

International Journal of Applied Mathematics in Control Engineering

Journal homepage: <http://www.ijamce.com>

Robust Sliding Mode Flight Controller for Quad-Rotor UAV with Image Recognition

Xiaole Zhang^a, Jian Fu^{a,*}, Dong Wang^a, Ruoyin Xie^a^a School of Energy and Power Engineering, Nanjing University of Sciences and Technology, Nanjing, 210094, CHN

ARTICLE INFO

Article history:

Received 5 September 2021

Accepted 20 December 2021

Available online 22 December 2021

Keywords:

Sliding mode controller

Image recognition

Trajectory tracking

Quad-rotor UAV

Flight simulation

ABSTRACT

Based on sliding mode control and image recognition method, a kind of trajectory tracking control systems is proposed in this paper for quad-rotor UAV with parameter uncertainty and external interference. This control system is divided into four different control loops which are position loop, velocity loop, attitude loop and angular rate loop according to time scale separation principle. And Hough Transform method is also utilized to analyze the tracking error between the centerline of track and flight position. Then, autonomous trajectory tracking control system of quad-rotor UAV is designed here. The simulation results show the effectiveness of the designed sliding mode controller and image recognition algorithm.

Published by Y.X.Union. All rights reserved.

1. Introduction

In recent years, four-axis aircraft has attracted great attention from researchers in many countries (e.g., Louro P. et al., 2019; Tapan et al., 2018; Bao P., et al., 2019). The advantages of quad-rotor unmanned aerial vehicles (UAVs) include small size, simple mechanical structure and low cost compared with traditional aircraft. They have unique maneuverability features like hovering, take off and landing vertically (e.g., Mofid O et al., 2018; Ullah S et al., 2020). However, considerable uncertainty and external disturbances also exists in the quad-rotor UAV system (e.g., Chen, Q et al., 2018), due to the strong state, input aerodynamic coupled, under-actuated and high nonlinear, etc. Then, it is necessary to find a robust control system to complete specific flight missions.

Sliding mode control, which is a robust nonlinear control method, has been proved available for quad-rotor UAV flight control system (e.g., Adilet T et al., 2021; Sun Y X et al., 1980; Feng K P et al., 2018). It ensures the dynamic performance of the system by designing the sliding mode dynamic surface with strong robustness (Shi S et al., 2018). This strategy is also widely combined with other control methods such as data-driven control (e.g. Yuyang T et al., 2020), fault-tolerant control (e.g. Jing W et al., 2018.), backstepping sliding mode algorithm (e.g. Yang, Z., et al., 2018) to design a suitable flight control system for quad-rotor UAVs. Based on image recognition and sliding mode control method, this paper focus on the design process

of robust tracking flight controller in quad-rotor UAVs.

The general control structure in this paper is given as follows: Firstly, the initial image information taken by the sensors is sent to the image recognition modules. Then, quad-rotor UAV control system makes the decision of next flight action by comparing the information with previous one. This decision will be transformed to the control signals and send to the sliding mode control systems. And the rest of this paper is organized as follows: In Section II, the mathematical model of the quad-rotor UAV is provided. Section III shows the details of the designed control system. Image recognition and path tracking is carried out in Section IV. The simulation results are obtained in Section V. And Section VI concerns the concluding remarks of this paper.

2. Mathematical model

The quad-rotor UAV consists of a flight control system, four equal-length booms, propellers, and two counter-rotating rotors. It is maneuvered by changing the control inputs of four propeller engine systems (e.g. Jing D et al., 2021). The development of variable-complex motion equations for quadrotor has been extensively studied (e.g., Gaitan T A et al., 2013; Das A et al., 2009; Bouabdallah S et al., 2004). Table 1 summarizes the main characteristics of quad-rotor UAV used in this study, and simulation model of the quad-rotor UAV can be shown as Fig.1.

* Corresponding author.

E-mail addresses: fujian@njust.edu.cn (J. Fu)

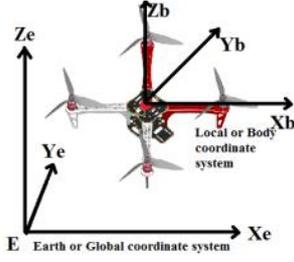


Fig.1. Cross configuration and motor orientation

Parameter	Value
Length	180 mm
Width	180 mm
Moment of Inertia	0.00005828, 0.00007169, 0.00010000
Length between the center of lift force and reference torque	0.044123

Table1 Main characteristics of quad-rotor UAV

The dynamical model (e.g. Wang C Y et al., 2021) of this system can be expressed as follow:

$$\begin{cases} \dot{x} = V_{ex} \\ \dot{y} = V_{ey} \\ \dot{z} = V_{ez} \end{cases} \quad (1)$$

$$\begin{cases} \dot{V}_{ex} = (\sin \psi \sin \phi + \cos \psi \sin \theta \cos \phi) \frac{U_1}{m} \\ \dot{V}_{ey} = (-\cos \psi \sin \phi + \sin \psi \sin \theta \cos \phi) \frac{U_1}{m} \\ \dot{V}_{ez} = (\cos \theta \cos \phi) \frac{U_1}{m} - g \end{cases} \quad (2)$$

$$\begin{cases} \dot{\phi} = p + (q \sin \phi + r \cos \phi) \cdot \tan \theta \\ \dot{\theta} = q \cos \phi - r \sin \phi \\ \dot{\psi} = \frac{q \sin \phi + r \cos \phi}{\cos \theta} \end{cases} \quad (3)$$

$$\begin{cases} \dot{p} = \frac{I_{yy} - I_{zz}}{I_{xx}} qr - \frac{J_{TP}}{I_{xx}} q\Omega + \frac{U_2 l}{I_{xx}} \\ \dot{q} = \frac{I_{zz} - I_{xx}}{I_{yy}} pr + \frac{J_{TP}}{I_{yy}} p\Omega + \frac{U_3 l}{I_{yy}} \\ \dot{r} = \frac{I_{xx} - I_{yy}}{I_{zz}} pq + \frac{U_4}{I_{zz}} \end{cases} \quad (4)$$

where x, y, z represent the positions along ground axis; ϕ, θ, ψ represent the Euler angles; V_{ex}, V_{ey}, V_{ez} are velocities along ground axis; p, q, r are angular rates; I_{xx}, I_{yy}, I_{zz} are the moments of inertia; m is the mass of the quadcopter; l represents the distance between UAV barycenter and rotary wing center; $U = [U_1, U_2, U_3, U_4]$ represents input control amount for each channel:

$$\begin{cases} U_1 = b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \\ U_2 = b(\Omega_4^2 - \Omega_2^2) \\ U_3 = b(\Omega_3^2 - \Omega_1^2) \\ U_4 = d(\Omega_2^2 + \Omega_4^2 - \Omega_1^2 - \Omega_3^2) \end{cases} \quad (5)$$

where b represents the lift coefficient; Ω_i represent the rotational

speed of the i the rotor; d represents the reverse torque coefficient.

3. Control system design

Consider a class of non-linear system as follow:

$$\dot{x} = f(x) + g(x) \cdot u \quad (6)$$

where $x = [x_1, \dots, x_n]^T \in R^n$ is system state, $u \in R^n$ is control input, invertible matrix $g(x) \in R^n$, and $f: D \rightarrow R^n$ is continuously differentiable mapping from $D \subset R^n$ to R^n .

The PI switching surface is designed as follow:

$$s(x_e) = x_e + k_1 \int x_e dt \quad (7)$$

where $s(x_e) = [s_1 \ s_2 \ s_3]^T$, $x_e = x - x_c$ is desired system state, coefficient $k_1 = \text{diag}\{k_{11}, \dots, k_{1m}\}$.

In order to reduce the chattering phenomenon in sliding mode control, saturation function is introduced to reaching law.

According to taking the derivative of Eq.(7), one can get

$$\dot{s}(x_e) = \dot{x}_e + k_1 \cdot x_e = -k_2 \cdot \tanh(s) \quad (8)$$

Lyapunov function is constructed as $V = \frac{1}{2} s^T s$, then it is noted that

$$\begin{aligned} \dot{V} &= s^T \dot{s} = s^T (-k_2 \cdot \tanh(s)) \\ &= -k_2 \cdot s^T \cdot \tanh(s) \\ &= -k_2 \cdot [s_1 \ s_2 \ s_3] \cdot \begin{bmatrix} \tanh(|s_1|) & 0 & 0 \\ 0 & \tanh(|s_2|) & 0 \\ 0 & 0 & \tanh(|s_3|) \end{bmatrix} \cdot \begin{bmatrix} \text{sgn}(s_1) \\ \text{sgn}(s_2) \\ \text{sgn}(s_3) \end{bmatrix} \\ &= -k_2 (s_1 \cdot \tanh(|s_1|) \cdot \text{sgn}(s_1) + s_2 \cdot \tanh(|s_2|) \cdot \text{sgn}(s_2) \\ &\quad + s_3 \cdot \tanh(|s_3|) \cdot \text{sgn}(s_3)) \\ &= -k_2 (|s_1| \cdot \tanh(|s_1|) + |s_2| \cdot \tanh(|s_2|) + |s_3| \cdot \tanh(|s_3|)) \leq 0 \end{aligned} \quad (9)$$

where hyperbolic tangent function

$$\tanh(s) = \frac{\exp(s) - \exp(-s)}{\exp(s) + \exp(-s)} \quad (10)$$

Basing on Lyapunov stability theory, the designed control systems are stable. The control systems have four independent control loops (e.g. Jian F et al., 2011) which are angular rate loop, velocity loop, attitude loop and position loop. And architecture of this flight system in this paper is shown as Fig.2.

3.1 Position Loop

Equations for this position loop are given in Eq.(1), and expressed as follow:

$$\dot{X}_p = f(X_p) + g(X_p) \cdot V \quad (11)$$

where position $X_p = [x; y; z]^T$ are given as states; velocity $V = [V_{ex}; V_{ey}; V_{ez}]^T$ are given as control inputs; $f(X_p) = [0; 0; 0]$ and $g(X_p) = \text{diag}\{1, 1, 1\}$; It means that the designed controller should send the reference velocity to next loop.

The tracking errors X_e between the desired outputs X_c and actual outputs X_c are defined as

$$X_e = X_p - X_c \quad (12)$$

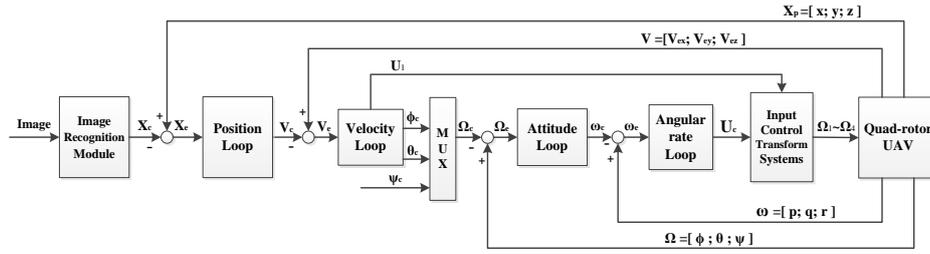


Fig.2. Control structure of quad-rotor UAV

It is noted that

$$\dot{X}_e = f(X_p) + g(X_p) \cdot V - \dot{X}_c \quad (13)$$

Then, the PI switching surfaces for the state X_p are given by:

$$s(X_p) = X_e + k_{p1} \int X_e dt \quad (14)$$

Hence, the control inputs V can be expressed as:

$$V = (g(X_p))^{-1} (-f(X_p) - k_{p1} \cdot X_e - k_{p2} \cdot \tanh(s) + \dot{X}_c) \quad (15)$$

where k_{p1} , k_{p2} are designed coefficients and $k_{p1} > 0$, $k_{p2} > 0$.

3.2 Velocity Loop

Equations for this loop are given in Eq.(2). In order to design a class of affine nonlinear controller, virtual control inputs are introduced as $\hat{u}_{v1} = \sin \theta U_1$, $\hat{u}_{v2} = \sin \theta \cos \phi U_1$ and $\hat{u}_{v3} = \cos \theta \cos \phi U_1$. The simplified velocity system is expressed as follow.

$$\begin{cases} \dot{V}_{ex} = \frac{(\hat{u}_{v1} \sin \psi + \hat{u}_{v2} \cos \psi)}{m} \\ \dot{V}_{ey} = \frac{(-\hat{u}_{v1} \cos \psi + \hat{u}_{v2} \sin \psi)}{m} \\ \dot{V}_{ez} = \frac{\hat{u}_{v3}}{m} - g \end{cases} \quad (16)$$

where

$$\theta = \arctan\left(\frac{\hat{u}_{v2}}{\hat{u}_{v3}}\right); \phi = \arctan\left(\frac{\hat{u}_{v1}}{1 + (\hat{u}_{v2})^2}\right); U_1 = \sqrt{\frac{\hat{u}_{v2}^2 + \hat{u}_{v3}^2}{2 \cos \phi}}$$

velocity $V = [V_{ex}; V_{ey}; V_{ez}]$ are given as states; $u_v = [\hat{u}_{v1}; \hat{u}_{v2}; \hat{u}_{v3}]$ are given as virtual control inputs; θ_c , ϕ_c are control inputs, and the combined form of control equations can be described as:

$$\dot{V} = f_v(V) + g_v(V) \cdot u_v \quad (17)$$

where

$$f_v(V) = \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} \quad g_v(V) = \begin{bmatrix} \frac{\sin \psi}{m} & \frac{\cos \psi}{m} & 0 \\ -\frac{\cos \psi}{m} & \frac{\sin \psi}{m} & 0 \\ 0 & 0 & \frac{1}{m} \end{bmatrix}$$

The tracking errors V_e between the desired outputs V and reference outputs V_c are defined as

$$V_e = V - V_c \quad (18)$$

It is noted that

$$\dot{V}_e = f_v(V) + g_v(V) \cdot u_v - \dot{V}_c \quad (19)$$

Then, the sliding mode surfaces used in this loop are expressed as follow:

$$s(V) = V_e + k_{v1} \int V_e dt \quad (20)$$

and the virtual control inputs u_v are given as:

$$u_v = (g_v(V))^{-1} [-f_v(V) - k_{v1} \cdot V_e - k_{v2} \cdot \tanh(s) + \dot{V}_c] \quad (21)$$

where designed coefficients $k_{v1} > 0$, $k_{v2} > 0$.

3.3 Attitude Loop

In this subsystem, $\Omega = [\phi; \theta; \psi]$ are given as states; angular rate $\omega = [p; q; r]$ are given as control inputs. Equations for this loop are given in Eq.(3), and can be also expressed as follow:

$$\dot{\Omega} = f_\Omega(\Omega) + g_\Omega(\Omega) \cdot \omega \quad (22)$$

where

$$f_\Omega(\Omega) = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad g_\Omega(\Omega) = \begin{bmatrix} 1 & \tan \theta \sin \phi & \tan \theta \cos \phi \\ 0 & \cos \phi & -\sin \phi \\ 0 & \frac{\sin \phi}{\cos \theta} & \frac{\cos \phi}{\cos \theta} \end{bmatrix}$$

The tracking errors Ω_e between the desired reference Ω and actual outputs Ω_c are defined as

$$\Omega_e = \Omega - \Omega_c \quad (23)$$

It is noted that

$$\dot{\Omega}_e = f_\Omega(\Omega) + g_\Omega(\Omega) \cdot \omega_c - \dot{\Omega}_c \quad (24)$$

Then, the PI sliding surfaces is expressed as follow:

$$s(\Omega_e) = \Omega_e + k_{\Omega 1} \int \Omega_e dt \quad (25)$$

The designed coefficients $k_{\Omega 1} > 0$, $k_{\Omega 2} > 0$ and the the control inputs ω are given as

$$\omega = g_\Omega(\Omega)^{-1} [-f_\Omega(\Omega) - k_{\Omega 1} \cdot \Omega_{e} - k_{\Omega 2} \cdot \tanh(s) + \dot{\Omega}_c] \quad (26)$$

3.4 Angular Rate Loop

In this subsystem, $\omega = [p; q; r]$ are given as states and $U_c = [U_2; U_3; U_4]$ are given as control inputs. Equations for this position

loop are given in Eq.(4), and expressed as follow:

$$\dot{\omega} = f_{\omega}(\omega) + g_{\omega}(\omega) \cdot U_c \quad (27)$$

where

$$f_{\omega}(\omega) = \begin{bmatrix} \frac{I_{YY} - I_{ZZ}}{I_{XX}} qr - \frac{J_{TP}}{I_{XX}} q\Omega \\ \frac{I_{ZZ} - I_{XX}}{I_{YY}} pr + \frac{J_{TP}}{I_{YY}} p\Omega \\ \frac{I_{XX} - I_{YY}}{I_{ZZ}} pq \end{bmatrix} \quad g_{\omega}(\omega) = \text{diag} \left\{ \frac{1}{I_{XX}}; \frac{1}{I_{YY}}; \frac{1}{I_{ZZ}} \right\}$$

The tracking errors ω_e between the desired outputs ω and actual outputs ω_c are defined as

$$\omega_e = \omega - \omega_c \quad (28)$$

It is noted that

$$\dot{\omega}_e = f_{\omega}(\omega) + g_{\omega}(\omega) \cdot U_c - \dot{\omega}_c \quad (29)$$

Then, the PI switching surfaces are given by:

$$s(\omega_e) = \omega_e + k_{\omega 1} \int \omega_e dt \quad (30)$$

Hence, the control inputs U_c can be expressed as:

$$U_c = g_{\omega}(\omega)^{-1} [-f_{\omega}(\omega) - k_{\omega 1} \cdot \omega_e - k_{\omega 2} \cdot \tanh(s) + \dot{\omega}_c] \quad (31)$$

where designed coefficients $k_{\omega 1} > 0, k_{\omega 2} > 0$.

4. Image recognition and path tracking

Simulated test track is generated and the quad-rotor UAV will tracing along the track centerline. The images taken by the bottom camera as shown in Fig.3 are converted to RGB color format, and three 120 * 160 matrices of red, green and blue are generated.

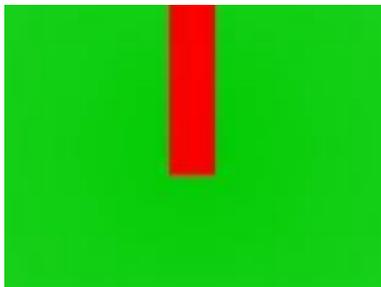


Fig. 3. Simulated track image captured by the bottom camera

Step1 Image Pre-processing

Due to the image obtained in the simulation environment is ideal, the pretreatment directly process the three matrices. The elements in the three matrices represent corresponding color element values constituting the camera image, and these values vary from 0 to 255.

Step2 Gray Processing and Binarization

Binarization is an image processing technology that separates the pixel values of a digital image into two groups: white and black. Image binarization is used in this paper to separate simulation track and ground as shown in Fig.4.

All the pixel matrices meet the requirements ($R > 200, G < 50, B < 50$) in the element matrix are extracted. These pixel matrices are combined to form a new image data matrix without changing the original size (e.g. Han H., 2014). If the value of the matrix element satisfies the proposed condition, then the new image data matrix element is 1, otherwise it is 0.



Fig.4. Image after gray processing and binarization

Step3 Hough Transform

According to the definition of Hough transformation, all points presenting a straight line in the original coordinate system, their slopes and intercepts are the same. The specific transformation process is as follows.

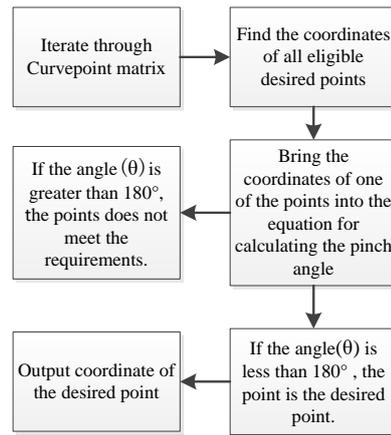


Fig.5. Coordinate transformation process

where θ is the angle between the correct expectation point and the intersection point at the previous moment.

In the image $x - y$ coordinate space, the straight line passing through the point (x_i, y_i) is expressed as:

$$y_i = ax_i + b \quad (32)$$

where the parameter a is the slope, b is the intercept moment.

If x_i and y_i are considered as constants and the original parameters a and b are considered as variables, Eq.(31) can be expressed as

$$b = -x_i a + y_i \quad (33)$$

Conversely, all lines that intersect at the same point in the parameter space have points that correspond to them in the image coordinate space. According to this property, given some edge points in the image coordinate space, the equation of the line connecting these points can be determined by the Hough transform.

In practical applications, there is no way to represent a straight-line equation of the form $y = kx + b$ for a line of the form $x = c$. In this paper, the parametric equation $p = x \cos \theta + y \sin \theta$ is used, so that a point on the image plane corresponds to a curve on the parametric $p - \theta$ plane. Input the binary image matrix into the internal functions, and then the image matrix containing only the center line of the track can be obtained. Fig.6 shows the straight path centerline identified by Hough transform.

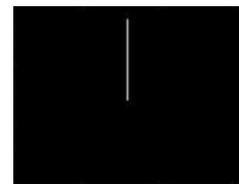


Fig.6. Centerline extraction using Hough transform

Mathematics in Control Engineering. 1(2),187-193.

Jing D., et al, 2021. Prescribed Performance Back-stepping Control of Fast Trajectory Tracking for Quad-rotor Aircraft PAN Shi-hua. Control Engineering of China. 28(11), 2119-2208.

Gaitan,T., et al, 2013. Modeling and Robust Attitude Control of a Quadrotor System. International Conference on Electrical Engineering, Computing Science and Automatic Control, IEEE. 10(9), 7-13.

Das,A., et al, 2009. Dynamic Inversion with Zero-Dynamics Stabilisation for Quadrotor Control. IET Control Theory & Applications. 3(3), 303-314.

Bouabdallah S., et al, 2004. Design and Control of an Indoor Micro Quadrotor. International Conference on Robotics & Automation, IEEE. 5(4), 4393-4398.

Wang,C,Y., et al, 2019. Design of Trajectory Tracking Control System for Quad-Rotor UAV. Electronics Optics & Control. 26(3), 103-107.

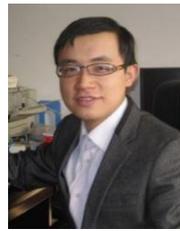
Jian,F., et al, 2011. Sliding mode control for a miniature helicopter. International Conference on Automation & Computing, IEEE. 17(9), 98-103.

Han,H., et al, 2014. Path Recognition Algorithm of Tracking Robots Based on Image Sensor. Journal of Donghua University (English Edition). 31(2), 137-140.

Huang,Z,Q., 2011. Design of Tracking Robots Section PID Control Algorithm Based on CCD Camera. Electronic Design Engineering. 19(2), 55-57.



Xiaole Zhang is currently pursuing her BS study at School of Energy and Power Engineering, Nanjing University of Sciences and Technology, Nanjing, China. Her research interest covers sliding mode control and flight control.



Jian Fu received his Ph.D. degree in control theory and control engineering from Nanjing University of Aeronautics and Astronautics. In the same year, he taught at Nanjing University of Science and Technology. Now, he is an associate professor at NJUST. He mainly engaged in nonlinear control, robust control, sliding mode control, adaptive observer design and so on.



Dong Wang is currently studying for his BS degree at the School of Energy and Power Engineering, Nanjing University of Science and Technology. His research interest covers aircraft control technology and Image recognition.



Ruoyin Xie is currently studying for his BS degree at the School of Energy and Power Engineering, Nanjing University of Science and Technology. She mainly engaged in dynamic system modeling and algorithm research.