cascade PID Control Algorithm[J]. Information Technology and Informatization, 2023, (1): 100-103.

[7] Guo Jie, Jin Hai, Shen Xinge. Optimized Control of Quadrotor UAV Based on Neural Network PID Algorithm[J]. Electronic Science and Technology, 2021, 34(10): 51-55.

[8] Wan Dongbao, Wang Huqi, Cong Peichao, et al. Fuzzy PID Control and Simulation Analysis of Lower Limb Exoskeleton Robot Based on Simulink[J]. Journal of Guangxi University of Science and Technology, 2023, 34(2): 91-99.