IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 14, 2019, accepted May 28, 2019, date of publication June 4, 2019, date of current version June 20, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2920716

Vision-Based Adaptive Neural Positioning Control of Quadrotor Aerial Robot

YI LYU[1,2], GUANYU LAI[2], CI CHEN[2,3], AND YUN ZHANG[2]

[1]School of Computer, University of Electronic Science and Technology of China Zhongshan Institute, Zhongshan 528400, China
[2]School of Automation, Guangdong University of Technology, Guangzhou 510006, China
[3]School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798

Corresponding author: Yi Lyu (yi.lyu@foxmail.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61803090 and Grant 61802070, in part by the China Postdoctoral Science Foundation under Grant 2017M620363 and Grant 2018T110850, and in part by the Science and Technology Program of Guangdong under Grant 2017A010101004.

ABSTRACT In this paper, a new vision-based adaptive control algorithm is proposed for the positioning of a quadrotor aerial robot (QAR) with an onboard pin-hole camera. First, the transformation between the position tracking error and image projection error is constructed through the spherical projection method, and then the regulation of the position error is achieved indirectly by stabilizing the image projection error. To overcome the challenge that the dynamics of QAR is physically underactuated, a backstepping-based approach that synthesizes the Lipschitz condition and natural saturation of the inverse tangent function is proposed. In the proposed adaptive controller, an optimized adaptive neural network (NN) means is designed, where only the square of the NN weight matrix's maximum singular value, not the weight matrix itself, is estimated. Moreover, to facilitate practical application, a novel inertial matrix estimator is introduced in the tuning laws, so that the accurate QAR rotation inertial information is not required. By Lyapunov theory, it is proved that the image projection error converges to an adjustable region of zero asymptotically. The effectiveness of the proposed algorithm has been confirmed by the experimental results.

INDEX TERMS Adaptive control, neural networks, underactuation, position tracking, quadrotor aerial robots, nonlinear systems.

I. INTRODUCTION

Over the last few decades, visual servoing which aims to control the motion of robots through visual feedback signals has been an active research topic (such as [1]–[7]). In general, visual servoing can be classified into two types, namely, position-based visual servoing (PBVS) [1]–[3] and image-based visual servoing (IBVS) [4]–[9]. The PBVS method involves the reconstruction of target pose and robot motion in a Cartesian space. Hence its control effect depends heavily on the 3D model accuracy in world-space coordinates and is relatively sensitive to the camera parameter biases [7]. Meanwhile, the IBVS method, which solves the servoing problem by directly controlling the position error in the image plan, is more robust to disturbances and parameter biases than PBVS. Weiss *et al.* [5] and Espiau *et al.* [4] take visual system as one special sensor, and introduced the image signal into classical feedback control structures to perform robot visual servoing. In [6], vision was considered in early modeling stage, and a novel vision-based kinematic modeling was proposed.

In classical IBVS frameworks, the depth information of each feature point is required to be estimated from visual signals, which is generally difficult or costly in practical implementation. To avoid this problem, various solutions have been proposed. In [10], a depth-independent interaction matrix was proposed to estimate the unknown camera parameters, and an adaptive control method was presented to minimize the actual and estimated projection errors of the feature points. Kai *et al.* [11] proposed an adaptive estimator to estimate the robot position online and then controlled a nonholonomic mobile robot to perform position tracking. Wang *et al.* [12] assumed that the camera parameters are not calibrated and use the pseudoinverse of the depth-independent interaction matrix to map the image errors onto the joint space of the manipulator.

Note that these studies mainly deal with the visual servoing of *fully-actuated* systems. However, quadrotor aerial robot (QAR) is a typical underactuated system, and thus

The associate editor coordinating the review of this manuscript and approving it for publication was Lifeng Ma.

may not be effectively handled by these previous approaches. To cope with the underactuated characteristic of the aerial robot, earlier researches were concentrated on the state-feedback strategy, which implies that the position and orientation information are required to be measured accurately as a *prior* knowledge for the development of the controllers, as shown in [13]–[19]. Despite tremendous efforts, the high-accuracy localization of the mobile robots remains one most challenging problem in robotics.

The visual servoing technique provides an effective framework in controlling the aerial mobile robots to further eliminate the requirement for the global position measurement and improve the localization accuracy, and some pioneering works have been reported in [20]–[26]. Specifically, Hamel and Mahony [20] proposed an IBVS algorithm for a class of underactuated rigid body systems by exploiting the structural passivity-like property, and the developed closed-loop system exhibits the robustness to the calibrated errors of the camera and target. As an extended research, Guenard *et al.* [24] further modified the visual error term used in their previous work [20] to compensate for the poor condition of the image Jacobian matrix so that the convergence performance of the closed-loop system is improved. Additional visual control algorithms similar to the above are shown in [21]–[23]. Subsequently, Carrillo *et al.* [25] presented a comparison of three control techniques: nested saturations, backstepping, and sliding modes for unmanned aerial vehicles (UAVs) when using visual feedback. In [26], a vision-based method for multi-UAV position estimation was presented, and its effectiveness was well validated by experiments. It is noticed, however, that these developed visual controllers commonly require explicit inertia information which is often difficult to obtain. Some inertia identification scenarios have been raised in [27]–[30]. However, most these methods are established on a simplified model of underactuated aerial robot [31], and none of them take the vision systems into account.

In this paper, we propose a vision-based adaptive control algorithm for positioning control of QAR systems. The main contributions are summarized as follows:

- **Without needing position measurement and suitable to indoor environment:** With the use of spherical projection, we construct an unnormalized spherical centroid, and establish its one-to-one relationship with the position of QAR system (see Theorem 1), so that the positioning control problem is well transformed into the image tracking control problem. In this sense, the proposed control algorithm does not need the accurate position information of quadrotor system, and hence is applicable to the indoor environment in which GPS devices generally cannot work well.
- **Equipping with an efficient adaptive neural-network system:** The proposed positioning controller contains an adaptive neural-network system to handle model uncertainties of quadrotor aircraft. Such a system is rather efficient as it only involves two adaptive parameters, and thus is computationally attractive.

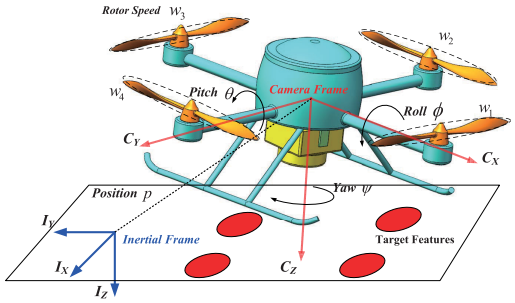

FIGURE 1. Coordinate frames of QAR with a pin-hole camera.

- **Without needing inertial parameters of quadrotor aircraft:** Most existing control algorithms for positioning of quadrotor aircraft require a completely accurate inertial matrix which may not be accessible for measurement in practice. To overcome this problem, the Cholesky decomposition method is used to construct an online estimator which can adaptively estimate the unknown inertial matrix, and hence our proposed positioning control algorithm is more feasible in practice.

The rest of this paper are arranged into four sections. Some preliminaries about the dynamic model of QAR and NNs are presented in Section II. In Section III, the adaptive IBVS control algorithm is designed and the corresponding stability verification is shown. Two simulated experiments are conducted in Section IV. The conclusion is given in Section V.

II. PRELIMINARIES OF DYNAMIC MODEL AND RADIAL BASIS FUNCTION NNS

An eye-in-hand setup visually depicted in Fig. 1, in which a camera is mounted on the body of the QAR to observe a number of feature points marked on the ground, is considered. Suppose that the camera used is a pin-hole camera with the perspective projection, and its optical point is located at the center of mass of the QAR; thus, the camera frame $\mathbb{C}$ is consistent with the body-fixed frame $\mathbb{B}$. In addition, the inertial reference frame is denoted by $\mathbb{I}$.

A. NOTATIONS

In this section, some notations throughout the literature are predefined. The notation " $\times$ " denotes the cross-product operator, and $sk(\cdot)$ represents the skew-symmetric matrix with $sk(a)b = a \times b$ for $a, b \in \mathbb{R}^3$ [15]. The representations $\lambda_{\min}(\cdot)$ and $\lambda_{\max}(\cdot)$ are the maximum eigenvalue and minimum eigenvalue, respectively. In addition, $||\cdot||$ denotes the Euclidean norm, and the denotation $||\cdot||_{\mathbf{F}}$ represents the Frobenius norm of matrix. Given a vector $\mathbf{v}(t) \in \mathbb{R}^3$, we additionally define $\tanh(\mathbf{v}) = (\tanh \mathbf{v}_1, \tanh \mathbf{v}_2, \tanh \mathbf{v}_3)^T \in \mathbb{R}^3$. For the defined error $\Delta\mathbf{v}$, its time differentiation is written as $\Delta\dot{\mathbf{v}}$, i.e., $\Delta\dot{\mathbf{v}} = d(\Delta\mathbf{v})/dt$. The notation "||" is used to denote "parallel" of two straight lines, e.g., $OA||OB$ means that the straight line OA is parallel to the straight line OB.

$$\mathbf{R}(\boldsymbol{\Theta}) = \mathbf{R} = \begin{pmatrix} \cos\psi\cos\theta & \cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi & \cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi \\ \sin\psi\cos\theta & \sin\psi\sin\theta\sin\phi + \cos\psi\cos\phi & \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{pmatrix} \tag{1}$$

B. DYNAMICS

As shown in Fig. 1, $\mathbb{I} = \{\mathbf{I}_x, \mathbf{I}_y, \mathbf{I}_z\}$ denotes the right-hand inertial frame, and $\mathbb{C} = \{\mathbf{C}_x, \mathbf{C}_y, \mathbf{C}_z\}$ denotes the right-hand camera frame which is consistent with QAR body frame. As argued in [14], [18], [19], [32], there exists a rotation matrix $\mathbf{R}(\Theta) \in \mathbf{SO}_3$, specifically given in (1), as shown at the top of this page, which can be used to fulfill the transformation from Frame $\mathbb{C}$ to Frame $\mathbb{I}$, where $\boldsymbol{\Theta} = (\phi, \theta, \psi)^T$ is QAR's body attitudes vector, whose members are the angle of roll, pitch, and yaw, respectively.

Let $\mathbf{p} \in \mathbb{R}^3$ denotes the position of QAR's centroid in the Frame $\mathbb{I}$; $\mathbf{V} \in \mathbb{I}$ represents the line velocity; and $\boldsymbol{\Omega} \in \mathbb{C}$ represents the angular velocity expressed in Frame $\mathbb{C}$. Then according to Newton's moving theory, QAR's dynamic model is

$$\begin{cases} \dot{\mathbf{p}} = \mathbf{R}(\boldsymbol{\Theta})\mathbf{V} \\ \dot{\boldsymbol{\Theta}} = \mathbf{M}(\boldsymbol{\Theta})\boldsymbol{\Omega} \\ \dot{\mathbf{R}}(\boldsymbol{\Theta}) = \mathbf{R}(\boldsymbol{\Theta})sk(\boldsymbol{\Omega}) \\ \mathbf{m}\dot{\mathbf{V}} = -\mathbf{m}\boldsymbol{\Omega} \times \mathbf{V} + \mathbf{T}\mathscr{C}_z + \mathbf{m}\mathbf{g}\mathbf{R}^T(\boldsymbol{\Theta})\mathscr{C}_z \\ \mathbf{J}\dot{\boldsymbol{\Omega}} = -\boldsymbol{\Omega} \times (\mathbf{J}\Omega) + \mathbf{F} \end{cases} \tag{2}$$

where $\mathbf{m}$ is the mass of QAR; $\mathscr{C}_z = (0, 0, 1)^T$; $\mathbf{J} \in \mathbb{R}^{3\times3}$ denotes the QAR's constant inertial matrix around its center of mass; $\mathbf{T} \in \mathbb{R}$ is the exogenous force along z-direction; $\mathbf{F} \in \mathbb{R}^3$ denotes the rotational torques in Frame $\mathbb{C}$. $\mathbf{F}$ and $\mathbf{T}$, provided by the QAR motors, are regarded as the control input variables of the adaptive controller in this paper. $\mathbf{M}(\boldsymbol{\Theta})$ is the transformation matrix of rotation velocity between Frame $\mathbb{C}$ and Frame $\mathbb{I}$ [15] given by

$$\mathbf{M}(\boldsymbol{\Theta}) = \begin{pmatrix} -\sin\theta & 0 & 1 \\ \cos\theta\sin\phi & \cos\phi & 0 \\ \cos\theta\cos\phi & -\sin\phi & 0 \end{pmatrix}^{-1}.$$

C. NEURAL NETWORKS APPROXIMATOR

In the intelligent control field, it is well known that the radial basis function neural network (RBFNN), as one of the universal approximators, has been frequently adopted to model the system uncertainties [32]–[38]. Specifically, let $\mathscr{F}(\mathbf{x}) : \mathbf{X} \to \mathbb{R}^n$ be the uncertain dynamics defined on a compact set $\mathbf{X} \subseteq \mathbb{R}^q$ and given any accuracy level ϵ, there exists a RBFNN $\Psi^T\mathbf{Y}(\mathbf{x})$ such that

$$\mathscr{F}(\mathbf{x}) = \Psi^T\mathbf{Y}(\mathbf{x}) + \delta(\mathbf{x}), \quad ||\delta(\mathbf{x})|| \leq \epsilon \tag{3}$$

where $\delta(\mathbf{X}) \in \mathbb{R}^n$ is the according approximation error which is bounded by the prescribed precision ϵ. In (3), the notation $\Psi \in \mathbb{R}^{N\times n}$ represents an idealized weight matrix, and $\mathbf{Y}(\mathbf{x}) = (\mathbf{Y}_1(\mathbf{x}), \ldots, \mathbf{Y}_N(\mathbf{x}))^T \in \mathbb{R}^N$ is the Gaussian radial basis function vector whose ith componentwise $\mathbf{Y}_i(\mathbf{x})$ commonly takes the exponential function

$$\mathbf{Y}_i(\mathbf{x}) = \exp\left[-\frac{(\mathbf{x} - \mathbf{x}_i^0)^T(\mathbf{x} - \mathbf{x}_i^0)}{\eta_i^2}\right], \quad i = 1, \ldots, N \tag{4}$$

where $\mathbf{x}_i^0 = (\mathbf{x}_{i1}^0, \ldots, \mathbf{x}_{iq}^0)^T \in \mathbb{R}^q$ and $\eta_i \in \mathbb{R}$ are the center and the width of the receptive field, respectively. Importantly note that the Gaussian basis function vector satisfies the following condition

$$||\mathbf{Y}(\mathbf{x})|| \leq \sqrt{N} \tag{5}$$

Next we introduce a basic definition which will be used later on.

Definition of Semi-Globally Uniformly Ultimately Bounded (SGUUB): The time-varying vector $\mathbf{x}(t) \in \mathbb{R}^q$ is SGUUB if for any compact set $\mathbf{X} \subseteq \mathbb{R}^q$ and all $\mathbf{x}(t_0) \in \mathbf{X}$, there exist an $\epsilon > 0$ and a time constant $T(\epsilon, \mathbf{x}(t_0))$ such that $||\mathbf{x}(t)|| < \epsilon$ for all $t > t_0 + T$, where t_0 is an initial time.

Remark 1: In the adaptive intelligent control field, it has been popularly assumed, e.g., [31], [33]–[38], that the approximator inputs always remain in a compact set due to two reasons. First, the size of such compact set mainly depends on the distribution of the centers and widths of Gaussian basis functions, and the number of neural network nodes, which implies that it is completely possible to construct a RBFNN to hold the approximator inputs. From the control engineering point of view, the state variables in physical systems and devices (e.g., the inverted pendulum [38], the active magnetic bearing [39] and the robot manipulator [10]–[12]) must be bounded due to the hard constraint. Consequently, such assumption is feasible in practical application although the result finally obtained is often SGUUB [19] and [36]. •

III. ADAPTIVE VISUAL CONTROL OF UNDERACTUATED QAR WITH UNKNOWN INERTIA

A. VISION SYSTEM BASED ON SPHERICAL PROJECTION

The image geometry of the QAR vision is of primary concern in this section. Here, we use a spherical projection method to build the equivalent transformation between the position error and the projected image error, as presented visually in Fig. 2. We use a constant vector $\bar{\mathbf{S}}_i$ to denote the coordinate of the ith target point in Frame $\mathbb{I}$, where $i = 1, \ldots, n$ and n is the number of target features. Using the formula (2), one obtains

$$\mathbf{S}_i = \mathbf{R}^T(\boldsymbol{\Theta})(\bar{\mathbf{S}}_i - \mathbf{p}) \tag{6}$$

where $\mathbf{S}_i = (X_i, Y_i, Z_i)^T \in \mathbb{C}$ represents the corresponding coordinate of the ith target point in Frame $\mathbb{C}$. So its perspective projection on the image plane is further derived as

$$\mathbf{s}_i^p = (u_i, v_i, f)^T = (fX_i/Z_i, fY_i/Z_i, f)^T \tag{7}$$

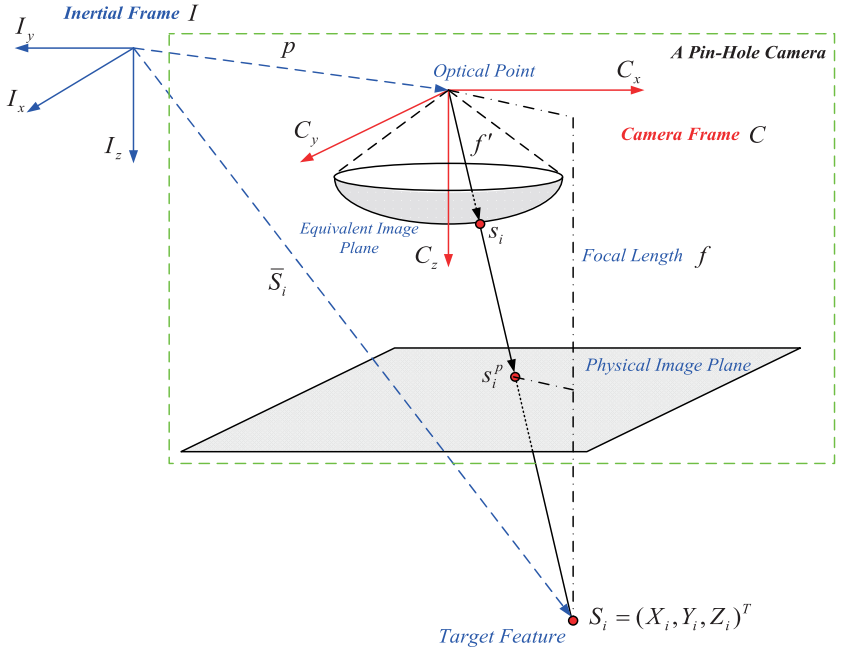

FIGURE 2. Equivalent spherical image geometry of a pin-hole camera.

where f is the focal length of the used pin-hole camera, and $(u_i, v_i, f)^T$ is a perspective projection which is considered to be measurable using the basic image detection techniques, i.e., $\mathbf{s}_i^p$ is a measurable vector. Subsequently, we project the target features to the spherical image plane, and the corresponding transformation is

$$\mathbf{s}_i = \frac{\mathbf{s}_i^p}{||\mathbf{s}_i^p||} = \frac{1}{r_i(\mathbf{S}_i)}\mathbf{S}_i \tag{8}$$

where $r_i(\mathbf{S}_i) = ||\mathbf{S}_i||/f'$ is the relative depth with respect to an unit focal length f'. Note that once $\mathbf{s}_i^p$ is available, then the above equivalent spherical projection $\mathbf{s}_i$ can be readily obtained using (8). Moreover, we can further derive the time differentiation of $\mathbf{s}_i$ as

$$\begin{aligned}\dot{\mathbf{s}}_i=&\frac{\dot{\mathbf{S}}_i r_i(\mathbf{S}_i) - \mathbf{S}_i \dot{r}_i(\mathbf{S}_i)}{r_i^2(\mathbf{S}_i)}\\ =&-\boldsymbol{\Omega}\times\mathbf{s}_i - \frac{1}{r_i(\mathbf{S}_i)}\mathbf{V} + \frac{1}{r_i(\mathbf{S}_i)}\mathbf{s}_i^T\mathbf{V}\mathbf{s}_i\\ =&-\boldsymbol{\Omega}\times\mathbf{s}_i - \frac{1}{r_i(\mathbf{S}_i)}\left(\mathbf{I}_3 - \mathbf{s}_i\mathbf{s}_i^T\right)\mathbf{V}\end{aligned} \tag{9}$$

where $\mathbf{I}_3 \in \mathbb{R}^{3\times 3}$ denotes an identity matrix. Let $\mathscr{H}_i = \mathbf{I}_3 - \mathbf{s}_i\mathbf{s}_i^T$, then (9) can be further formulated as

$$\dot{\mathbf{s}}_i = -\boldsymbol{\Omega}\times\mathbf{s}_i - \frac{\mathscr{H}_i}{r_i(\mathbf{S}_i)}\mathbf{V} \tag{10}$$

Then we employ an unnormalized spherical centroid (USC) to build the relationship between the image projection and the QAR position, and give its definition as follows

$$\mathbf{s} = \sum_{i=1}^{n}\mathbf{s}_i \tag{11}$$

Through differentiating the USC of all target points with respect time t, one obtains

$$\begin{aligned}\dot{\mathbf{s}}=&-\boldsymbol{\Omega}\times\mathbf{s} - \left(\sum_{i=1}^{n}\frac{\mathscr{H}_i}{r(\mathbf{S}_i)}\right)\mathbf{V}\\ =&-\boldsymbol{\Omega}\times\mathbf{s} - \mathbf{Q}(\mathbf{S}_1,\ldots,\mathbf{S}_n)\mathbf{V}\end{aligned} \tag{12}$$

Note that $\mathbf{Q}(\mathbf{S}_1,\ldots,\mathbf{S}_n)$ is an image Jacobian matrix. According to the definition of $\mathscr{H}_i$, It is readily seen that $\mathbf{Q}(\mathbf{S}_1,\ldots,\mathbf{S}_n)$ is a symmetric matrix. Furthermore, as is pointed out in [24], when there are more than two effective target features (i.e., $n \geq 2$), $\mathbf{Q}(\mathbf{S}_1,\ldots,\mathbf{S}_n)$ becomes a positive definite matrix.

Note that the main task of this paper is to control the QAR moving to the desired position, that means to make $\mathbf{p}(t)$ asymptotically converge to $\mathbf{p}^*$ with the visual feedback. Due to there is no position information, the transformation relationship between the image projection error and position tracking error should be initially established. Then the positioning task can be fulfilled by controlling the image error's convergence to zero. Subsequently, a theorem is provided to build the one-to-one relationship.

Theorem 1: Assuming that the onboard camera can observe at least two target features. Let $\mathbf{p}^*(t)$ represents the target location of QAR, and $\mathscr{Q}^*$ denotes the corresponding USC in Frame $\mathbb{I}$. Then if $\mathbf{R}(\boldsymbol{\Theta})\mathbf{s}$ is asymptotically convergent to $\mathscr{Q}^*$, then the position $\mathbf{p}(t) \to \mathbf{p}^*$, and vice versa.

Proof: If the position $\mathbf{p}$ of QAR is convergent to the point $\mathbf{p}^*$, it is clearly obtained from the definition of $\mathscr{Q}^*$ that $\mathbf{R}(\boldsymbol{\Theta})\mathbf{s} \in \mathbb{I} \to \mathscr{Q}^*$. Subsequently, we intensively discuss its converse, i.e, is there only one position $\mathbf{p}^*$ corresponding to

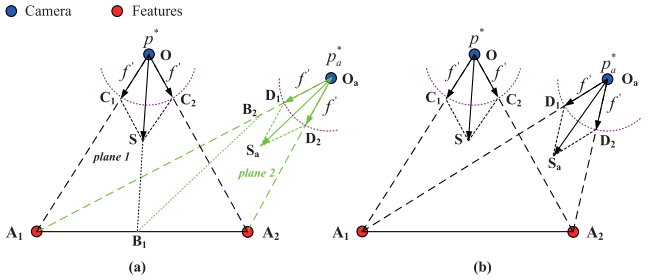

FIGURE 3. Two possible locations of $\mathbf{p}_a^*$: Non-coplanar (a) and coplanar (b).

the desired USC $\mathscr{Q}^*$. To respond this problem, the following two cases should be taken into full consideration.

Case 1: Assume that there exists another position $\mathbf{p}_a^*$ such that $\mathbf{R}(\boldsymbol{\Theta})\mathbf{s} = \mathscr{Q}^*$, and $\mathbf{p}_a^*$ is not in the plane constructed by two features $\mathbf{A}_1$, $\mathbf{A}_2$, and $\mathbf{p}^*$, see in Fig. 3(a). With this assumption, one gets $\mathbf{OS} \parallel \mathbf{OS}_a$. Note that the extension of the vector $\mathbf{OS}$ and the line $\mathbf{A}_1\mathbf{A}_2$ intersect at the point $\mathbf{B}_1$ on the plane 1. In addition, on the plane 2, there must exist a line, e.g, $\mathbf{B}_1\mathbf{B}_2$ which is parallel with $\mathbf{OS}_a$, i.e., $\mathbf{B}_1\mathbf{B}_2 \parallel \mathbf{O}_a\mathbf{S}_a$. Then a contradiction result is derived as

$$\mathbf{B}_1\mathbf{B}_2 \parallel \mathbf{OB}_1 \tag{13}$$

which implies that the position $\mathbf{p}_a^*$ must be in the plane constructed by $\mathbf{OS}$ and $\mathbf{A}_1\mathbf{A}_2$, i.e., the plane 1 in Fig. 3(a).

Case 2: According to the discussion of above Case 1, it is further assumed that there exists a position $\mathbf{p}_a^*$ which is coplanar with $\mathbf{p}^*$ such that $\mathbf{R}(\boldsymbol{\Theta})\mathbf{s} = \mathscr{Q}^*$, see in Fig. 3(b). In other words, we obtain the result $\mathbf{OS} = \mathbf{O}_a\mathbf{S}_a$. In addition, it is noticeable from above Fig. 3(b) that

$$||\mathbf{OC}_1|| = ||\mathbf{OC}_2|| = ||\mathbf{O}_a\mathbf{D}_1|| = ||\mathbf{O}_a\mathbf{D}_2|| \tag{14}$$

So it will also be derived a contradiction conclusion that

$$\mathbf{O}_a\mathbf{A}_1 \parallel \mathbf{OA}_1 \quad \text{and } \mathbf{O}_a\mathbf{A}_2 \parallel \mathbf{OA}_2 \tag{15}$$

Based on the detailed discussion of above two cases, it can be concluded that the possible point $\mathbf{p}_a^*$ must coincide with the desired point $\mathbf{p}^*$, which sufficiently verifies the reasonability of Theorem 1. ■

Remark 2: Note that it was commonly assumed, in most of the traditional position regulation algorithms for UAVs, e.g., [13]–[19], that the position information has to be available for measurement. To relax this rigorous condition, a new adaptive control algorithm based on above vision system is raised in this paper. It is observed from the one-by-one correspondence established above that the regulation of the position error $\Delta\mathbf{p}$ can be achieved indirectly by stabilizing visual error so that the realtime position measurement is not required by the developed adaptive controller. •

B. VISUAL-BASED ADAPTIVE POSITION TRACKING CONTROL

The main task of this section is to design a new adaptive controller to guarantee $\mathbf{s} \to \mathbf{R}^T(\boldsymbol{\Theta})\mathscr{Q}*$. Then the visual error criterion is defined as

$$\Delta\mathbf{s}(t) = \mathbf{s} - \mathbf{R}^T(\boldsymbol{\Theta})\mathscr{Q}* \tag{16}$$

By further implementing the time differentiation of the visual error $\Delta\mathbf{s}$, it gets

$$\Delta\dot{\mathbf{s}}(t) = -\boldsymbol{\Omega} \times \Delta\mathbf{s} - \mathbf{Q}(\mathbf{S}_1, \ldots, \mathbf{S}_n)\mathbf{V} \tag{17}$$

Until now, the position tracking control problem has been transformed into the stabilization problem of the image error $\Delta\mathbf{s}$. Synthetically analyze above (17) and the earlier (2), we propose an adaptive controller, and the detailed design procedure will be discussed in subsequent analysis. First, we use the translational velocity $\mathbf{V}$ to stabilize the image error $\Delta\mathbf{s}$ in (17). Denote the error between the real line velocity $\mathbf{V}$ and the virtual one $\mathbf{V}^*$ by

$$\Delta\mathbf{V} = \frac{1}{\mathbf{k}_1}\left(\mathbf{V} - \mathbf{V}^*\right) \tag{18}$$

where $\mathbf{k}_1$ is a scalar factor. Substituting above (18) into (17) to get

$$\Delta\dot{\mathbf{s}} = -\boldsymbol{\Omega} \times \Delta\mathbf{s} - \mathbf{Q}(\mathbf{S}_1, \ldots, \mathbf{S}_n)\left(\mathbf{k}_1\Delta\mathbf{V} + \mathbf{V}^*\right) \tag{19}$$

Design the virtual control velocity $\mathbf{V}^*$ as

$$\mathbf{V}^* = \mathbf{k}_1\Delta\mathbf{s} \tag{20}$$

Thus, we further obtain

$$\begin{aligned}\Delta\dot{\mathbf{s}} =& -\boldsymbol{\Omega} \times \Delta\mathbf{s} - \mathbf{k}_1\mathbf{Q}(\mathbf{S}_1, \ldots, \mathbf{S}_n)\Delta\mathbf{V} \\ =& -\mathbf{k}_1\mathbf{Q}(\mathbf{S}_1, \ldots, \mathbf{S}_n)\Delta\mathbf{s}\end{aligned} \tag{21}$$

Subsequently, it is necessary to stabilize the velocity error $\Delta\mathbf{V}$ so that the image error $\Delta\mathbf{s}$ can be convergent to zero. By analyzing its change rate, and recalling the 4th sub-equation in (2), it can be derived that

$$\begin{aligned}\Delta\dot{\mathbf{V}} =& \frac{1}{\mathbf{k}_1}\left(\dot{\mathbf{V}} - \dot{\mathbf{V}}^*\right) \\ =& -\frac{1}{\mathbf{k}_1}\boldsymbol{\Omega} \times \mathbf{V} + \mathbf{g}\mathbf{R}^T(\boldsymbol{\Theta})\mathscr{C}_z + \frac{1}{\mathbf{m}}\mathbf{T}\mathscr{C}_z - \Delta\dot{\mathbf{s}} \\ =& -\boldsymbol{\Omega} \times \Delta\mathbf{V} + \mathbf{g}\mathbf{R}^T(\boldsymbol{\Theta})\mathscr{C}_z + \frac{1}{\mathbf{m}}\mathbf{T}\mathscr{C}_z \\ &+ \mathbf{k}_1\mathbf{Q}(\mathbf{S}_1, \ldots, \mathbf{S}_n)\Delta\mathbf{V} + \mathbf{k}_1\mathbf{Q}(\mathbf{S}_1, \ldots, \mathbf{S}_n)\Delta\mathbf{s}\end{aligned} \tag{22}$$

Remark 3: It is important to note that above error dynamics is underactuated since $\mathscr{C}_z$ is a unit vector, which implies that it is physically impossible to directly stabilize it only using the thrust $\mathbf{T} \in \mathbb{R}$. From control theory point of view, such error system can be regarded as an extended case of the strict-feedback nonlinear system previously investigated by [39]–[43], etc. For this reason, the backstepping technique is introduced from these theoretical works to address an engineering problem, i.e., the underactuation problem. •

Remark 4: Different from the standard strict-feedback nonlinear system, the attitude information of the error system (22) is wrapped in a rotational matrix $\mathbf{R}(\boldsymbol{\Theta})$, and such coupling phenomenon takes a challenging difficulty in the development of the backstepping-based controller. To overcome it,

an approach that synthesizes the Lipschitz condition and the natural saturation property of the inverse tangent function is proposed and will be discussed hereinafter. •

Note that the translational velocity error system described in (22) is underactuated so that it is required to exploit the attitude information to stabilize such error system. Afterward, a virtual attitude controller is defined as $\boldsymbol{\Theta}^* = \boldsymbol{\Theta} - \Delta\boldsymbol{\Theta}$. Without loss of generality, the desired yaw angle is selected as $\boldsymbol{\Theta}_3^* = 0$ in the control design. Then the above (22) is further analyzed as

$$\begin{aligned}\Delta\dot{\mathbf{V}}=&-\boldsymbol{\Omega}\times\Delta\mathbf{V}+\mathbf{g}\Big(\mathbf{R}^T(\boldsymbol{\Theta})-\mathbf{R}^T(\boldsymbol{\Theta}^*)\Big)\mathscr{C}_z\\&+\mathbf{k}_2\begin{pmatrix}\cos\theta^*\Delta\mathbf{V}_1\\ \cos\phi^*\Delta\mathbf{V}_2\\ \Delta\mathbf{V}_3\end{pmatrix}+\mathbf{k}_1\mathbf{Q}(\mathbf{S}_1,\ldots,\mathbf{S}_n)(\Delta\mathbf{V}+\Delta\mathbf{s})\\&+\mathbf{g}\begin{pmatrix}-\sin\theta^*\\ \sin\phi^*\cos\theta^*\\ \cos\phi^*\cos\theta^*\end{pmatrix}+\frac{1}{\mathbf{m}}\begin{pmatrix}0\\0\\\mathbf{T}\end{pmatrix}-\mathbf{K}_2\Delta\mathbf{V}\end{aligned}\tag{23}$$

where $\mathbf{K}_2 = \mathrm{diag}(\mathbf{k}_2\cos\theta^*, \mathbf{k}_2\cos\phi^*, \mathbf{k}_2) \in \mathbb{R}^{3\times3}$, and $\mathbf{k}_2$ is a scalar factor. The virtual controllers θ^* and ϕ^* must be constrained in $(-\pi/2, \pi/2)$ due to the natural saturation constraint of the tangent function, which infers that $\mathbf{K}_2$ is a symmetric positive-definite matrix. And then, the intermediate controller $\boldsymbol{\Theta}^*$ is designed as follows

$$\boldsymbol{\Theta}^* = (\phi^*, \theta^*, \psi^*)^T\tag{24}$$

where $\psi^* = 0$, $\theta^* = \arctan(\mathbf{k}_1\mathbf{k}_2\Delta\mathbf{V}_1/\mathbf{g})$, and $\phi^* = -\arctan(\mathbf{k}_1\mathbf{k}_2\Delta\mathbf{V}_2/\mathbf{g}\cos\theta^*)$. In addition, the desired thrust $\mathbf{T} \in \mathbb{R}$ is subsequently constructed as

$$\mathbf{T} = \mathbf{m}\Big(-\mathbf{g}\cos\phi^*\cos\theta^* - \mathbf{k}_2\Delta\mathbf{V}_3\Big)\tag{25}$$

Then (23) can be reformulated as

$$\begin{aligned}\Delta\dot{\mathbf{V}} =& -\boldsymbol{\Omega}\times\Delta\mathbf{V}+\mathbf{g}\Big(\mathbf{R}^T(\boldsymbol{\Theta})-\mathbf{R}^T(\boldsymbol{\Theta}^*)\Big)\mathscr{C}_z\\&-\mathbf{K}_2\Delta\mathbf{V}+\mathbf{k}_1\mathbf{Q}(\mathbf{S}_1,\ldots,\mathbf{S}_n)(\Delta\mathbf{V}+\Delta\mathbf{s})\end{aligned}\tag{26}$$

It is noticeable from above (26) that the stability of the velocity error system $\Delta\mathbf{V}$ mainly depends on the convergence of the attitude error $\Delta\boldsymbol{\Theta}$. Once the condition $\boldsymbol{\Theta} \to \boldsymbol{\Theta}^*$ is satisfied, it can be derived that $\mathbf{R}^T(\boldsymbol{\Theta}) \to \mathbf{R}^T(\boldsymbol{\Theta}^*)$ so that the convergence of the translational velocity error $\Delta\mathbf{V}$ is further guaranteed. Subsequently, the main task intensively focused on is the stabilization of the attitude error system $\Delta\boldsymbol{\Theta}$. By differentiating $\Delta\boldsymbol{\Theta}$ and recalling the second sub-equation in (2), it obtains

$$\begin{aligned}\Delta\dot{\boldsymbol{\Theta}}=&\dot{\boldsymbol{\Theta}}-\dot{\boldsymbol{\Theta}}^*=\mathbf{M}(\boldsymbol{\Theta})\boldsymbol{\Omega}-\dot{\boldsymbol{\Theta}}^*\\=&\mathbf{M}(\boldsymbol{\Theta})\boldsymbol{\Omega}-\frac{\partial\boldsymbol{\Theta}^*}{\partial\Delta\mathbf{V}}\Delta\dot{\mathbf{V}}\\=&\underbrace{\Big(\mathbf{M}(\boldsymbol{\Theta})-\frac{\partial\boldsymbol{\Theta}^*}{\partial\Delta\mathbf{V}}\mathbf{S}(\Delta\mathbf{V})\Big)}_{\bar{\mathbf{M}}(\boldsymbol{\Theta})}\boldsymbol{\Omega}-\mathbf{g}\frac{\partial\boldsymbol{\Theta}^*}{\partial\Delta\mathbf{V}}\Big[\Big(\mathbf{R}^T(\boldsymbol{\Theta})\\&-\mathbf{R}^T(\boldsymbol{\Theta}^*)\Big)\mathscr{C}_z+\mathbf{k}_1\mathbf{Q}(\mathbf{S}_1,\ldots,\mathbf{S}_n)(\Delta\mathbf{V}+\Delta\mathbf{s})\\&-\mathbf{K}_2\Delta\mathbf{V}\Big]\\=&\bar{\mathbf{M}}(\boldsymbol{\Theta})(\Delta\boldsymbol{\Omega}+\boldsymbol{\Omega}^*)+\Xi_1(\boldsymbol{\Theta},\Delta\mathbf{V},\mathbf{S}_1,\ldots,\mathbf{S}_n)\end{aligned}\tag{27}$$

where $\Delta\boldsymbol{\Omega} = \boldsymbol{\Omega} - \boldsymbol{\Omega}^*$ denotes the difference between the real angular velocity and the virtual one. The partial derivative $\partial\boldsymbol{\Theta}^*/\partial\Delta\mathbf{V}$ is specified as

$$\frac{\partial\boldsymbol{\Theta}^*}{\partial\Delta\mathbf{V}}=\begin{pmatrix}\partial\phi^*/\partial\Delta\mathbf{V}_1 & \partial\phi^*/\partial\Delta\mathbf{V}_2 & \partial\phi^*/\partial\Delta\mathbf{V}_3\\ \partial\theta^*/\partial\Delta\mathbf{V}_1 & \partial\theta^*/\partial\Delta\mathbf{V}_2 & \partial\theta^*/\partial\Delta\mathbf{V}_3\\ 0 & 0 & 0\end{pmatrix}\tag{28}$$

Note that in above (27), the exact value of the image Jacobian matrix $\mathbf{Q}(\cdot)$ is unknown since the relative depth $r_i(\mathbf{S}_i)$ cannot be available for the realtime measurement (see in (12)). In other words, the combined nonlinearity $\Xi_1(\boldsymbol{\Theta}, \Delta\mathbf{V}, \mathbf{S}_1, \ldots, \mathbf{S}_n)$ is also unknown hence it cannot be cancelled by the virtual angular velocity. To address this problem, a RBFNN is adopted to approximate the uncertain combined term $\Xi_1(\boldsymbol{\Theta}, \Delta\mathbf{V}, \mathbf{S}_1, \ldots, \mathbf{S}_n)$, which is as

$$\Xi_1(\mathscr{Z}_1) = \Phi_1^T\mathbf{S}_1(\mathscr{Z}_1) + \mathbf{b}_1(\mathscr{Z}_1)\tag{29}$$

where $\mathscr{Z}_1 = (\boldsymbol{\Theta}^T, \Delta^T\mathbf{V}, \mathbf{S}_1, \ldots, \mathbf{S}_n)^T$ is the neural network inputs. $\Phi_1 \in \mathbb{R}^{N_1\times3}$ is an idealized weight matrix, and $\mathbf{S}_1(\mathscr{Z}_1) \in \mathbf{R}^{N_1}$ is the Gaussian radial basis function vector and its componentwise takes the exponential form as can be seen from the (4). N_1 is the number of the hidden neural network nodes. $\mathbf{b}_1(\mathscr{Z}_1)$ is the approximation error which can sufficiently be reduced less than any given bound $\bar{\mathbf{b}}_1$, i.e., $||\mathbf{b}_1(\mathscr{Z}_1)|| \le \bar{\mathbf{b}}_1$. In light of the (27), to stabilize the attitude error, intermediate controller $\boldsymbol{\Omega}^*$ can be constructed as

$$\boldsymbol{\Omega}^* = \bar{\mathbf{M}}_1^{-1}(\boldsymbol{\Theta})\Big(-\mathbf{k}_3\Delta\boldsymbol{\Theta} - \frac{1}{2\mathbf{a}_1^2}\Delta\boldsymbol{\Theta}\hat{\mathbf{W}}_1 - \bar{\hat{\mathbf{b}}}_1\tanh\frac{\Delta\boldsymbol{\Theta}}{\varsigma_1}\Big)\tag{30}$$

where $\mathbf{a}_1$ and ς_1 are two adjustable parameters. Substituting the developed intermediate controller $\boldsymbol{\Omega}^*$ into (27), one obtains

$$\begin{aligned}\Delta\dot{\boldsymbol{\Theta}} =& \bar{\mathbf{M}}(\boldsymbol{\Theta})\Delta\boldsymbol{\Omega} - \mathbf{k}_3\Delta\boldsymbol{\Theta} + \Phi_1^T\mathbf{S}_1(\mathscr{Z}_1) + \mathbf{b}_1(\mathscr{Z}_1)\\&-\frac{1}{2\mathbf{a}_1^2}\Delta\boldsymbol{\Theta}\hat{\mathbf{W}}_1 - \hat{\bar{\mathbf{b}}}_1\tanh\frac{\Delta\boldsymbol{\Theta}}{\varsigma_1}\end{aligned}\tag{31}$$

where $\mathbf{a}_1$ and ς_1 are two positive designed parameters. In above equation, $\hat{\mathbf{W}}_1(t)$ is an online estimation of the idealized constant $\mathbf{W}_1$ defined as

$$\mathbf{W}_1 = N_1\cdot\Big(\sqrt{\lambda_{\max}(\Phi_1^T\Phi_1)}\Big)^2\tag{32}$$

In fact, it is noticed from above definition that $\mathbf{W}_1$ is proportional to the square of the maximum singular value of the idealized weight matrix Φ_1.

Remark 5: Recalling the traditional neural-networks-based adaptive control algorithms, e.g., [18], [19], [31], [33], [34] and, the idealized weight matrix Φ_1 was directly estimated by the corresponding adaptive matrix $\hat{\Phi}_1(t) \in \mathbb{R}^{3\times 3}$. As the neurons increase, it will lead to a burdensome online computation which casts a shadow on application. In this paper, we propose a novel optimized adaptive neural structure to remove such computation burden. As clearly shown in (31)–(32), instead of the weight matrix itself, the square of its maximum singular value is estimated by the adaptive tuning law so that only one online parameter is left, which significantly alleviates the burdensome online computation. •

In above expression (31), the stabilization of the attitude error $\Delta\boldsymbol{\Theta}$ mainly relies on the convergence of the angular velocity error $\Delta\boldsymbol{\Omega}$ to zero. Thus, the subsequent task is to stabilize the angular velocity error $\Delta\boldsymbol{\Omega}$. First, we give the rate of change of $\Delta\boldsymbol{\Omega}$ as follows

$$\begin{aligned}\Delta\dot{\boldsymbol{\Omega}}=&-\mathbf{J}^{-1}\boldsymbol{\Omega}\times(\mathbf{J}\Omega)+\mathbf{J}^{-1}\mathbf{F}-\dot{\boldsymbol{\Omega}}^{*}\\ =&\mathbf{J}^{-1}\mathbf{F}-\underbrace{\mathbf{J}^{-1}\boldsymbol{\Omega}\times(\mathbf{J}\Omega)}_{\Xi_{21}}\\ &\underbrace{-\frac{\partial\boldsymbol{\Omega}^{*}}{\partial\hat{\mathbf{W}}_1}\dot{\hat{\mathbf{W}}}_1-\frac{\partial\boldsymbol{\Omega}^{*}}{\partial\hat{\hat{\mathbf{b}}}_1}\dot{\hat{\hat{\mathbf{b}}}}_1-\frac{\partial\boldsymbol{\Omega}^{*}}{\partial\Delta\mathbf{V}}\Delta\dot{\mathbf{V}}-\frac{\partial\boldsymbol{\Omega}^{*}}{\partial\Delta\boldsymbol{\Theta}}\Delta\dot{\boldsymbol{\Theta}}}_{\Xi_{22}}\\ =&\mathbf{J}^{-1}\mathbf{F}+\Xi_2(\mathscr{Z}_2)\end{aligned}\tag{33}$$

where $\Xi_2(\mathscr{Z}_2)=\Xi_{21}+\Xi_{22}$ is a combined nonlinear function and $\mathscr{Z}_2=(\Delta^T\boldsymbol{\Theta},\mathscr{Z}_1^T,\hat{\hat{\mathbf{b}}}_1,\hat{\mathbf{W}}_1,\Delta^T\mathbf{V})^T$. Similar to the process in (29), a RBFNN is used to approximate the uncertain function $\Xi_2(\mathscr{Z}_2)$. Thus, we have

$$\Xi_2(\mathscr{Z}_2)=\Phi_2^T\mathbf{S}_2(\mathscr{Z}_2)+\mathbf{b}_2(\mathscr{Z}_2)\tag{34}$$

In this approximator, $||\mathbf{b}_2(\mathscr{Z}_2)||\le\bar{\mathbf{b}}_2$ for a given constant $\bar{\mathbf{b}}_2$. In addition, the notations Φ_2 and $\mathbf{S}_2(\mathscr{Z}_2)$ denote the idealized weight matrix and the Gaussian basis function vector, respectively. Follow the definition of $\mathbf{W}_1$ in (32), a constant proportional to the square of the maximum singular value of idealized weight matrix Φ_2 is defined as

$$\mathbf{W}_2=N_2\cdot\left(\sqrt{\lambda_{\max}(\Phi_2^T\Phi_2)}\right)^2\tag{35}$$

where N_2 is the number of the neural network nodes.

Also, denote its online estimation by $\hat{\mathbf{W}}_2(t)$. Afterward, by substituting the approximation (34) into (33), one has

$$\Delta\dot{\boldsymbol{\Omega}}=\mathbf{J}^{-1}\mathbf{F}+\Phi_2^T\mathbf{S}_2(\mathscr{Z}_2)+\mathbf{b}_2(\mathscr{Z}_2)\tag{36}$$

From the practical point of view, it is often difficult to explicitly measure the inertial matrix $\mathbf{J}$. Also, its inverse $\mathbf{J}^{-1}$ is not accessible for the control design. Thus, a challenging problem, i.e, the unknown control gain matrix, appears in stabilizing above error system $\Delta\boldsymbol{\Omega}$. To address such challenge, a new inertial matrix estimator based on the Cholesky decomposition is technically designed as in (44). Afterward, the estimation error is defined as $\tilde{\mathbf{J}}=\mathbf{J}-\hat{\mathbf{J}}$. The desired control torque $\mathbf{F}\in\mathbf{R}^3$ is subsequently constructed as

$$\mathbf{F}=\hat{\mathbf{J}}\bar{\mathbf{F}}\tag{37}$$

where $\bar{\mathbf{F}}$ is an auxiliary controller which is further specified as

$$\bar{\mathbf{F}}=-\mathbf{k}_4\Delta\boldsymbol{\Omega}-\frac{1}{2\mathbf{a}_2^2}\Delta\boldsymbol{\Omega}\hat{\mathbf{W}}_2\mathbf{S}_2^T(\mathbf{Z}_2)\mathbf{S}_2(\mathbf{Z}_2)-\hat{\bar{\mathbf{b}}}_2\tanh\frac{\Delta\boldsymbol{\Omega}}{\varsigma_2}\tag{38}$$

where $\mathbf{a}_2$ and ς_2 are two designed parameters. Subsequently, we give the following property.

Property 1: The inverse matrix $\mathbf{J}^{-1}$ can be decomposed into $\mathbf{J}^{-1}=\mathscr{L}\mathscr{L}^T$, where $\mathscr{L}$ is a lower triangular matrix whose all diagonal elements are strictly positive.

Proof: Note that the inertial matrix $\mathbf{J}$ is symmetric positive-definite (SPD), which infers that its all eigenvalues are positive. Denote one of them by $\lambda_i\in\mathbb{R}^+$ corresponding to the eigenvector $\mathscr{X}_i$. Then one gets

$$\mathbf{J}\mathscr{X}_i=\lambda_i\mathscr{X}_i\Rightarrow\mathbf{J}^{-1}\mathscr{X}_i=\frac{1}{\lambda_i}\mathscr{X}_i.\tag{39}$$

So it is clear that all the eigenvalues of $\mathbf{J}^{-1}$ are also positive. In addition, note that the following result holds.

$$(\mathbf{J}^{-1})^T=(\mathbf{J}^T)^{-1}=\mathbf{J}^{-1}.\tag{40}$$

which implies that the inversion $\mathbf{J}^{-1}$ is a symmetric matrix. Combined with above two aspects, it can be concluded that $\mathbf{J}^{-1}$ is a symmetric positive-definite matrix. Using the Cholesky decomposition of $\mathbf{J}^{-1}$, one finally obtains

$$\mathbf{J}^{-1}=\mathscr{L}\mathscr{L}^T\tag{41}$$

where $\mathscr{L}$ is a lower triangular matrix, and its inversion is denoted by $\mathscr{L}^{-1}$. ■

Note that the matrix $\mathscr{L}$ has an inversion. Afterward, three auxiliary matrices are denoted by $\bar{\mathbf{J}}=\mathbf{J}^T\mathscr{L}$, $\hat{\bar{\mathbf{J}}}=\hat{\mathbf{J}}^T\mathscr{L}$, and $\tilde{\bar{\mathbf{J}}}=\tilde{\mathbf{J}}^T\mathscr{L}$, respectively. With these definitions, the (36) is further derived as

$$\Delta\dot{\boldsymbol{\Omega}}=\bar{\mathbf{F}}-\mathscr{L}\tilde{\bar{\mathbf{J}}}^T\bar{\mathbf{F}}+\Phi_2^T\mathbf{S}_2(\mathscr{Z}_2)+\mathbf{b}_2(\mathscr{Z}_2)\tag{42}$$

Substituting the proposed adaptive control torque $\mathbf{F}$ in (38) into (42), it gets

$$\begin{aligned}\Delta\dot{\boldsymbol{\Omega}}=&-\mathbf{k}_4\Delta\boldsymbol{\Omega}-\mathscr{L}\tilde{\bar{\mathbf{J}}}^T\bar{\mathbf{F}}+\mathbf{b}_2(\mathscr{Z}_2)-\hat{\bar{\mathbf{b}}}_2\tanh\frac{\Delta\boldsymbol{\Omega}}{\varsigma_2}\\ &+\Phi_2^T\mathbf{S}_2(\mathscr{Z}_2)-\frac{1}{2\mathbf{a}_2^2}\Delta\boldsymbol{\Omega}\hat{\mathbf{W}}_2\end{aligned}\tag{43}$$

where $\mathbf{a}_2$ and ς_2 are two positive scalars.

Until now, the overall procedure of the adaptive controller design is almost completed. Subsequently, we further provide the update laws versus the adaptive parameters in (27)

and (33) as follows

$$\begin{cases} \dot{\hat{\mathbf{W}}}_1 = \dfrac{\mathbf{r}_1}{2\mathbf{a}_1^2}\Delta^T\boldsymbol{\Theta}\Delta\boldsymbol{\Theta} - \mathbf{k}_{01}\hat{\mathbf{W}}_1 \\ \dot{\hat{\mathbf{W}}}_2 = \dfrac{\mathbf{r}_2}{2\mathbf{a}_2^2}\Delta^T\boldsymbol{\Omega}\Delta\boldsymbol{\Omega} - \mathbf{k}_{02}\hat{\mathbf{W}}_2 \\ \dot{\hat{\bar{\mathbf{b}}}}_1 = \mathbf{h}_1\Delta^T\boldsymbol{\Theta}\tanh\dfrac{\Delta\boldsymbol{\Theta}}{\varsigma_1} - \mathbf{h}_{10}\hat{\bar{\mathbf{b}}}_1 \\ \dot{\hat{\bar{\mathbf{b}}}}_2 = \mathbf{h}_2\Delta^T\boldsymbol{\Theta}\tanh\dfrac{\Delta\boldsymbol{\Theta}}{\varsigma_2} - \mathbf{h}_{20}\hat{\bar{\mathbf{b}}}_2 \end{cases} \tag{44}$$

where $\mathbf{r}_i$, $\mathbf{k}_{0i}$, $\mathbf{h}_i$, and $\mathbf{h}_{0i}(i = 1, 2)$ are some positive scalars. The inertial matrix estimator is technically designed as

$$\dot{\hat{\mathbf{J}}} = (\mathbf{d}\Delta\boldsymbol{\Omega}\bar{\mathbf{F}}^T - \mathbf{d}_0\hat{\mathbf{J}})\Gamma \tag{45}$$

where $\mathbf{d}$ and $\mathbf{d}_0$ are two positive scalar factors, and $\boldsymbol{\Gamma}$ is a SPD matrix. In the subsequent analysis, the stability of the aerial robot system under the control of the proposed controller will be testified by the strict Lyapunov argument.

C. STABILITY ANALYSIS BASED ON LYAPUNOV SYNTHESIS

In this subsection, we first utilize Lyapunov theory to prove the asymptotic convergence of the visual tracking error $\Delta\mathbf{s}$, and then further analyze stability of QAR's position tracking error with the adaptive laws list in (20), (24), (25), (30), and (37), and turning laws in (44) and (45).

A quadratic positive-definite Lyapunov function is considered as the following form

$$\begin{aligned}\mathbf{L} = &\frac{1}{2}\Delta^T\mathbf{s}\Delta\mathbf{s} + \frac{1}{2}\Delta^T\mathbf{V}\Delta\mathbf{V} + \frac{1}{2}\Delta^T\boldsymbol{\Theta}\Delta\boldsymbol{\Theta} + \frac{1}{2}\Delta^T\boldsymbol{\Omega}\Delta\boldsymbol{\Omega} \\ &+ \frac{1}{2\mathbf{d}}\mathbf{Tr}(\tilde{\bar{\mathbf{J}}}^T\Gamma^{-1}\tilde{\bar{\mathbf{J}}}) + \sum_{i=1}^{2}\frac{1}{2\mathbf{h}_i}\tilde{\bar{\mathbf{b}}}_i^2 + \sum_{i=1}^{2}\frac{1}{2\mathbf{r}_i}\tilde{\mathbf{W}}_i^2\end{aligned} \tag{46}$$

where Γ is a SPD matrix, $\mathbf{r}_i$, $\mathbf{h}_i$, $\mathbf{d} \in \mathbb{R}^+$, and $\tilde{\bar{\mathbf{b}}}_i = \bar{\mathbf{b}}_i - \hat{\bar{\mathbf{b}}}_i$, $\tilde{\mathbf{W}}_i = \mathbf{W}_i - \hat{\mathbf{W}}_i$, and $\tilde{\bar{\mathbf{J}}} = \bar{\mathbf{J}} - \hat{\bar{\mathbf{J}}}$. Deriving (46) and recalling (21), (26), (31) and (43),

$$\begin{aligned}\dot{\mathbf{L}} = &-\mathbf{k}_1\Delta^T\mathbf{s}\mathbf{Q}(\mathbf{S}_1,\ldots,\mathbf{S}_n)\Delta\mathbf{s} - \Delta^T\mathbf{V}\mathbf{K}_2\Delta\mathbf{V} \\ &+ \mathbf{k}_1\Delta^T\mathbf{V}\mathbf{Q}\Delta\mathbf{V} + \Delta^T\boldsymbol{\Theta}\bar{\mathbf{M}}\Delta\boldsymbol{\Omega} - \mathbf{k}_3\Delta^T\boldsymbol{\Theta}\Delta\boldsymbol{\Theta} \\ &+ \underbrace{\Delta^T\boldsymbol{\Theta}\Phi_1^T\mathbf{S}_1(\mathscr{Z}_1) - \frac{1}{2\mathbf{a}_1^2}\Delta^T\boldsymbol{\Theta}\Delta\boldsymbol{\Theta}\hat{\mathbf{W}}_1}_{\mathbf{C}_1} - \mathbf{k}_4\Delta^T\boldsymbol{\Omega}\Delta\boldsymbol{\Omega} \\ &+ \underbrace{\Delta^T\boldsymbol{\Omega}\Phi_2^T\mathbf{S}_2(\mathscr{Z}_2) - \frac{1}{2\mathbf{a}_2^2}\Delta^T\boldsymbol{\Omega}\Delta\boldsymbol{\Omega}\hat{\mathbf{W}}_2}_{\mathbf{C}_2} - \Delta^T\boldsymbol{\Omega}\mathscr{L}\tilde{\bar{\mathbf{J}}}^T\bar{\mathbf{F}} \\ &+ \underbrace{\Delta^T\boldsymbol{\Theta}\left(\mathbf{b}_1(\mathscr{Z}_1) - \hat{\bar{\mathbf{b}}}_1\tanh\frac{\Delta\boldsymbol{\Theta}}{\varsigma_1}\right)}_{\mathbf{C}_3} - \sum_{i=1}^{2}\frac{1}{\mathbf{r}_i}\tilde{\mathbf{W}}_i\dot{\hat{\mathbf{W}}}_i \\ &+ \underbrace{\Delta^T\boldsymbol{\Omega}\left(\mathbf{b}_2(\mathscr{Z}_2) - \hat{\bar{\mathbf{b}}}_2\tanh\frac{\Delta\boldsymbol{\Omega}}{\varsigma_2}\right)}_{\mathbf{C}_4} - \frac{1}{\mathbf{d}}\mathbf{Tr}(\tilde{\bar{\mathbf{J}}}^T\Gamma^{-1}\dot{\hat{\bar{\mathbf{J}}}}) \\ &+ \underbrace{\mathbf{g}\Delta^T\mathbf{V}\left(\mathbf{R}^T(\boldsymbol{\Theta}) - \mathbf{R}^T(\boldsymbol{\Theta}^*)\right)\mathscr{C}_z}_{\mathbf{C}_5} - \sum_{i=1}^{2}\frac{1}{\mathbf{h}_i}\tilde{\bar{\mathbf{b}}}_i\dot{\hat{\bar{\mathbf{b}}}}_i\end{aligned} \tag{47}$$

In above derivative (47), it is necessary to analyze some complicated terms (i.e., $\mathbf{C}_i$) in closer detail. Using the Young's inequality, the Cauchy-Schwarz inequality, and the corresponding preliminaries of the matrix theory, and recalling the adaptive laws, one has

$$\begin{aligned}\mathbf{C}_1 - \frac{1}{\mathbf{r}_1}\tilde{\mathbf{W}}_1\dot{\hat{\mathbf{W}}}_1 \le &\frac{1}{2\mathbf{a}_1^2}\Delta^T\boldsymbol{\Theta}\Delta\boldsymbol{\Theta}\mathbf{S}_1^T(\mathscr{Z}_1)\Phi_1\Phi_1^T\mathbf{S}_1(\mathscr{Z}_1) \\ &+ \frac{1}{2}\mathbf{a}_1^2 - \frac{1}{\mathbf{r}_1}\tilde{\mathbf{W}}_1\dot{\hat{\mathbf{W}}}_1 - \frac{1}{2\mathbf{a}_1^2}\Delta^T\boldsymbol{\Theta}\Delta\boldsymbol{\Theta}\hat{\mathbf{W}}_1 \\ \le &\frac{1}{2\mathbf{a}_1^2}\Delta^T\boldsymbol{\Theta}\Delta\boldsymbol{\Theta}\tilde{\mathbf{W}}_1 + \frac{1}{2}\mathbf{a}_1^2 - \frac{1}{\mathbf{r}_1}\tilde{\mathbf{W}}_1\dot{\hat{\mathbf{W}}}_1 \\ \le &-\frac{\mathbf{k}_{01}}{2\mathbf{r}_1}\tilde{\mathbf{W}}_1^2 + \frac{\mathbf{k}_{01}}{2\mathbf{r}_1}\mathbf{W}_1^2 + \frac{1}{2}\mathbf{a}_1^2\end{aligned} \tag{48}$$

where $\mathbf{k}_{01}$ is a positive modification parameter. In the adaptation law $\dot{\hat{\mathbf{W}}}_1$, the introduction of the modified term $\mathbf{k}_{01}\hat{\mathbf{W}}_1$ is to suppress the drift phenomenon of the adaptive parameter.

Following the above process again, it is derived that

$$\mathbf{C}_2 - \frac{1}{\mathbf{r}_2}\tilde{\mathbf{W}}_2\dot{\hat{\mathbf{W}}}_2 \le -\frac{\mathbf{k}_{02}}{2\mathbf{r}_2}\tilde{\mathbf{W}}_2^2 + \frac{\mathbf{k}_{02}}{2\mathbf{r}_2}\mathbf{W}_2^2 + \frac{1}{2}\mathbf{a}_2^2 \tag{49}$$

where $\mathbf{k}_{01}$ and $\mathbf{k}_{02}$ are positive designed parameters.

Furthermore,

$$\begin{aligned}\mathbf{C}_3 - &\frac{1}{\mathbf{h}_1}\tilde{\bar{\mathbf{b}}}_1\dot{\hat{\bar{\mathbf{b}}}}_1 \\ \le &\bar{\mathbf{b}}_1\left(||\Delta\boldsymbol{\Theta}|| - \Delta^T\boldsymbol{\Theta}\tanh\frac{\Delta\boldsymbol{\Theta}}{\varsigma_1}\right) + \tilde{\bar{\mathbf{b}}}_1\Delta^T\boldsymbol{\Theta}\tanh\frac{\Delta\boldsymbol{\Theta}}{\varsigma_1} \\ &- \frac{1}{\mathbf{h}_1}\tilde{\bar{\mathbf{b}}}_1\dot{\hat{\bar{\mathbf{b}}}}_i \\ \le &-\frac{\mathbf{h}_{10}}{2\mathbf{h}_1}\tilde{\bar{\mathbf{b}}}_1^2 + \frac{\mathbf{h}_{10}}{2\mathbf{h}_1}\bar{\mathbf{b}}_1^2 + 0.8355\bar{\mathbf{b}}_1\varsigma_1\end{aligned} \tag{50}$$

Similarly, one obtains

$$\mathbf{C}_4 - \frac{1}{\mathbf{h}_2}\tilde{\bar{\mathbf{b}}}_2\dot{\hat{\bar{\mathbf{b}}}}_2 \le -\frac{\mathbf{h}_{20}}{2\mathbf{h}_2}\tilde{\bar{\mathbf{b}}}_2^2 + \frac{\mathbf{h}_{20}}{2\mathbf{h}_2}\bar{\mathbf{b}}_2^2 + 0.8355\bar{\mathbf{b}}_2\varsigma_2 \tag{51}$$

As for the term $\mathbf{C}_5$, by using the algebraic operation, it gets

$$\begin{aligned}\mathbf{C}_5 \le &\mathbf{g}||\Delta\mathbf{V}|| \cdot \left|\left|\left(\mathbf{R}^T(\boldsymbol{\Theta}) - \mathbf{R}^T(\boldsymbol{\Theta}^*)\right)\mathscr{C}_z\right|\right| \\ &\le \mathbf{g}\mathfrak{L}||\Delta\boldsymbol{\Theta}||^2 + \frac{1}{4}||\Delta\mathbf{V}||^2\end{aligned} \tag{52}$$

Importantly, it should be highlighted how to dispose the term $\Delta^T\boldsymbol{\Omega}\mathscr{L}\tilde{\bar{\mathbf{J}}}^T\bar{\mathbf{F}}$ in equation (47).

$$\begin{aligned}&\Delta^T\boldsymbol{\Omega}\mathscr{L}\tilde{\bar{\mathbf{J}}}^T\bar{\mathbf{F}} - \frac{1}{\mathbf{d}}\mathbf{Tr}(\tilde{\bar{\mathbf{J}}}^T\Gamma^{-1}\dot{\hat{\bar{\mathbf{J}}}}) \\ &= \mathbf{Tr}(\tilde{\bar{\mathbf{J}}}^T\bar{\mathbf{F}}\Delta^T\boldsymbol{\Omega}\mathscr{L}) - \frac{1}{\mathbf{d}}\mathbf{Tr}(\tilde{\bar{\mathbf{J}}}^T\Gamma^{-1}\dot{\hat{\bar{\mathbf{J}}}}) \le \frac{\mathbf{d}_0}{\mathbf{d}}\mathbf{Tr}(\tilde{\bar{\mathbf{J}}}^T\hat{\bar{\mathbf{J}}})\end{aligned}$$

$$\leq -\frac{\mathbf{d}_0}{2\mathbf{d}}||\tilde{\bar{\mathbf{J}}}||_{\mathbf{F}}^2 + \frac{\mathbf{d}_0}{2\mathbf{d}}||\bar{\mathbf{J}}||_{\mathbf{F}}^2$$
$$\leq -\frac{\mathbf{d}_0}{2\mathbf{d}\lambda_{\max}(\Gamma^{-1})}\mathbf{Tr}(\tilde{\bar{\mathbf{J}}}^T\Gamma^{-1}\tilde{\bar{\mathbf{J}}}) + \frac{\mathbf{d}_0}{2\mathbf{d}}||\bar{\mathbf{J}}||_{\mathbf{F}}^2 \quad (53)$$

In terms of the detailed derivations of (48)–(53), the time derivative $\dot{\mathbf{L}}$ in (47) is further formulated as

$$\begin{aligned}\dot{\mathbf{L}} \leq & -\mathbf{k}_1\lambda_{\min}(\mathbf{Q})||\Delta\mathbf{s}||^2 - \big(\lambda_{\min}(\mathbf{K}_2) - \mathbf{k}_1\lambda_{\max}(\mathbf{Q}) \\ & - \frac{1}{4}\big)||\Delta\mathbf{V}||^2 - \big(\mathbf{k}_3 - ||\bar{\mathbf{M}}||_{\mathbf{F}}^2 - \mathbf{g}\mathfrak{L}\big)||\boldsymbol{\Theta}||^2 \\ & - \big(\mathbf{k}_4 - \frac{1}{4}\big)||\boldsymbol{\Omega}||^2 - \sum_{i=1}^{2}\frac{\mathbf{k}_{0i}}{2\mathbf{r}_i}\tilde{\mathbf{W}}_i^2 - \sum_{i=1}^{2}\frac{\mathbf{h}_{i0}}{2\mathbf{h}_i}\tilde{\mathbf{b}}_i^2 \\ & - \frac{\mathbf{d}_0}{2\mathbf{d}\lambda_{\max}(\Gamma^{-1})}\mathbf{Tr}(\tilde{\bar{\mathbf{J}}}^T\Gamma^{-1}\tilde{\bar{\mathbf{J}}}) + \frac{\mathbf{d}_0}{2\mathbf{d}}||\bar{\mathbf{J}}||_{\mathbf{F}}^2 \\ & + \sum_{i=1}^{2}\left(\frac{\mathbf{k}_{0i}}{2\mathbf{r}_i}\mathbf{W}_i^2 + \frac{\mathbf{h}_{i0}}{2\mathbf{h}_i}\bar{\mathbf{b}}_i^2 + \frac{\mathbf{a}_i^2}{2} + 0.8355\bar{\mathbf{b}}_i\varsigma_i\right) \quad (54)\end{aligned}$$

Assuming that there are two constants $\mathbf{K}$ and $\mathbf{C}$ defined as follows

$$\begin{aligned}\mathbf{K} = \min\Big\{ & 2\mathbf{k}_1\lambda_{\min}(\mathbf{Q}), 2\big(\lambda_{\min}(\mathbf{K}_2) - \mathbf{k}_1\lambda_{\max}(\mathbf{Q}) - \frac{1}{4}\big), \\ & 2\big(\mathbf{k}_3 - ||\bar{\mathbf{M}}||_{\mathbf{F}}^2 - \mathbf{g}\mathfrak{L}\big), 2\mathbf{k}_4 - \frac{1}{2}, \mathbf{k}_{0i}, \mathbf{h}_{i0}, \frac{\mathbf{d}_0}{\lambda_{\max}(\Gamma^{-1})}\Big\} \quad (55)\end{aligned}$$

$$\mathbf{C} = \sum_{i=1}^{2}\left(\frac{\mathbf{k}_{0i}}{2\mathbf{r}_i}\mathbf{W}_i^2 + \frac{\mathbf{h}_{i0}}{2\mathbf{h}_i}\bar{\mathbf{b}}_i^2 + \frac{\mathbf{a}_i^2}{2} + 0.8355\bar{\mathbf{b}}_i\varsigma_i\right) + \frac{\mathbf{d}_0}{2\mathbf{d}}||\bar{\mathbf{J}}||_{\mathbf{F}}^2 \quad (56)$$

Then inequality (54) can be further formulated as

$$\dot{\mathbf{L}}(t) \leq -\mathbf{K}\mathbf{L}(t) + \mathbf{C} \quad (57)$$

And thus the solution of inequality (57) can be obtained as follows

$$\mathbf{L}(t) \leq \big(\mathbf{L}(t_0) - \frac{\mathbf{C}}{\mathbf{K}}\big)\exp\Big(-\mathbf{K}(t - t_0)\Big) + \frac{\mathbf{C}}{\mathbf{K}} \quad (58)$$

where $t > t_0$ and t_0 is the initial time. Therefore, base on the Lyapunov theory, we know that the visual tracking error $\Delta\mathbf{s}$ is asymptotically convergent and its asymptotic performance is derived as

$$\lim_{t\to\infty}||\Delta\mathbf{s}(t)|| \leq \sqrt{\frac{2\mathbf{C}}{\mathbf{K}}} \quad (59)$$

Furthermore, it is clear that all the signals in the closed-loop system remain SGUUB. Invoking the Theorem 1, we can now draw the conclusion that, with the proposed adaptive controller, the QAR's position tracking error can be made arbitrarily small through appropriate design parameters.

Remark 6: Note, from (59) and (55), that the norm of the visual tracking error $\Delta\mathbf{s}(t)$ converges ultimately to an adjustable bound $\sqrt{\frac{2\mathbf{C}}{\mathbf{K}}}$ which may be reduced to some extent by increasing the design parameters k_i, k_{0i}, and h_{i0} for $i = 1, \ldots, 4$. So, in practical implementation, we first choose a set of proper design parameters k_i, k_{0i}, and h_{i0} and then increase them until the visual tracking error $\Delta\mathbf{s}(t)$ meets the desired accuracy requirement. •

Remark 7: The positioning control algorithm proposed in this paper is a robust adaptive approach which can accommodate external disturbances effectively in the sense that when disturbances appear, the closed-loop signal boundedness can be ensured, and when disturbances vanish, a desired tracking performance can be recovered. In other words, if disturbances appear after implementing the proposed control algorithm, the stability of the quadrotor system can be maintained but both the transient and steady-state tracking performances will be affected to some extent. When disturbances vanish, a desired tracking performance can be recovered automatically. On the other hand, before implementing the controller, we may predict the range of potential disturbances and choose a set of reasonable design parameters. With such a choice, the desired transient and steady-state tracking performances can also be ensured even when in the presence of external disturbances, provided that the appeared disturbances do not go beyond the predicted range. •

Remark 8: Many important nonlinear control techniques such as backstepping, neural networks and adaptive control are used in the study. They are rather useful for the positioning control design of quadrotor systems. For instances, the quadrotor system is indeed an underactuated control system and hence we need the backstepping technique for its control design; neural networks and adaptive control are able to handle the model uncertainties of quadrotor systems. More nonlinear control techniques which are useful for the extension of this research can be found in [44]–[50]. •

IV. PERFORMANCE RESULTS

We have implemented the proposed adaptive neural visual control algorithm in a QAR at the intelligent control and robot laboratory of Guangdong University of Technology, see in Fig. 4. Subsequently, some results are collected to evaluate the positioning control performance.

A. MECHANICAL SETUP AND COMMUNICATION STRUCTURE

To practically assess the tracking control performance of the adaptive controller, we have developed an hardware system of the underactuated QAR as shown in Fig.4. In this flight platform, the desired thrust $\mathbf{T}$ and the control torque $\mathbf{F}$ are physically generated by the rotor speeds $\mathscr{W}_i (i = 1, \ldots, 4)$ of four brushless direct current (BLDC) motors, and their inherent relation is specified as

$$\begin{cases}\mathbf{T} = -f_t\sum_{i=1}^{4}\mathscr{W}_i^2 \\ \mathbf{F} = \Big[df_t(\mathscr{W}_2^2 - \mathscr{W}_4^2), df_t(\mathscr{W}_1^2 - \mathscr{W}_3^2), \\ \qquad f_d\sum_{i=1}^{4}(-1)^{i+1}\mathscr{W}_i^2\Big]^T\end{cases} \quad (60)$$

where d represents the distance from the center of the robot to the axis of the rotor. f_t and f_d are two positive scalars

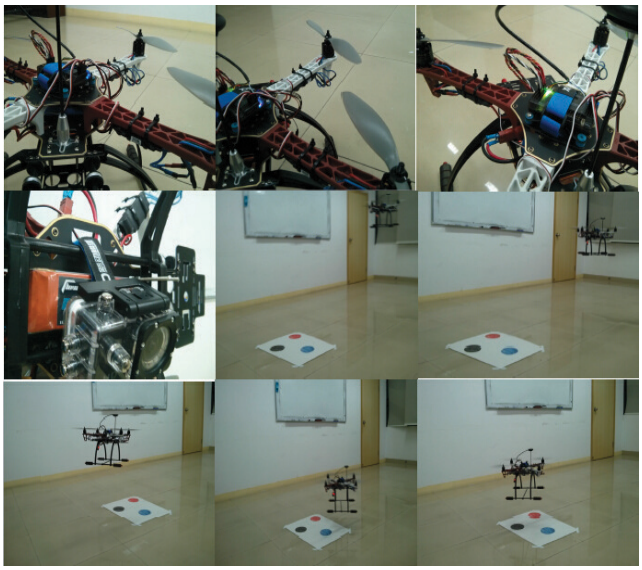

FIGURE 4. Hardware system of the QAR.

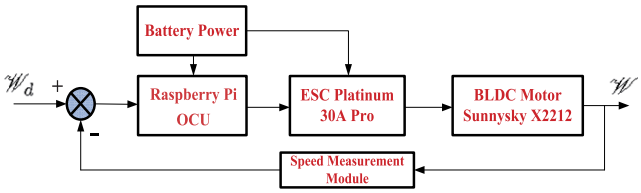

FIGURE 5. Visualization of the rotor speed control of BLDC motors.

denoting by the thrust factor and drag factor, respectively. It is observed from (60) that once the desired control $(\mathbf{T}, \mathbf{F}^T)$ is determined, the required rotor speeds of the BLDC motors can be calculated accordingly. Thus, it is necessary to further take into account the speed tracking control of the motors. In the developed mechanical setup, the types of four BLDC motors are all selected as the Sunnysky X2212 whose maximum speed nearly approaches to 5000rpm and, at this time, it can provide 680N force to the aerial robot under the power 102W. The power used for the actuation of these four BLDC motors is provided by the lithium-polymer battery which has the capacity of 2200mAh. To achieve speed tracking control, an electronic speed controller (ESC) Platinum 30A Pro is mounted on the QAR to connect the BLDC motors with the onboard control unit (OCU) Arduino MEGA ATmega 2560. Also, the realtime output speed feeds back to the OCU by using LM393-based speed measurement module as a specific sensor. Subsequently, a standard motor speed proportion-integration (PI) control algorithm is integrated in the OCU so that the rotor speed vector $\mathscr{W} = [\mathscr{W}_1, \mathscr{W}_2, \mathscr{W}_3, \mathscr{W}_4]^T$ are fast convergent to the desired values $\mathscr{W}_d \in \mathbb{R}^4$ which are computed from (60). A visualization about the motor speed control is presented in Fig. 5.

It is observed from (60) that the desired rotor speed $\mathscr{W}_d$ can be determined once the thrust and control torque are designed. In this experiment, the desired thrust $\mathbf{T}$ and control torque $\mathbf{F}$ are obtained from the adaptive controller.

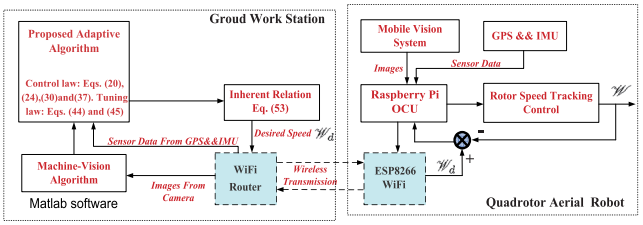

FIGURE 6. Block diagram of the ground work station and QAR.

Subsequently, we focus on the construction of the control forces in mechanical setup. First, a camera is embedded in the QAR to observe the target features marked on ground with the rate of 20FPS and, at the same time, the images are sent to the ground station computer with an ESP8266 WiFi module. Afterward, a machine-vision algorithm is run in the ground computer to extract the feature points from the background of the images and calculate the pixel coordinate with respect to the calibrated frame. Thus, the USC $\mathbf{s}(t)$ can be computed according to (8) and (11). In addition, the Raspberry Pi OCU integrates an inertial measurement unit (IMU) so that the attitude and velocities can be measured accordingly. The QAR is equipped a global position system (GSP) Ublox NEO-6M for a rough estimation of the position. Lastly, these collected sensor data are transmitted from the OCU to the ground computer via a WiFi module. After collecting the realtime sensor data from the QAR and computing the spherical centroid, the proposed adaptive visual control algorithm is implemented in ground station computer, and then the intermediate controllers $\mathbf{V}^*$, $\mathbf{\Theta}^*$ and $\mathbf{\Omega}^*$ are numerically computed so that the desired control forces $\mathbf{T}$ and $\mathbf{F}$ are finally constructed. Subsequently, these control points are re-transmitted to the OCU for achieving the rotors speed tracking control of four BLDC motors in QAR. The persistent cooperation between the ground station computer and the onboard controller under the designed algorithms guarantees the asymptotic convergence of the aerial robot to the desired position. The block diagram of control flow between ground work station and QAR is shown in Fig. 6.

B. CONTROL IMPLEMENTATION AND PERFORMANCE RESULTS

At the beginning of the flight test, we use joystick to control the aerial robot such that its onboard camera can observe the marked target features. At this moment, the airplane mode is switched to the autonomous flight which is driven implicitly by the proposed adaptive visual control algorithm. In this experiment, the switched moment is recorded as an initial actuation time t_0. In addition, the QAR mass is $\mathbf{m} = 0.78$kg and the initial value of inertial matrix is assumed to be an identity matrix. We assume that all target features are always in the view field of the onboard pin-hole camera. The focal length is $f = 10$mm. To obtain the image projection of target points to the desired position, we give a general formula as

$$\mathscr{Q}^* = \sum_{i=1}^{n} \frac{(\bar{\mathbf{S}}_i - \mathbf{p}^*)}{||\bar{\mathbf{S}}_i - \mathbf{p}^*||} \tag{61}$$

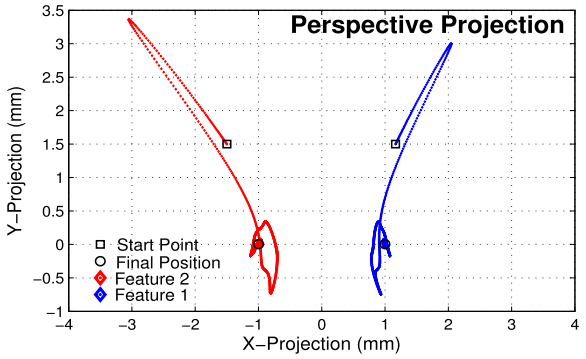

FIGURE 7. Position tracking path on physical image plane.

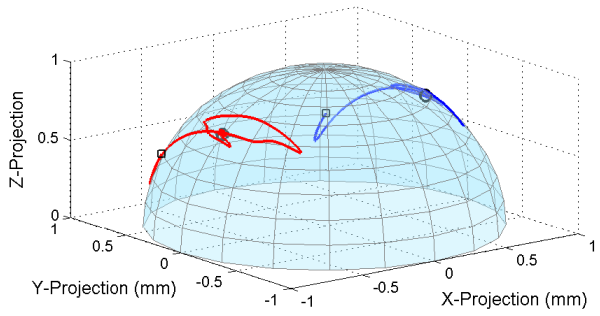

FIGURE 8. Position tracking path on spherical projective image plane.

where n represents the number of the total target features, $\bar{\mathbf{S}}_i$ is the coordinate of ith target point in Frame $\mathbb{I}$, which is defined in (6).

Control Using Two Features: In the first implementation, two feature points will be used to develop the proposed control method. The two target points' coordinates are $\bar{\mathbf{S}}_1 = (3\text{m}, 3\text{m}, 0\text{m})^T$ and $\bar{\mathbf{S}}_2 = (-3\text{m}, 3\text{m}, 0\text{m})^T$, respectively. The desired position's coordinate is chosen as $\mathbf{p}^* = (0\text{m}, 3\text{m}, -3\text{m})^T$, which is 3m above the midpoint between the two target points. Through (61), one obtains the desired centroid as $\mathscr{Q}^* = (0, 0, \sqrt{2})^T$. In addition, we set the initial position of QAR as $\mathbf{p}(t_0) = (0.4\text{m}, -0.4\text{m}, -2\text{m})^T$. The initial values of QAR's velocity $\mathbf{V}$, attitude $\mathbf{\Theta}$, and angular velocity $\mathbf{\Omega}$ are set as zero. In addition, to construct the proposed adaptive controller, we set the corresponding coefficient as $\mathbf{k}_1 = 6$, $\mathbf{k}_2 = 2$, $\mathbf{k}_3 = 10$, and $\mathbf{k}_4 = 0.9$. The control results of position tracking are shown in Fig. 7-10, and a further analysis is provided hereinafter.

Discussion: **1)** As shown in Fig. 9, the position error $\Delta\mathbf{p}$ asymptotically goes to zero after about 15s, that verifies the validity of the proposed adaptive controller. Moreover, it is notable that both of the projective image error $\Delta\mathbf{s}$ and $\Delta\mathbf{p}$ are asymptotically convergent to zero. Such result demonstrates the reasonability of Theorem 1.

2) Note that the roll angle ϕ and the pitch angle θ are always remained within $(-0.5\text{rad}, 0.5\text{rad})$, which indicates that no singular phenomena occurs over the entire flying process of the aerial robot. Such result further validates the practical feasibility of the proposed adaptive controller.

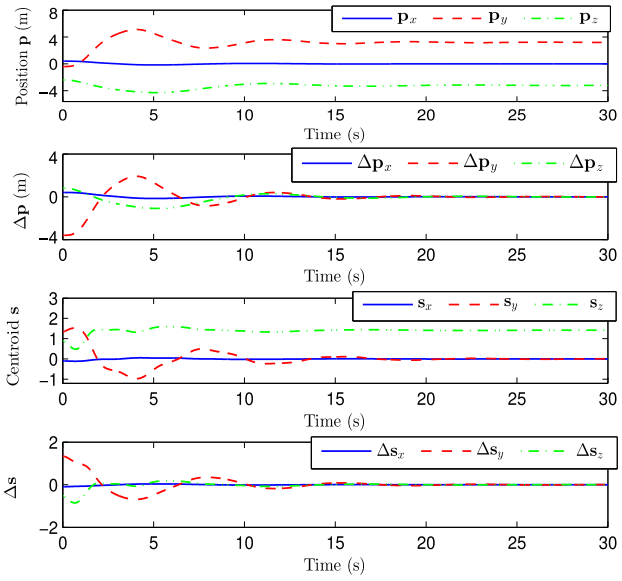

FIGURE 9. Positioning control result.

3) When the QAR hovers in the desired position $\mathbf{p}^*$ which is in fact an equilibrium point, the thrust satisfy the condition $\mathbf{T} = \mathbf{mg}$ as can be seen from the profile of the thrust in Fig. 10.

Control Using Four Features: To further testify the positioning behavior of the closed-loop system under the control of the proposed adaptive controller, a simulation with the four target features is implemented in the same dynamic model. Specifically, the target features' coordinates in Frame $\mathbb{I}$ are given as $\bar{\mathbf{S}}_1 = (3\text{m}, 3\text{m}, 0\text{m})^T$, $\bar{\mathbf{S}}_2 = (-3\text{m}, 3\text{m}, 0\text{m})^T$, $\bar{\mathbf{S}}_3 = (-3\text{m}, -3\text{m}, 0\text{m})^T$ and $\bar{\mathbf{S}}_4 = (3\text{m}, -3\text{m}, 0\text{m})^T$. The desired position of the aerial robot is

$$\mathbf{p}^* = \begin{pmatrix} 0 \\ 0 \\ -3\sqrt{2} \end{pmatrix} \text{m}. \tag{62}$$

Subsequently, adopting the calculation formula (61) of the centroid, it obtains

$$\mathscr{Q}^* = (0, 0, 2\sqrt{2})^T \tag{63}$$

Thus, an adaptive controller with the same parameters as those in the above simulation is constructed, and applied to the same QAR. In Fig. 11, the realtime perspective projection of four target features on physical image plane are depicted, whereas their equivalent spherical projections on image plane are shown in Fig. 12.

In light of the performance results shown in Fig. 13, it is clear that the image projection error $\Delta\mathbf{s}$ almost approaches zero after about 20s. Accordingly, the position tracking error $\Delta\mathbf{p}$ also converges to zero from now on. In addition, it is noted that all the physical variables including the position, translation velocity, attitude, and angular velocity remain bounded for any time. Compared with the evolution of the attitude using two target features, the attitude using four target

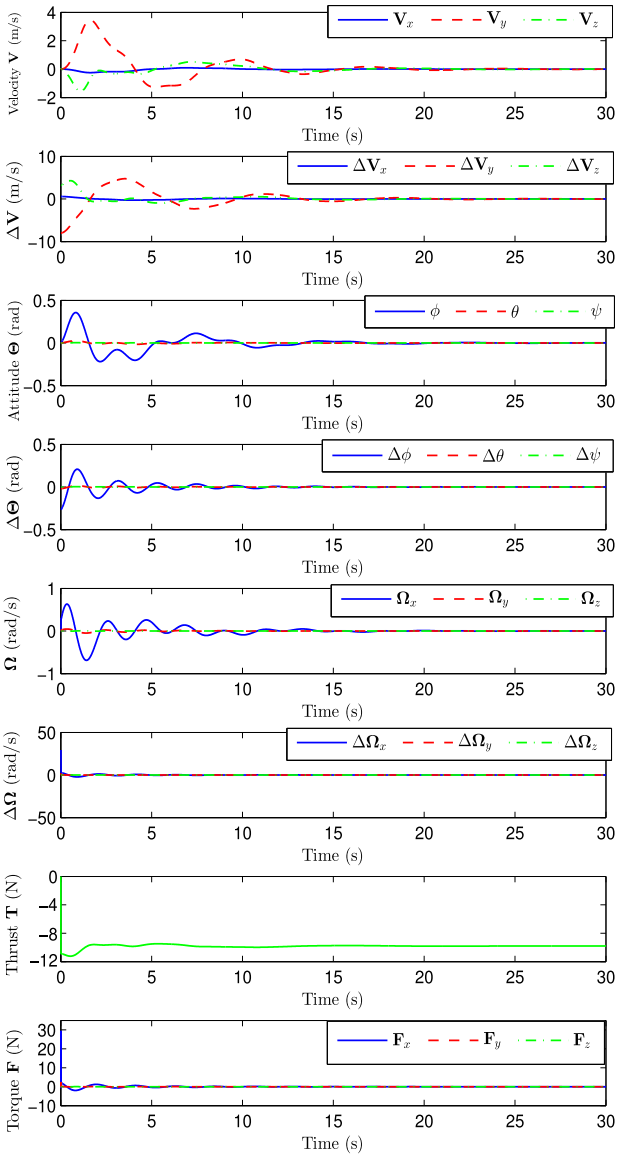

FIGURE 10. Profiles of physical variables (i.e, the translational velocity V, the attitude Θ, the angular velocity Ω, the thrust T, and the torque F).

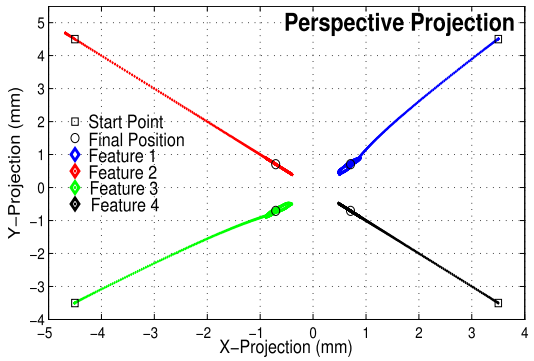

FIGURE 11. Perspective projection of the four features on physical image plane.

features varies smaller so that the singular phenomenon can be avoided more possibly. Note that the position of the QAR finally locates at the desire position which can be regarded as

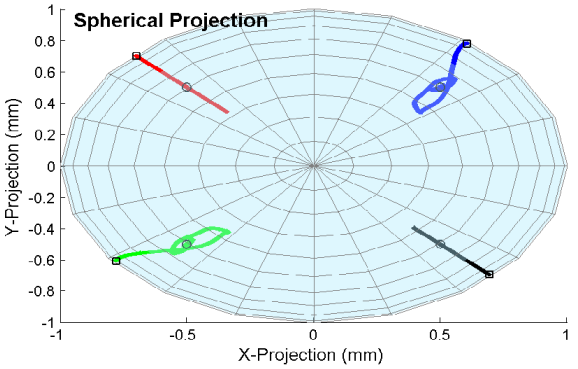

FIGURE 12. Spherical projection of the four features on equivalent image plane.

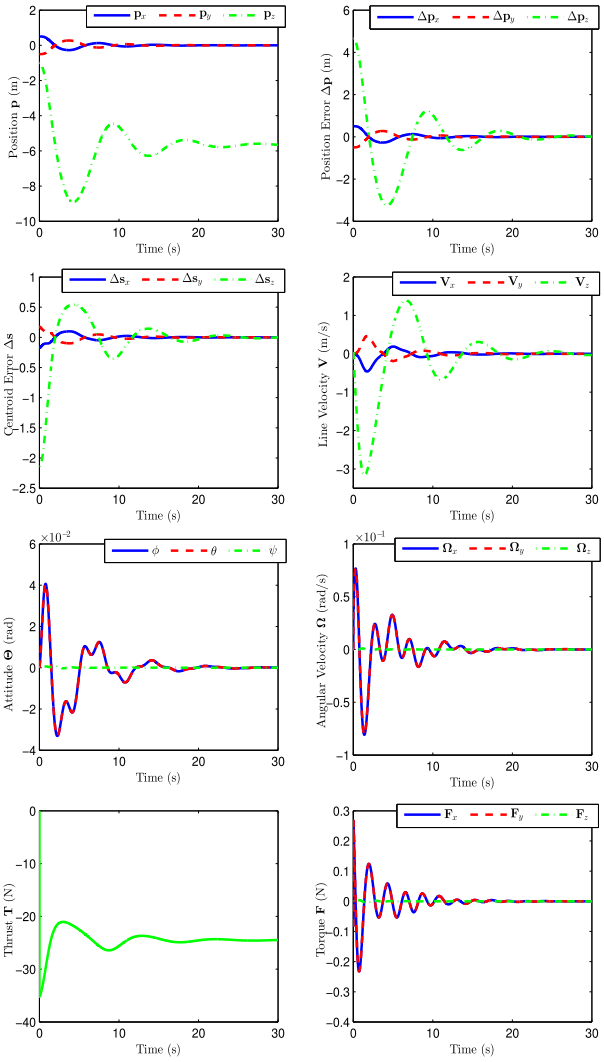

FIGURE 13. Positioning control results and profiles of physical variables.

an equilibrium point. At this time, the thrust should be equal to the gravity of the aerial robot, i.e, $\mathbf{T} = \mathbf{mg}$. According to the profile of the thrust, it is clear that such condition is well satisfied by the proposed adaptive controller.

V. CONCLUSION

QARs are widely used in various applications due to their special advantages, such as simple mechanical structure, easy maintenance and excellent maneuvering. For the underactuation, QAR visual servoing remains a challenging topic in the robotics field. In this paper, we present an adaptive controller to fulfill the visual position tracking control for a QAR. The transformation relationship between the position tracking error and image projection error is constructed to eliminate the requirement of position information. In the adaptive controller, a backstepping-based method that synthesizes the Lipschitz condition and the natural saturation property of the inverse tangent function is designed to deal with the underactuation feature of QAR. Moreover, an optimized adaptive NN mechanism is proposed to reduce the computation burden in the online estimation of the weight matrix. In addition, the corresponding stability analysis is performed using Lyapunov theory. Lastly, two simulated results are presented to illustrate the validity and control performance of our proposed method. Note that the Round-Robin protocol [51], [52] and stochastic communication protocol [53], [54] are useful in circumventing the data collisions caused by limited communication resources. Based on such an observation and the research of this paper, it is possible to extend our proposed control method to the teleoperation application of QAR in our future study.

REFERENCES

[1] V. Lippiello, B. Siciliano, and L. Villani, "Position-based visual servoing in industrial multirobot cells using a hybrid camera configuration," *IEEE Trans. Robot.*, vol. 23, no. 1, pp. 73–86, Feb. 2007.
[2] D.-H. Park, J.-H. Kwon, and I.-J. Ha, "Novel position-based visual servoing approach to robust global stability under field-of-view constraint," *IEEE Trans. Ind. Electron.*, vol. 59, no. 12, pp. 4735–4752, Dec. 2012.
[3] C. Yang, Z. Qu, and J. Han, "Decoupled-space control and experimental evaluation of spatial electrohydraulic robotic manipulators using singular value decomposition algorithms," *IEEE Trans. Ind. Electron.*, vol. 61, no. 7, pp. 3427–3438, Jul. 2014.
[4] B. Espiau, F. Chaumette, and P. Rives, "A new approach to visual servoing in robotics," *IEEE Trans. Robot. Autom.*, vol. 8, no. 3, pp. 313–326, Jun. 1992.
[5] L. Weiss, A. Sanderson, and C. Neuman, "Dynamic sensor-based control of robots with visual feedback," *IEEE J. Robot. Autom.*, vol. RA-3, no. 5, pp. 404–417, Oct. 1987.
[6] N. Andreff and P. Martinet, "Unifying kinematic modeling, identification, and control of a Gough–Stewart parallel robot into a vision-based framework," *IEEE Trans. Robot.*, vol. 22, no. 6, pp. 1077–1086, Dec. 2006.
[7] J.-J. Slotine and W. Li, *Applied Nonlinear Control: United States Edition*. London, U.K.: Pearson, 1990.
[8] C. Y. Tsai, C. C. Wong, C. J. Yu, C. C. Liu, and T. Y. Liu, "A hybrid switched reactive-based visual servo control of 5-DOF robot manipulators for pick-and-place tasks," *IEEE Syst. J.*, vol. 9, no. 1, pp. 119–130, Mar. 2015.
[9] Y. Wang, H. Lang, and C. W. D. Silva, "A hybrid visual servo controller for robust grasping by wheeled mobile robots," *IEEE/ASME Trans. Mechatronics*, vol. 15, no. 5, pp. 757–769, Oct. 2010.
[10] Y.-H. Liu, H. Wang, C. Wang, and K. K. Lam, "Uncalibrated visual servoing of robots using a depth-independent interaction matrix," *IEEE Trans. Robot.*, vol. 22, no. 4, pp. 804–817, Aug. 2006.
[11] K. Wang, Y. Liu, and L. Li, "Vision-based tracking control of nonholonomic mobile robots without position measurement," in *Proc. IEEE Int. Conf. Robot. Automat.*, May 2013, pp. 5265–5270.
[12] H. Wang, Y.-H. Liu, and D. Zhou, "Dynamic visual tracking for manipulators using an uncalibrated fixed camera," *IEEE Trans. Robot.*, vol. 23, no. 3, pp. 610–617, Jun. 2007.
[13] Z. T. Dydek, A. M. Annaswamy, and E. Lavretsky, "Adaptive control of quadrotor UAVs: A design trade study with flight evaluations," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 4, pp. 1400–1406, Jul. 2013.
[14] T. Madani and A. Benallegue, "Control of a quadrotor mini-helicopter via full state backstepping technique," in *Proc. IEEE Conf. Decis. Control*, Dec. 2006, pp. 1515–1520.
[15] T. Madani and A. Benallegue, "Sliding mode observer and backstepping control for a quadrotor unmanned aerial vehicles," in *Proc. Amer. Control Conf.*, Jul. 2007, pp. 5887–5892.
[16] A. Roberts and A. Tayebi, "Adaptive position tracking of VTOL UAVs," *IEEE Trans. Robot.*, vol. 27, no. 1, pp. 129–142, Feb. 2011.
[17] J. B. Vance, B. C. Kaul, S. Jagannathan, and J. A. Drallmeier, "Output feedback controller for operation of spark ignition engines at lean conditions using neural networks," *IEEE Trans. Control Syst. Technol.*, vol. 16, no. 2, pp. 214–228, Mar. 2008.
[18] T. Dierks and S. Jagannathan, "Output feedback control of a quadrotor UAV using neural networks," *IEEE Trans. Neural Netw.*, vol. 21, no. 1, pp. 50–66, Jan. 2010.
[19] D. Nodland, H. Zargarzadeh, and S. Jagannathan, "Neural network-based optimal adaptive output feedback control of a helicopter UAV," *IEEE Trans. Neural Netw.*, vol. 24, no. 7, pp. 1061–1073, Jul. 2013.
[20] T. Hamel and R. Mahony, "Visual servoing of an under-actuated dynamic rigid-body system: An image-based approach," *IEEE Trans. Robot. Autom.*, vol. 18, no. 2, pp. 187–198, Apr. 2002.
[21] R. Mahony and T. Hamel, "Image-based visual servo control of aerial robotic systems using linear image features," *IEEE Trans. Robot.*, vol. 21, no. 2, pp. 227–239, Apr. 2005.
[22] O. Bourquardez, R. Mahony, N. Guenard, F. Chaumette, T. Hamel, and L. Eck, "Image-based visual servo control of the translation kinematics of a quadrotor aerial vehicle," *IEEE Trans. Robot.*, vol. 25, no. 3, pp. 743–749, Jun. 2009.
[23] T. Hamel, R. Mahony, R. Lozano, and J. Ostrowski, "Dynamic modelling and configuration stabilization for an x4-flyer," *IFAC Proc. Volumes*, vol. 35, no. 1, pp. 217–222, 2002.
[24] N. Guenard, T. Hamel, and R. Mahony, "A practical visual servo control for an unmanned aerial vehicle," *IEEE Trans. Robot.*, vol. 24, no. 2, pp. 331–340, Apr. 2008.
[25] L. R. Garcia-Carrillo, A. Dzul, R. Lozano, A. Sanche, and E. Rondon, "Hovering quad-rotor control: A comparison of nonlinear controllers using visual feedback," in *Proc. IEEE Conf. Decis. Control*, Dec. 2012, pp. 1662–1667.
[26] L. Merino, J. Wiklund, F. Caballero, A. Moe, J. R. M. De Dios, P.-E. Forssen, K. Nordberg, and A. Ollero, "Vision-based multi-UAV position estimation," *IEEE Robot. Autom. Mag.*, vol. 13, no. 3, pp. 53–62, Sep. 2006.
[27] T. Lee, "Robust adaptive attitude tracking on SO(3) with an application to a quadrotor UAV," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 5, pp. 1924–1930, Sep. 2013.
[28] N. A. Chaturvedi, A. K. Sanyal, M. Chellappa, J. L. Valk, N. H. McClamroch, and D. S. Bernstein, "Adaptive tracking of angular velocity for a planar rigid body with unknown models for inertia and input nonlinearity," *IEEE Trans. Control Syst. Technol.*, vol. 14, no. 4, pp. 613–627, Jul. 2006.
[29] J. Ahmed, V. T. Coppola, and D. S. Bernstein, "Asymptotic tracking of spacecraft attitude motion with inertia matrix identification," in *Proc. 36th IEEE Conf. Decision Control*, vol. 3, Dec. 1997, pp. 2471–2476.
[30] N. A. Chaturvedi, D. S. Bernstein, J. Ahmed, F. Bacconi, and N. H. McClamroch, "Globally convergent adaptive tracking of angular velocity and inertia identification for a 3-DOF rigid body," *IEEE Trans. Control Syst. Technol.*, vol. 14, no. 5, pp. 841–853, Sep. 2006.
[31] Z. Chen, S. S. Ge, Y. Zhang, and Y. Li, "Adaptive neural control of MIMO nonlinear systems with a block-triangular pure-feedback control structure," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 11, pp. 2017–2029, Nov. 2014.
[32] H. Shi, X. Li, K.-S. Hwang, W. Pan, and G. Xu, "Decoupled visual servoing with fuzzy Q-learning," *IEEE Trans Ind. Informat.*, vol. 14, no. 1, pp. 241–252, Jan. 2018.
[33] W. Chen, L. Jiao, J. Li, and R. Li, "Adaptive NN backstepping output-feedback control for stochastic nonlinear strict-feedback systems with time-varying delays," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 40, no. 3, pp. 939–950, Jun. 2010.
[34] H. Q. Wang, X. P. Liu, and K. F. Liu, "Robust adaptive neural tracking control for a class of stochastic nonlinear interconnected systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 3, pp. 510–523, Apr. 2016.

[35] W. Chen, S. S. Ge, J. Wu, and M. Gong, "Globally stable adaptive backstepping neural network control for uncertain strict-feedback systems with tracking accuracy knowna priori," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 9, pp. 1842–1854, Sep. 2015.
[36] Q. Zhou, P. Shi, H. Liu, and S. Xu, "Neural-network-based decentralized adaptive output-feedback control for large-scale stochastic nonlinear systems," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 6, pp. 1608–1619, Dec. 2012.
[37] D. Wang and J. Huang, "Neural network-based adaptive dynamic surface control for a class of uncertain nonlinear systems in strict-feedback form," *IEEE Trans. Neural Netw.*, vol. 16, no. 1, pp. 195–202, Jan. 2005.
[38] S. Yang, Z. Li, R. Cui, and B. Xu, "Neural network-based motion control of an underactuated wheeled inverted pendulum model," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 11, pp. 2004–2016, Nov. 2014.
[39] Y. Li and S. Tong, "Adaptive fuzzy control with prescribed performance for block-triangular-structured nonlinear systems," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 3, pp. 1153–1163, Jun. 2018. doi: 10.1109/TFUZZ.2017.2710950.
[40] S. I. Han and J. M. Lee, "Improved prescribed performance constraint control for a strict feedback non-linear dynamic system," *IET Control Theory Appl.*, vol. 7, no. 14, pp. 1818–1827, Sep. 2013.
[41] S. I. Han and J. M. Lee, "Fuzzy echo state neural networks and funnel dynamic surface control for prescribed performance of a nonlinear dynamic system," *IEEE Trans. Ind. Electron.*, vol. 61, no. 2, pp. 1099–1112, Feb. 2014.
[42] S. C. Tong, S. Sui, and Y. Li, "Fuzzy adaptive output feedback control of MIMO nonlinear systems with partial tracking errors constrained," *IEEE Trans. Fuzzy Syst.*, vol. 23, no. 4, pp. 729–742, Aug. 2015.
[43] Y.-X. Li and G.-H. Yang, "Fuzzy adaptive output feedback fault-tolerant tracking control of a class of uncertain nonlinear systems with nonaffine nonlinear faults," *IEEE Trans. Fuzzy Syst.*, vol. 24, no. 1, pp. 223–234, Feb. 2016.
[44] L. Ma, Z. Wang, Y. Liu, and F. E. Alsaadi, "A note on guaranteed cost control for nonlinear stochastic systems with input saturation and mixed time-delays," *Int. J. Robust Nonlinear Control*, vol. 27, no. 18, pp. 4443–4456, Dec. 2017.
[45] G. Lai, C. Wen, Z. Liu, Y. Zhang, C. L. P. Chen, and S. Xie, "Adaptive compensation for infinite number of actuator failures based on tuning function approach," *Automatica*, vol. 87, pp. 365–374, Jan. 2018.
[46] L. Ma, Z. Wang, Y. Liu, and F. E. Alsaadi, "Exponential stabilization of nonlinear switched systems with distributed time-delay: An average dwell time approach," *Eur. J. Control*, vol. 37, pp. 34–42, Sep. 2017.
[47] C. Chen, H. Modares, K. Xie, F. L. Lewis, Y. Wan, and S. Xie, "Reinforcement learning-based adaptive optimal exponential tracking control of linear systems with unknown dynamics," *IEEE Trans. Autom. Control*, to be published.
[48] L. Ma, Z. Wang, Q.-L. Han, and Y. Liu, "Dissipative control for nonlinear Markovian jump systems with actuator failures and mixed time-delays," *Automatica*, vol. 98, pp. 358–362, Dec. 2018.
[49] C. Chen, F. L. Lewis, S. Xie, H. Modares, Z. Liu, S. Zuo, and A. Davoudi, "Resilient adaptive and H_∞ controls of multi-agent systems under sensor and actuator faults," *Automatica*, vol. 102, pp. 19–26, Apr. 2019.
[50] L. Ma, Z. Wang, Q.-L. Han, and H.-K. Lam, "Envelope-constrained H_∞ filtering for nonlinear systems with quantization effects: The finite horizon case," *Automatica*, vol. 93, pp. 527–534, 2018.
[51] X. Wan, Z. Wang, M. Wu, and X. Liu, "H_∞ state estimation for discrete-time nonlinear singularly perturbed complex networks under the round-robin protocol," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 2, pp. 415–426, Feb. 2019.
[52] X. Wan, Z. Wang, M. Wu, and X. Liu, "State estimation for discrete time-delayed genetic regulatory networks with stochastic noises under the round-robin protocols," *IEEE Trans. Nanobiosci.*, vol. 17, no. 2, pp. 145–154, Apr. 2018.
[53] Q. Li, B. Shen, Y. Liu, and F. E. Alsaadi, "Event-triggered H_∞ state estimation for discrete-time stochastic genetic regulatory networks with Markovian jumping parameters and time-varying delays," *Neurocomputing*, vol. 174, pp. 912–920, Jan. 2016.
[54] X. Wan, Z. Wang, Q.-L. Han, and M. Wu, "Finite-time H_∞ state estimation for discrete time-delayed genetic regulatory networks under stochastic communication protocols," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 65, no. 10, pp. 3481–3491, Oct. 2018.

YI LYU received the B.S. degree in automatic engineering from the Taiyuan University of Technology, Taiyuan, China, in 2003, and the M.S. degree from the Guangdong University of Technology, Guangzhou, China, in 2006, where he is currently pursuing the Ph.D. degree with the Department of Automation. He has been an Associate Professor with the Department of Computer, University of Electronic Science and Technology of China Zhongshan Institute, since 2014. His research interests include reliability modeling using stochastic process, burn-in, accelerated degradation test, maintenance optimization, and remaining useful life estimation for complex systems.

GUANYU LAI received the B.S. degree in electrical engineering and automation and the Ph.D. degree in control science and engineering from the Guangdong University of Technology, Guangdong, China, in 2012 and 2017, respectively. He was a Research Assistant with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, from 2015 to 2016. He was a Visiting Assistant Professor with the Department of Electrical and Computer Engineering, University of Virginia, USA, from 2017 to 2018. His research interests include the robust adaptive control of uncertain systems, and its applications to robotic systems and networked systems.

CI CHEN received the B.E. and Ph.D. degrees from the School of Automation, Guangdong University of Technology, Guangzhou, China, in 2011 and 2016, respectively. He was a Research Assistant with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, from 2015 to 2016. From 2016 to 2018, he has been with The University of Texas at Arlington and The University of Tennessee at Knoxville as a Research Associate. He is currently with the School of Automation, Guangdong University of Technology and also with the School of Electrical and Electronic Engineering, Nanyang Technological University. His research interests include reinforcement learning, nonlinear system control, resilient control, and computational intelligence. He is an Editor of the *International Journal of Robust and Nonlinear Control* and an Associate Editor of *Advanced Control for Applications: Engineering and Industrial Systems*.

YUN ZHANG received the B.S. and M.S. degrees from Hunan University, Changsha, China, in 1982 and 1986, respectively, and the Ph.D. degree from the South China University of Science and Technology, Guangzhou, China, in 1998, all in automatic engineering. He is currently a Full Professor with the School of Automation, Guangdong University of Technology, Guangzhou. His research interests include intelligent control systems, multi-agent systems, neural networks, and signal processing.