



Energy-Efficiency-Oriented Vision Feedback Control of QCSP Systems: Linear ADRC Approach

Shengming Li^{1,2} and Lin Feng^{1,2*}

¹School of Computer Science and Technology, Dalian University of Technology, Dalian, China, ²School of Innovation and Entrepreneurship, Dalian University of Technology, Dalian, China

How to save the energy of unmanned aerial vehicles (UAVs) and then enable long-distance transport is a very real and difficult task. However, for UAVs, the classic object detection algorithm, such as the deep convolutional neural network-based object detection algorithm and the classic flight control algorithm, such as the PID-based position control algorithm, require significant energy, which limits the application scenarios of the UAV system. In view of this problem, this paper proposes a lightweight object detection network and a linear active disturbance rejection controller (LADRC) for the quadrotor with the cable-suspended payload (QCSP) system to improve energy efficiency. The system uses a YOLOV3 network and embeds it into the Jesson NX mobile platform to accurately detect the target position. Furthermore, a nonlinear velocity controller with a cable-suspended structure to control the velocity of the payload, a LADRC algorithm is adopted to achieve fast and accurate control of the payload position. Simulations and real flight experiments show that the proposed object detection algorithm and the LADRC control strategy can save the energy of drone effectively.

Keywords: cable-suspended payload, quadrotor UAV, energy efficiency, object detection, linear active disturbance rejection controller, model compression

OPEN ACCESS

Edited by:

Xun Shen,

Tokyo Institute of Technology, Japan

Reviewed by:

Hongxu Zhang,

Harbin University of Science and
Technology, China

Guoguang Wen,

Beijing Jiaotong University, China

*Correspondence:

Lin Feng

fenglin@dlut.edu.cn

Specialty section:

This article was submitted to

Smart Grids,

a section of the journal

Frontiers in Energy Research

Received: 29 January 2022

Accepted: 08 February 2022

Published: 15 March 2022

Citation:

Li S and Feng L (2022) Energy-Efficiency-Oriented Vision Feedback Control of QCSP Systems: Linear ADRC Approach.

Front. Energy Res. 10:865069.

doi: 10.3389/fenrg.2022.865069

1 INTRODUCTION

With the development of unmanned aerial vehicle (UAV) technology (Wu et al., 2018), drone transport has become an important branch of UAV applications. The quadrotor with the cable-suspended payload (QCSP) (Lv et al., 2020; Lv et al., 2021) equipped with a camera and an embedded platform, is of great relevance to the realization of rescue and transport tasks. The QCSP actively adjusts the UAV's own attitude to quickly reduce the oscillation of the suspended load and then runs the vision algorithm through the embedded platform to process the images from the camera to obtain an accurate target position for the drone. Based on the target position information, the QCSP needs to reach the target position quickly and stably. However, in the transportation process, in addition to the energy required for drone flight, the object detection algorithm and the QCSP flight control strategy also consume great energy. Therefore, by considering the limited battery capacity of the drone, it is important to improve the energy efficiency of the QCSP systems.

Recent decades witness great progress in object detection with the development of convolutional neural networks (CNNs). To obtain a powerful network, numerous efforts are made to build large and complex architecture with high computation and energy consumption, which restricts its application on embedding devices such as a drone. To get the light network, Z. Liu et al. (Liu et al., 2017) utilize scaling factors to value the significance of connections and remove these under a

threshold. This efficient method works well for classification networks. However, it lacks effectiveness for detection networks. To make the network compression method suitable to detection network, Z. Xie *et al.* (Xie *et al.*, 2020) introduce location-aware loss for network compression, which helps in preserving the comprehensive ability of the detection network. These methods are not optimal for saving energy because they do not adopt an energy-aware function during compressing.

To realize the path following for the QCSP, Qian, *et al.* (Qian and Liu, 2019) propose a controller based on uncertainty and an interference estimator. Hao, *et al.* (Hao *et al.*, 2021) propose a nonlinear, robust, fault-tolerant, position-tracking, control law for a tilt tri-rotor UAV, thus avoiding rear servo's stuck fault together with parametric uncertainties and unknown external disturbances. To enable a multirotor UAV to achieve static hovering, Mochida, *et al.* (Mochida *et al.*, 2021) propose a geometric method that reveals the relationship between the position of the center of mass (CoM) and the rotor placement of a multirotor UAV with upward-oriented rotors. These methods can effectively help UAVs accomplish their tasks, but they do not take into account the energy limitation of UAVs; the algorithm is complex and not applicable to the QCSP.

Although many researchers have done meaningful work and achieved results, there are still some challenges in vision processing and position control in the energy-efficiency-oriented QCSP, mainly as follows: 1) Object detection technique-based CNNs can obtain accurate target information for a UAV, but the improved detection performance of deep neural networks also brings huge energy consumption, which is not friendly to the QCSP. 2) The QCSP needs to control the load stably and reach the target position quickly, whereas the traditional PID controller usually needs a long adjustment time for the UAV to reach the target position, which is also not conducive to the QCSP system to save energy for long-distance transportation.

To improve the energy efficiency of the QCSP systems, we propose a lightweight object detection algorithm and an LADRC payload position control strategy for the QCSP. Specifically, the object detection model is compressed by network scaling factors and an energy-aware penalty, which enables the YOLOV3 network to run on the Jetson NX embedded platform of the QCSP with low energy consumption. In addition, an efficient control strategy in the form of a string stage is used to overcome the under-actuated characteristics of the QCSP, which includes attitude, swing angle, load velocity, and load position subcontrollers. The contributions of this paper to the energy saving of QCSP mainly include 1) a new QCSP experimental platform with embedded vision detection is constructed, and a lightweight object detection network is used to obtain position information; 2) an LADRC algorithm is used to control the payload position quickly and efficiently.

The remainder of this paper is structured as follows: **Section 2** introduces the dynamic model and object detection algorithm of the QCSP in detail; controller design, including the LADRC position control algorithm for the QCSP is introduced in **Section 3**; in **Section 4**, the effectiveness of the proposed QCSP system is verified through experiments. Conclusions are drawn in **Section 5**.

2 DYNAMIC MODEL AND OBJECT DETECTION ALGORITHM OF THE QCSP

There are three reference frames to describe the QCSP (Lv *et al.*, 2020) system (see left of **Figure 1**): the inertial frame $\mathcal{I}\{X_i, Y_i, Z_i\}$, the quadrotor body frame $\mathcal{B}\{X_b, Y_b, Z_b\}$, and the payload body frame $\mathcal{B}_p\{X_p, Y_p, Z_p\}$. What needs to be mentioned is that the inertial frame follows the North-East-Down (NED) notation. For the quadrotor body frame, Z_b points down, the X_b toward the front direction, and the Y_b toward the right direction. Based on the reference frames, some variables are defined. The generalized coordinates $q = [\xi^T \ \eta^T \ \sigma^T]^T \in \mathbb{R}^8$, where $\xi = [x \ y \ z]^T \in \mathbb{R}^3$ denotes the coordinate of the quadrotor's CoG under the inertial frame \mathcal{I} ; $\eta = [\phi \ \theta \ \psi]^T \in \mathbb{R}^3$ denotes the attitude angle of the quadrotor in the Euler coordinate system, and ϕ means the roll angle, θ means pitch angle, and ψ means yaw angle; $\sigma = [\alpha \ \beta]^T \in \mathbb{R}^2$ denotes the swing angle of the payload, where α and β are the roll and pitch angles of the cable, respectively. The boundaries of the quadrotor attitude and the swing angle are limited as

$$\phi, \theta, \alpha, \beta \in (-\pi/2, \pi/2). \quad (1)$$

The coordinate $\delta = [x_p \ y_p \ z_p]^T$ of the payload's CoG in inertial frame \mathcal{I} can be given by ξ and σ :

$$x_p = x + l \cos \alpha \beta, \quad y_p = y - l \sin \alpha, \quad z_p = z + l \cos \alpha \beta, \quad (2)$$

and the velocity of the payload is expressed by $\dot{\delta} = [\dot{x}_p \ \dot{y}_p \ \dot{z}_p]^T$. In addition, I_n and $\mathbf{0}_{m \times n}$ represent the n -dimensional identity matrix and $m \times n$ dimensional null matrix, respectively. c and s are used to represent \cos and \sin , respectively.

Following previous work (Lv *et al.*, 2020), the dynamic model of the QCSP system is described by the following equations:

$$\ddot{\delta} = (\mathbf{R}_G F_l - m_q \ddot{\xi} + m_q \mathbf{g} + \mathbf{D}_\xi + \mathbf{D}_\delta) / m_p + \mathbf{g}, \quad (3a)$$

$$\ddot{\eta} = J_q^{-1} (\tau_\eta + \mathbf{A} - \dot{J}_q \dot{\eta} + \mathbf{D}_\eta), \quad (3b)$$

$$\ddot{\sigma} = -M_1^{-1} (M_2 \ddot{\xi} + V_{d\sigma} - C - \mathbf{D}_\sigma), \quad (3c)$$

where $F_l = \mathbf{R}_G F_l - m_q \ddot{\xi} + m_q \mathbf{g} + \mathbf{D}_\xi$ is the tensile force of the cable on the payload.

For dynamic model described by **Eq. 3a**, m_q is the mass of the quadrotor, m_p is the mass of the payload, \mathbf{g} is the gravity acceleration, \mathbf{D}_ξ and \mathbf{D}_δ are the air drag forces that act on the quadrotor and the payload, respectively. \mathbf{R}_G is the projection vector in the inertial frame \mathcal{I} of the unit vector on the axis Z_b .

For the dynamic submodel described by **Eq. (3b)**, \mathbf{D}_η denotes the aerodynamic drag torque on the quadrotor. What needs to be mentioned is that $\mathbf{A} = [A_\phi \ A_\theta \ 0]^T$ is given in **Eq. 12a** of (Lv *et al.*, 2020), and the inertial matrix J_q is given in **Eq. 4** of (Lv *et al.*, 2020).

For the dynamic model described by **Eq. 3c**, the drag torque \mathbf{D}_σ on the payload is given by

$$\mathbf{D}_\sigma = \mathbf{I}_{2 \times 3} (\mathbf{I} \times \mathbf{R}_i^{bp} \mathbf{D}_\sigma), \quad (4)$$

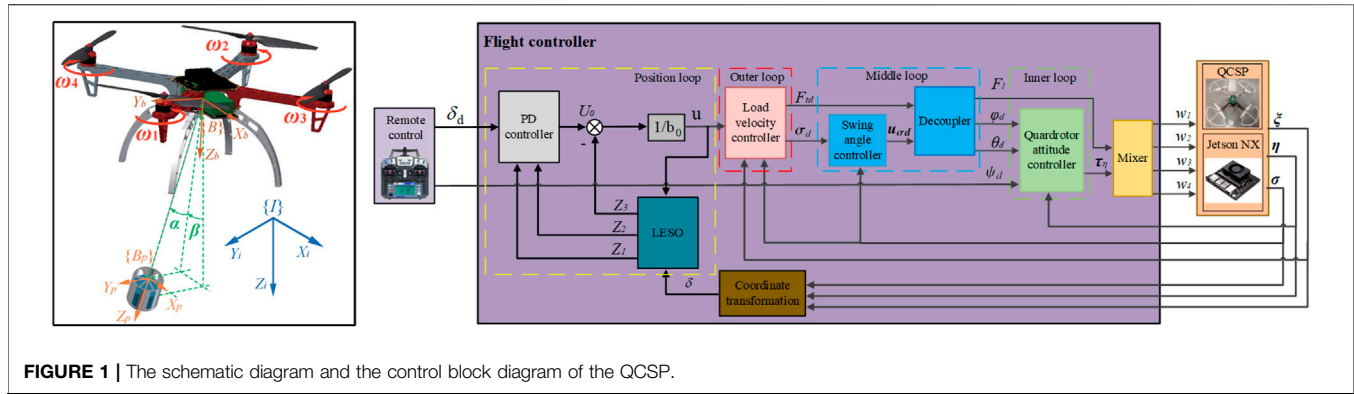


FIGURE 1 | The schematic diagram and the control block diagram of the QCSP.

where $\mathbf{I}_{2 \times 3} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$, $\mathbf{l} = [0 \ 0 \ l]^T$, l is the length of cable. $\mathbf{M}_1 = m_p l^2 \text{diag}(1, c^2 \alpha)$, $\mathbf{M}_2 = -m_p l \begin{bmatrix} sas\beta & c\alpha & sac\beta \\ -cac\beta & 0 & cas\beta \end{bmatrix}$, $\mathbf{V}_{d\sigma} = [m_p g l sac\beta \ m_p g l cas\beta]^T$, $\mathbf{C} = [C_\alpha \ C_\beta]^T$, with $C_\alpha = -m_p l^2 sac\alpha\beta^2$, $C_\beta = 2m_p l^2 sac\alpha\dot{\alpha}\dot{\beta}$. To facilitate the controller design, the dynamic model (see Eq. 3) is rewritten as

$$\ddot{\delta} = (\mathbf{F}_t + \mathbf{D}_\sigma) / m_p + \mathbf{g}, \quad (5a)$$

$$\ddot{\eta} = \mathbf{J}_q^{-1} (\boldsymbol{\tau}_\eta + \boldsymbol{\tau}_{\eta e}), \quad (5b)$$

$$\ddot{\sigma} = \mathbf{M}_\sigma \ddot{\xi} + \mathbf{F}_{\sigma e} + \mathbf{M}_1^{-1} \mathbf{D}_\sigma, \quad (5c)$$

where $a = 1/m_p$, $\boldsymbol{\tau}_{\eta e} = \mathbf{A} - \dot{\mathbf{J}}_q \dot{\eta} + \mathbf{D}_\eta$,

$$\mathbf{F}_{\sigma e} = \mathbf{M}^{-1} (\mathbf{C} - \mathbf{V}_{d\sigma}) = \begin{bmatrix} -sac\alpha\dot{\beta}^2 - sas\beta g / l \\ 2s\alpha / c\alpha\dot{\alpha}\dot{\beta} - sac\beta g / l \end{bmatrix},$$

$$\mathbf{M}_\sigma = -\mathbf{M}_1^{-1} \mathbf{M}_2 = \begin{bmatrix} sas\beta & c\alpha & sac\beta \\ -c\beta/c\alpha & 0 & s\beta/c\alpha \end{bmatrix} / l.$$

It can be found that \mathbf{M}_σ , $\mathbf{F}_{\delta e}$, and $\mathbf{F}_{\sigma e}$ do not contain a .

Apart from dynamic model controlling the basic attitude of the quadrotor, the motion of the quadrotor depends on the guide of the object detection network (ODN). Currently, because of accuracy and effectiveness, YOLOV3 (Redmon and Farhadi, 2018) is adopted in a growing number of real-world situations. However, this ODN method is computationally expensive; hence, it creates huge energy consumption, which is not friendly to the QCSP system. Therefore, compressing the ODN to obtain a lightweight ODN is essential for deploying YOLOV3 on the quadrotor. As mentioned, preserving computational performance and saving energy cost simultaneously are challenging issues. For preserving performance of the network, we utilize the sparsity-induced penalty to retrain a sparsity network indicated by scaling factors. Then, these low-significance connections distinguished by scaling factors are removed to achieve network compression. Considering the energy consumption, we add an energy-aware penalty to supervise the compression process. Specifically, the retrain objective is given by

$$L = \sum_{\mathbf{X}, \mathbf{Y}} l(f(\mathbf{X}, \mathbf{W}), \mathbf{Y}) + \lambda \sum_{\gamma \in \Gamma} \|\gamma\|_1 + \alpha \sum_{i \in N} (E_{comp}^{(i)} + E_{data}^{(i)}), \quad (6)$$

where $l(\cdot)$ denotes the supervised training loss; for YOLOV3, $l(\cdot)$ is the detection loss proposed in (Redmon and Farhadi, 2018). (\mathbf{X}, \mathbf{Y}) denote the retrain input and label, \mathbf{W} denotes the learnable weights, $\|\cdot\|_1$ denotes the L1-norm function, γ is a scaling factor and Γ is the scaling factors set, $\|\gamma\|_1$ is used as a sparsity-induced penalty. In practice, the learnable γ in batch normalization (Ioffe and Szegedy, 2015) is widely adopted as a scaling factor. $E_{comp}^{(i)}$ denotes the energy consumption for computation of the i th layer, whereas $E_{data}^{(i)}$ denotes the energy consumption for data access of the i th layer, N means the whole network consisting of N layers. $E_{comp}^{(i)} + E_{data}^{(i)}$ is utilized as the energy-aware penalty. λ and α balance three items. Following the principles of (Yang et al., 2018), the $E_{comp}^{(i)}$ of the normal convolutional layer is given by

$$\begin{cases} E_{comp}^{(i)} = e_{MAC} \cdot h' \cdot w' \cdot \|\mathbf{W}^{(i)}\|_0 \\ h' = \lfloor (h + 2p - r) / s \rfloor + 1 \\ w' = \lfloor (w + 2p - r) / s \rfloor + 1 \end{cases}, \quad (7)$$

where e_{MAC} denotes the energy consumption of one systolic array MAC (Kung, 1982) (a kind of hardware widely used in GPU or TPU) operation, whereas h and w denote the height and weight of the convolutional layer input, $\|\cdot\|_0$ denotes the L0-norm function, p, r, s denote the convolutional arguments, i.e., padding, kernel size, and stride. Data access energy $E_{data}^{(i)}$ depends on the hardware architecture, i.e., systolic array, which is complex and not helpful for understanding our method. We just describe it as a function of $\mathbf{X}^{(i)}$, h, w, p, r, s : $E_{data}^{(i)} = E_{data}^{(i)}(\mathbf{X}^{(i)}, h, w, p, r, s)$, where $\mathbf{X}^{(i)}$ denotes the input of the i th convolutional layer. The other omitted items depend on the specific architecture of the hardware, e.g., bus bandwidth.

Obviously, Eq. 6 gives a meaningful objective. However, it is hard to optimize because of $E_{data}^{(i)}$ and $E_{comp}^{(i)}$ are not functions of scaling factors γ . To make the energy-aware penalty influence γ , we redescribe Eq. 6 as

$$L' = \sum_{\mathbf{X}, \mathbf{Y}} l(f(\mathbf{X}, \mathbf{W}), \mathbf{Y}) + \lambda \sum_{i \in N} (E_{comp}^{(i)} + E_{data}^{(i)}) \cdot \|\mathbf{y}^{(i)}\|_1,$$

where $\mathbf{y}^{(i)}$ is a scaling factor vector of the i th layer. Then, we optimize the above equation to obtain a sparsity distribution of

scaling factors and remove these low-significance connections. After that, the compact network is fine-tuned for several iterations to resume.

Finally, a network deployed on a computation and energy-limited platform could be accessed. We utilize this compact network to provide location and category information of the target object to the quadrotor as a basis for flight adjustment.

3 CONTROLLER DESIGN

Because of the underactuated character of the QCSP, the proposed controller mainly consists of two parts: the cascade controller for attitude self-stabilization and the active disturbance rejection controller for position control (see the right subfigure of **Figure 1**). Referring to (Lv et al., 2020), the design process of the cascade controller mainly consists of three parts: inner-loop attitude, middle-loop swing angle, and outer-loop velocity subcontrollers.

3.1 Tracking Errors

Errors associated with the dynamics of the QCSP are given as follows:

$$\mathbf{e}_{\eta, p_\eta} = [\mathbf{e}_\eta^\top \mathbf{e}_{p_\eta}^\top]^\top, \quad (8)$$

$$\mathbf{e}_\eta = \boldsymbol{\eta}_d - \boldsymbol{\eta}, \quad \mathbf{e}_{p_\eta} = \dot{\boldsymbol{\eta}}_d - \dot{\boldsymbol{\eta}} + \mathbf{K}_\eta \mathbf{e}_\eta, \quad (9)$$

$$\mathbf{e}_{\sigma, p_\sigma} = [\mathbf{e}_\sigma^\top \mathbf{e}_{p_\sigma}^\top]^\top, \quad (10)$$

$$\mathbf{e}_\sigma = \boldsymbol{\sigma}_d - \boldsymbol{\sigma}, \quad \mathbf{e}_{p_\sigma} = \dot{\boldsymbol{\sigma}}_d - \dot{\boldsymbol{\sigma}} + \mathbf{K}_\sigma \mathbf{e}_\sigma, \quad (11)$$

$$\mathbf{e}_\delta = \boldsymbol{\delta}_d - \boldsymbol{\delta}, \quad \dot{\mathbf{e}}_\delta = \dot{\boldsymbol{\delta}}_d - \dot{\boldsymbol{\delta}}, \quad (12)$$

where $\boldsymbol{\delta}_d = [\dot{x}_{pd} \ \dot{y}_{pd} \ \dot{z}_{pd}]^\top$, $\boldsymbol{\sigma}_d = [\alpha_d \ \beta_d]^\top$, and $\boldsymbol{\eta}_d = [\phi_d \ \theta_d \ \psi_d]^\top$ with the desired attitude $\boldsymbol{\eta}_d$, and the desired position $\boldsymbol{\delta}_d$, which can be determined by the object detection algorithm proposed in the above section. The positive definite diagonal matrixes $\mathbf{K}_\sigma = \text{diag}(k_\alpha, k_\beta)$ and $\mathbf{K}_\eta = \text{diag}(k_\phi, k_\theta, k_\psi)$. The attitude $\boldsymbol{\eta}$ in (9) and the velocity $\dot{\boldsymbol{\xi}}$ in (12) of the quadrotor are measured by the IMU integrated in the flight control system. The payload's velocity $\dot{\boldsymbol{\delta}}$ in (12) can be calculated by $\boldsymbol{\eta}$ and $\dot{\boldsymbol{\xi}}$. From **Eq. 9**, the attitude error dynamic of the quadrotor is obtained as

$$\dot{\mathbf{e}}_\eta = \mathbf{e}_{p_\eta} - \mathbf{K}_\eta \mathbf{e}_\eta, \quad \dot{\mathbf{e}}_{p_\eta} = \dot{\boldsymbol{\eta}}_d - \dot{\boldsymbol{\eta}} + \mathbf{K}_\eta (\mathbf{e}_{p_\eta} - \mathbf{K}_\eta \mathbf{e}_\eta). \quad (13)$$

The swing angle error dynamics of the payload are deduced from **Eq. 11**:

$$\dot{\mathbf{e}}_\sigma = \mathbf{e}_{p_\sigma} - \mathbf{K}_\sigma \mathbf{e}_\sigma, \quad \dot{\mathbf{e}}_{p_\sigma} = \dot{\boldsymbol{\sigma}}_d - \dot{\boldsymbol{\sigma}} + \mathbf{K}_\sigma (\mathbf{e}_{p_\sigma} - \mathbf{K}_\sigma \mathbf{e}_\sigma). \quad (14)$$

3.2 Load Velocity Controller

3.2.1 Inner-Loop Attitude Controller

Considering the subsystem (see **Eq. 5b**), the inner-loop subcontroller (see the right of **Figure 1**) is used to control the attitude $\boldsymbol{\eta}$ of the quadrotor, which is measured by the inertial measurement unit (IMU) integrated in the flight control system of the quadrotor. The control torque $\boldsymbol{\tau}_\eta \in \mathbb{R}^3$ of the inner-loop controller is given by

$$\boldsymbol{\tau}_\eta = \mathbf{J}_q [(I_3 - \mathbf{K}_\eta^2) \mathbf{e}_\eta + (\mathbf{K}_\eta + \mathbf{K}_{p_\eta}) \mathbf{e}_{p_\eta}] - \boldsymbol{\tau}_{\eta e}, \quad (15)$$

where $\mathbf{K}_{p_\eta} = \text{diag}(k_{p_\phi}, k_{p_\theta}, k_{p_\psi})$ denotes a constant positive definite matrix.

3.2.2 Middle-Loop Swing Angle Controller

Referring to **Eq. 16** in (Lv et al., 2020), the adaptive swing angle controller is applied to make $\boldsymbol{\sigma}$ follow the desired $\boldsymbol{\sigma}_d$. Noting **Eq. 5c**, $\mathbf{M}_\sigma \ddot{\boldsymbol{\xi}}$ is taken as the visual control input. The desired visual control input is designed as

$$\mathbf{M}_\sigma \ddot{\boldsymbol{\xi}}_d = (\mathbf{I}_2 - \mathbf{K}_\sigma^2) \mathbf{e}_\sigma + (\mathbf{K}_\sigma + \mathbf{K}_{p_\sigma}) \mathbf{e}_{p_\sigma} - \mathbf{F}_{\sigma e} - \mathbf{M}_1^{-1} \mathbf{D}_\sigma, \quad (16)$$

where $\mathbf{K}_{p_\sigma} = \text{diag}(k_{p_\alpha}, k_{p_\beta})$ is constant positive definite.

3.3.3 Decoupler

For the desired acceleration $\mathbf{M}_\sigma \ddot{\boldsymbol{\xi}}_d$ in **Eq. 16** generated by the aforementioned adaptive swing angle controller, the quadrotor's lift force F_l and the desired attitude ϕ_d can be decoupled by $\mathbf{M}_\sigma \ddot{\boldsymbol{\xi}}_d$ and F_{td} . Considering the constraint given in (1) and the mechanisms of the quadrotor maneuvers, we have $F_{lzd} > 0$, $\theta_d, \phi_d \in (-\pi/2, \pi/2)$. The decoupling result is given by

$$\theta_d = \arctan\left(\frac{F_{lxd} \mathbf{c}\boldsymbol{\psi} + F_{lyd} \mathbf{s}\boldsymbol{\psi}}{F_{lzd}}\right), \quad (17a)$$

$$\phi_d = -\arctan\left(\frac{-F_{lxd} \mathbf{s}\boldsymbol{\psi} + F_{lyd} \mathbf{c}\boldsymbol{\psi}}{F_{lzd}}\right) \mathbf{c}\theta_d, \quad (17b)$$

$$F_l = F_{lzd} / (\mathbf{c}\phi_d \mathbf{c}\theta_d). \quad (17c)$$

3.3.4 Outer-Loop Velocity Controller

As illustrated in the outer-loop part of the right subfigure in **Figure 1**, the outer-loop velocity controller is utilized to track the desired velocity $\dot{\boldsymbol{\delta}}_d$ for the translational dynamic (see **Eq. 5a**) of the payload. The desired tensile force $\mathbf{F}_{td} = [F_{txd} \ F_{tyd} \ F_{tzd}]^\top$ is designed as

$$\mathbf{F}_{td} = \mathbf{K}_\delta \mathbf{e}_\delta - \mathbf{F}_{de} - m_p \mathbf{g}, \quad (18)$$

with a constant positive definite diagonal matrix $\mathbf{K}_\delta = \text{diag}(k_{\dot{x}_p}, k_{\dot{y}_p}, k_{\dot{z}_p})$. Considering the constraints in (1) and that the cable is always taut and there is tensile force on the cable, $F_{tzd} > 0$, $\alpha_d, \beta_d \in (-\pi/2, \pi/2)$. Referring to **Eq. 26** in (Lv et al., 2020), the desired magnitude F_{td} of tensile force and the desired swing angles α_d and β_d are given by

$$F_{td} = F_{tzd} / (\mathbf{c}\alpha_d \mathbf{c}\beta_d), \quad (19a)$$

$$\beta_d = \arctan(F_{txd} / F_{tzd}), \quad (19b)$$

$$\alpha_d = -\arctan(F_{tyd} \mathbf{c}\beta_d / F_{tzd}). \quad (19c)$$

3.3 LADRC Based Position Controller

LADRC is utilized for the position control of quadrotors. There are two parts in the LADRC, including the linear extended state observer (LESO) and PD controller. The LESO estimates the internal and external disturbances of the system through an extended state, which is called total disturbance, and compensates the control variables. Therefore, the integrator used in traditional PID to eliminate static error under

constant disturbance is no longer necessary. The system can be stabilized by PD controller.

Referring to the controller built by Gao in (Han, 2009) and taking δ_d as the expected input of the controller and δ as feedback, the designed LADRC block diagram is as follows:

3.3.1 LESO

Compared with ESO (Han, 2009), LESO introduces the frequency domain method. It connects the parameters with the observer bandwidth, making the parameter tuning more convenient.

The LESO for the position control of the QCSP is designed as

$$\dot{z} = [A - LC]z - [B, L]u_c, \quad (20a)$$

$$y_c = z, \quad (20b)$$

where $u_c = [u \ \delta]^T$ is the input of LESO, and y_c is the output.

Besides this, $A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$, $B = \begin{bmatrix} 0 \\ b_0 \\ 0 \end{bmatrix}$, $C = [1 \ 0 \ 0]$,

$L = [3\omega_0 \ 3\omega_0^2 \ \omega_0^3]^T$. Here, b_0 can be adjusted according to the step response of the system. ω_0 is the observer bandwidth.

3.3.2 PD Controller

Under the action of LESO, the linear PD controller can stabilize the system. Besides this, the proportional coefficient and differential time constant are related to the controller bandwidth, which simplifies the tuning of the controller.

The PD controller is designed as

$$u_0 = k_p(\delta_d - z_1) - k_d z_2 \quad (21)$$

where δ_d is the expected input of the controller. z_1 and z_2 are observer states from LESO. k_p and k_d are the parameters of the controller gain matrix $K = [k_d \ k_p]^T$ to be designed. We choose $k_p = \omega_c^2$, $k_d = 2\omega_c$ with the controller bandwidth ω_c .

Finally, the control quantity u_0 with the total disturbance z_3 has to be compensated, and the control quantity u is $u = \frac{u_0 - z_3}{b}$, where b is the gain factor.

For the position control, it is necessary to obtain the position coordinates of the target point, but the camera feeds back the pixel coordinates, which should be compensated by attitude angle and height information. Besides this, due to the relative displacement between the quadcopter and the load, the coordinates of the target point collected by the camera relative to the quadcopter should be transformed into the coordinates relative to the load.

4 EXPERIMENT

To verify the effectiveness of the proposed algorithm, a QCSP experimental platform was created. The payload is connected to the bottom of the F450 quadcopter by a Cadan joint, and the Jetson NX board is fixed to the bottom plate with the camera as shown in **Figure 3A**.

Before the flight test, we simulated the designed LADRC control strategy and compared it with the conventional PID algorithm as shown in **Figure 2**. The parameters of LADRC and PID are obtained

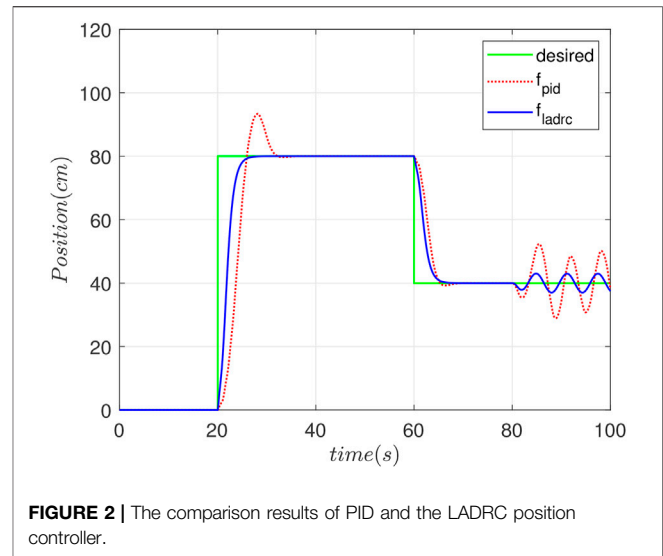


FIGURE 2 | The comparison results of PID and the LADRC position controller.

by many experiments according to overshoot and response time. Among them, “desired” is the target position curve after transformation, “ f_{pid} ” is the position curve of the payload under PID control. “ f_{ladrc} ” is the position curve of the payload under LADRC control. It can be seen that, at 20 s, given a target position of 80 cm, the payload achieved steady state in 7 s without overshoot under the LADRC control, while the state of the system oscillated and took more than three times as long to stabilize under the PID controller. When a sin disturbance signal is added in 80 s, the LADRC can obviously suppress the disturbance. As a result, it can be deduced that the LADRC controller is effective in saving QCSP energy.

The QCSP vision deployment hardware is Jetson NX, which runs aarch64 Ubuntu 18.04 as the operating system. PyTorch (Paszke et al., 2019) is used as the retraining, fine-tuning, and inference deep learning software. For efficient inference, popular object detection model YOLOV3 (Redmon and Farhadi, 2018) is adopted, whose backbone is replaced by MobileNet (Howard et al., 2017) from DarkNet. The image size requires 416×416 . The original network is retrained on a PASCAL VOC data set (Everingham et al., 2010) for 50,000 iterations and removes the connections whose scaling factors are lower than the threshold 0.01. Then, the compact network is fine-tuned for 12,000 iterations. The retraining and fine-tuning, which cost a large amount of computing power, are carried out on NVIDIA RTX 2080Ti, and only inference is done on the embedding Jetson NX platform. The target pattern is the helicopter landing area-“H”, and the recognition effect is shown in part (b) of **Figure 3**, where four objects are detected with confidence 0.89, 0.95, 0.98, and 0.97.

Compression results are reported in **Table 1**. Through our method, we saved about 52% energy of the whole network with only 0.7 mean average precision (mAP) dropping. In addition, we also demonstrate the necessity of using sparsity-induced and energy-aware penalties simultaneously. In the case of only the sparsity-induced penalty used, the energy saved is not satisfactory (43% saved), but performance drops a lot (1.4) when only the energy-aware penalty is used.

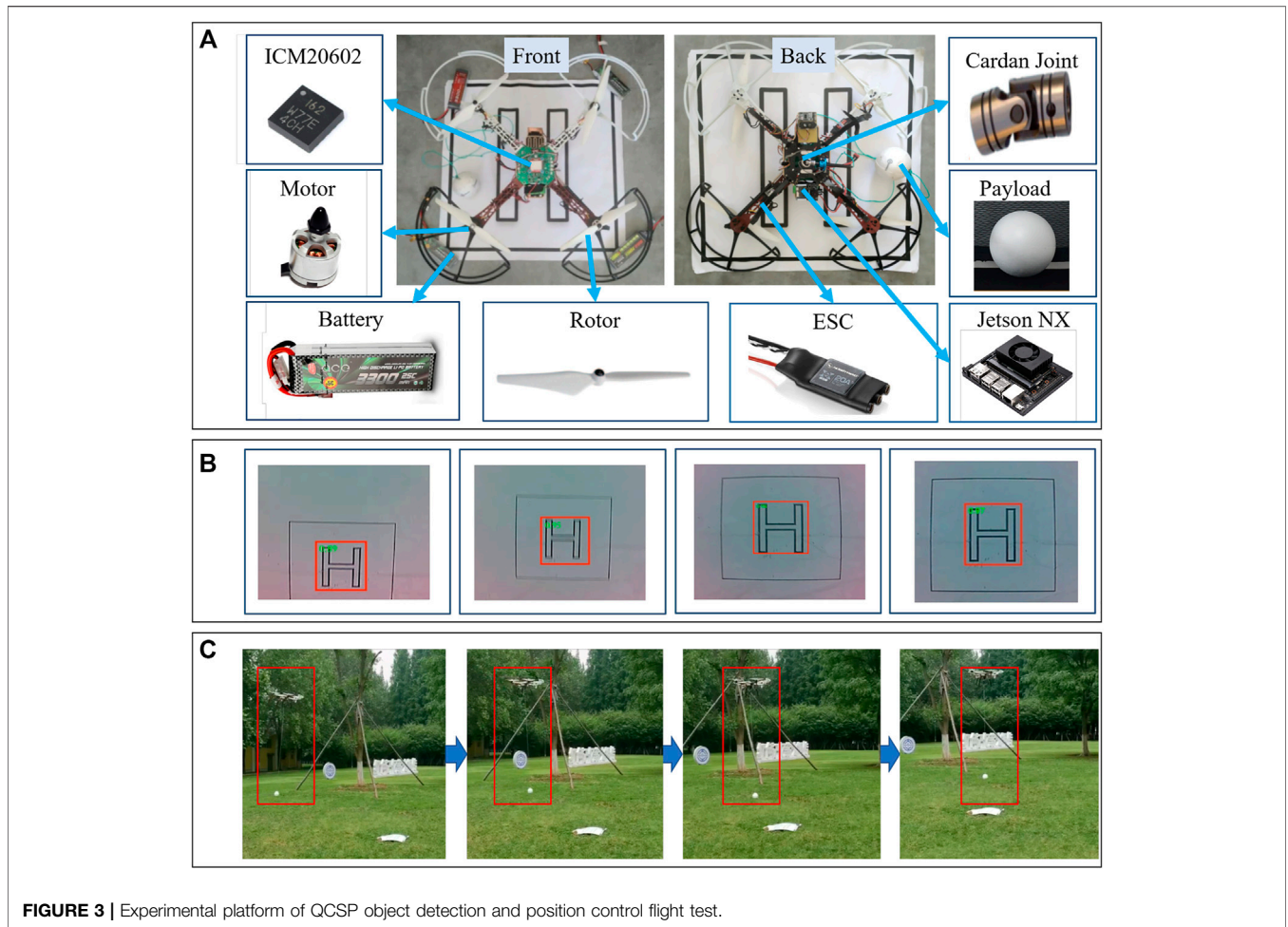


TABLE 1 | Object detection performance on PASCAL VOC. “Penalty” denotes the penalty item that we added for training loss. “Energy” denotes the energy cost for detecting one image. “Energy ↓” denotes the energy saved comparing to original network whose backbone is MobileNet; “mAP” is a common indicator evaluating the performance of detection network.

Backbone	Penalty	Energy (J)	Energy ↓	mAP
DarkNet Redmon and Farhadi (2018)	—	6.61	—	76.1
MobileNet Howard et al. (2017)	—	0.21	—	76.8
MobileNet Howard et al. (2017)	sapsrity-induced Liu et al. (2017)	0.12	43%	75.9
MobileNet Howard et al. (2017)	energy-aware Yang et al. (2018)	0.10	52%	75.4
MobileNet Howard et al. (2017)	Ours	0.10	52%	76.1

During the actual flight experiment, the quadcopter flew along the positive direction of the X axis at the speed of 20 cm/s. When the target point is identified by Jetson NX, it turn into position control mode as shown in **Figure 3C**. From the experiment, it can be seen that the QCSP system can control the payload stably, and when the target position is detected, it can reach the destination quickly and remain stable.

5 CONCLUSION

Energy-efficiency plays a crucial role in the development of UAVs. In this paper, a lightweight YOLOV3 object detection

network with a LADRC-based position controller is proposed to reduce the energy consumption of the QCSP system. The experimental results show that the compressed network can save more than 50% of energy compared with the original network with little accuracy loss, and the LADRC controller has three times faster stabilization time and no overshoot compared with the classic PID controller and has a suppression effect on disturbing signals. Therefore, the work done in this paper can effectively save the energy of the QCSP and improve its range, anti-interference performance, and robustness.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

SL contributed to the architecture, property, and training algorithm, and design of feedback controller of the QCSP

REFERENCES

- Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., and Zisserman, A. (2010). The Pascal Visual Object Classes (Voc) challenge. *Int. J. Comput. Vis.* 88 (2), 303–338. doi:10.1007/s11263-009-0275-4
- Han, J. (2009). From Pid to Active Disturbance Rejection Control. *IEEE Trans. Ind. Electron.* 56 (3), 900–906. doi:10.1109/tie.2008.2011621
- Hao, W., Xian, B., and Xie, T. (2021). Fault Tolerant Position Tracking Control Design for an Tilt Tri-rotor Unmanned Aerial Vehicle. *IEEE Trans. Ind. Electron.* 99, 1. doi:10.1109/TIE.2021.3050384
- Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., et al. (2017). *Mobilenets: Efficient Convolutional Neural Networks for mobile Vision Applications*. California: Google Inc. arXiv preprint arXiv:1704.04861. Available at: <https://arxiv.53yu.com/pdf/1704.04861.pdf%EF%BC%89>.
- Ioffe, S., and Szegedy, C. (2015). “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift,” in International conference on machine learning (PMLR), Lille, France, July 7–9, 2015, 448–456.
- Kung, H.-T. (1982). Why Systolic Architectures? *Computer* 15 (01), 37–46. doi:10.1109/mc.1982.1653825
- Liu, Z., Li, J., Shen, Z., Huang, G., Yan, S., and Zhang, C. (2017). “Learning Efficient Convolutional Networks through Network Slimming,” in Proceedings of the IEEE international conference on computer vision, Venice, Italy, October 22–29, 2017, 2736–2744. doi:10.1109/iccv.2017.298
- Lv, Z.-Y., Li, S., Wu, Y., and Wang, Q.-G. (2021). Adaptive Control for a Quadrotor Transporting a cable-suspended Payload with Unknown Mass in the Presence of Rotor Downwash. *IEEE Trans. Veh. Technol.* 70 (9), 8505–8518. doi:10.1109/tvt.2021.3096234
- Lv, Z.-Y., Wu, Y., and Rui, W. (2020). Nonlinear Motion Control for a Quadrotor Transporting a cable-suspended Payload. *IEEE Trans. Veh. Technol.* 69 (8), 8192–8206. doi:10.1109/tvt.2020.2997733
- Mochida, S., Matsuda, R., Ibuki, T., and Sampei, M. (2021). A Geometric Method of Hoverability Analysis for Multirotor Uavs with Upward-Oriented Rotors. *IEEE Trans. Robotics* 99, 1–15. doi:10.1109/tro.2021.3064101
- Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., et al. (2019). “PyTorch: An Imperative Style, High-Performance Deep Learning Library,” in *Advances in Neural Information Processing Systems*. Editors Wallach, H., Larochelle, H., Beygelzimer, A., d\textquotesingle Alch\{e}-Buc, F., Fox, E., and Garnett, R. (Curran Associates, Inc.) 32. Available at: <https://proceedings.neurips.cc/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf>.
- Qian, L., and Liu, H. (2019). Path Following Control of a Quadrotor Uav with a cable Suspended Payload under Wind Disturbances. *IEEE Trans. Ind. Electron.* 67, 1. doi:10.1109/TIE.2019.2905811
- Redmon, J., and Farhadi, A. (2018). *Yolov3: An Incremental Improvement*. Washington D.C.: University of Washington. Available at: <https://arxiv.53yu.com/pdf/1804.02767.pdf>.
- Wu, Y., Hu, K., and Sun, X.-M. (2018). Modeling and Control Design for Quadrotors: A Controlled Hamiltonian Systems Approach. *IEEE Trans. Vehicular Tech.* 67 (12), 11 365. doi:10.1109/tvt.2018.2877440
- Xie, Z., Zhu, L., Zhao, L., Tao, B., Liu, L., and Tao, W. (2020). Localization-aware Channel Pruning for Object Detection. *Neurocomputing* 403, 400–408. doi:10.1016/j.neucom.2020.03.056
- Yang, H., Zhu, Y., and Liu, J. (2018). *Energy-constrained Compression for Deep Neural Networks via Weighted Sparse Projection and Layer Input Masking*. New York, NY: Cornell University. arXiv preprint arXiv:1806.04321.

systems. SL, and LF drafted the work and contributed to the experiments and conclusions. All authors agree to be accountable for the content of the work.

FUNDING

National Natural Science Foundation of China No. 61972064; LiaoNingRevitalizationTalentsProgram No. XLYC1806006.

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher’s Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Li and Feng. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.