



Image-based finite-time visual servoing of a quadrotor for tracking a moving target

Wei He · Liang Yuan

Received: 23 June 2022 / Accepted: 12 November 2022 / Published online: 30 November 2022
© The Author(s), under exclusive licence to Springer Nature B.V. 2022

Abstract This paper proposes an image-based visual servoing control method for a moving target of a quadrotor UAV (QUAV). Firstly, the dynamic image model with moving target parameters is established based on the image moment features in the virtual camera plane. For the unpredictability of the moving target in space, we use a high-order differentiator to estimate the state parameters of the moving target. In order to solve the problem of image depth information caused by a monocular camera, we derive a nonlinear finite-time linear velocity observer from the virtual image plane, which can not only estimate the linear velocity information of QUAV but also avoid the measurement of image depth. Based on the above information, we design the global finite-time controller and use Lyapunov theory to prove the finite-time stability of the system. Finally, the numerical simulations verify the convergence of the proposed control scheme, and the ROS

gazebo simulations demonstrate the improved performance of the proposed control scheme in tracking error.

Keywords Image-based visual servoing · Back-stepping · Finite-time observer · Global finite-time controller · Quadrotor UAV

1 Introduction

Quadrotor UAV (QUAV) has been widely used in autonomous detection [1,2], payload transportation [3], target tracking [4], and other missions because of its vertical takeoff and landing characteristics. In these missions, the QUAV needs to have the ability of autonomous flight, and the primary task to achieve the autonomous flight of the QUAV is to obtain their spatial position information. The traditional QUAV combines low-cost GPS (Global Positioning System) and low-cost IMU (inertial measurement unit) to obtain its position information. However, this positioning method will not provide reliable position feedback when the QUAV is in an indoor, low latitude, and complex urban environment [5]. The most common way to solve this problem is to combine it with visual technology. Under this scheme, QUAV usually carries a low-cost monocular camera. The rich visual information provided by the camera can help us achieve QUAV target tracking [6,7].

Using images to obtain the position and attitude information of the robot is called visual servoing [8].

Liang Yuan have contributed equally to this work.

W. He · L. Yuan (✉)
School of Mechanical Engineering, Xinjiang University,
No. 777 Huarui street, Shuimogou District, Wulumuqi
830046, Xinjiang, China
e-mail: yuanliang_1232021@163.com

W. He
e-mail: hewei442993802@stu.xju.edu.cn

L. Yuan
School of Information Science and Technology, Beijing University of Chemical Technology, 15 Beisanhuan Dong Lu, Chaoyang District, Beijing 100029, China

Visual servoing technology has two branches: position-based visual servoing (PBVS) and image-based visual servoing (IBVS). PBVS requires complete spatial data for 3D reconstruction, which requires high computational power and is unsuitable for small QAV. IBVS calculates the position and attitude information according to the selected image features. IBVS does not require the accuracy of the camera and has better robustness to calibration error and image noise [9], so it is suitable for small QAV. It should be pointed out that due to the underactuated characteristics of QAV, the dynamics and image of QAV are coupled, so the IBVS of QAV is more challenging than the full actuated system [10].

The earliest IBVS of QAV was started by Nicolas [9], who realized image-based visual servoing of static targets using the optical flow feature. Due to optical flow features, the coupling degree of the QAV visual servoing will be improved [11]. Researchers begin to try to use invariant features, such as image moments [12, 13]. Image moments contain many features, such as point features and line features. In addition, image moments are scale invariant. Taking advantage of this feature and introducing the concept of a virtual camera plane, Hamed and Giuseppe [14] eliminate the coupling of pitch angle and roll angle, reducing the complexity of controller design and designing an adaptive controller. Subsequently, image moment features in the virtual camera plane are widely used [15–17].

Because most QAVs use low-cost monocular cameras for image-based visual servo, it will lead to the inability to obtain image depth information accurately. To solve this problem, Hamed Jabbari and Jungwon estimated the linear velocity of QAV in the virtual camera plane [16]. This method uses an observer to compensate for the depth information of the image. After that, Zheng and Wang discussed the IBVS of QAV under slope conditions [17]. They designed a nonlinear velocity observer to estimate the linear velocity information of QAV. The linear velocity observer derived from the virtual plane can avoid acquiring depth information. Then they use the backstepping method to design the controller and prove the global asymptotic stability for the first time.

The above methods are designed for static targets. However, QAV has an urgent need for tracking moving targets. Due to the lack of accurate prior parameters of moving targets, it is more challenging to track unpredictable targets. S. Masoud and A. Abdollah designed

an indirect adaptive neural controller for QAV using a radial basis function (RBF) neural network to enable QAV to track moving targets [18]. After that, Masoud and Hamed Jabbari [19] used an artificial neural network to process the speed of moving targets and considered the influence of external interference. Ning [20] uses error symbol compensated robust integral (DCRISE) feedback control to design a visual servoing controller for moving targets, which improves the system's robustness. Zhiqiang and Xuchao [21] use a higher-order differentiator to estimate the motion parameters of the moving target and transfer the parameters as feedforward to the design stage of the controller. This method eliminates the coupling of the controller to the motion parameters. Moreover, it should be noted that the QAV often needs better system response capability when facing autonomous flight missions [22].

In the current research, the discussion on the convergence of the system controller is mainly asymptotic convergence. As time tends to infinity, the system state reaches the equilibrium point. Alexis [23] proposed a finite-time controller for position control, which realized the hover control of QAV in a finite time. Tian et al. [24] applied a multivariable hyper twist algorithm to realize the attitude control of QAV and proved the stability through theory and experiments. Harshavarthini [25] uses Lyapunov–Krasovskii functional to design a finite-time fault-tolerant controller for the attitude of the QAV. Gajbhiye [26] considered the finite-time control of the slung payload transportation of QAV and proposed a geometric finite-time inner–outer loop control strategy. Wenwu [27] uses the homogeneous theory to design the finite-time position controller and the finite-time attitude controller of QAV so that QAV can realize hovering control in a finite time. Guanglei and Guangbo [28] designed a finite-time controller using the nonsingular terminal sliding mode to realize the finite-time control of QAV image-based visual servoing. We use a table to summarize the above research work, as shown in Table 1. We consider these methods through three dimensions: the convergence rate of the controller, whether to consider the depth information of targets, and whether to discuss external disturbances.

We can see that no control scheme can simultaneously satisfy the requirements of tracking a moving target, finite-time convergence, considering the target's depth information, and verifying the disturbance. We

Table 1 Comparison of control methods

Methods	Convergence rate	Target depth information	Disturbance verification
Adaptive neural network [18]	Asymptotically stable	✓	✓
RBFNN [19]	Asymptotically stable	✓	✓
DCRISE [20]	Asymptotically stable	✓	✓
VO-backstepping [21]	Exponentially stable	×	✓
NFTSM* [28]	Finite-time stable	×	✓

*This method has not been verified on moving targets

propose a control scheme based on global finite-time stability to solve this problem. First, we use a nonlinear high-order tracking differentiator to estimate the parameters of moving objects. Using the backstepping method, we design a novel nonlinear global finite-time linear velocity observer for image depth information acquisition. Finally, we use the backstepping method to design the global finite-time controller of the system.

The main contributions of this paper are as follows:

- (1) We propose the global finite-time control scheme used for the image-based visual servoing of a moving target for the first time. In this scheme, we use a nonlinear tracking differentiator to estimate the parameters of the moving target. Then a global finite-time nonlinear linear velocity observer is designed using the backstepping method. Finally, we use the backstepping method to design the global finite-time controller of the QUAV.
- (2) To solve the problem of acquiring the image depth information of a monocular camera in the image-based visual servoing of a moving target, we propose a global finite-time nonlinear linear velocity observer. The depth information of the QUAV relative to the target is estimated by estimating the linear velocity of the altitude axis in the virtual camera plane.
- (3) The numerical simulation and ROS gazebo simulation results show high error convergence performance and better tracking control performance.

The rest of this paper is organized as follows. In Sect. 2, we introduce the QUAV and the image moment dynamics. In Sect. 3, we introduce the design of the controller. In Sect. 4, the simulation experiment of the proposed control scheme is carried out. Finally, our conclusions are given in Sect. 5.

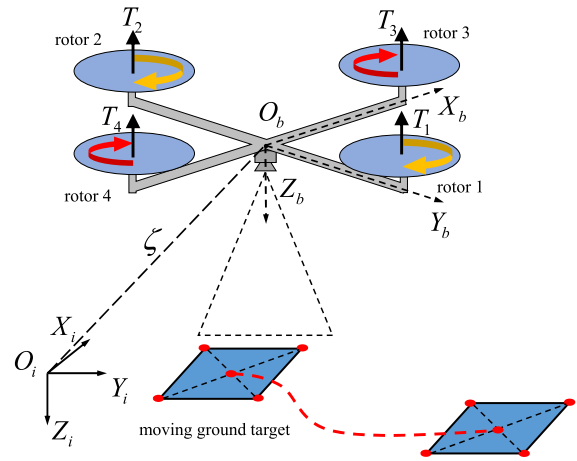


Fig. 1 Inertial frame I and body frame B

2 Modeling of QUAV and image moment

In this section, we will introduce the dynamic modeling of QUAV and image moment feature dynamics.

2.1 Quadrotor model

This section considers using two frames to describe the translation and rotation motion of the QUAV (see Fig. 1). The inertial frame $I = \{O_i, X_i, Y_i, Z_i\}$ is assumed to be fixed at a point on the earth. We assume that the mass center of the quadrotor UAV is the origin of the body-fixed frame $B = \{O_b, X_b, Y_b, Z_b\}$. Moreover, we assume that the quadrotor UAV is always a rigid body. The position of the origin of the body frame B in the inertial frame is expressed as $\zeta = (x, y, z)^T$. We use three Euler angles ϕ , θ , and ψ to represent the rotation between two frames. $\mathbf{R} : B \rightarrow I$ is the rotation matrix between two frames.

Assumption 1 The roll and pitch angles belong to $(-\frac{\pi}{2}, \frac{\pi}{2})$, and the yaw angle belongs to $(-\pi, \pi)$.

The mass of the QUAUV is m and the inertia is $\mathbf{J} = \text{diag}\{J_{xx}, J_{yy}, J_{zz}\}$. The linear velocity and angular velocity in the body-fixed frame of the QUAUV are expressed as $\mathbf{V} \in \mathbb{R}^3$ and $\omega = (\omega_1, \omega_2, \omega_3)^T \in \mathbb{R}^3$. According to [29–31], the kinematics and dynamics equations used to describe QUAUV with disturbance are expressed as follows:

$$\dot{\zeta} = \mathbf{R}\mathbf{V} \tag{1}$$

$$\dot{\mathbf{R}} = \mathbf{R}\mathbf{sk}(\omega) \tag{2}$$

$$m\dot{\mathbf{V}} = -m\omega \times \mathbf{V} + \mathbf{F} \tag{3}$$

$$\mathbf{J}\dot{\omega} = -\omega \times \mathbf{J}\omega + \tau \tag{4}$$

$$\mathbf{F} = -U_1\mathbf{E}_3 + mg\mathbf{R}^T\mathbf{e}_3 \tag{5}$$

where $\mathbf{sk}(\omega)$ is skew symmetric matrix, which means that $\mathbf{sk}(\mathbf{a})\mathbf{b} = \mathbf{a} \times \mathbf{b}$, $\mathbf{a} \in \mathbb{R}^3$ and $\mathbf{b} \in \mathbb{R}^3$ are arbitrary vectors, and \times denotes the vector cross product. g is the gravity acceleration. $\mathbf{F} \in \mathbb{R}^3$ is the force. $\tau \in \mathbb{R}^3$ is the torque. U_1 is the total thrust.

2.2 Image dynamics

This section also uses the image moment feature based on a virtual camera. First, we assume a camera frame $C = \{O_C, X_C, Y_C, Z_C\}$, which coincides with the body-fixed frame of the QUAUV. We assume that the camera is fixed at the mass center of the quadrotor and has a downward FOV. The virtual camera frame ν is the same as the actual camera frame. However, the roll angle and pitch angle of the virtual camera frame are always zero, and the yaw angle is consistent with the actual orientation of the quadrotor (see Fig. 2).

We assume that there is a fixed point ${}^I\mathbf{P} = [{}^Ix, {}^Iy, {}^Iz]^T$ in the inertia frame, and it is represented as ${}^C\mathbf{P} = [{}^Cx, {}^Cy, {}^Cz]^T$ in the camera frame and as ${}^\nu\mathbf{P} = [{}^\nu x, {}^\nu y, {}^\nu z]^T$ in the virtual camera frame. Therefore, we get

$${}^\nu\mathbf{P}(t) = \mathbf{R}_\psi^T(t) \left({}^I\mathbf{P} - O_\nu(t) \right) \tag{6}$$

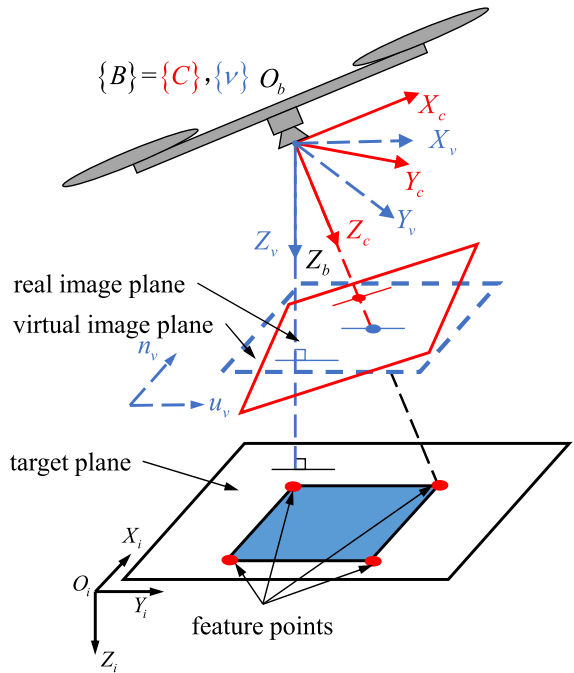


Fig. 2 Camera frame C and the virtual camera frame ν

where \mathbf{R}_ψ^T is a rotation matrix used to describe rotation around the Z -axis, ψ denotes yaw angle. Then, we get

$$\begin{aligned} {}^\nu\dot{\mathbf{P}} &= \frac{d\mathbf{R}_\psi^T}{dt} \left({}^I\mathbf{P} - O_\nu \right) - \mathbf{R}_\psi^T \frac{dO_\nu}{dt} + \mathbf{R}_\psi^T \frac{d}{dt} {}^I\mathbf{P} \\ &= -sk(\dot{\psi}e_3) \mathbf{R}_\psi^T \left({}^I\mathbf{P} - O_\nu \right) - \mathbf{R}_\psi^T \dot{O}_\nu + \mathbf{R}_\psi^T {}^I\dot{\mathbf{P}} \\ &= -sk(\dot{\psi}e_3) \mathbf{R}_\psi^T \left({}^I\mathbf{P} - O_\nu \right) - \mathbf{v} + \mathbf{d} \end{aligned} \tag{7}$$

where \dot{O}_ν denotes the linear velocities of camera and virtual frameworks in the inertia frame. $\mathbf{v} = [{}^\nu v_x, {}^\nu v_y, {}^\nu v_z]^T$ is the linear velocity of camera framework in the virtual frame, and $\mathbf{d}(t) = [v_x^d, v_y^d, v_z^d]^T$ is the velocity vector of a moving point in the virtual frame.

According to the perspective projection model, the projection of point \mathbf{P} onto the virtual camera plane can be expressed as

$$\begin{cases} {}^\nu u = \lambda \frac{{}^\nu x}{{}^\nu z} \\ {}^\nu n = \lambda \frac{{}^\nu y}{{}^\nu z} \end{cases} \tag{8}$$

where λ is focal length and $({}^v u, {}^v n)$ are the point coordinates in the virtual camera frame. From (7) and (8), one obtains

$$\begin{bmatrix} {}^v \dot{u} \\ {}^v \dot{n} \end{bmatrix} = \begin{bmatrix} -\frac{\lambda}{v_z} & 0 & \frac{v_u}{v_z} \\ 0 & -\frac{\lambda}{v_z} & \frac{v_n}{v_z} \end{bmatrix} \begin{bmatrix} {}^v v_x - v_x^d \\ {}^v v_y - v_y^d \\ {}^v v_z - v_z^d \end{bmatrix} + \begin{bmatrix} v_n \\ -v_u \end{bmatrix} \dot{\psi}. \tag{9}$$

Suppose that there are N stationary points in a level plane in the inertial frame, which are subject to the following assumptions.

Assumption 2 The observed target is a planar object that locates at a level plane of inertial frame and its binary image is obtained by segmentation algorithm.

Assumption 3 Image points are always in the field of view(FOV) of the camera.

Assumption 4 The sensor can get an accurate measurement value in the controller design stage.

Remark 1 Assumptions 3 and 4 are difficult to guarantee in the natural environment. These two assumptions are added here only to ensure that in the simulation experiment, the image features will always remain in the camera FOV, and each sensor can also accurately give feedback.

Then according to [13], we can get that the image moment features are as follows:

$$q_x = q_z \frac{{}^v u_g}{\lambda}, q_y = q_z \frac{{}^v n_g}{\lambda}, q_z = \sqrt{\frac{a^*}{a}} \tag{10}$$

where ${}^v u_g = \frac{1}{N} \sum_{k=1}^N {}^v u_k$ and ${}^v n_g = \frac{1}{N} \sum_{k=1}^N {}^v n_k$. ${}^v u_k$ and ${}^v n_k$ are the two components of the k th point. $a = {}^v \mu_{02} + {}^v \mu_{20}$ and ${}^v \mu_{ij} = \sum_{k=1}^N ({}^v u_k - {}^v u_g)^i ({}^v n_k - {}^v n_g)^j$. The desired value of a is a^* .

Based on (9) and (10), the dynamics of the image features are defined as follows [14]:

$$\dot{\mathbf{q}} = -\mathbf{sk} (\dot{\psi} \mathbf{e}_3) \mathbf{q} - \frac{1}{z^*} \mathbf{v} + \frac{1}{z^*} \mathbf{d} \tag{11}$$

where $\mathbf{q} = [q_x, q_y, q_z]^T$. z^* is the desired altitude.

In order to control the yaw motion of quadrotor UAV, we select the image feature q_ψ to describe the corre-

sponding motion according to [12], which is defined as follows:

$$q_\psi = \frac{1}{2} \arctan \left(\frac{2 {}^v \mu_{11}}{{}^v \mu_{20} - {}^v \mu_{02}} \right) \tag{12}$$

where the time derivative of q_ψ is $\dot{q}_\psi = -\dot{\psi} + \Delta_\psi$, and Δ_ψ is the equivalent of an undefined term that indicates the velocity of the target in the yaw direction.

3 Controller design

This section first gives the closed-loop error equation of the system and then gives the design of the target trajectory observer and the design of the linear velocity finite-time observer in the plane of the virtual camera. Finally, we give the design process of the global finite-time controller. Figure 3 shows the block diagram of the whole system.

The following lemmas are useful to derive our main results.

Lemma 1 [32] Consider the nonlinear system $\dot{x} = f(x)$, $f(0) = 0$, $x \in \mathbb{R}^n$, where $f(\cdot) : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is continuous function. Suppose that there exists a positive definite continuous function $V(x)$ such that

$$\dot{V}(x) + cV^\alpha(x) \leq 0,$$

where $c > 0$ and $\alpha \in (0, 1)$. Then, the system is finite-time stable. In addition, the finite convergence time $T(x)$ satisfies that

$$T(x) \leq \frac{V^{1-\alpha}(x(0))}{c(1-\alpha)}.$$

Lemma 2 [33] For any real numbers $x_i, i = 1, \dots, n$ and $b \in (0, 1]$ the following inequality holds:

$$(|x_1| + \dots + |x_n|)^b \leq |x_1|^b + \dots + |x_n|^b.$$

When $b = p/q \leq 1$, where $p > 0, q > 0$ are odd integers,

$$|x^b - y^b| \leq 2^{1-b} |x - y|^b.$$

Lemma 3 [34] For real variables x, y , and any positive constants a, b, c , the following inequality is true.

$$|x|^a |y|^b \leq \frac{a}{a+b} c |x|^{a+b} + \frac{b}{a+b} c^{-\frac{a}{b}} |y|^{a+b}.$$

3.1 Image feature error dynamics

We define the desired image features as follows:

$$\mathbf{q}_d = (q_x^d, q_y^d, q_z^d)^T = (0, 0, 1)^T, q_\psi^d = 0.$$

Therefore, the image moment feature errors of translational motion are as follows:

$$\mathbf{q}_1 = \mathbf{q} - (0, 0, 1)^T.$$

Taking the derivative of the above formula and using (11), we can obtain the following dynamics

$$\dot{\mathbf{q}}_1 = -\mathbf{sk}(\dot{\psi}\mathbf{e}_3)\mathbf{q}_1 - \frac{1}{z^*}\mathbf{v} + \frac{1}{z^*}\mathbf{d} \tag{13}$$

$$\dot{\mathbf{v}} = -\mathbf{sk}(\dot{\psi}\mathbf{e}_3)\mathbf{v} + \mathbf{f} \tag{14}$$

$$\dot{\mathbf{R}}_{\phi\theta} = \mathbf{R}_{\phi\theta}\mathbf{sk}(\omega) - \mathbf{sk}(\dot{\psi}\mathbf{e}_3)\mathbf{R}_{\phi\theta} \tag{15}$$

$$\mathbf{J}\dot{\omega} = -\omega \times \mathbf{J}\omega + \tau \tag{16}$$

where $\mathbf{f} = -\mathbf{R}_{\phi\theta}U_1\mathbf{E}_3/m + g\mathbf{e}_3 = [f_x, f_y, f_z]^T$, $\mathbf{R}_{\phi\theta} = \mathbf{R}_\theta\mathbf{R}_\phi$.

3.2 Trajectory observer

Before designing the controller, we need to estimate the trajectory parameters. The standard method uses a high-order differentiator, but the high-order differentiator is sensitive to noise. Therefore, we use the nonlinear tracking differentiator proposed by Han and Wang, proving its stability. The general form of the nonlinear tracking differentiator with the input $v(t)$ is as follows [35]:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dots \\ \dot{x}_{n-1} = x_n \\ \dot{x}_n = R^n f\left(x_1 - v(t), \frac{x_2}{R}, \dots, \frac{x_n}{R^{n-1}}\right) \end{cases} \tag{17}$$

where x_1 tracks the origin signal $v(t)$ and $x_{i+1}(i > 0)$ is the estimation of i th-order derivative. R is a coefficient to determine the rate of convergence for (17). The function $f(\cdot)$ is suggested by Han and Wang as follows [35]:

$$f(\cdot) = \text{fal}(x_1, \alpha_1, \Delta) + \text{fal}(x_2, \alpha_2, \Delta) + \dots + \text{fal}(x_n, \alpha_n, \Delta) \tag{18}$$

where

$$\text{fal}(x_i, \alpha_i, \Delta) = \begin{cases} |x_i|^{\alpha_i} \text{sign}(x_i), & |x_i| > \Delta \\ z/\Delta^{1-\alpha_i}, & |x_i| < \Delta, \Delta > 0 \end{cases} \tag{19}$$

α_i is a coefficient to reflect the degree of the nonlinearity, and $\alpha_i = 1$ is corresponding to the linear case. Δ is used to determine the linear interval, which can prevent vibration when the system is in the neighborhood of origin point.

For our system, the velocity of the target object is \mathbf{d} , its estimated value is $\hat{\mathbf{d}}$, and the estimation error is $\tilde{\mathbf{d}}$. Therefore, the following observer is designed as follows:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = x_3 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = R^4 f\left(x_1 - \tilde{d}, \frac{x_2}{R}, \frac{x_3}{R^2}, \frac{x_4}{R^3}\right) \end{cases} \tag{20}$$

where $x_1 = \hat{\mathbf{d}}, x_2 = \dot{\hat{\mathbf{d}}}, x_3 = \ddot{\hat{\mathbf{d}}}, x_4 = \dddot{\hat{\mathbf{d}}}$.

3.3 Finite-time velocity observer

In order to compensate for the depth information of the monocular camera, estimate the linear velocity in the virtual plane, and ensure the convergence performance of the observer, we design a finite-time linear velocity observer (FTO).

Theorem 1 The velocity observer and the corresponding update law are defined as follows:

$$\dot{\mathbf{q}}_1 = -\mathbf{sk}(\dot{\psi}\mathbf{e}_3)\mathbf{q}_1 - \frac{1}{z^*}\hat{\mathbf{v}} + \frac{1}{z^*}\mathbf{d} + k_1\tilde{\mathbf{q}}_1^{\frac{5}{7}} \tag{21}$$

$$\dot{\hat{\mathbf{v}}} = -\mathbf{sk}(\dot{\psi}\mathbf{e}_3)\hat{\mathbf{v}} + \mathbf{f} - \frac{1}{z^*}\tilde{\mathbf{q}}_1 + \frac{k_2}{z^*}\tilde{\mathbf{v}}^{\frac{5}{7}} \tag{22}$$

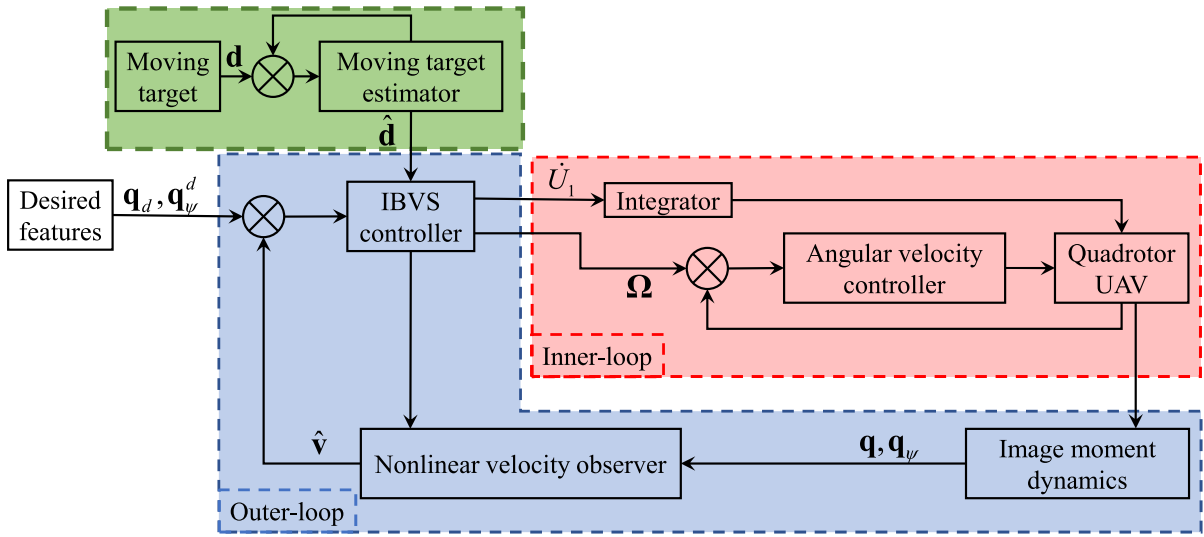


Fig. 3 Structure diagram of the system

where $\hat{\mathbf{q}}_1$ and $\hat{\mathbf{v}}$ are the estimated values of \mathbf{q}_1 and \mathbf{v} , respectively. k_1 and k_2 are positive constant. $\tilde{\mathbf{q}}_1^{\frac{5}{7}} = [\tilde{q}_{11}^{\frac{5}{7}}, \tilde{q}_{12}^{\frac{5}{7}}, \tilde{q}_{13}^{\frac{5}{7}}]^T$ and $\tilde{\mathbf{v}}^{\frac{5}{7}} = [\tilde{v}_1^{\frac{5}{7}}, \tilde{v}_2^{\frac{5}{7}}, \tilde{v}_3^{\frac{5}{7}}]^T$. $\tilde{\mathbf{q}}_1$ and $\tilde{\mathbf{v}}$ are the corresponding estimation errors, which define as follows:

$$\tilde{\mathbf{q}}_1 = \mathbf{q}_1 - \hat{\mathbf{q}}_1 \tag{23}$$

$$\tilde{\mathbf{v}} = \mathbf{v} - \hat{\mathbf{v}}. \tag{24}$$

This linear velocity observer is globally finite-time stable.

Proof We first take the time derivatives of (23) and (24) then substitute (13) and (14), respectively. We can get the following expression:

$$\dot{\tilde{\mathbf{q}}}_1 = \dot{\mathbf{q}}_1 - \dot{\hat{\mathbf{q}}}_1 = -\frac{1}{z^*} \tilde{\mathbf{v}} - k_1 \tilde{\mathbf{q}}_1^{\frac{5}{7}} \tag{25}$$

$$\dot{\tilde{\mathbf{v}}} = \dot{\mathbf{v}} - \dot{\hat{\mathbf{v}}} = -\mathbf{sk}(\dot{\psi} \mathbf{e}_3) \tilde{\mathbf{v}} + \frac{1}{z^*} \tilde{\mathbf{q}}_1 - \frac{k_2}{z^*} \tilde{\mathbf{v}}^{\frac{5}{7}}. \tag{26}$$

Now we choose a Lyapunov candidate function

$$L = \frac{1}{2} \tilde{\mathbf{q}}_1^T \tilde{\mathbf{q}}_1 + \frac{1}{2} \tilde{\mathbf{v}}^T \tilde{\mathbf{v}} = \frac{1}{2} \left(\sum_{i=1}^3 \tilde{q}_{1i}^2 + \sum_{i=1}^3 \tilde{v}_i^2 \right). \tag{27}$$

Taking the derivative of time for (27). Then substitute (21) and (22) into to (27), we can get the following:

$$\begin{aligned} \dot{L} &= -\tilde{\mathbf{q}}_1^T \frac{1}{z^*} \tilde{\mathbf{v}} - k_1 \sum_{i=1}^3 \tilde{q}_{1i}^r + \tilde{\mathbf{v}}^T \frac{1}{z^*} \tilde{\mathbf{q}}_1 - \frac{k_2}{z^*} \sum_{i=1}^3 \tilde{v}_i^r \\ &= -k_1 \sum_{i=1}^3 \tilde{q}_{1i}^r - \frac{k_2}{z^*} \sum_{i=1}^3 \tilde{v}_i^r \end{aligned} \tag{28}$$

where r is a design parameter. (28) is definite negative. Inspired by [17] and [33], we can select a set of design parameters as follows:

$$\beta_1 = 2 \min \left\{ k_1, \frac{k_2}{z^*} \right\} \tag{29}$$

$$n = 3 \tag{30}$$

$$r = \frac{4n}{2n+1} = \frac{12}{7} \tag{31}$$

$$\alpha = \frac{2n}{2n+1} = \frac{6}{7} \tag{32}$$

where $n = 3$ represents the order of the system. Then, we use the parameter (29) in (28), and we can get

$$\dot{L} = -k_1 \sum_{i=1}^3 \tilde{q}_{1i}^r - \frac{k_2}{z^*} \sum_{i=1}^3 \tilde{v}_i^r$$

$$\leq -\beta_1 \left(\sum_{i=1}^3 \tilde{q}_{1i}^r + \sum_{i=1}^3 \tilde{v}_i^r \right) \leq 0 \tag{33}$$

So far, we have been able to prove that the observer can be globally asymptotically stable (GAS). We continue to prove that it can be globally finite-time stable (GFTS). Scaling the (27), we can get

$$\begin{aligned} L &= \frac{1}{2} \left(\sum_{i=1}^3 \tilde{q}_{1i}^2 + \sum_{i=1}^3 \tilde{v}_i^2 \right) \\ &\leq 2 \left(\sum_{i=1}^3 \tilde{q}_{1i}^2 + \sum_{i=1}^3 \tilde{v}_i^2 \right). \end{aligned} \tag{34}$$

Combining (33) and (34), and using Lemma 2, we can get

$$\begin{aligned} L^\alpha &\leq 2^\alpha \left(\sum_{i=1}^3 \tilde{q}_{1i}^2 + \sum_{i=1}^3 \tilde{v}_i^2 \right)^\alpha \\ &\leq 2 \left(\sum_{i=1}^3 \tilde{q}_{1i}^2 + \sum_{i=1}^3 \tilde{v}_i^2 \right)^\alpha \leq 2 \left(\sum_{i=1}^3 \tilde{q}_{1i}^r + \sum_{i=1}^3 \tilde{v}_i^r \right) \end{aligned} \tag{35}$$

where $r = 2\alpha$. And then, we select a function as follows:

$$\begin{aligned} \dot{L} + \frac{\beta_1}{4} L^\alpha &\leq -\beta_1 \left(\sum_{i=1}^3 \tilde{q}_{1i}^r + \sum_{i=1}^3 \tilde{v}_i^r \right) \\ &\quad + \frac{\beta_1}{2} \left(\sum_{i=1}^3 \tilde{q}_{1i}^r + \sum_{i=1}^3 \tilde{v}_i^r \right) \\ &\leq -\frac{\beta_1}{2} \left(\sum_{i=1}^3 \tilde{q}_{1i}^r + \sum_{i=1}^3 \tilde{v}_i^r \right) \leq 0. \end{aligned} \tag{36}$$

The result means that $\dot{L} \leq -\frac{\beta_1}{4} L^\alpha$ holds. That is, we can find a Lyapunov function satisfying Lemma 1. So far, the proposed velocity observer is globally finite-time stable (GFTS). \square

3.4 Finite-time controller

We mainly use the backstepping scheme to design the IBVS controller of QUAV, and we need to use the proposed velocity observer.

We define the first Lyapunov function:

$$V_1 = \frac{1}{2} \mathbf{q}_1^T \mathbf{q}_1.$$

By substituting (13), we can get the time derivative of the Lyapunov function

$$\begin{aligned} \dot{V}_1 &= \mathbf{q}_1^T \left(-\mathbf{sk}(\dot{\psi} \mathbf{e}_3) \mathbf{q}_1 - \frac{1}{z^*} \mathbf{v} + \frac{1}{z^*} \mathbf{d} \right) \\ &= -\mathbf{q}_1^T \frac{1}{z^*} (\tilde{\mathbf{v}} + \hat{\mathbf{v}} + \mathbf{d}). \end{aligned} \tag{37}$$

We treat $\hat{\mathbf{v}}$ as a virtual control input, and choose $\hat{\mathbf{v}}^d = c_1 \mathbf{q}_1^{\frac{5}{7}} + \mathbf{d}$, $c_1 > 0$, $\mathbf{q}_1^{\frac{5}{7}} = [q_{11}^{\frac{5}{7}}, q_{12}^{\frac{5}{7}}, q_{13}^{\frac{5}{7}}]^T$. According to the standard backstepping scheme, it is necessary to continue to define new error terms \mathbf{q}_2

$$\mathbf{q}_2 = \mathbf{q}_1^{\frac{5}{7}} - \frac{1}{c_1} \hat{\mathbf{v}} + \frac{1}{c_1} \mathbf{d}. \tag{38}$$

Update \dot{V}_1 in (37) using (38)

$$\dot{V}_1 = -\mathbf{q}_1^T \frac{1}{z^*} \tilde{\mathbf{v}} - \frac{c_1}{z^*} \sum_{i=1}^3 q_{1i}^r + \mathbf{q}_1^T \frac{c_1}{z^*} \mathbf{q}_2. \tag{39}$$

Then we define the second Lyapunov function

$$V_2 = V_1 + \frac{1}{2} \mathbf{q}_2^T \mathbf{q}_2 + L. \tag{40}$$

Since we need to get the time derivative of (40), we need to get the time derivative of \mathbf{q}_2 first. Taking the time derivative of (38), then substituting (13) and (22) into it, we can get

$$\begin{aligned} \dot{\mathbf{q}}_2 &= -\frac{5}{7} \mathbf{diag} \left(\mathbf{q}_1^{-\frac{2}{7}} \right) \mathbf{sk}(\dot{\psi} \mathbf{e}_3) \mathbf{q}_1 + \frac{1}{c_1} \dot{\mathbf{d}} \\ &\quad - \frac{5}{7z^*} \mathbf{diag} \left(\mathbf{q}_1^{-\frac{2}{7}} \right) \mathbf{v} - \frac{k_2}{c_1 z^*} \tilde{\mathbf{v}}^{\frac{5}{7}} + \frac{1}{c_1 z^*} \dot{\tilde{\mathbf{q}}}_1 \\ &\quad + \frac{5}{7z^*} \mathbf{diag} \left(\mathbf{q}_1^{-\frac{2}{7}} \right) \mathbf{d} - \frac{1}{c_1} \mathbf{f} + \mathbf{sk}(\dot{\psi} \mathbf{e}_3) \frac{1}{c_1} \hat{\mathbf{v}} \end{aligned} \tag{41}$$

where $\mathbf{q}_1^{-\frac{2}{7}} = [q_{11}^{-\frac{2}{7}}, q_{12}^{-\frac{2}{7}}, q_{13}^{-\frac{2}{7}}]^T$. Now we can get the time derivative of the second Lyapunov function. Taking the time derivative of (40), then substituting (39), (41) and (33) into it, we can get

$$\begin{aligned}
 \dot{V}_2 &= \dot{V}_1 + \mathbf{q}_2^T \dot{\mathbf{q}}_2 + \dot{L} \\
 &= -\mathbf{q}_1^T \frac{1}{z^*} \tilde{\mathbf{v}} - \mathbf{q}_2^T \frac{k_2}{c_1 z^*} \tilde{\mathbf{v}}^{\frac{5}{7}} \\
 &\quad - \frac{c_1}{z^*} \sum_{i=1}^3 q_{1i}^r - k_1 \sum_{i=1}^3 \tilde{q}_{1i}^r - \frac{k_2}{z^*} \sum_{i=1}^3 \tilde{v}_i^r \\
 &\quad + \mathbf{q}_2^T \left(-\frac{5}{7} \text{diag} \left(\mathbf{q}_1^{-\frac{2}{7}} \right) \mathbf{sk} (\dot{\psi} \mathbf{e}_3) \mathbf{q}_1 \right. \\
 &\quad \left. - \frac{5}{7z^*} \text{diag} \left(\mathbf{q}_1^{-\frac{2}{7}} \right) \mathbf{v} + \frac{5}{7z^*} \text{diag} \left(\mathbf{q}_1^{-\frac{2}{7}} \right) \mathbf{d} \right. \\
 &\quad \left. + \frac{c_1}{z^*} \mathbf{q}_1 + \mathbf{sk} (\dot{\psi} \mathbf{e}_3) \frac{1}{c_1} \hat{\mathbf{v}} - \frac{1}{c_1} \mathbf{f} \right. \\
 &\quad \left. + \frac{1}{c_1 z^*} \tilde{\mathbf{q}}_1 + \frac{1}{c_1} \dot{\mathbf{d}} \right) \tag{42}
 \end{aligned}$$

We can also prove that (42) is also GFTS.

Theorem 2 Considering \mathbf{f} as a virtual control input and design \mathbf{f} as follows:

$$\begin{aligned}
 \frac{1}{c_1} \mathbf{f} \mathbf{d} &= \frac{5}{7} \text{diag} \left(\mathbf{q}_1^{-\frac{2}{7}} \right) \left(-\mathbf{sk} (\dot{\psi} \mathbf{e}_3) \mathbf{q}_1 - \frac{1}{z^*} \mathbf{v} + \frac{1}{z^*} \mathbf{d} \right) \\
 &\quad + k_3 \mathbf{q}_2^{\frac{5}{7}} + \frac{c_1}{z^*} \mathbf{q}_1 + \mathbf{sk} (\dot{\psi} \mathbf{e}_3) \frac{1}{c_1} \hat{\mathbf{v}} \\
 &\quad + \frac{1}{c_1} \dot{\mathbf{d}} + \frac{1}{c_1 z^*} \tilde{\mathbf{q}}_1 \tag{43}
 \end{aligned}$$

\dot{V}_2 is definite negative, and the system is GFTS.

Proof Substituting (43) into (42), we can obtain

$$\begin{aligned}
 \dot{V}_2 &= -\mathbf{q}_1^T \frac{1}{z^*} \tilde{\mathbf{v}} - \mathbf{q}_2^T \frac{k_2}{c_1 z^*} \tilde{\mathbf{v}}^{\frac{5}{7}} - \frac{c_1}{z^*} \sum_{i=1}^3 q_{1i}^r \\
 &\quad - k_1 \sum_{i=1}^3 \tilde{q}_{1i}^r - \frac{k_2}{z^*} \sum_{i=1}^3 \tilde{v}_i^r - k_3 \sum_{i=1}^3 q_{2i}^r. \tag{44}
 \end{aligned}$$

Using Lemma 3 to scale (44). The parameters are $a = b = c = 1$, then we can get

$$\begin{aligned}
 \dot{V}_2 &\leq -\frac{c_1}{z^*} \sum_{i=1}^3 q_{1i}^r - k_1 \sum_{i=1}^3 \tilde{q}_{1i}^r - \frac{k_2}{z^*} \sum_{i=1}^3 \tilde{v}_i^r \\
 &\quad - k_3 \sum_{i=1}^3 q_{2i}^r - \frac{1}{2z^*} \left(\mathbf{q}_1^T \mathbf{q}_1 + \tilde{\mathbf{v}}^T \tilde{\mathbf{v}} \right) \\
 &\quad - \frac{k_2}{2c_1 z^*} \left(\mathbf{q}_2^T \mathbf{q}_2 + \sum_{i=1}^3 \tilde{v}_i^{\frac{10}{7}} \right) \\
 &\leq -\frac{c_1}{z^*} \sum_{i=1}^3 q_{1i}^r - k_1 \sum_{i=1}^3 \tilde{q}_{1i}^r - \frac{k_2}{z^*} \sum_{i=1}^3 \tilde{v}_i^r - k_3 \sum_{i=1}^3 q_{2i}^r. \tag{45}
 \end{aligned}$$

where $-\mathbf{q}_1^T \frac{1}{z^*} \tilde{\mathbf{v}} \leq -\frac{1}{2z^*} \left(\mathbf{q}_1^T \mathbf{q}_1 + \tilde{\mathbf{v}}^T \tilde{\mathbf{v}} \right)$, $z^* < 0$.

Now use Lemma 2 for (45) and scale (40) simultaneously, then combine the results of the two equations, and we can get

$$\begin{aligned}
 \dot{V}_2 &\leq -\frac{c_1}{z^*} \sum_{i=1}^3 q_{1i}^r - k_1 \sum_{i=1}^3 \tilde{q}_{1i}^r \\
 &\quad - \frac{k_2}{z^*} \sum_{i=1}^3 \tilde{v}_i^r - k_3 \sum_{i=1}^3 q_{2i}^r \\
 &\leq -\beta_2 \left(\sum_{i=1}^3 q_{1i}^r + \sum_{i=1}^3 \tilde{q}_{1i}^r + \sum_{i=1}^3 \tilde{v}_i^r + \sum_{i=1}^3 q_{2i}^r \right) \tag{46}
 \end{aligned}$$

$$\begin{aligned}
 V_2^\alpha &= \left(\frac{1}{2} \sum_{i=1}^3 q_{1i}^2 + \frac{1}{2} \sum_{i=1}^3 q_{2i}^2 \right. \\
 &\quad \left. + \frac{1}{2} \sum_{i=1}^3 \tilde{q}_{1i}^2 + \frac{1}{2} \sum_{i=1}^3 \tilde{v}_i^2 \right)^\alpha \\
 &\leq 2^\alpha \left(\sum_{i=1}^3 q_{1i}^2 + \sum_{i=1}^3 q_{2i}^2 + \sum_{i=1}^3 \tilde{q}_{1i}^2 + \sum_{i=1}^3 \tilde{v}_i^2 \right)^\alpha \\
 &\leq 2 \left(\sum_{i=1}^3 q_{1i}^2 + \sum_{i=1}^3 q_{2i}^2 + \sum_{i=1}^3 \tilde{q}_{1i}^2 + \sum_{i=1}^3 \tilde{v}_i^2 \right)^\alpha \\
 &\leq 2 \left(\sum_{i=1}^3 q_{1i}^r + \sum_{i=1}^3 q_{2i}^r + \sum_{i=1}^3 \tilde{q}_{1i}^r + \sum_{i=1}^3 \tilde{v}_i^r \right) \tag{47}
 \end{aligned}$$

$$\begin{aligned}
 \dot{V}_2 + \frac{\beta_2}{4} V_2^\alpha &\leq -\beta_2 \left(\sum_{i=1}^3 q_{1i}^r + \sum_{i=1}^3 \tilde{q}_{1i}^r + \sum_{i=1}^3 \tilde{v}_i^r + \sum_{i=1}^3 q_{2i}^r \right) \\
 &\quad + \frac{\beta_2}{2} \left(\sum_{i=1}^3 q_{1i}^r + \sum_{i=1}^3 q_{2i}^r + \sum_{i=1}^3 \tilde{q}_{1i}^r + \sum_{i=1}^3 \tilde{v}_i^r \right) \\
 &= -\frac{\beta_2}{2} \left(\sum_{i=1}^3 q_{1i}^r + \sum_{i=1}^3 q_{2i}^r + \sum_{i=1}^3 \tilde{q}_{1i}^r + \sum_{i=1}^3 \tilde{v}_i^r \right) \\
 &\leq 0. \tag{48}
 \end{aligned}$$

The result means that $\dot{V}_2 \leq -\frac{\beta_2}{4} V_2^\alpha$ holds, where $\beta_2 = 2 \min \left\{ \frac{c_1}{z^*}, k_1, \frac{k_2}{z^*}, k_3 \right\}$. So far, we have proved that V_2 is GFTS. \square

Remark 2 If we use the small angle assumption, we can get the desired attitude of the QUAV through the following equation

$$\theta_d = \arctan\left(\frac{f_x^d}{f_z^d - g}\right),$$

$$\phi_d = \arctan\left(-\frac{\cos\theta_d f_y^d}{f_z^d - g}\right).$$

However, this paper does not adopt this assumption, so it is necessary to continue the backstepping design to obtain the desired angular velocity.

Continue to define the third error term

$$\mathbf{q}_3 = \frac{1}{k_3 c_1} \mathbf{f} - \frac{1}{k_3 c_1} \mathbf{f}_d. \tag{49}$$

Update \dot{V}_2 in (42) using (43)

$$\begin{aligned} \dot{V}_2 = & -\mathbf{q}_1^T \frac{1}{z^*} \tilde{\mathbf{v}} - \mathbf{q}_2^T \frac{k_2}{c_1 z^*} \tilde{\mathbf{v}}^{\frac{5}{7}} - k_3 \mathbf{q}_3^T \mathbf{q}_2 \\ & - \frac{c_1}{z^*} \sum_{i=1}^3 q_{1i}^r - k_1 \sum_{i=1}^3 \tilde{q}_{1i}^r \\ & - k_3 \sum_{i=1}^3 q_{2i}^r - \frac{k_2}{z^*} \sum_{i=1}^3 \tilde{v}_i^r. \end{aligned} \tag{50}$$

Then we define the third Lyapunov function

$$V_3 = V_2 + \frac{1}{2} \mathbf{q}_3^T \mathbf{q}_3. \tag{51}$$

Using (50) and (49), we can get the time derivative of V_3

$$\begin{aligned} \dot{V}_3 = & \dot{V}_2 + \mathbf{q}_3^T \dot{\mathbf{q}}_3 \\ = & -\mathbf{q}_1^T \frac{1}{z^*} \tilde{\mathbf{v}} - \mathbf{q}_2^T \frac{k_2}{c_1 z^*} \tilde{\mathbf{v}}^{\frac{5}{7}} - \frac{c_1}{z^*} \sum_{i=1}^3 q_{1i}^r \\ & - k_1 \sum_{i=1}^3 \tilde{q}_{1i}^r - k_3 \sum_{i=1}^3 q_{2i}^r - \frac{k_2}{z^*} \sum_{i=1}^3 \tilde{v}_i^r \\ & + \mathbf{q}_3^T \left(\frac{1}{k_3 c_1} \dot{\mathbf{f}} - \frac{1}{k_3 c_1} \dot{\mathbf{f}}_d - k_3 \mathbf{q}_2 \right) \end{aligned} \tag{52}$$

where $\frac{1}{c_1} \dot{\mathbf{f}}_d$ is expressed as follows:

$$\begin{aligned} \frac{1}{c_1} \dot{\mathbf{f}}_d = & \frac{10}{49} \mathbf{diag}\left(\mathbf{q}_1^{-\frac{9}{7}}\right) \mathbf{diag}(\dot{\mathbf{q}}_1) \mathbf{sk}(\dot{\psi} \mathbf{e}_3) \mathbf{q}_1 \\ & + \frac{10}{49} \mathbf{diag}\left(\mathbf{q}_1^{-\frac{9}{7}}\right) \mathbf{diag}(\dot{\mathbf{q}}_1) \frac{1}{z^*} \mathbf{v} \end{aligned}$$

$$\begin{aligned} & + \frac{10}{49} \mathbf{diag}\left(\mathbf{q}_1^{-\frac{9}{7}}\right) \mathbf{diag}(\dot{\mathbf{q}}_1) \frac{1}{z^*} \mathbf{d} \\ & - \frac{5}{7} \mathbf{diag}\left(\mathbf{q}_1^{-\frac{2}{7}}\right) \frac{1}{z^*} \dot{\mathbf{v}} + \frac{1}{c_1 z^*} \dot{\mathbf{q}}_1 \\ & + \frac{5}{7} \mathbf{diag}\left(\mathbf{q}_1^{-\frac{2}{7}}\right) \frac{1}{z^*} \dot{\mathbf{d}} + \frac{1}{c_1} \ddot{\mathbf{d}} \\ & - \frac{5}{7} \mathbf{diag}\left(\mathbf{q}_1^{-\frac{2}{7}}\right) \mathbf{sk}(\dot{\psi} \mathbf{e}_3) \mathbf{q}_1 \\ & - \frac{5}{7} \mathbf{diag}\left(\mathbf{q}_1^{-\frac{2}{7}}\right) \mathbf{sk}(\dot{\psi} \mathbf{e}_3) \dot{\mathbf{q}}_1 \\ & + \frac{5}{7} k_3 \mathbf{q}_2^{-\frac{2}{7}} \dot{\mathbf{q}}_2 + \mathbf{sk}(\dot{\psi} \mathbf{e}_3) \frac{1}{c_1} \dot{\mathbf{v}} \\ & + \frac{c_1}{z^*} \dot{\mathbf{q}}_1 + \mathbf{sk}(\dot{\psi} \mathbf{e}_3) \frac{1}{c_1} \dot{\mathbf{v}}. \end{aligned} \tag{53}$$

Theorem 3 Considering $\dot{\mathbf{f}}$ as the control input and design $\dot{\mathbf{f}}$ as follows:

$$\dot{\mathbf{f}} = -c_1 k_3 k_4 \mathbf{q}_3^{\frac{5}{7}} + c_1 k_3^2 \mathbf{q}_2 + \dot{\mathbf{f}}_d \tag{54}$$

\dot{V}_3 is definite negative, the system is GFTS.

Proof Substituting (54) into (52), we can get

$$\begin{aligned} \dot{V}_3 = & \dot{V}_2 + \mathbf{q}_3^T \dot{\mathbf{q}}_3 \\ = & -\mathbf{q}_1^T \frac{1}{z^*} \tilde{\mathbf{v}} - \mathbf{q}_2^T \frac{k_2}{c_1 z^*} \tilde{\mathbf{v}}^{\frac{5}{7}} - \frac{c_1}{z^*} \sum_{i=1}^3 q_{1i}^r \\ & - k_1 \sum_{i=1}^3 \tilde{q}_{1i}^r - k_3 \sum_{i=1}^3 q_{2i}^r \\ & - \frac{k_2}{z^*} \sum_{i=1}^3 \tilde{v}_i^r - k_4 \sum_{i=1}^3 q_{3i}^r. \end{aligned} \tag{55}$$

Using Lemma 3 to scale (55), we can get

$$\begin{aligned} \dot{V}_3 \leq & -\frac{c_1}{z^*} \sum_{i=1}^3 q_{1i}^r - k_1 \sum_{i=1}^3 \tilde{q}_{1i}^r \\ & - \frac{k_2}{z^*} \sum_{i=1}^3 \tilde{v}_i^r - k_3 \sum_{i=1}^3 q_{2i}^r - k_4 \sum_{i=1}^3 q_{3i}^r \\ & - \frac{1}{2z^*} \left(\mathbf{q}_1^T \mathbf{q}_1 + \tilde{\mathbf{v}}^T \tilde{\mathbf{v}} \right) \\ & - \frac{k_2}{2c_1 z^*} \left(\mathbf{q}_2^T \mathbf{q}_2 + \sum_{i=1}^3 \tilde{v}_i^{\frac{10}{7}} \right) \end{aligned}$$

$$\begin{aligned} &\leq -\frac{c_1}{z^*} \sum_{i=1}^3 q_{1i}^r - k_1 \sum_{i=1}^3 \tilde{q}_{1i}^r \\ &\quad - \frac{k_2}{z^*} \sum_{i=1}^3 \tilde{v}_i^r - k_3 \sum_{i=1}^3 q_{2i}^r - k_4 \sum_{i=1}^3 q_{3i}^r. \end{aligned} \tag{56}$$

Using Lemma 2 for (56) and scaling (51), we can obtain

$$\begin{aligned} \dot{V}_3 &\leq -\frac{c_1}{z^*} \sum_{i=1}^3 q_{1i}^r - k_1 \sum_{i=1}^3 \tilde{q}_{1i}^r \\ &\quad - \frac{k_2}{z^*} \sum_{i=1}^3 \tilde{v}_i^r - k_3 \sum_{i=1}^3 q_{2i}^r - k_4 \sum_{i=1}^3 q_{3i}^r \\ &\leq \beta_3 \left(-\sum_{i=1}^3 q_{1i}^r - \sum_{i=1}^3 \tilde{q}_{1i}^r \right. \\ &\quad \left. - \sum_{i=1}^3 \tilde{v}_i^r - \sum_{i=1}^3 q_{2i}^r - \sum_{i=1}^3 q_{3i}^r \right) \end{aligned} \tag{57}$$

$$\begin{aligned} V_3^\alpha &= \left(\frac{1}{2} \sum_{i=1}^3 q_{1i}^2 + \frac{1}{2} \sum_{i=1}^3 q_{2i}^2 + \frac{1}{2} \sum_{i=1}^3 q_{3i}^2 \right. \\ &\quad \left. + \frac{1}{2} \sum_{i=1}^3 \tilde{q}_{1i}^2 + \frac{1}{2} \sum_{i=1}^3 \tilde{v}_i^2 \right)^\alpha \\ &\leq 2 \left(\sum_{i=1}^3 q_{1i}^2 + \sum_{i=1}^3 q_{2i}^2 + \sum_{i=1}^3 q_{3i}^2 \right. \\ &\quad \left. + \sum_{i=1}^3 \tilde{q}_{1i}^2 + \sum_{i=1}^3 \tilde{v}_i^2 \right)^\alpha \end{aligned} \tag{58}$$

$$\begin{aligned} \dot{V}_3 + \frac{\beta_3}{4} V_3^\alpha &\leq -\beta_3 \left(\sum_{i=1}^3 q_{1i}^r + \sum_{i=1}^3 \tilde{q}_{1i}^r + \sum_{i=1}^3 \tilde{v}_i^r \right. \\ &\quad \left. + \sum_{i=1}^3 q_{2i}^r + \sum_{i=1}^3 q_{3i}^r \right) \\ &\quad + \frac{\beta_3}{2} \left(\sum_{i=1}^3 q_{1i}^r + \sum_{i=1}^3 \tilde{q}_{1i}^r + \sum_{i=1}^3 \tilde{v}_i^r \right. \\ &\quad \left. + \sum_{i=1}^3 q_{2i}^r + \sum_{i=1}^3 q_{3i}^r \right) \\ &= -\frac{\beta_3}{2} \left(\sum_{i=1}^3 q_{1i}^r + \sum_{i=1}^3 \tilde{q}_{1i}^r + \sum_{i=1}^3 \tilde{v}_i^r \right. \\ &\quad \left. + \sum_{i=1}^3 q_{2i}^r + \sum_{i=1}^3 q_{3i}^r \right) \leq 0. \end{aligned} \tag{59}$$

The result means that $\dot{V}_3 \leq -\frac{\beta_3}{4} V_3^\alpha$ holds, where $\beta_3 = 2 \min \left\{ \frac{c_1}{z^*}, k_1, \frac{k_2}{z^*}, k_3, k_4 \right\}$. The V_3 is GFTS. \square

So far, we have obtained the controller, but we also need to obtain the angular velocity and thrust that can be directly used to control the QUAUV. Since in the virtual camera frame, the force \mathbf{f} of the QUAUV is expressed as

$$\mathbf{f} = \frac{-\mathbf{R}_{\phi\theta} U_1 \mathbf{E}_3}{m} + g \mathbf{e}_3. \tag{60}$$

By taking the time derivative of (60) and substituting (15), we can get

$$\mathbf{sk}(\dot{\psi} \mathbf{e}_3) (m g \mathbf{e}_3 - m \mathbf{f}) - m \dot{\mathbf{f}} = \mathbf{R}_{\phi\theta} \begin{bmatrix} \omega_2 U_1 \\ -\omega_1 U_1 \\ \dot{U}_1 \end{bmatrix}. \tag{61}$$

Finally, we only need to combine (61) and (54) to obtain the desired thrust U_1 and desired angular velocity ω_1, ω_2 respectively.

Remark 3 This paper only studies the controller of translational motion obtained from the image moment features in the virtual camera plane. When we get the desired angular velocity, we can use a PD or PID controller to realize the translation control of the QUAUV in the horizontal and altitude directions.

After the above process, we get the controller to control the translational motion, and we also need to get the controller to control the yaw motion. The image feature error is defined as

$$q_4 = q_\psi - q_\psi^d$$

where q_ψ^d is the desired value. According to (12), we can get

$$\dot{q}_4 = -\dot{\psi} + \Delta_\psi. \tag{62}$$

Theorem 4 Design the control input as

$$\omega_3 = \left(k_5 q_4^{\frac{1}{3}} + \Delta_\psi - \frac{\sin \phi}{\cos \theta} \omega_2 \right) \frac{\cos \theta}{\cos \phi} \tag{63}$$

with $k_5 > 0$, the image feature error q_4 while converge to zero in a finite-time.

Proof The relationship between the time derivative of Euler angles and angular velocity is as follows:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & \sin \phi \\ 0 & \sin \phi / \cos \theta & \cos \phi / \cos \theta \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix}. \quad (64)$$

From (64), we have

$$\dot{\psi} = \frac{\sin \phi}{\cos \theta} \omega_2 + \frac{\cos \phi}{\cos \theta} \omega_3. \quad (65)$$

Now we choose a Lyapunov function

$$V_\psi = \frac{1}{2} q_4^2. \quad (66)$$

Then we take the time derivative of (66) and substitute (63) (65) into it, we get

$$\dot{V}_\psi = q_4 (-\dot{\psi} + \Delta_\psi) = -k_5 q_4^r. \quad (67)$$

By scaling (66) and combining the results with (67), we can get

$$V_\psi^\alpha = \left(\frac{1}{2} q_4^2\right)^\alpha \leq \left(2q_4^2\right)^\alpha \leq 2q_4^r, \quad (68)$$

$$\dot{V}_\psi + \frac{k_5}{4} V_\psi^\alpha = -k_5 q_4^r + \frac{k_5}{2} q_4^r = -\frac{k_5}{2} q_4^r \leq 0. \quad (69)$$

The result means that $\dot{V}_\psi \leq -\frac{k_5}{4} V_\psi^\alpha$ holds, and the V_ψ is GFTS. Finally, we can get that q_4 will converge to zero in a finite time. \square

Remark 4 In the actual QUAV system, yaw motion control is usually regarded as an independent channel. Therefore, according to the characteristics of its dynamic model and image moment error model, the design parameters of the yaw motion controller are selected as follows:

$$n = 1, r = \frac{4n}{2n + 1} = \frac{4}{3}, \alpha = \frac{2n}{2n + 1} = \frac{2}{3}.$$

Remark 5 After the above controller design process, we can use Lemma 1 to describe the convergence time of the system quantitatively. Recall that convergence time $T(x)$ as follows:

$$T(x) \leq \frac{V^{1-\alpha}(x(0))}{c(1-\alpha)}. \quad (70)$$

Then the convergence time of the translation motion and the yaw motion can be expressed as follows:

$$T_1(x) \leq \frac{V_3^{\frac{1}{7}}(\mathbf{q}_1(0))}{\beta_3/28} \quad (71)$$

$$T_2(x) \leq \frac{V_\psi^{\frac{1}{3}}(q_4(0))}{k_5/12} \quad (72)$$

where $\mathbf{q}_1(0)$ and $q_4(0)$ represent the corresponding initial state of the system, respectively.

Remark 6 In the controller design stage, the rotor of QUAV is assumed to be ideal. For the control allocation of the actuator, since the four rotors can produce a single thrust U_1 and a full torque vector $\tau = (\tau_1, \tau_2, \tau_3)^T$ for rotation, we can use the following equation to obtain the desired angular velocity of the four rotors.

$$F = -U_1 E_3 + mgR^T e_3 = \begin{bmatrix} 0 \\ 0 \\ b(n_1^2 + n_2^2 + n_3^2 + n_4^2) \end{bmatrix} + mgR^T e_3 \quad (73)$$

$$\tau = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} = \begin{bmatrix} bl(n_4^2 - n_2^2) \\ bl(n_3^2 - n_1^2) \\ dl(n_2^2 + n_4^2 - n_1^2 - n_3^2) \end{bmatrix} \quad (74)$$

where $n_1, n_2, n_3,$ and n_4 denote the angular velocity of the front, right, rear, and left rotor, respectively. b and d are the thrust and drag factors. l is the distance between each rotor center and the center of mass of the QUAV.

4 Simulation

In order to verify the effectiveness of the proposed controller, we set up four groups of simulations. In the first group, we apply the proposed method to static targets to ensure that our method can be applied in static scenarios. We apply the proposed method to a moving target in the second group. We compare the proposed method with the previous methods in the third group. In the fourth group, we conduct a simulation comparison in Robot Operating System (ROS) gazebo environment. The model parameters of the QUAV and the camera are shown in Table 2.

The first three groups of simulation are numerical simulations conducted in MATLAB R2019b of Windows 10×64. The fourth group of ROS Gazebo moving

Table 2 Simulation parameters

Contents	Parameters
m	2 kg
g	9.81 m/s ²
\mathbf{J}	diag {0.0081, 0.0081, 0.0142} kg · m ² /rad ²
Initial position (x, y, z)	(3, 2, -8)m
Initial pose (ϕ, θ, ψ)	(0, 0, 0.174)rad
Focal length λ	3.2 mm
Pixel length ^a	1.4 × 10 ⁻⁶ m
Fixed points ^b	(0.25, 0.5, 0) m, (-0.25, 0.5, 0) m, (-0.25, -0.5, 0) m, (0.25, -0.5, 0) m
a^*	8 × 10 ⁻⁷
\mathbf{q}_d	(0, 0, 1, 0)

^aWe assume that the pixels of the camera are square

^bThese points are located in the inertial frame I

target simulation was conducted under Ubuntu 18.04 amd64. The simulation calculation platform parameter is Intel i7-8700k with 64GB RAM.

4.1 Numerical simulation of the stationary target

The selection of control parameters of the controller is shown in Table 3. It should be noted that the current desired image moment can be obtained when the QUAV is at position (0, 0, -4)m in the inertial frame I with attitude (0, 0, 0)rad. Therefore, the desired height is $z^* = -4$ m. The numerical simulation is divided into two groups: one without external disturbance and the other with external disturbance $\mathbf{d}\mathbf{t} = [0.2 \sin(t), 0.2 \sin(t), 0.2 \sin(t)]^T \text{m} \cdot \text{s}^{-2}$.

Figure 4 shows the performance of the proposed control method on static targets without external disturbance. Figure 4a shows that QUAV finally hovers at the position of (0, 0, -4)m, and Fig. 4b shows that the attitude of QUAV is (0, 0, 0)rad. Figure 4c shows the convergence of the image moment characteristic error in the plane of the virtual camera. It can be seen that the proposed control method can make the system reach the desired state, and it can be seen from the yaw feature that the system is stable in a finite time. Figure 4d shows

Table 3 Control parameters

Contents	Parameters	Contents	Parameters
k_1	4	k_3	1
k_2	2	k_4	0.5
c_1	1	k_5	0.1
α_1	0.94	Δ	0.05
R	25		

the linear velocity in the virtual plane. We can see that the designed linear observer can accurately estimate the actual linear velocities of QUAV. Figure 4e, f shows the trajectories of feature points in the virtual camera plane and the actual camera plane, respectively. Through the axis of the trajectories, we can find that theoretically, using a low-cost 1080p resolution camera in practice is enough to deal with the current visual servoing task. Figure 4g, h shows the thrust input and torque input of the system, respectively. Figure 4i, j, k shows the observation state of the target motion observer in three axes, respectively. Figure 4l shows the space trajectory of the QUAV. We can see that the target observer can accurately estimate the motion state of the target.

Figure 5 shows the performance of the proposed control method on static targets with external disturbance. We can see that although we impose external disturbances on the system, the system is robust to external disturbances.

4.2 Numerical simulation of the moving target

Firstly, we give the motion constraint equation of the target, and its trajectory in space is a square in the xOy plane. The simulation results are shown in Fig. 6.

The motion constraint equation as follows:

$$\begin{cases} v_x = 0.2 \text{ m/s}, v_y = 0 \text{ m/s}, & t \in (0, 10] \text{ s} \\ v_x = 0 \text{ m/s}, v_y = 0.2 \text{ m/s}, & t \in (10, 20] \text{ s} \\ v_x = -0.2 \text{ m/s}, v_y = 0 \text{ m/s}, & t \in (20, 30] \text{ s} \\ v_x = 0 \text{ m/s}, v_y = -0.2 \text{ m/s}, & t \in (30, 40] \text{ s} \end{cases} \quad (75)$$

The target is always kept in the xOy plane during movement, i.e., $v_z = 0$ m/s. At the same time, the target does not spin at any angle, that is, $\omega_z = 0$ rad/s. The numerical simulation is divided into two groups, one without

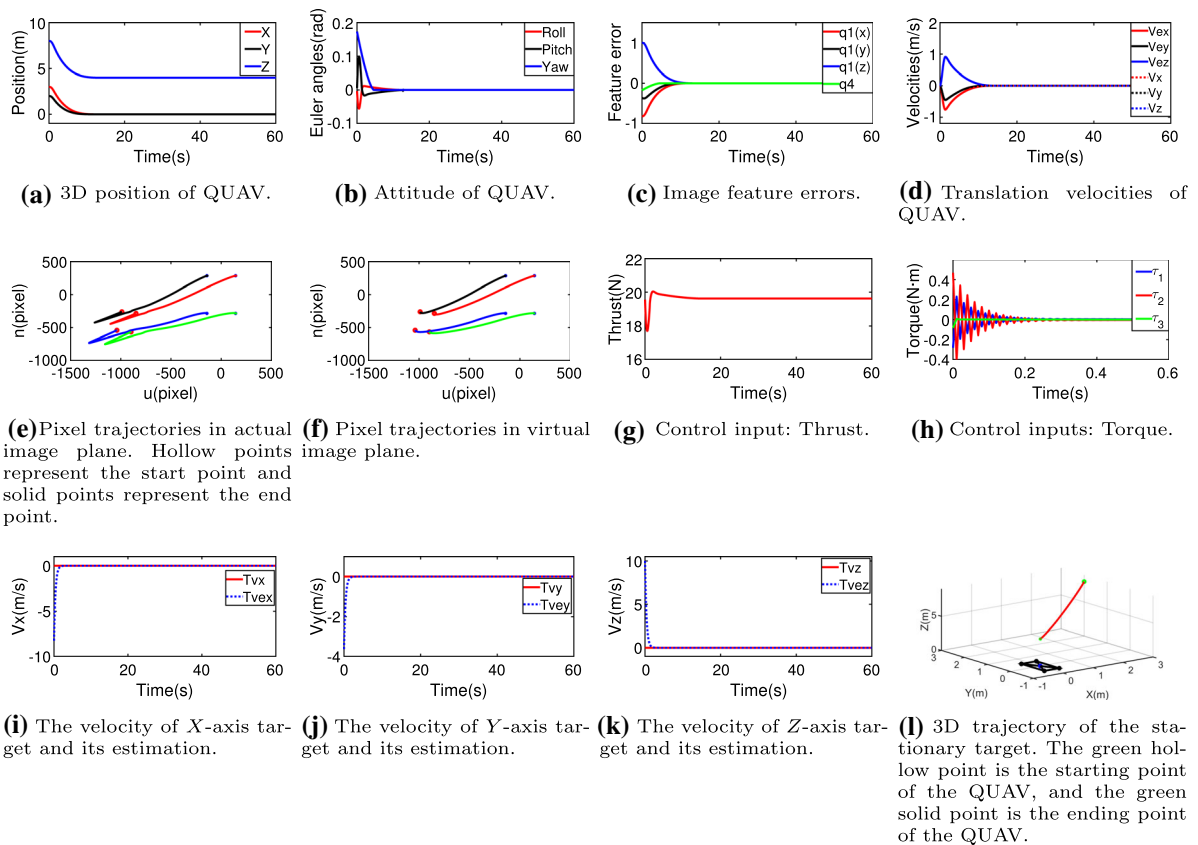


Fig. 4 Numerical simulation of the stationary target without external disturbance

external disturbance and the other with external disturbance $dt = [0.2 \sin(t), 0.2 \sin(t), 0.2 \sin(t)]^T m \cdot s^{-2}$.

We focus on the following aspects for this simulation:

- (1) Image moment feature error.
- (2) Linear velocity observer.
- (3) Target trajectory observer.

Figure 6c shows the convergence of image moment characteristic error. We can see some weak fluctuations in the figure due to the QAV tracking the target to the inflection point, but they can quickly converge to the desired value. Figure 6d shows the linear velocity in the plane of the virtual camera. We can see that the proposed finite-time linear velocity observer can also accurately estimate the linear velocity of QAV in the scene of target motion. Figure 6i, j, k shows the situation of the target trajectory observer. We can see that in the scene of target motion, the target trajectory observer can also accurately estimate the motion parameters of the target. Figure 7 shows the performance of the pro-

posed control method on moving targets with external disturbance. We can also see that the system is robust to external disturbances. Finally, Fig. 7l shows the spatial trajectory of QAV. We can find that there is no time delay in the proposed control method for tracking the moving target.

To further illustrate the effectiveness of our proposed method, we also carried out a group of nonlinear target tracking simulations. The target trajectory parameters are as follows:

$$v_x(t) = -\sin(t) \text{ m/s}, v_y(t) = \cos(t) \text{ m/s}. \quad (76)$$

Figure 8 shows the numerical simulation results for nonlinear moving targets. Because we use a nonlinear tracking differentiator, we can get the motion trajectory estimation parameters of the target, so the system shows adaptability to nonlinear moving targets. Therefore, we can verify the effectiveness of the proposed control method for tracking nonlinear moving targets.

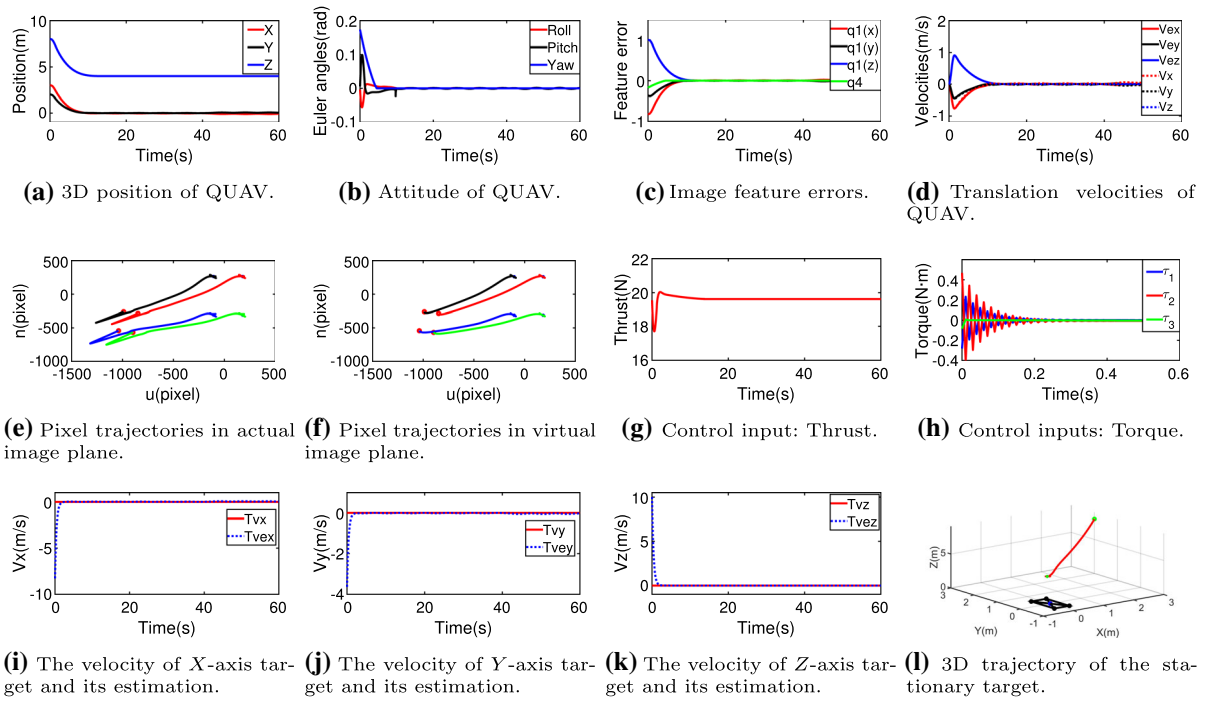


Fig. 5 Numerical simulation of the stationary target with external disturbance

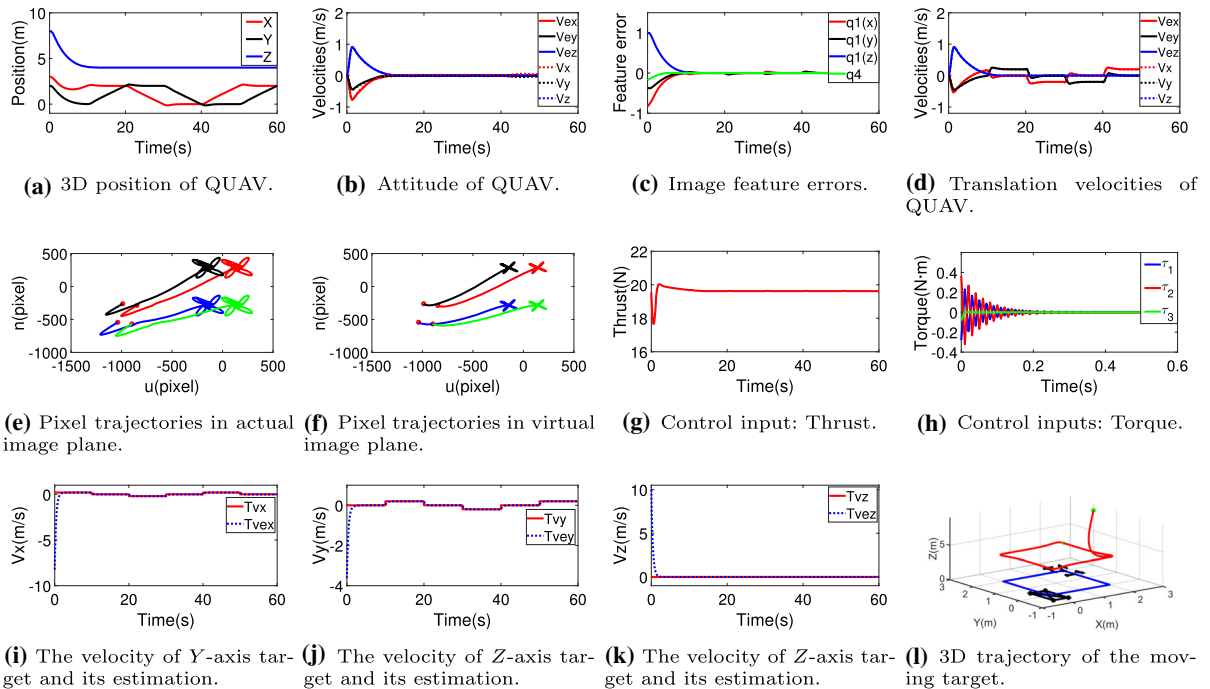


Fig. 6 Numerical simulation of the moving target without external disturbance

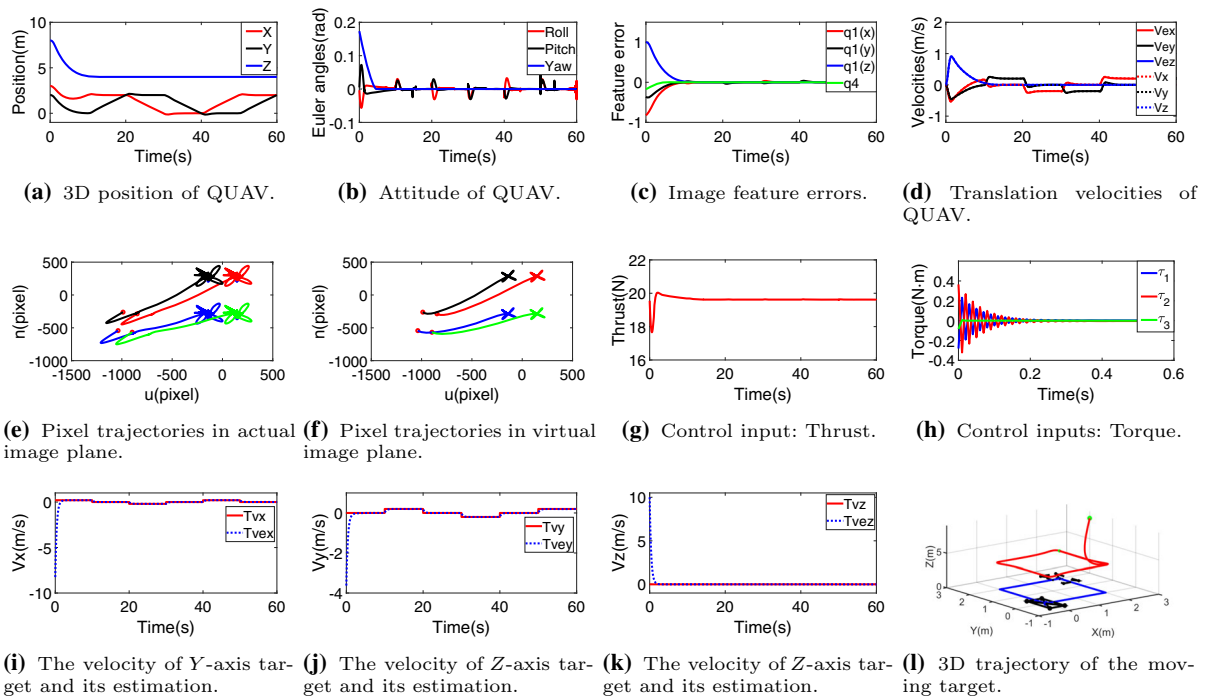


Fig. 7 Numerical simulation of the moving target with external disturbance

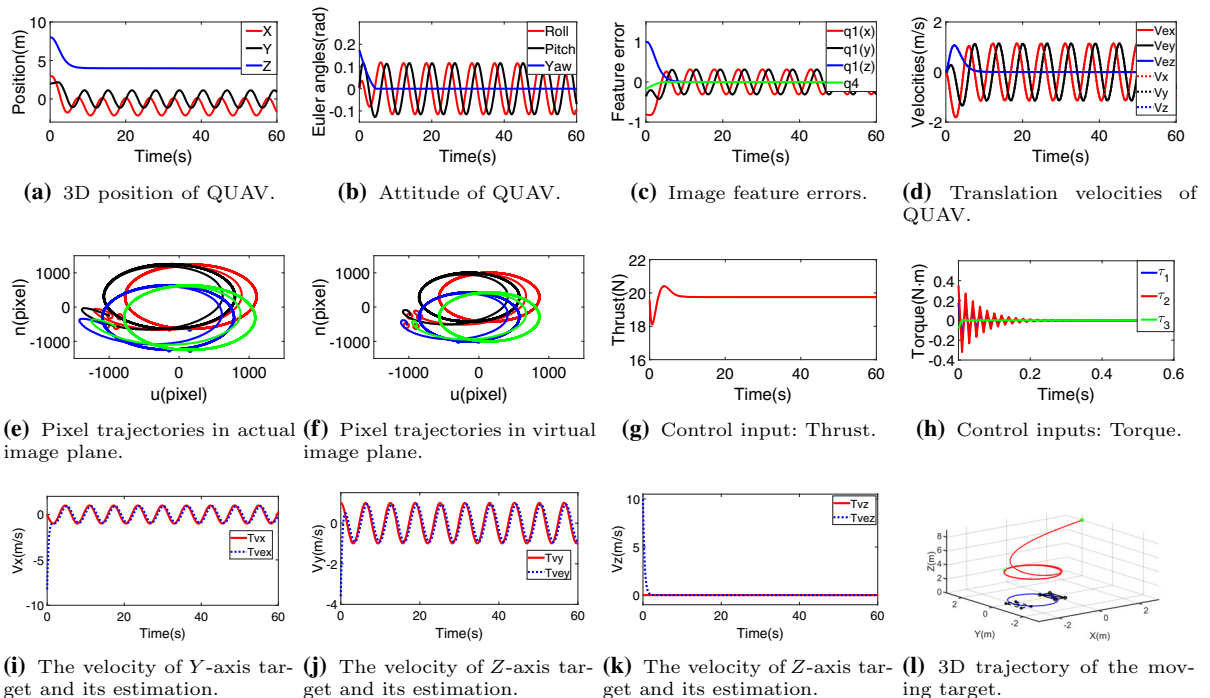


Fig. 8 Numerical simulation of the nonlinear moving target

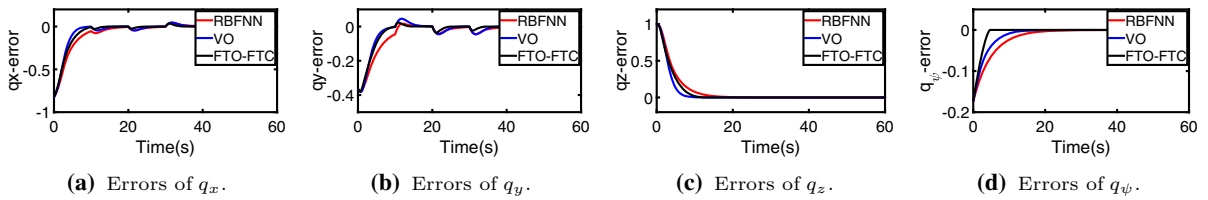


Fig. 9 Image moment feature errors under three control methods. “VO” means “VE-backstepping” method

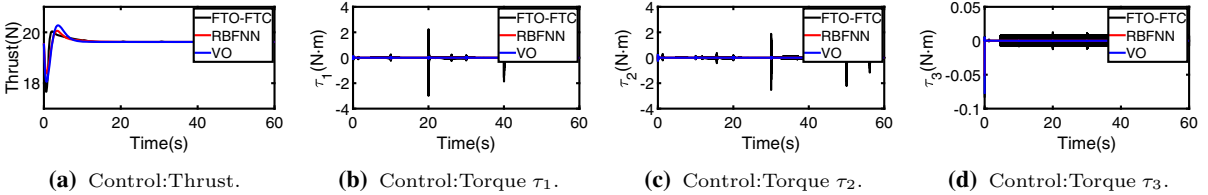


Fig. 10 Control inputs under three control methods

Table 4 IAE and ISDE of numerical simulation

Methods	Error	IAE	ISDE
RBFNN	q_x	0.0711 (+)	0.0245 (+)
	q_y	0.0437 (+)	0.0071 (+)
	q_z	0.0750 (+)	0.0401 (+)
	q_ψ	0.0289 (+)	0.0017 (+)
VE-	q_x	0.0521	0.0198 (-)
	q_y	0.0365	0.0063
Backstepping	q_z	0.0518 (-)	0.0335 (-)
	q_ψ	0.0097	7.4903×10^{-4}
VE-FTO-FTC	q_x	0.0487 (-)	0.0201 (-)
	q_y	0.0285 (-)	0.0057 (-)
(Our method)	q_z	0.0614	0.0354
	q_ψ	0.0054 (-)	5.6275×10^{-4} (-)

(+) and (-) represent the maximum and minimum values of the item, respectively, which is the same as later

4.3 Comparative simulation of moving targets

In order to better illustrate the performance of the proposed method, we conducted a set of comparative experiments. The methods involved in the comparison include the artificial neural network method proposed by Masoud [19], which is noted as RBFNN. The other method uses the target observer proposed by Zhiqiang [21], which is noted as VE-backstepping. The methods selected here all use the same dynamic model and image moment features in the virtual camera plane. At the same time, to quantitatively illustrate the perfor-

mance of the proposed method, we use the four indicators proposed by Jing and Qiang [36, 37] to analyze the control method. Our method is noted as VE-FTO-FTC.

The four indices are as follows:

- (1) Integrated Absolute Error (IAE). It is used to measure the tracking performance of the control method.

$$IAE = \int |e(t)| dt.$$

Table 5 IAU and ISDU of numerical simulation

Methods	Input	IAU	ISDU
RBFNN	U_1	19.6210 (-)	0.0329 (-)
	τ_1	5.5290×10^{-4} (-)	2.1887×10^{-5} (-)
	τ_2	6.4639×10^{-4} (-)	3.5541×10^{-5} (-)
	τ_3	9.6328×10^{-6} (-)	1.1416×10^{-7} (-)
VE-	U_1	19.6214	0.0586
	τ_1	5.7101×10^{-4}	2.5954×10^{-5}
Backstepping	τ_2	7.4470×10^{-4}	6.3480×10^{-5}
	τ_3	2.6737×10^{-5}	1.0280×10^{-6}
VE-FTO-FTC	U_1	19.6221 (+)	0.0526 (+)
	τ_1	0.0212 (+)	0.0099 (+)
(Ours)	τ_2	0.0271 (+)	0.0113(+)
	τ_3	0.0070 (+)	5.3911×10^{-5} (+)

- (2) Integrated Square Error (ISDE). It shows the fluctuation degree of tracking error.

$$\text{ISDE} = \int (e(t) - \bar{e}(t))^2 dt$$

where \bar{e} is the mean of the error.

- (3) Integrated Absolute Control (IAU). It shows the intervention of the controller.

$$\text{IAU} = \int |u(t)| dt.$$

- (4) Integrated Square Control (ISDU). It is used to measure the fluctuation degree of control signal.

$$\text{ISDU} = \int (u(t) - \bar{u}(t))^2 dt$$

where \bar{u} is the mean of the control input.

Remark 7 IAE represents the accumulation of absolute error, and the smaller this indicator is, the stronger the control effect is. ISDE represents the variance of error, which describes the fluctuation of error. When the control effect of the controller is more robust, the value is smaller. Similarly, IAU describes the accumulation of controllers. The higher the value, the stronger the control effect. ISDU is the variance of the controller. The more significant the value, the better the control intervention effect.

Figures 9 and 10 show the image moment feature errors and control inputs under three control methods, respectively. Figure 10 shows the control effect of our proposed method. However, the superiority of our proposed control method cannot be better demonstrated only by the data plots. Therefore, we use the above four indicators and obtain Tables 4 and 5.

As shown in Table 4, our control method has smaller IAE and ISDE, which means that our method can make the system reach the desired value as soon as possible. In addition, Table 5 shows that our proposed control method has large IAU and ISDE, which means that our method has a better control effect.

4.4 Simulation experiment in ROS gazebo

The simulation parameters in ROS gazebo are shown in Table 6. And Table 7 shows the control parameters in ROS gazebo. Since the environment of the ROS gazebo is close to the actual environment, we suggest selecting relatively small control parameters. The simulation environment we use is Prometheus framework [38], which is developed based on MAVROS. We need to build our scenario to use this environment, as shown in Fig. 9a. For the requirements of the visual servoing task, we design the ground markers as shown in Fig. 9b.

In the ROS gazebo, the sensor parameters of QUAV are provided by the IMU module of the gazebo. It should be noted that the module can add Gaussian white noise to simulate the measurement noise. These param-

Table 6 ROS gazebo simulation parameters

Contents	Parameters
m^a	1.47 kg
g^a	9.81 m/s ²
\mathbf{J}^a	diag {0.011, 0.015, 0.021} kg · m ² /rad ²
Initial position (x, y, z)	(0, 0, -3)m
Initial pose (ϕ, θ, ψ)	(0, 0, 0.174)rad
Focal length λ^b	369.502083 pixel
Horizontal field of view ^b	2.0944 rad
Camera resolution ^b	1280 × 720
Fixed points ^c	(0.35, 0.65, 0.01) m, (-0.35, 0.5, 0.01) m, (-0.35, -0.65, 0.01) m, (0.35, -0.65, 0.01) m, (0.0, 0.0, 0.0) m
z^*	-1.5m
a^*	137905.00
\mathbf{q}_d	(0, 0, 1, 0)

^a Mass m , gravity acceleration g and inertia \mathbf{J} are approximate values

^bThe focal length λ of the camera can be calculated by field of view angle and camera resolution. The actual camera needs to be calibrated

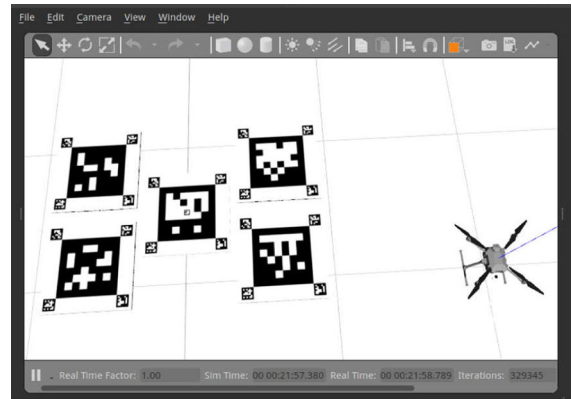
^c These points are located in the inertial frame I . Moreover, these five points are the center points of the QR code image

Table 7 Control parameters in ROS gazebo

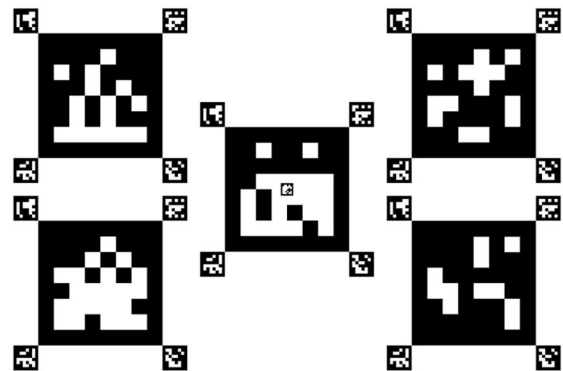
Contents	Parameters	Contents	Parameters
k_1	2	k_3	1
k_2	1	k_4	0.1
c_1	0.5	k_5	0.1
α_1	0.94	Δ	0.03
R	10		

Table 8 IMU parameters in ROS gazebo

Sensors	Noise density	Bias correlation time	Drift lower bound	Drift upper bound
Magnetometer	0.0004	600	–	–
Gyroscope	3.394×10^{-4}	1000	3.8785×10^{-5}	0.0087
Accelerometer	0.004	300	0.006	0.196



(a) Simulation scene in ROS gazebo.



(b) Ground markers.

Fig. 11 ROS gazebo simulation environment

eters are ROS default, and we have not modified these parameters. Table 8 shows the parameters of the sensors.

In the simulation process, we first make QUAV fly to the initial position and maintain the initial attitude. After that, the QUAV will hover in this position and wait for the command of the visual servoing. After receiving the start command, the QUAV starts to enter the IBVS task independently. At the same time, the system will record the current data of the QUAV for subsequent analysis. The motion constraint equation as follows:

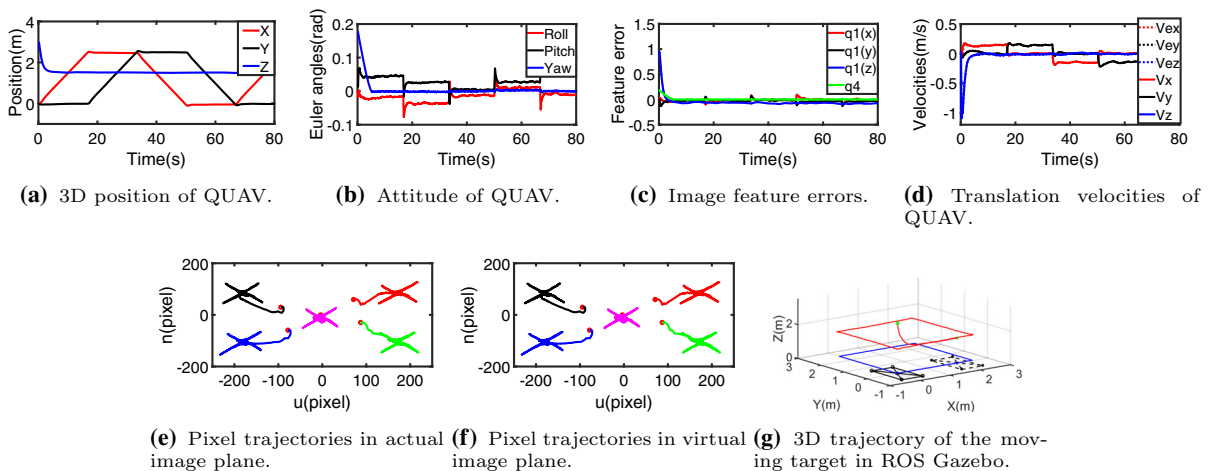


Fig. 12 Simulation in ROS gazebo

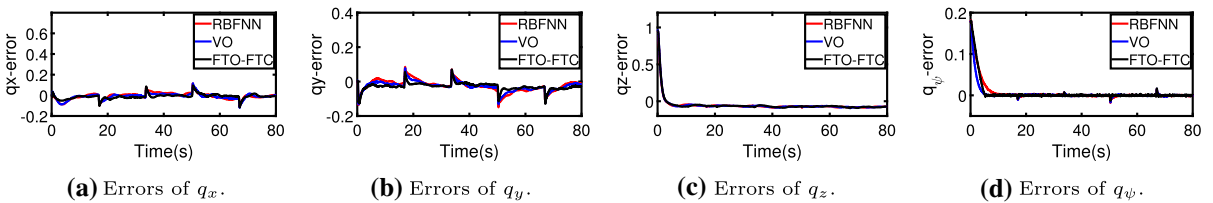


Fig. 13 Time charts of image moment feature error under three control schemes

$$\begin{cases} v_x = 0.5 \text{ m/s}, v_y = 0 \text{ m/s}, & t \in (0, 17] \text{ s} \\ v_x = 0 \text{ m/s}, v_y = 0.5 \text{ m/s}, & t \in (17, 34] \text{ s} \\ v_x = -0.5 \text{ m/s}, v_y = 0 \text{ m/s}, & t \in (34, 51] \text{ s} \\ v_x = 0 \text{ m/s}, v_y = -0.5 \text{ m/s}, & t \in (51, 68] \text{ s} \end{cases}$$

The target is always kept in the xOy plane during movement, i.e., $v_z = 0 \text{ m/s}$. At the same time, the target does not spin at any angle, that is, $\omega_z = 0 \text{ rad/s}$.

As shown in Fig. 12c, the proposed control method can also converge the image feature error to the expected value in the ROS gazebo. Figure 12d shows that the designed finite-time observer can estimate the linear velocity of QUAUV more accurately. Figure 12g shows the spatial trajectory of QUAUV. It can be seen that the control effect is ideal. It should be noted that in Fig. 12b, the attitude angle in the figure has oscillated, but its range is within 2 degrees. Therefore, we think the QUAUV can be in hovering mode.

We also conducted comparative experiments in ROS gazebo, and the comparative methods are still RBFNN and VE-backstepping. Figure 13 shows the variation

Table 9 IAE and ISDE of ROS gazebo simulation

Methods	Error	IAE	ISDE
RBFNN	q_x	0.0229 (+)	9.5138×10^{-4} (+)
	q_y	0.0326 (+)	0.0012 (+)
	q_z	0.0778 (+)	0.0401 (+)
	q_ψ	0.0070 (+)	5.6282×10^{-4} (+)
VE-	q_x	0.0187 (-)	5.4594×10^{-4} (-)
	q_y	0.0298	4.5750×10^{-4} (-)
Backstepping	q_z	0.0784	0.0091
	q_ψ	0.0059	5.6282×10^{-4}
VE-FTO-FTC	q_x	0.0211	8.0148×10^{-4}
	q_y	0.0289 (-)	7.6505×10^{-4}
(Our method)	q_z	0.0781 (-)	0.0088 (-)
	q_ψ	0.0040 (-)	3.2289×10^{-4} (-)

of image moment feature error with time under three control schemes. We also use IAE and ISDE to analyze the performance of the controller. Table 9 shows the result.

The above simulation experiments show that the proposed control method has better control convergence, and the designed finite-time observer can also accurately estimate the linear velocity in the virtual image plane of the QUAUV. However, parameters such as system quality, inertia, and camera focal length are not easy to obtain in practice. Therefore, we need to further discuss these problems in the future.

5 Conclusion

In this paper, we propose a novel image-based visual servoing control scheme combining target motion differentiator, finite-time observer, and finite-time controller, and we apply the scheme to the research of moving target tracking of QUAUV. We use the differentiator to estimate the target motion parameters and transfer the parameters to the controller to ensure that the system can track the target in real time. Aiming to acquire camera depth information and the linear velocity of QUAUV, we design a finite-time observer to compensate for this information. Finally, we design the global finite-time controller of the system using the simplified backstepping method. We have made assumptions about the visibility of the target, but in practice, the target is likely to be out of the field of view of the camera. At the same time, we do not conduct real machine experiments. Therefore, we plan to consider the target visibility constraint in future research and real machine verification.

Author contributions WH involved in writing—original draft, validation and software. LY involved in writing—review and editing, and supervision.

Funding The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: The authors would like to thank the National Natural Science Foundation of China under Grant [52275003] and Grant [U1813220], in part by the Fundamental Research Funds for the Central Universities under Grant [buctrc202105] for their support in this research.

Data availability All data are available upon request at the authors' email address.

Declarations

Conflict of interest The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Code availability Custom code is available upon request at Liang Yuan email address.

References

- Shao, X., Wang, L., Li, J., Liu, J.: High-order eso based output feedback dynamic surface control for quadrotors under position constraints and uncertainties. *Aerosp. Sci. Technol.* **89**, 288–298 (2019)
- Guerreiro, B.J., Silvestre, C., Cunha, R., Cabecinhas, D.: Lidar-based control of autonomous rotorcraft for the inspection of pierlike structures. *IEEE Trans. Control Syst. Technol.* **26**(4), 1430–1438 (2017)
- Guerrero-Sánchez, M.E., Mercado-Ravell, D.A., Lozano, R., García-Beltrán, C.D.: Swing-attenuation for a quadrotor transporting a cable-suspended payload. *ISA Trans.* **68**, 433–449 (2017)
- Tomic, T., Schmid, K., Lutz, P., Domel, A., Kassecker, M., Mair, E., Grix, I.L., Ruess, F., Suppa, M., Burschka, D.: Toward a fully autonomous uav: research platform for indoor and outdoor urban search and rescue. *IEEE Robot. Autom. Mag.* **19**(3), 46–56 (2012)
- Sani, M.F., Shoaran, M., Karimian, G.: Automatic landing of a low-cost quadrotor using monocular vision and Kalman filter in gps-denied environments. *Turk. J. Electr. Eng. Comput. Sci.* **27**(3), 1821–1838 (2019)
- Chen, J., Liu, T., Shen, S.: Tracking a moving target in cluttered environments using a quadrotor. In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 446–453. IEEE (2016)
- Beyeler, A., Zufferey, J.-C., Floreano, D.: Vision-based control of near-obstacle flight. *Auton. Robot.* **27**(3), 201–219 (2009)
- Chaumette, F., Hutchinson, S.: Visual servo control. i. Basic approaches. *IEEE Robot. Autom. Mag.* **13**(4), 82–90 (2006)
- Guenard, N., Hamel, T., Mahony, R.: A practical visual servo control for an unmanned aerial vehicle. *IEEE Trans. Rob.* **24**(2), 331–340 (2008)
- Yesildirek, A., Imran, B.: Nonlinear control of quadrotor using multi lyapunov functions. In: 2014 American Control Conference, pp. 3844–3849. IEEE (2014)
- Lee, D., Lim, H., Kim, H.J., Kim, Y., Seong, K.J.: Adaptive image-based visual servoing for an underactuated quadrotor system. *J. Guid. Control. Dyn.* **35**(4), 1335–1353 (2012)
- Chaumette, F.: Image moments: a general and useful set of features for visual servoing. *IEEE Trans. Rob.* **20**(4), 713–723 (2004). <https://doi.org/10.1109/TRO.2004.829463>
- Tahri, O., Chaumette, F.: Point-based and region-based image moments for visual servoing of planar objects. *IEEE Trans. Robot.* **21**(6), 1116–1127 (2005). <https://doi.org/10.1109/TRO.2005.853500>
- Jabbari, H., Oriolo, G., Bolandi, H.: An adaptive scheme for image-based visual servoing of an underactuated uav. *Int. J. Robot. Autom.* **29**(1), 92–104 (2014)
- Xie, H., Lynch, A.F.: State transformation-based dynamic visual servoing for an unmanned aerial vehicle. *Int. J. Control* **89**(5), 892–908 (2016)
- Asl, H.J., Yoon, J.: Adaptive vision-based control of an unmanned aerial vehicle without linear velocity measurements. *ISA Trans.* **65**, 296–306 (2016)

17. Zheng, D., Wang, H., Wang, J., Chen, S., Chen, W., Liang, X.: Image-based visual servoing of a quadrotor using virtual camera approach. *IEEE/ASME Trans. Mechatron.* **22**(2), 972–982 (2016)
18. Shirzadeh, M., Amirkhani, A., Jalali, A., Mosavi, M.R.: An indirect adaptive neural control of a visual-based quadrotor robot for pursuing a moving target. *ISA Trans.* **59**, 290–302 (2015)
19. Shirzadeh, M., Asl, H.J., Amirkhani, A., Jalali, A.A.: Vision-based control of a quadrotor utilizing artificial neural networks for tracking of moving targets. *Eng. Appl. Artif. Intell.* **58**, 34–48 (2017)
20. Liu, N., Shao, X.: Desired compensation rise-based ibvs control of quadrotors for tracking a moving target. *Nonlinear Dyn.* **95**(4), 2605–2624 (2019)
21. Cao, Z., Chen, X., Yu, Y., Yu, J., Liu, X., Zhou, C., Tan, M.: Image dynamics-based visual servoing for quadrotors tracking a target with a nonlinear trajectory observer. *IEEE Trans. Syst. Man Cybern. Syst.* **50**(1), 376–384 (2017)
22. Arif, A., Wang, H., Liu, Z., Castañeda, H., Wang, Y.: Adaptive visual servo control law for finite-time tracking to land quadrotor on moving platform using virtual reticle algorithm. *Robot. Auton. Syst.* **141**, 103764 (2021)
23. Alexis, K., Nikolakopoulos, G., Tzes, A.: Experimental constrained optimal attitude control of a quadrotor subject to wind disturbances. *Int. J. Control Autom. Syst.* **12**(6), 1289–1302 (2014)
24. Tian, B., Liu, L., Lu, H., Zuo, Z., Zong, Q., Zhang, Y.: Multi-variable finite time attitude control for quadrotor uav: theory and experimentation. *IEEE Trans. Industr. Electron.* **65**(3), 2567–2577 (2017)
25. Harshavarthini, S., Sakthivel, R., Ahn, C.K.: Finite-time reliable attitude tracking control design for nonlinear quadrotor model with actuator faults. *Nonlinear Dyn.* **96**(4), 2681–2692 (2019)
26. Gajbhiye, S., Cabecinhas, D., Silvestre, C., Cunha, R.: Geometric finite-time inner-outer loop trajectory tracking control strategy for quadrotor slung-load transportation. *Nonlinear Dyn.* **107**(3), 2291–2308 (2022)
27. Zhu, W., Du, H., Cheng, Y., Chu, Z.: Hovering control for quadrotor aircraft based on finite-time control algorithm. *Nonlinear Dyn.* **88**(4), 2359–2369 (2017)
28. Zhao, G., Chen, G., Chen, J., Hua, C.: Finite-time control for image-based visual servoing of a quadrotor using non-singular fast terminal sliding mode. *Int. J. Control Autom. Syst.* **18**(9), 2337–2348 (2020)
29. Cabecinhas, D., Cunha, R., Silvestre, C.: A globally stabilizing path following controller for rotorcraft with wind disturbance rejection. *IEEE Trans. Control Syst. Technol.* **23**(2), 708–714 (2014)
30. Islam, S., Liu, P.X., El Saddik, A.: Robust control of four-rotor unmanned aerial vehicle with disturbance uncertainty. *IEEE Trans. Industr. Electron.* **62**(3), 1563–1571 (2014)
31. Amirkhani, A., Shirzadeh, M., Papageorgiou, E.I., Mosavi, M.R.: Visual-based quadrotor control by means of fuzzy cognitive maps. *ISA Trans.* **60**, 128–142 (2016)
32. Bhat, S.P., Bernstein, D.S.: Finite-time stability of continuous autonomous systems. *SIAM J. Control. Optim.* **38**(3), 751–766 (2000)
33. Huang, X., Lin, W., Yang, B.: Global finite-time stabilization of a class of uncertain nonlinear systems. *Automatica* **41**(5), 881–888 (2005)
34. Qian, C., Lin, W.: Non-lipschitz continuous stabilizers for nonlinear systems with uncontrollable unstable linearization. *Syst. Control Lett.* **42**(3), 185–200 (2001)
35. Han, J., Wang, W.: Nonlinear tracking-differentiator (in chinese). *J. Syst. Sci. Math. Sci.* **14**(2), 177–183 (1994)
36. Na, J., Ren, X., Herrmann, G., Qiao, Z.: Adaptive neural dynamic surface control for servo systems with unknown dead-zone. *Control. Eng. Pract.* **19**(11), 1328–1343 (2011)
37. Chen, Q., Ren, X., Na, J., Zheng, D.: Adaptive robust finite-time neural control of uncertain pmsm servo system with nonlinear dead zone. *Neural Comput. Appl.* **28**(12), 3725–3736 (2017)
38. Amovlab: Prometheus autonomous UAV opensource project. <https://github.com/amov-lab/Prometheus> (2019)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.