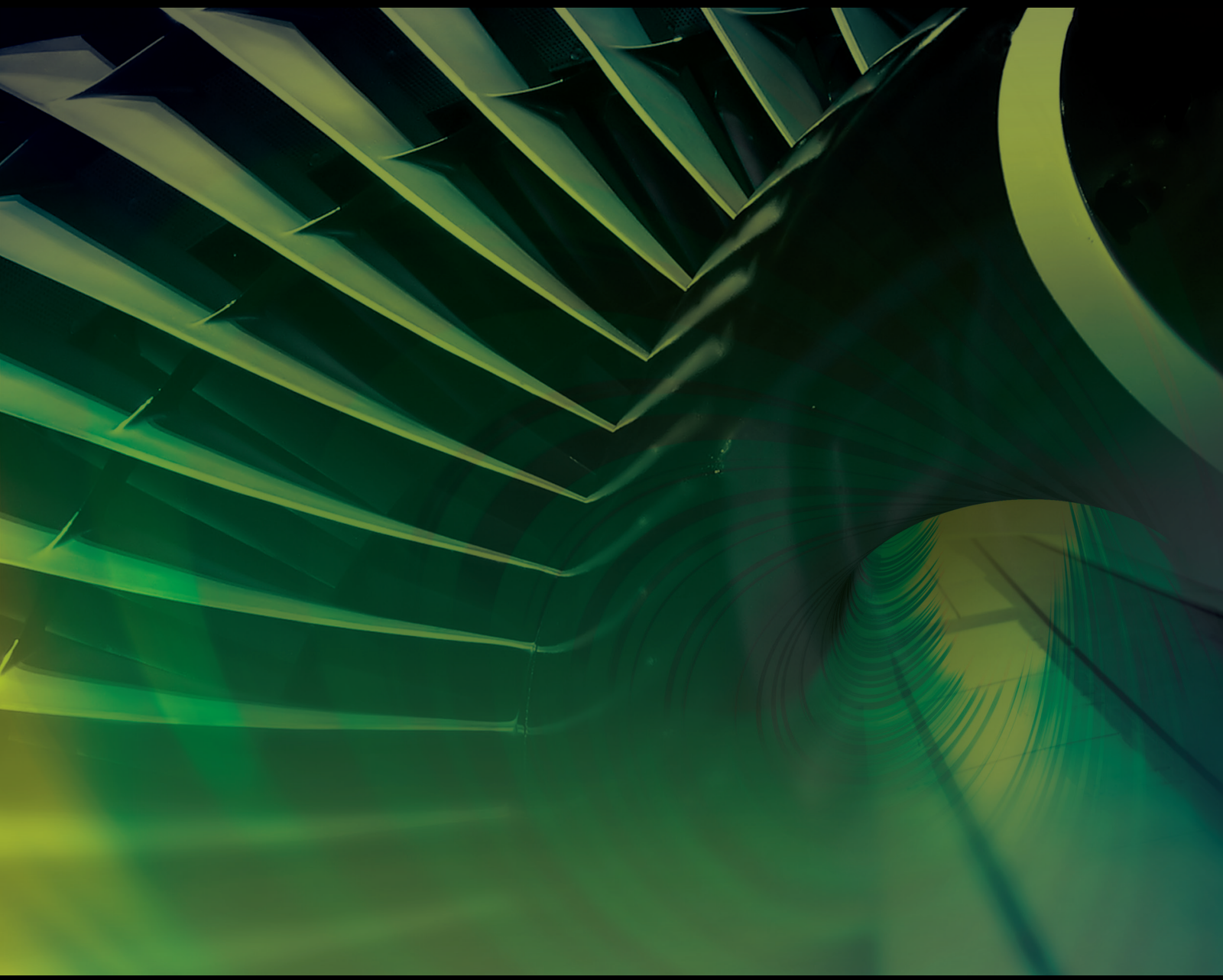
International Journal of Aerospace Engineering

Human-Robot Interaction and Collaboration for Space On-Orbit Servicing

Lead Guest Editor: Xin Zhang
Guest Editors: Qing Gao, Haijun Peng, and Zhaojie Ju



Human-Robot Interaction and Collaboration for Space On-Orbit Servicing

International Journal of Aerospace Engineering

Human-Robot Interaction and Collaboration for Space On-Orbit Servicing

Lead Guest Editor: Xin Zhang
Guest Editors: Qing Gao, Haijun Peng, and Zhaojie Ju

Contents

Hindawi
International Journal of Aerospace Engineering
Volume 2023, Article ID 7435217, 14 pages
https://doi.org/10.1155/2023/7435217

Research Article

Dynamic Modelling and Continuous Trajectory Tracking Control of Space Robots Based on Lie Group SE(3)

Yong Wang,[1] **Kejie Gong**,[2] **Yurui Duan**,[2] **Boyong He**,[1] **and Hong Ma**[1]

[1]*State Key Laboratory of Astronautic Dynamics, Xi'an 710043, China*
[2]*Key Laboratory for Fault Diagnosis and Maintenance of Spacecraft in-Orbit, Xi'an 710043, China*

Correspondence should be addressed to Yong Wang; wangyong08@nudt.edu.cn

Received 21 April 2022; Revised 13 December 2022; Accepted 4 April 2023; Published 19 April 2023

Academic Editor: Xin Zhang

For space robots, it is difficult to track continuous time-varying manifolds on SE(3) by using traditional closed-loop control strategies, which are designed to track the position and the attitude separately. Therefore, the dynamics model should be rebuilt, and the corresponding control strategy should be redesigned. Firstly, the dynamics equations for a space robot in the joint space and workspace are established separately in the framework of Lie group SE(3) and screw theory based on the Lagrange principle. Secondly, based on the proposed feedback form, a PD (proportional derivative) control law of output force on the end-effector is designed, and a closed-loop continuous tracking control strategy is proposed using the force Jacobian matrix and the kinematic model. The simulation results show that the control scheme has good performance when the system state changes gently. Furthermore, a robust sliding mode tracking control scheme is designed. The simulation results show that the proposed robust control law has better accuracy than the PD control law because the system state changes wildly. Finally, a robust fuzzy sliding mode tracking control scheme is designed to deal with the chattering phenomenon. The simulation results show that the proposed robust fuzzy control law can eliminate the chattering well and decrease the joint control torque significantly. The robustness of the proposed robust fuzzy control law is also verified by numerical simulation.

1. Introduction

The emergence of on-orbit assembly stems from the growing need to build large space structures [1]. Related technology has been greatly promoted through the development of on-orbit service technology, especially the on-orbit demonstration of space robot technology. However, the process of assembling large space structures is so elaborate that it requires an accurate and reliable operation ability for space robots. Accordingly, a proper motion planning method and corresponding tracking control strategy should be introduced. Due to the advantages beyond traditional theory, Lie group SE(3) and screw theory have been applied to research in space robotics, such as kinematic modelling [2] and trajectory planning. However, compatible dynamics and tracking control strategy are not considered in detail.

The dynamics of space robots belong to the category of multibody system dynamics. Thus, the modelling is mainly based on the Newton-Euler method and the Lagrange method. The former analyzes the forces of each rigid body based on classical Newtonian mechanics and then obtains dynamic equations in iterative form according to the relationship of internal forces [3]. The advantages of such a method are the clear physical meaning and the relatively small calculation [4]. However, the equations will become more complicated as the number of rigid bodies increases [5]. The Lagrange method is based on the Hamiltonian principle, and the dynamic relationship between the system state variables and the generalized forces is built based on the conservation of energy [6]. The advantages of such a method are that modelling process is relatively easy, and the dynamic equation is in the analytic form [7], but the physical meaning is not clear, and the partial differential calculation is more complex. A detailed comparison of these two methods is presented in [8].

In addition, the Kane method is also introduced in robotic dynamics research [9]. This method subtly uses the partial velocity as the generalized coordinate, which is

combined with the concept of generalized force to establish the dynamic equations according to the Darenbel-Lagrange principle [10]. The Kane method combines the advantages of the Newton-Euler and Lagrange methods, which have high calculation efficiency and concise form. Some improvement methods are proposed [11, 12], which further enhance the performance of the Kane method. Xu and Huang [13] compare classical methods with the Kane method and deduce the dynamic equations of a five-DOF robot using both methods. Yin and Ge [14] derive the dynamic equations of a dual-arm space robot using the Kane-Huston method and verify the feasibility with simple simulation.

Traditional dynamics modelling, which is based on geometric relationships, makes the derivation process relatively complicated. It brings a lot of convenience by introducing the Lie group and Lie algebra, which have a concise and unified form and decreases the difficulty of theoretical derivation. The theory of the Lie group has been applied to the dynamics of fixed-base robots [15, 16]. Besides, Liu et al. [17] combine the screw theory and the Kane method and propose a new concept of space robot dynamics.

Trajectory tracking control is crucial to accomplishing the task for the space robots, and the performance of the tracking controller determines the accuracy and reliability of the task. Many trajectory-tracking control strategies are designed based on the dynamic model. And some control theories for fixed-base ground robots can be introduced to tracking control of space robots, such as decomposition rate control [18], calculating torque control [19], robustness control [20], and reinforcement learning control [21]. Traditional dynamics modelling, which is based on geometric relationships, makes the derivation process relatively complicated.

Trajectory tracking of space robots can be performed either in joint space or in cartesian space. According to tracking targets, it can be further classified into point-to-point and continuous trajectory tracking. The latter means that the end-effector is required to move along the desired continuous trajectory, such as peg-in-hole insertion [22], which makes tracking control relatively difficult. Li and Liang [23] establish the kinematic equation of the space robot system by using the D-H (Denavit-Hartenberg) method and design the continuous trajectory control algorithm according to the momentum conservation law. Such an algorithm could be extended to a space robot with an arbitrary tree structure. Galicki [24] defines a nonsingular terminal sliding model in the workspace and proposes a robustness controller based on the Jacobian transfer, which can effectively eliminate the uncertainty of dynamics. Su et al. [25] define an approximately fixed-time convergence terminal sliding model and design a nonsingular sliding model tracking control strategy, but it is carried out in the joint space. In addition, most existing trajectory tracking control laws use position or linear velocity as feedback variables. There are also attempts that use quaternion [26] or dual quaternion [27] as feedback variables, but they are all difficult to apply to the tracking of the continuous time-varying manifold on SE(3).

Above all, this paper will focus on the dynamics modelling under the framework on Lie group SE(3) and the control problem of tracking the continuous time-varying trajectory of the end-effector on SE(3). Highlights of this paper are addressed below.

(1) The dynamic model of a multiarm space robot is established by combining the Lie group SE(3) with the Lagrange method. The model has a concise and unified form, which can decrease the difficulty of theoretical derivation and calculation

(2) A novel form is proposed to describe the feedback error in the tracking control problem of the space robot. Lie group SE(3) and screw theory are used to build the form, which can assemble the position and attitude, with their derivative, of the end-effector in a unified variable. It can significantly reduce the calculation in the process of closed-loop control

(3) A robust fuzzy sliding mode tracking control strategy is proposed to track the continuous and time-varying trajectory, including position and attitude together, of the end-effector. The tracking control strategy has good control accuracy, good robustness, and no chatter

2. Dynamic Modelling

2.1. Dynamic Equations of the Space Robot System. For a dual-arm space robot, as provided in Figure 1, the kinetic energy of the link of the arm in the barycentre-fixed inertial coordinate system is

$$T_i^k = \frac{1}{2}\left[\left(\boldsymbol{\omega}_{ki}^B\right)^{\mathrm{T}} \boldsymbol{I}_{C_i^k} \boldsymbol{\omega}_{ki}^B + m_i^k \left({}^I\dot{\boldsymbol{p}}_i^k\right)^{\mathrm{T}I} \dot{\boldsymbol{p}}_i^k\right] \left(i = 0, 1, \cdots, n^k ; k = 1, 2\right), \tag{1}$$

where ω_{ki}^B, $\mathrm{I}_{C_i^k}$, m_i^k, and ${}^I\dot{\mathrm{p}}_i^k$ are the body angular velocity, the inertia, the mass, and the velocity, respectively. $i = 0$ refers to the base.

According to the definition of body velocity in screw theory, formula (1) can be transformed into the following form:

$$\begin{aligned} T_i^k &= \frac{1}{2}\left[\left(\boldsymbol{\omega}_{ki}^B\right)^{\mathrm{T}} I_{C_i^k} \boldsymbol{\omega}_{ki}^B + m_i^k \left(\left({}^I\ R_i^k\right)^{\mathrm{T}I} \dot{\mathrm{p}}_i^k\right)^{\mathrm{T}} \left({}^I\ R_i^k\right)^{\mathrm{T}I} \dot{\mathrm{p}}_i^k\right] \\ &= \frac{1}{2}\begin{bmatrix} \boldsymbol{\omega}_{ki}^B \\ \left({}^I\ R_i^k\right)^{\mathrm{T}I} \dot{\mathbf{p}}_i^k \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} I_{C_i^k} & \mathbf{0} \\ \mathbf{0} & m_i^k E \end{bmatrix} \begin{bmatrix} \boldsymbol{\omega}_{ki}^B \\ \left({}^I\ R_i^k\right)^{\mathrm{T}I} \dot{\mathbf{p}}_i^k \end{bmatrix} \\ &= \frac{1}{2}\left({}^I_{C_i^k}\boldsymbol{V}^B\right)^{\mathrm{T}} \begin{bmatrix} I_{C_i^k} & \mathbf{0} \\ \mathbf{0} & m_i^k E \end{bmatrix} {}^I_{C_i^k}\boldsymbol{V}^B, \end{aligned} \tag{2}$$

where ${}^I_{C_i^k}\boldsymbol{V}^B$ is the body velocity, and ${}^IR_i^k$ represents the attitude rotation matrix in the inertial coordinate system.

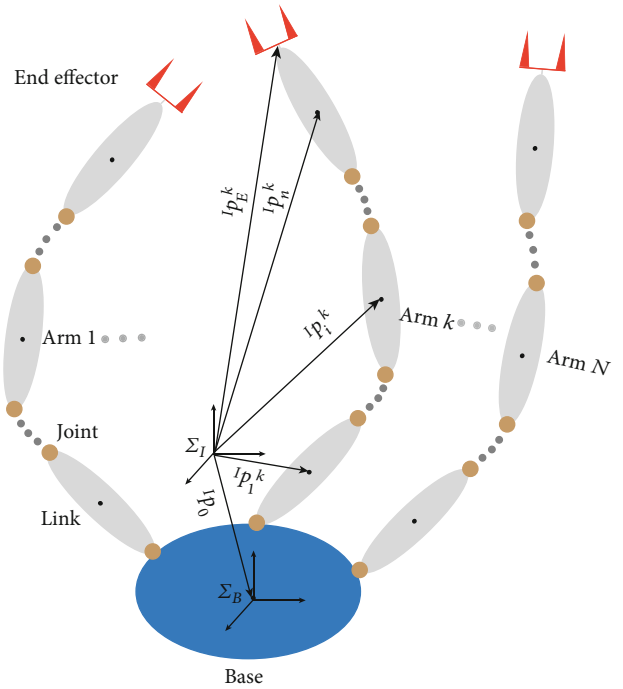


FIGURE 1: The model of a dual-arm space robot, which has two arms, and each arm has n^k joints.

According to the relationship between body velocity and spatial velocity, formula (2) is transformed into

$$\begin{aligned} T_i^k &= \frac{1}{2}\left(Ad_{(g_{ik})}\,{}^{I}_{C_i^k}\mathbf{V}^S\right)^{\mathrm{T}}\begin{bmatrix}\mathbf{I}_{C_i^k} & \mathbf{0}\\ \mathbf{0} & m_i^k E\end{bmatrix}Ad_{(g_{ik})}\,{}^{I}_{C_i^k}\mathbf{V}^S \\ &= \frac{1}{2}\left({}^{I}_{C_i^k}\mathbf{V}^S\right)^{\mathrm{T}}\left(Ad_{(g_{ik})}\right)^{\mathrm{T}}\begin{bmatrix}\mathbf{I}_{C_i^k} & \mathbf{0}\\ \mathbf{0} & m_i^k E\end{bmatrix}Ad_{(g_{ik})}\,{}^{I}_{C_i^k}\mathbf{V}^S \\ &= \frac{1}{2}\left({}^{I}_{C_i^k}\mathbf{V}^S\right)^{\mathrm{T}}\begin{bmatrix}{}^{I}R_i^k\mathbf{I}_{C_i^k}\left({}^{I}R_i^k\right)^{\mathrm{T}} - m_i^{kI}\,\hat{\mathbf{p}}_i^{kI}\,\hat{\mathbf{p}}_i^k & m_i^{kI}\,\hat{\mathbf{p}}_i^k\\ -m_i^{kI}\,\hat{\mathbf{p}}_i^k & m_i^k E\end{bmatrix}{}^{I}_{C_i^k}\mathbf{V}^S \\ &= \frac{1}{2}\left({}^{I}_{C_i^k}\mathbf{V}^S\right)^{\mathrm{T}}K_i^{kI}\,{}^{I}_{C_i^k}\mathbf{V}^S, \end{aligned} \tag{3}$$

where ${}^{I}_{C_i^k}\boldsymbol{V}^S$ is the spatial velocity, $Ad_{(g_{ik})}$ is the adjoint transformation, $\hat{\cdot}$ represents the antisymmetric matrix of the corresponding vector, and $\boldsymbol{K}_i^k$ means generalized inertia matrix.

Therefore, the total kinetic energy of the dual-arm space robot is,

$$\begin{aligned} T &= \frac{1}{2}\left[\sum_{k=1}^{2}\sum_{i=1}^{n^k}\left({}^{I}_{C_i^k}\mathbf{V}^S\right)^{\mathrm{T}}\mathbf{K}_i^{kI}\,{}^{I}_{C_i^k}\mathbf{V}^S + \left({}^{I}_{B}\mathbf{V}^S\right)^{\mathrm{T}}\mathbf{K}_{0B}\,{}^{I}_{B}\mathbf{V}^S\right] \\ &= \frac{1}{2}\left[\sum_{k=1}^{n}\sum_{i=1}^{n^k}\left({}^{I}_{B}\mathbf{V}^S + Ad_{{}^{I}_{B}g\,{}^{B}_{C_i^k}}\mathbf{V}^S\right)^{\mathrm{T}}\mathbf{K}_i^k\left({}^{I}_{B}\mathbf{V}^S + Ad_{{}^{I}_{B}g\,{}^{B}_{C_i^k}}\mathbf{V}^S\right)\right) + \left({}^{I}_{B}\mathbf{V}^S\right)^{\mathrm{T}}\mathbf{K}_{0B}\,{}^{I}_{B}\mathbf{V}^S\right] \\ &= \frac{1}{2}\left[{}^{I}_{B}\mathbf{V}^S D + \sum_{k=1}^{n}\sum_{i=1}^{n^k}\left(Ad_{{}^{I}_{B}g\,{}^{B}_{C_i^k}}\mathbf{V}^S\right)^{\mathrm{T}}\mathbf{K}_i^k\left({}^{I}_{B}\mathbf{V}^S + Ad_{{}^{I}_{B}g\,{}^{B}_{C_i^k}}\mathbf{V}^S\right)\right], \end{aligned} \tag{4}$$

where D represents the total momentum of the dual-arm space robot.

As is known, $D = 0$ is workable in the case of a free-floating base. Substituting GJM into formula (4) yields

$$T = \frac{1}{2}\left[\sum_{k=1}^{2}\sum_{i=1}^{n^k}\left(Ad_{I_Bg}\boldsymbol{J}_i^k\dot{\boldsymbol{\theta}}\right)^{\mathrm{T}}\boldsymbol{K}_i^k\left(\boldsymbol{J}_B\dot{\boldsymbol{\theta}} + Ad_{I_Bg}\boldsymbol{J}_i^k\dot{\boldsymbol{\theta}}\right)\right] = \frac{1}{2}\dot{\boldsymbol{\theta}}^T\boldsymbol{H}\dot{\boldsymbol{\theta}}, \tag{5}$$

where $\boldsymbol{J}_i^k$ are the GJM of the link and the base, respectively. $\dot{\boldsymbol{\theta}}$ is the angular velocity vector of all joints. $\boldsymbol{H}$ represents the generalized inertia tensor of the dual-arm space robot, the specific expression is as follows:

$$\boldsymbol{H} = \sum_{k=1}^{2}\sum_{i=1}^{n^k}\left(Ad_{{}_{B}^{I}g}\boldsymbol{J}_i^k\right)^{\mathrm{T}}\boldsymbol{K}_i^k\left(\boldsymbol{J}_B + Ad_{{}_{B}^{I}g}\boldsymbol{J}_i^k\right) = \boldsymbol{J}_N^{\mathrm{T}}\tilde{\boldsymbol{K}}_{\mathrm{ver}}\boldsymbol{J}_B + \boldsymbol{J}_N^{\mathrm{T}}\tilde{\boldsymbol{K}}_{\mathrm{dia}}\boldsymbol{J}_N, \tag{6}$$

where

$$\tilde{\boldsymbol{K}}_{\mathrm{ver}} = \begin{bmatrix}\boldsymbol{K}_1^1\\ \vdots\\ \boldsymbol{K}_{n^1}^1\\ \boldsymbol{K}_1^2\\ \vdots\\ \boldsymbol{K}_{n^2}^2\end{bmatrix} \in \mathbb{R}^{6n_d\times 6},. \tag{7}$$

$$\tilde{\boldsymbol{K}}_{\mathrm{dia}} = \begin{bmatrix}\boldsymbol{K}_1^1 & & & & & \\ & \ddots & & & & \\ & & \boldsymbol{K}_{n^1}^1 & & & \\ & & & \boldsymbol{K}_1^2 & & \\ & & & & \ddots & \\ & & & & & \boldsymbol{K}_{n^2}^2\end{bmatrix} \in \mathbb{R}^{6n_d\times 6n_d}.$$

If the gravity is ignored and the joint elasticity is not considered, this system satisfies the Lagrange equation in the following form:

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial T}{\partial\dot{\boldsymbol{\theta}}} - \frac{\partial T}{\partial\boldsymbol{\theta}} = \boldsymbol{\tau}. \tag{8}$$

Substituting formula (5) into the above formula, then

$$\boldsymbol{H}\ddot{\boldsymbol{\theta}} + \dot{\boldsymbol{H}}\dot{\boldsymbol{\theta}} - \frac{\partial}{\partial\boldsymbol{\theta}}\left(\frac{1}{2}\dot{\boldsymbol{\theta}}^{\mathrm{T}}\boldsymbol{H}\dot{\boldsymbol{\theta}}\right) = \boldsymbol{\tau}. \tag{9}$$

It can be rewritten as

$$\boldsymbol{H}\ddot{\boldsymbol{\theta}} + \boldsymbol{C}\dot{\boldsymbol{\theta}} = \boldsymbol{\tau}, \tag{10}$$

where

$$\boldsymbol{C}\dot{\boldsymbol{\theta}} = \dot{\boldsymbol{H}}\dot{\boldsymbol{\theta}} - \frac{\partial}{\partial \boldsymbol{\theta}}\left(\frac{1}{2}\dot{\boldsymbol{\theta}}^{\mathrm{T}}\boldsymbol{H}\dot{\boldsymbol{\theta}}\right), \tag{11}$$

which represents the centrifugal force and Coriolis force.

For a single-arm space robot, the above derivation process is also applicable.

2.2. Static Force Jacobian Matrix. Static force Jacobian matrix is also a fundamental concept in the dynamics of space robots. Similar to the speed Jacobian matrix, it describes the mapping relationship between the output force on the end effector and the joint torque. This derivation is based on the equivalence principle of work and energy. Taking a single-arm space robot as an example, and assuming that the output force on the end effector is $\boldsymbol{F}^S$, then the work is

$$W_\tau = \int_{t1}^{t2} \dot{\boldsymbol{\theta}}^{\mathrm{T}}\boldsymbol{\tau}\mathrm{d}t, \tag{12}$$

where ${}^I_E\mathbf{V}^S$ is the spatial velocity of the end effector. Simultaneously, the work done by the joint torque$\boldsymbol{\tau}$ is

$$W_\tau = \int_{t1}^{t2} \dot{\boldsymbol{\theta}}^{\mathrm{T}}\boldsymbol{\tau}\mathrm{d}t. \tag{13}$$

If joint friction is not considered, then W_F and W_τ should be equal and time-independent. Thus,

$$\left({}^I_E\boldsymbol{V}^S\right)^{\mathrm{T}}\boldsymbol{F}^S = \dot{\boldsymbol{\theta}}^{\mathrm{T}}\boldsymbol{\tau}. \tag{14}$$

As is known, $\boldsymbol{V}^S = \boldsymbol{J}_E\dot{\boldsymbol{\theta}}$, where $\boldsymbol{J}_E$is the speed GJM of the end-effector for the joints. So

$$\boldsymbol{\tau} = (\boldsymbol{J}_E)^{\mathrm{T}}\boldsymbol{F}^S, \tag{15}$$

where $(\boldsymbol{J}_E)^{\mathrm{T}}$ is the static force Jacobian matrix.

2.3. Dynamic Equations in the Workspace. When performing trajectory tracking control of the end effector, it is also necessary to establish dynamic equations in the workspace. Firstly, according to the GJM of the end-effector, there are

$$\dot{\boldsymbol{\theta}} = \boldsymbol{J}_E^{-1}{}^I_E\boldsymbol{V}^S, {}^I_E\dot{\boldsymbol{V}}^S = \boldsymbol{J}_E\ddot{\boldsymbol{\theta}} + \dot{\boldsymbol{J}}_E\dot{\boldsymbol{\theta}}. \tag{16}$$

Accordingly,

$$\ddot{\boldsymbol{\theta}} = \boldsymbol{J}_E^{-1}\left({}^I_E\dot{\boldsymbol{V}}^S - \dot{\boldsymbol{J}}_E\boldsymbol{J}_E^{-1}{}^I_E\boldsymbol{V}^S\right). \tag{17}$$

Substituting formula (17) into formula (10),

$$\boldsymbol{H}\boldsymbol{J}_E^{-1}\left({}^I_E\dot{\boldsymbol{V}}^S - \dot{\boldsymbol{J}}_E\boldsymbol{J}_E^{-1}{}^I_E\boldsymbol{V}^S\right) + \boldsymbol{C}\boldsymbol{J}_E^{-1}{}^I_E\boldsymbol{V}^S = \boldsymbol{\tau}. \tag{18}$$

It can be rewritten as

$$\boldsymbol{H}\boldsymbol{J}_E^{-1}{}^I_E\dot{\boldsymbol{V}}^S + \left(\boldsymbol{C} - \boldsymbol{H}\boldsymbol{J}_E^{-1}\dot{\boldsymbol{J}}_E\right)\boldsymbol{J}_E^{-1}{}^I_E\boldsymbol{V}^S = \boldsymbol{\tau}. \tag{19}$$

Then, substituting formula (15) into the above formula,

$$\boldsymbol{H}_g{}^I_E\dot{\boldsymbol{V}}^S + \boldsymbol{C}_g{}^I_E\boldsymbol{V}^S = \boldsymbol{F}^S, \tag{20}$$

where $\boldsymbol{H}_g = (\boldsymbol{J}_E)^{-\mathrm{T}}\boldsymbol{H}\boldsymbol{J}_E^{-1}$ and $\boldsymbol{C}_g = (\boldsymbol{J}_E)^{-\mathrm{T}}(\boldsymbol{C} - \boldsymbol{H}\boldsymbol{J}_E^{-1}\dot{\boldsymbol{J}}_E)\boldsymbol{J}_E^{-1}$. It can be proved that the above dynamic equation satisfies,

(1) $\boldsymbol{H}_g$ is a symmetric positive definite matrix

(2) $\dot{\boldsymbol{H}}_g - 2\boldsymbol{C}_g$ is an antisymmetric matrix

2.4. Feedback Calculation Model Based on SE(3). As the desired trajectory is expressed on SE(3), compatible feedback should be designed first. Supposing thatg_i and $g_{di}$are the actual and desired pose of the end effector, respectively, then the pose error can be written as

$$\boldsymbol{g}_e = \boldsymbol{g}_{di}^{-1}\boldsymbol{g}_i, \tag{21}$$

where $\boldsymbol{g}_e \in \mathrm{SE}(3)$.

The tracking error in exponential coordinate is presented as

$$\widehat{\boldsymbol{\xi}}_e = \log_g\left(\boldsymbol{g}_{di}^{-1}\boldsymbol{g}_i\right), \tag{22}$$

where $\log_g(\cdot)$ represents the logarithmic mapping from Lie group SE(3) to its Lie algebra se(3).

The above equation can be written in the form of screw coordinates as

$$\boldsymbol{\xi}_e^B = \left(\log_g\left(\boldsymbol{g}_{di}^{-1}\boldsymbol{g}_i\right)\right)^\vee. \tag{23}$$

It is used as feedback at the position level. It should be noted that $\boldsymbol{\xi}_e^B$ is with respect to the body-fixed coordinate system of the end effector.

The feedback at the velocity level can be expressed as

$$\boldsymbol{V}_e^B = \boldsymbol{V}_i^B - \boldsymbol{V}_{di}^B, \tag{24}$$

where $\boldsymbol{V}_i^B = (\boldsymbol{g}_i^{-1}\dot{\boldsymbol{g}}_i)^\vee$ is the actual body velocity, and $\boldsymbol{V}_{di}^B = (\boldsymbol{g}_{di}^{-1}\dot{\boldsymbol{g}}_{di})^\vee$ is the desired body velocity.

In fact, the following relationship exists:

$$\dot{\boldsymbol{\xi}} = \boldsymbol{G}(\boldsymbol{\xi})\boldsymbol{V}^B. \tag{25}$$

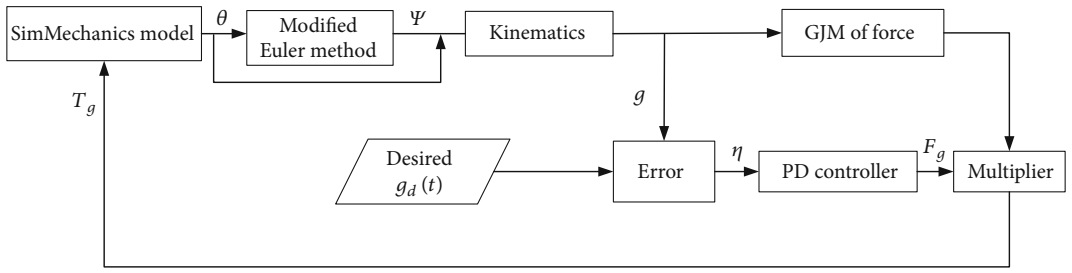


Figure 2: PD control block diagram based on the Jacobi matrix.

The expression of $\boldsymbol{G}(\boldsymbol{\xi})$ in the above formula is presented in [28] and omitted here.

Accordingly, the feedback of pose expressed on SE(3) can be expressed as

$$\boldsymbol{\eta}^B = \begin{bmatrix} \boldsymbol{\xi}_e^B & \boldsymbol{V}_e^B \end{bmatrix}. \tag{26}$$

Transforming it into inertial coordinate system yields

$$\boldsymbol{\eta} = Ad_{g_i}\boldsymbol{\eta}^B = [\boldsymbol{\xi}_e \quad \boldsymbol{V}_e]. \tag{27}$$

3. Continuous Trajectory Tracking Control Strategy

3.1. PD Controller Based on Jacobian Matrix. According to the feedback $\boldsymbol{\eta}$ obtained in the previous section, the following PD control law is designed:

$$\boldsymbol{F}_g^S = -\boldsymbol{K}_p\boldsymbol{\xi}_e - \boldsymbol{K}_v\boldsymbol{V}_e, \tag{28}$$

where $\boldsymbol{F}_g^S$ is the output force on the end effector with respect to the inertial coordinate system, and $\boldsymbol{K}_p$ and $\boldsymbol{K}_v$ are the position and velocity gains, respectively.

Then, the joint control torque can be computed as

$$\boldsymbol{\tau}_g = (\boldsymbol{J}_E)^{\mathrm{T}}\mathbf{F}_g^S. \tag{29}$$

Accordingly, detailed steps of a closed-loop tracking control strategy are given as follows (as shown in Figure 2):

Step 1. Set the initial joint angle $\boldsymbol{\theta}_0$ of the arm and the attitude $\boldsymbol{\Psi}_0$ of the base. Set the desired pose $g(t)$

Step 2. Calculate the current pose of the end effector according to the kinematics equation, and then obtain the force Jacobian matrix

Step 3. Calculate the current state feedback $\boldsymbol{\eta}$ according to formula (27)

Step 4. Calculate the joint control torque at the current moment according to formula (29)

Step 5. Calculate the joint angle$\boldsymbol{\theta}_i$ and joint angle rate$\dot{\boldsymbol{\theta}}_i$ at the next moment according to the dynamic equation, and then calculate the base attitude angle$\boldsymbol{\Psi}_i$ at the next moment

Step 6. Repeat steps 2-5 till to the end time.

The controllers proposed in this research include the PD controller, the robust sliding mode controller (RSMC), and the robust fuzzy sliding mode controller (RFSMC). RSMC or RFSMC can replace the PD controller module in Figure 2, and then the corresponding control block diagram is generated.

3.2. Robust Sliding Mode Tracking Controller. The PD control has a simple form and can meet general needs. However, the control accuracy will be significantly reduced when the state changes fast. Considering the advantages of sliding mode control, such as the high control accuracy and strong robustness against external disturbances, a controller based on robust sliding mode control theory with the feedback calculation model above is proposed in this section.

3.2.1. Linear Reaching Law-Based Sliding Mode Control Law. According to the tracking feedback at the velocity level, the sliding mode surface is designed as follows:

$$\boldsymbol{s} = \boldsymbol{V}_e + \Lambda\boldsymbol{\xi}_e = \boldsymbol{V} - \boldsymbol{V}_d + \boldsymbol{\Lambda}\boldsymbol{\xi}_e, \tag{30}$$

where Λ is a positive definite diagonal matrix. A control law can be given

$$\boldsymbol{F}_g = \boldsymbol{H}_g(\boldsymbol{V}_d - \Lambda\boldsymbol{\xi}_e) + \boldsymbol{C}_g(\boldsymbol{V}_d - \boldsymbol{\Lambda}\boldsymbol{\xi}_e) - \boldsymbol{K}_v\boldsymbol{s}, \tag{31}$$

where $\boldsymbol{K}_v$ is a positive definite diagonal matrix.

Theorem 1. *For the dynamics of a space robot system, applying sliding mode surface (30) and control law (31), the system will gradually converge to the equilibrium point.*

Proof. Define the Lyapunov function as follows:

$$L = \frac{1}{2}\boldsymbol{s}^{\mathrm{T}}\boldsymbol{H}_g\boldsymbol{s}. \tag{32}$$

According to the sliding mode surface (30),

$$\boldsymbol{V} = \boldsymbol{s} + \boldsymbol{V}_d - \boldsymbol{\Lambda}\boldsymbol{\xi}. \tag{33}$$

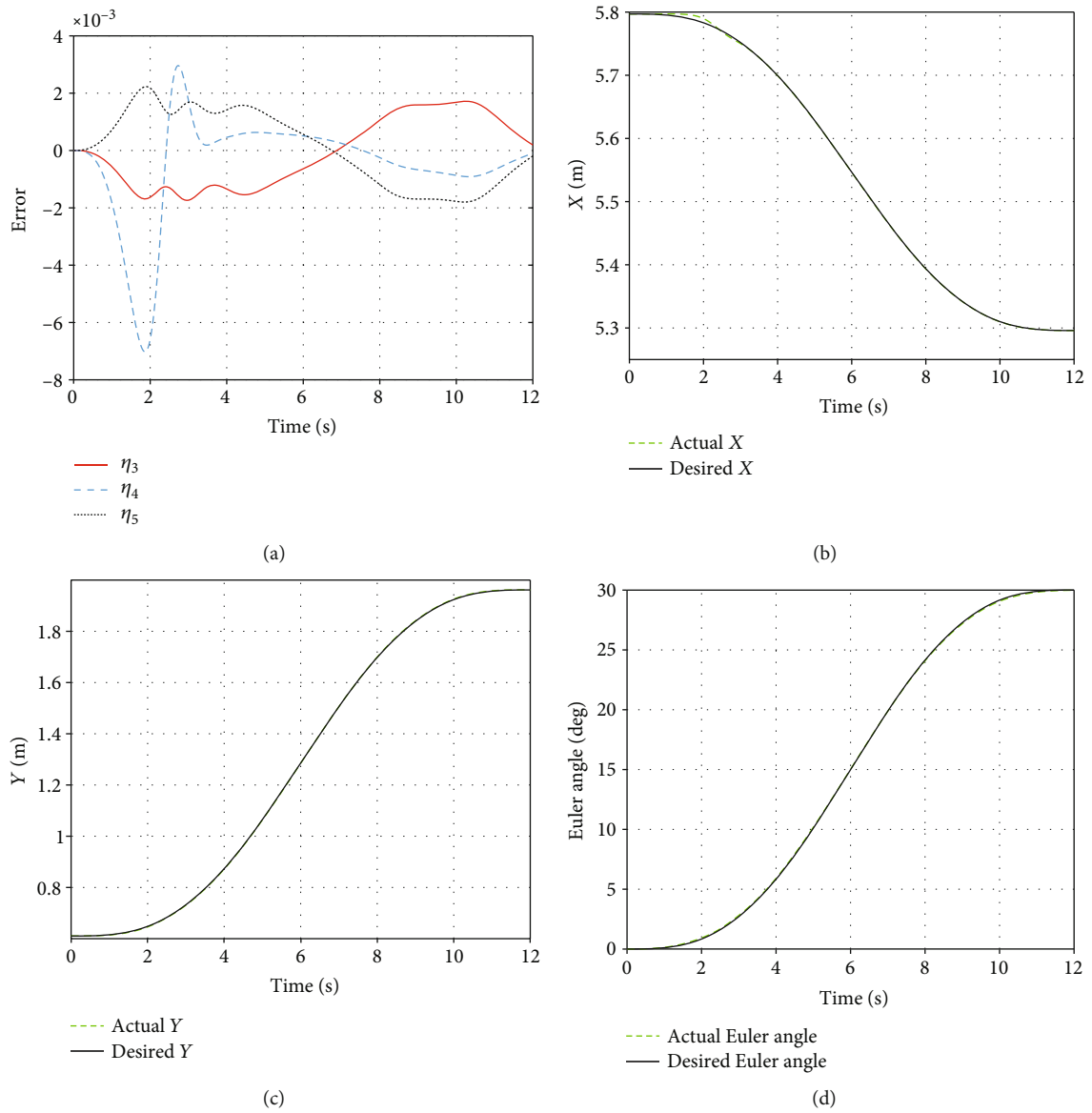


Figure 3: Error and pose components in the tracking process.

Calculating the derivation of L and substituting (28) into it yields

$$\begin{aligned}\dot{L} &= \boldsymbol{s}^{\mathrm{T}}\boldsymbol{H}_g\dot{\boldsymbol{s}} + \frac{1}{2}\boldsymbol{s}^{\mathrm{T}}\dot{\boldsymbol{H}}_g\boldsymbol{s} = \boldsymbol{s}^{\mathrm{T}}\boldsymbol{H}_g\left(\dot{\boldsymbol{V}}_d - \dot{\boldsymbol{V}}_e + \boldsymbol{\Lambda}\dot{\boldsymbol{\xi}}_e\right) + \frac{1}{2}\boldsymbol{s}^{\mathrm{T}}\dot{\boldsymbol{H}}_g\boldsymbol{s} \\ &= \boldsymbol{s}^{\mathrm{T}}\left(-\boldsymbol{H}_g\left(\dot{\boldsymbol{V}}_d - \boldsymbol{\Lambda}\dot{\boldsymbol{\xi}}_e\right) + \boldsymbol{H}_g\dot{\boldsymbol{V}}\right) + \frac{1}{2}\boldsymbol{s}^{\mathrm{T}}\dot{\boldsymbol{H}}_g\boldsymbol{s} \\ &= \boldsymbol{s}^{\mathrm{T}}\left(-\boldsymbol{H}_g\left(\dot{\boldsymbol{V}}_d - \boldsymbol{\Lambda}\dot{\boldsymbol{\xi}}_e\right) - \boldsymbol{C}_g\boldsymbol{V} + \boldsymbol{F}_g^S\right) + \frac{1}{2}\boldsymbol{s}^{\mathrm{T}}\dot{\boldsymbol{H}}_g\boldsymbol{s} \\ &= \boldsymbol{s}^{\mathrm{T}}\left(-\boldsymbol{H}_g\left(\dot{\boldsymbol{V}}_d - \boldsymbol{\Lambda}\dot{\boldsymbol{\xi}}_e\right) - \boldsymbol{C}_g(\boldsymbol{V}_d - \boldsymbol{\Lambda}\boldsymbol{\xi}_e) - \boldsymbol{C}_g\boldsymbol{s} + \boldsymbol{F}_g^S\right) \\ &\quad + \frac{1}{2}\boldsymbol{s}^{\mathrm{T}}\dot{\boldsymbol{H}}_g\boldsymbol{s} = \boldsymbol{s}^{\mathrm{T}}\left(-\boldsymbol{H}_g\left(\dot{\boldsymbol{V}}_d - \boldsymbol{\Lambda}\dot{\boldsymbol{\xi}}_e\right) - \boldsymbol{C}_g(\boldsymbol{V}_d - \boldsymbol{\Lambda}\boldsymbol{\xi}_e) + \boldsymbol{F}_g^S\right) \\ &\quad + \frac{1}{2}\boldsymbol{s}^{\mathrm{T}}\left(\dot{\boldsymbol{H}}_g - 2\boldsymbol{C}_g\right)\boldsymbol{s} = -\boldsymbol{s}^{\mathrm{T}}\boldsymbol{K}_v\boldsymbol{s} \leqslant -\lambda L,\end{aligned} \tag{34}$$

where $\lambda = \lambda_{\min}(K_v)/\lambda_{\max}(H_g)$. $\lambda_{\min}(K_v)$ is the smallest eigenvalue of $\boldsymbol{K}_v$, and $\lambda_{\max}(H_g)$ is the largest eigenvalue of $\boldsymbol{H}_g$. It can be seen that the sliding mode surface $\boldsymbol{s}$ converges to zero driven by the control law (31). If $\boldsymbol{s} \equiv 0$, there is $\boldsymbol{V}_e = -\boldsymbol{\Lambda}\boldsymbol{\xi}_e$, then $\boldsymbol{\xi}_e, \boldsymbol{V}_e$ progressively converge to the equilibrium point. Proof completed. □

3.2.2. Robust Sliding Mode Control Law. When the dynamic model can be accurately established, the sliding mode control law (31) can be substituted into the PD control strategy to replace the PD control law. However, as the dynamic model of the space robot is very complicated, if control law (31) is directly adopted not only will the modelling be complex but also the amount of calculation will be huge. Let

$$\mathbf{f} = -\boldsymbol{H}_g(\boldsymbol{V}_d - \boldsymbol{\Lambda}\dot{\boldsymbol{\eta}}_e) - \boldsymbol{C}_g(\boldsymbol{V}_d - \boldsymbol{\Lambda}\boldsymbol{\eta}_e). \tag{35}$$

Regard this term as a disturbance and assume that the term is bounded and satisfies the following inequality:

$$-\boldsymbol{H}_g(\boldsymbol{V}_d - \boldsymbol{\Lambda}\dot{\boldsymbol{\eta}}_e) - \boldsymbol{C}_g(\boldsymbol{V}_d - \boldsymbol{\Lambda}\boldsymbol{\eta}_e) \leq D. \tag{36}$$

In order to overcome the disturbance term, the following sliding mode control law is designed:

$$\boldsymbol{F}_g^S = -[\boldsymbol{K}_v\boldsymbol{s} + \rho\ \mathrm{sgn}\ (\boldsymbol{s})], \tag{37}$$

where $\rho$is the robust coefficient and$\rho \leq D$ and ρ sgn $(\boldsymbol{s})$ is the robust term to overcome the disturbance.

Theorem 2. *For the dynamics of a space robot system, applying sliding mode surface (30) and control law (37), the system will gradually converge to the equilibrium point.*

Proof. Define the Lyapunov function as follows:

$$L = \frac{1}{2}\boldsymbol{s}^{\mathrm{T}}\boldsymbol{H}_g\boldsymbol{s}. \tag{38}$$

Similar to Theorem (1),

$$\begin{aligned}\dot{L} &= \boldsymbol{s}^{\mathrm{T}}\left(-\boldsymbol{H}_g\left(\dot{\boldsymbol{V}}_d - \boldsymbol{\Lambda}\dot{\boldsymbol{\xi}}_e\right) - \boldsymbol{C}_g(\boldsymbol{V}_d - \boldsymbol{\Lambda}\boldsymbol{\xi}_e) + \boldsymbol{F}_g^s\right)\\ &= \boldsymbol{s}^{\mathrm{T}}\left(-\boldsymbol{H}_g\left(\dot{\boldsymbol{V}}_d - \boldsymbol{\Lambda}\dot{\boldsymbol{\xi}}_e\right) - \boldsymbol{C}_g(\boldsymbol{V}_d - \boldsymbol{\Lambda}\boldsymbol{\xi}_e) - [\boldsymbol{K}_v\boldsymbol{s} + \rho\ \mathrm{sgn}\ (\boldsymbol{s})]\right)\\ &\leqslant -\boldsymbol{s}^{\mathrm{T}}\boldsymbol{K}_v\boldsymbol{s} + (D-\rho)\|\boldsymbol{s}\| \leqslant -\lambda L.\end{aligned} \tag{39}$$

It can be proven like Theorem (1).
Proof completed. □

3.3. Robust Fuzzy Sliding Mode Tracking Controller considering Uncertainty. The robust sliding mode tracking controller has better control accuracy; however, when there are uncertainties with the parameters of the system model, the controller must choose a more significant robust coefficient or amplify the gain to ensure robust performance and control accuracy which may lead to the obvious chattering of the control torque. In order to solve this problem, fuzzy ideas are introduced into to sliding mode control law to approximate the dynamic model with uncertain parameters, which can realize robust and stable control of the space robot.

3.3.1. Fuzzy Approximation Method. Taking into account the unavoidable external interference, introduce the interference force $\boldsymbol{F}_{\boldsymbol{g}}$to the dynamic model as follow.

$$\boldsymbol{H}_{\boldsymbol{g}}\dot{\boldsymbol{V}} + \tilde{\boldsymbol{C}}_{\boldsymbol{g}}\dot{\boldsymbol{V}} = \boldsymbol{F}_{\boldsymbol{g}} + \boldsymbol{F}_{\boldsymbol{d}}. \tag{40}$$

The following fuzzy sliding mode control law is designed

$$\boldsymbol{F}_{\boldsymbol{g}} = -\hat{\boldsymbol{f}} - \boldsymbol{K}_{\boldsymbol{v}}\boldsymbol{s}, \tag{41}$$

where $\hat{\boldsymbol{f}}$is the estimated value of the fuzzy system. Substituting the control law into equation (40) to obtain

$$\boldsymbol{H}_{\boldsymbol{g}}\dot{\boldsymbol{s}} = -\boldsymbol{C}\boldsymbol{s} - \hat{\boldsymbol{f}} - \boldsymbol{K}_{\boldsymbol{v}}\boldsymbol{s} + \boldsymbol{f} + \boldsymbol{F}_{\boldsymbol{d}} = -(\boldsymbol{C} + \boldsymbol{K}_{\boldsymbol{v}})\boldsymbol{s} - \tilde{\boldsymbol{f}} + \boldsymbol{F}_{\boldsymbol{d}} = -(\boldsymbol{C} + \boldsymbol{K}_{\boldsymbol{v}})\boldsymbol{s} + \boldsymbol{\zeta}_0, \tag{42}$$

where $\tilde{\boldsymbol{f}} = \hat{\boldsymbol{f}} - \boldsymbol{f}$ and $\boldsymbol{\zeta}_{\boldsymbol{0}} = -\tilde{\boldsymbol{f}} + \boldsymbol{F}_{\boldsymbol{d}}$.

The fuzzy approximation can be used to estimate the function $\boldsymbol{f}$ in equation (35),

$$\boldsymbol{f} = -\boldsymbol{H}_{\boldsymbol{g}}(\boldsymbol{V}_{\boldsymbol{d}} - \boldsymbol{\Lambda}\dot{\boldsymbol{\eta}}_{\boldsymbol{e}}) - \boldsymbol{C}_{\boldsymbol{g}}(\boldsymbol{V}_{\boldsymbol{d}} - \boldsymbol{\Lambda}\boldsymbol{\eta}_{\boldsymbol{e}}) = \boldsymbol{W}^*\boldsymbol{\beta} + \boldsymbol{\epsilon}, \tag{43}$$

where $\boldsymbol{W}^*$is the optimal weight matrix of the fuzzy approximation system, $\boldsymbol{\epsilon}$is the approximation error of the system, and $\boldsymbol{\beta}$ is the basis function.

Let $\widehat{\boldsymbol{W}}$be the estimation of $\boldsymbol{W}^*$ and let the sliding mode surface $\boldsymbol{s}$ as the input variable of the basis function, and then the dynamic compensation part of the space robot control law is estimated to be

$$\hat{\boldsymbol{f}} = \widehat{\boldsymbol{W}}\boldsymbol{\beta} + \boldsymbol{\epsilon}. \tag{44}$$

3.3.2. Robust Fuzzy Sliding Mode Tracking Control Law. Firstly, assuming that

(1) The output of the fuzzy approximation system is bounded, and the total external disturbance of the estimated system is bounded, which is expressed as

$$\|\hat{\boldsymbol{f}}\| \leq f_M. \tag{45}$$

(2) The estimation error of the fuzzy approximation system is bounded, which is expressed as

$$\|\boldsymbol{\epsilon}\| \leq \epsilon_M, \tag{46}$$

where ϵ_M is a positive constant.

(3) The estimated weight of the fuzzy approximation system is bounded, which is expressed as

$$tr\left(\boldsymbol{W}^{*\mathrm{T}}\boldsymbol{W}^*\right) \leq W_M, \tag{47}$$

where W_M is a positive constant.

The input variable of the fuzzy system is designed as

$$\boldsymbol{x} = \left[\boldsymbol{s}^{\mathrm{T}}, \dot{\boldsymbol{s}}^{\mathrm{T}}\right]^{\mathrm{T}}. \tag{48}$$

Then, the control law can be reshaped as

$$\boldsymbol{F}_{\boldsymbol{g}} = -\widehat{\boldsymbol{W}}\boldsymbol{\beta}(\boldsymbol{x})\boldsymbol{K}_{\boldsymbol{v}}\boldsymbol{s} - \boldsymbol{v}, \tag{49}$$

where $\boldsymbol{v}$ is a robust term used to overcome approximation errors and interference. Let $\tilde{\boldsymbol{W}} = \hat{\boldsymbol{W}} - \boldsymbol{W}^*$ be the estimation error of the optimal weight matrix.

According to the above control law, there is

$$\begin{aligned}\boldsymbol{H}_g\dot{\boldsymbol{s}} &= -(\boldsymbol{C}+\boldsymbol{K}_v)\boldsymbol{s} - \hat{\boldsymbol{W}}\boldsymbol{\beta} - \boldsymbol{\epsilon} + \boldsymbol{f} + \boldsymbol{F}_d - \boldsymbol{v} \\ &= -(\boldsymbol{C}+\boldsymbol{K}_v)\boldsymbol{s} - \tilde{\boldsymbol{W}}\boldsymbol{\beta} - \boldsymbol{\epsilon} + \boldsymbol{F}_d - \boldsymbol{v} = -(\boldsymbol{C}+\boldsymbol{K}_v)\boldsymbol{s} + \boldsymbol{\zeta}_1,\end{aligned} \tag{50}$$

where $\boldsymbol{\zeta}_1 = -\tilde{\boldsymbol{W}}\boldsymbol{\beta} - \boldsymbol{\epsilon} + \boldsymbol{F}_d - \boldsymbol{v}$, and $\|\boldsymbol{\epsilon}\| \le \epsilon_M$, $\|\boldsymbol{F}_d\| \le F_{dM}$.

The robust term $\boldsymbol{v}$ is designed as

$$\boldsymbol{v} = (\epsilon_M + \boldsymbol{F}_{dM})\boldsymbol{sgn}(\boldsymbol{s}). \tag{51}$$

The adaptive updating law of the optimal weight matrix is taken as

$$\dot{\hat{\boldsymbol{W}}} = \sigma\boldsymbol{s}\boldsymbol{\beta}^{\mathrm{T}}, \tag{52}$$

where σ is a positive constant.

Theorem 3. *For the dynamics of a space robot system considering interference, the fuzzy approximation method is used to estimate the uncertainty of the system, and the weight matrix updating law is set as formula (52), then the system under the action of the control law (49) will gradually converge to the equilibrium point.*

Proof. Define the Lyapunov function as follows:

$$L_{fu} = \frac{1}{2}\boldsymbol{s}^{\mathrm{T}}\boldsymbol{H}_g\boldsymbol{s} + \frac{1}{2\sigma}\mathrm{tr}\left(\tilde{\boldsymbol{W}}^{\mathrm{T}}\tilde{\boldsymbol{W}}\right). \tag{53}$$

Derivation of the above formula is

$$\dot{L}_{fu} = \boldsymbol{s}^{\mathrm{T}}\boldsymbol{H}_g\dot{\boldsymbol{s}} + \frac{1}{2}\boldsymbol{s}^{\mathrm{T}}\dot{\boldsymbol{H}}_g\boldsymbol{s} + \frac{1}{\sigma}\mathrm{tr}\left(\tilde{\boldsymbol{W}}^{\mathrm{T}}\dot{\tilde{\boldsymbol{W}}}\right). \tag{54}$$

Substituting equation (50) into the above equation,

$$\begin{aligned}\dot{L}_{fu} &= -\boldsymbol{s}^{\mathrm{T}}\boldsymbol{K}_v\dot{\boldsymbol{s}} + \frac{1}{2}\boldsymbol{s}^{\mathrm{T}}\left(\dot{\boldsymbol{H}}_g - 2\boldsymbol{C}_g\right)\boldsymbol{s} + \frac{1}{\sigma}\mathrm{tr}\left(\tilde{\boldsymbol{W}}^{\mathrm{T}}\dot{\tilde{\boldsymbol{W}}}\right) \\ &\quad + \boldsymbol{s}^{\mathrm{T}}(-\boldsymbol{\epsilon} + \boldsymbol{F}_d - \boldsymbol{v}) = -\boldsymbol{s}^{\mathrm{T}}\boldsymbol{K}_v\dot{\boldsymbol{s}} + \frac{1}{\sigma}\mathrm{tr}\left(\tilde{\boldsymbol{W}}^{\mathrm{T}}\dot{\tilde{\boldsymbol{W}}}\right) \\ &\quad + \boldsymbol{s}^{\mathrm{T}}\left(-\tilde{\boldsymbol{W}}\boldsymbol{\beta}\right) + \boldsymbol{s}^{\mathrm{T}}(-\boldsymbol{\epsilon} + \boldsymbol{F}_d - \boldsymbol{v}) = -\boldsymbol{s}^{\mathrm{T}}\boldsymbol{K}_v\dot{\boldsymbol{s}} \\ &\quad + \frac{1}{\sigma}\mathrm{tr}\left(\tilde{\boldsymbol{W}}^{\mathrm{T}}\left(\dot{\tilde{\boldsymbol{W}}} - \sigma\boldsymbol{s}\boldsymbol{\beta}^{\mathrm{T}}\right)\right) + \boldsymbol{s}^{\mathrm{T}}(-\boldsymbol{\epsilon} + \boldsymbol{F}_d - \boldsymbol{v}).\end{aligned} \tag{55}$$

Substituting the updating law (52) into the above equation,

$$\dot{L}_{fu} = -\boldsymbol{s}^{\mathrm{T}}\boldsymbol{K}_v\dot{\boldsymbol{s}} + \boldsymbol{s}^{\mathrm{T}}(\boldsymbol{\epsilon} + \boldsymbol{F}_d - \boldsymbol{v}), \tag{56}$$

due to

$$\begin{aligned}\boldsymbol{s}^{\mathrm{T}}(-\boldsymbol{\epsilon} + \boldsymbol{F}_d - \boldsymbol{v}) &= \boldsymbol{s}^{\mathrm{T}}(-\boldsymbol{\epsilon} + \boldsymbol{F}_d) - \boldsymbol{s}^{\mathrm{T}}\boldsymbol{v} \\ &= \boldsymbol{s}^{\mathrm{T}}(-\boldsymbol{\epsilon} + \boldsymbol{F}_d) - \boldsymbol{s}^{\mathrm{T}}(\epsilon_M + F_{dM})\ \mathrm{sgn}\ (\boldsymbol{s}) \\ &= \boldsymbol{s}^{\mathrm{T}}(-\boldsymbol{\epsilon} + \boldsymbol{F}_d) - \|\boldsymbol{s}\|(\epsilon_M + F_{dM}) \leqslant 0\end{aligned} \tag{57}$$

Then, there is

$$\dot{L}_{fu} \leqslant -\boldsymbol{s}^{\mathrm{T}}\boldsymbol{K}_v\dot{\boldsymbol{s}} \leqslant 0. \tag{58}$$

If $\dot{L}_{fu} \equiv 0$, then $\boldsymbol{s} \equiv 0$. According to the LaSalle invariance principle, the closed-loop system has the property of gradual stability as $t \longrightarrow \infty$, $\boldsymbol{s} \longrightarrow 0$. Thus, the error of the pose in exponential form and the velocity will gradually converge to zero as $\boldsymbol{\eta} \longrightarrow 0$, $\boldsymbol{V}_e \longrightarrow 0$; so the pose and velocity of the end-effector will gradually converge to desired values as $\boldsymbol{g} \longrightarrow \boldsymbol{g}_d$, $\boldsymbol{V} \longrightarrow \boldsymbol{V}_d$.

Proof completed. □

4. Simulation Example

Considering a planar, three-DOF space robot as a simulation case and verifying the effectiveness and control performance of the proposed control strategies. The initial pose and the final pose are given as follows:

$$\begin{aligned}\boldsymbol{g}_0 &= \begin{bmatrix} & & 5.797 \\ & \boldsymbol{E}_{3\times3} & 0.6105 \\ & & 0 \\ 0 & & 1 \end{bmatrix}, \\ \boldsymbol{g}_n &= \begin{bmatrix} & & 5.296 \\ & \boldsymbol{R}_z(30^\circ) & 1.9615 \\ & & 0 \\ 0 & & 1 \end{bmatrix}.\end{aligned} \tag{59}$$

The desired trajectory $\boldsymbol{g}_d(t)$ is planned based on the drive transformation method and screw theory.

4.1. Simulation Results of PD Controller. Simulations of the PD controller are performed in MATLAB/Simulink, and the results are obtained as follows. The state error varying with time is shown in Figure 3(a), as is seen in which the maximum error is about 8×10^{-3}, and the steady state error is less than 2×10^{-3}. Figures 3(b)–3(d) show the comparison between the actual and desired pose components. It can be seen that the tracking process is ideal.

The output force on the end effector and the joint control torque obtained by the PD control law is shown in Figures 4 and 5 separately. The joint control torque is relatively small, and no saturation occurs. The space robot's joint motions are shown in Figures 6 and 7. It can be seen

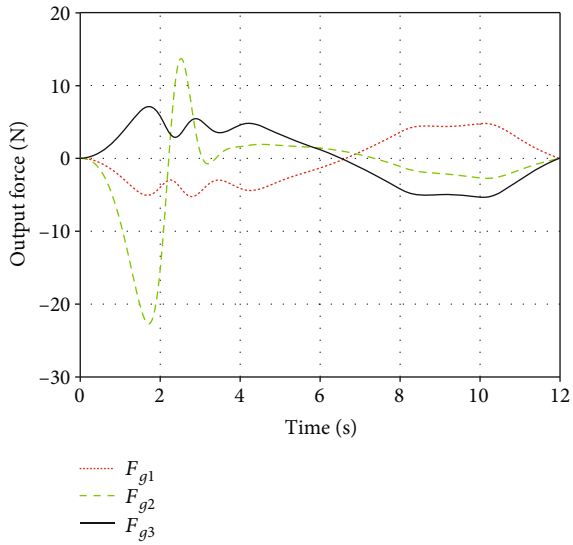


Figure 4: Output force.

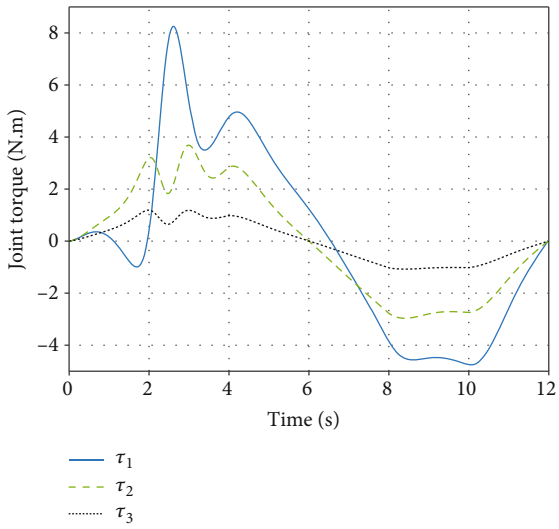


Figure 5: Joint control torque.

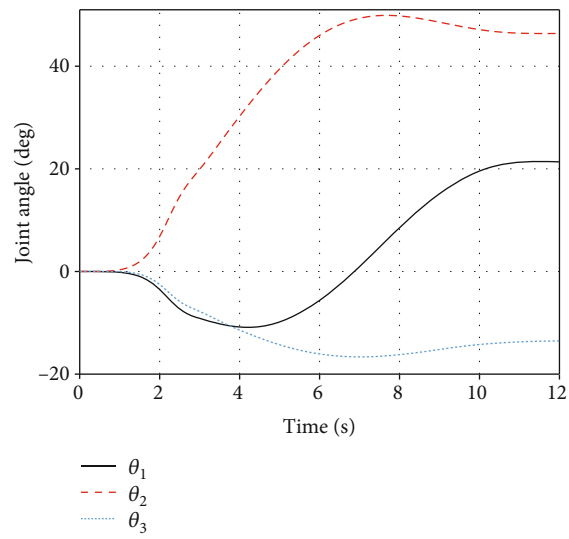


Figure 6: Joint angles.

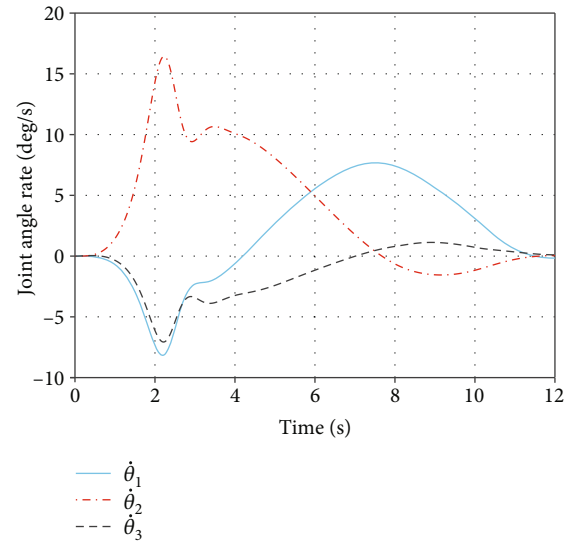


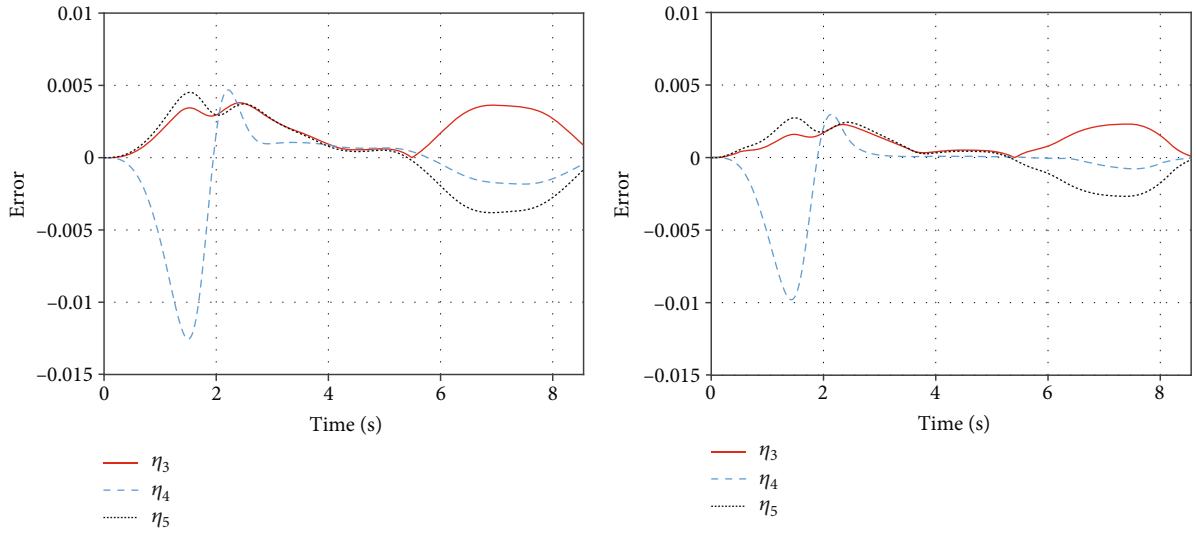
Figure 7: Angular velocity.

that the joint motions are continuous and smooth without sudden change. To sum up, the designed PD controller performs well under the gentle motion state.

4.2. Simulation Results of the Robust Sliding Mode Controller. The simulation object remains the same as in the previous section, and the initial and final poses remain unchanged. The desired trajectory is replanned to move faster. The parameters of the sliding mode controller are set as $\boldsymbol{K}_v = \text{diag}([0, 0, 667, 667, 667, 0])$, $\boldsymbol{\Lambda} = \text{diag}([0, 0, 4.20, 4.20, 4.20, 0])$, and $\rho = 3.3$.

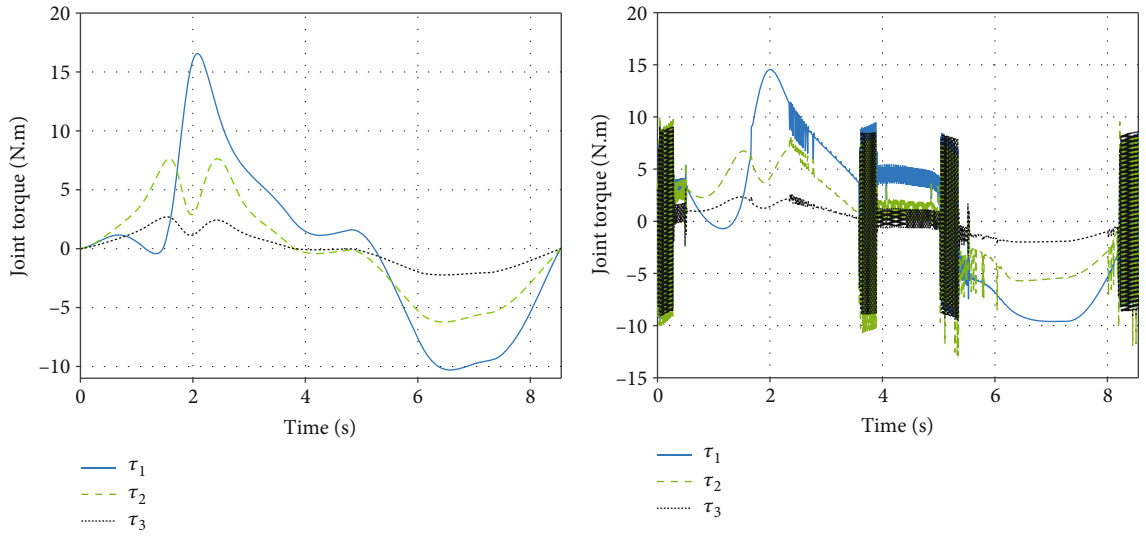
The PD control strategy and the robust sliding mode tracking control strategy are simulated, respectively, and simulation results are obtained as follows. From comparisons of Figures 8(a) and 3(a), it can be seen that accuracy of PD control has been significantly reduced, the maximum error has reached 0.013, and the steady-state error also increases. In the same situation, the accuracy of the robust sliding mode control is higher, the maximum error is about 9×10^{-3}, and the steady-state error remains within 2×10^{-3} as in Figure 8(b). Figures 9(a) and 9(b) show the joint control torque of the two control algorithms. In contrast, the control torque of sliding mode control has a specific chattering phenomenon.

4.3. Simulation Results of the Robust Fuzzy Sliding Mode Controller. The simulation object remains the same as in the previous section, and the initial and final poses remain unchanged. Set the parameters of the robust sliding mode controller as $\boldsymbol{K}_v = \text{diag}([0, 0, 1067, 967, 967, 0])$,$\boldsymbol{\Lambda} = \text{diag}([0, 0, 4.20, 4.20, 4.20, 0])$,and $\rho = 5$. Seven fuzzy



(a) PD controller (b) Robust sliding mode controller

Figure 8: Comparison of tracking error of two control algorithms.



(a) PD controller (b) Robust sliding mode controller

Figure 9: Comparison of joint control torque of the two control algorithms.

subjection functions of the fuzzy sliding mode controller are as follows:

$$\mu_{A_k^1}(x) = \frac{1}{1 + e^{5(x+(\pi/4))}},$$

$$\mu_{A_k^2}(x) = e^{-0.5\left(\frac{x+1}{0.25}\right)^2},$$

$$\mu_{A_k^3}(x) = e^{-0.5\left(\frac{x+0.5}{0.25}\right)^2},$$

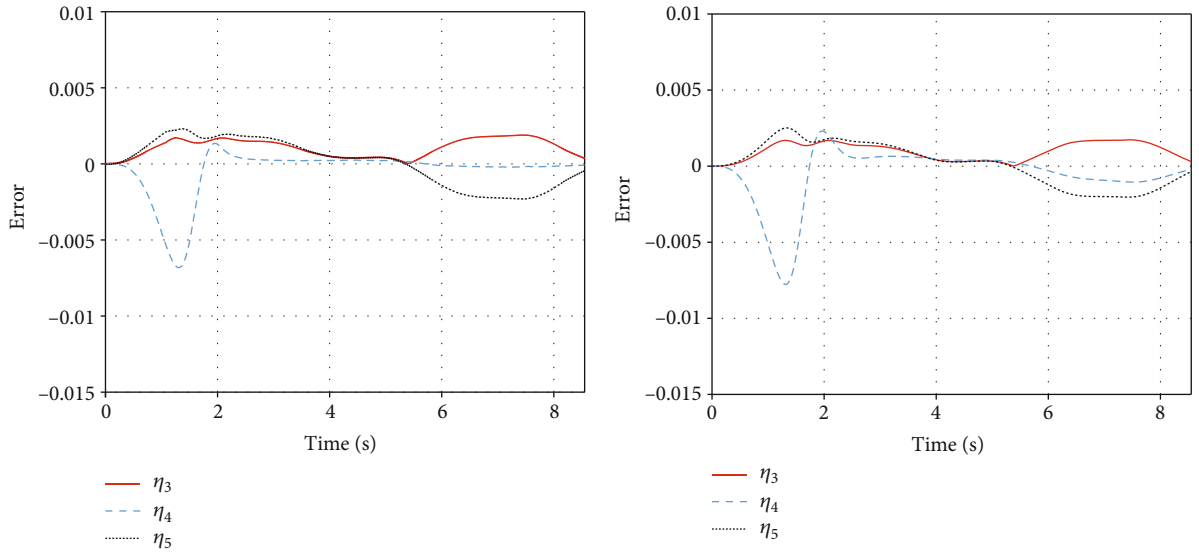
$$\mu_{A_k^4}(x) = e^{-0.5\left(\frac{x}{0.25}\right)^2},$$

$$\mu_{A_k^5}(x) = e^{-0.5\left(\frac{x-0.5}{0.25}\right)^2},$$

$$\mu_{A_k^6}(x) = e^{-0.5\left(\frac{x-1}{0.25}\right)^2},$$

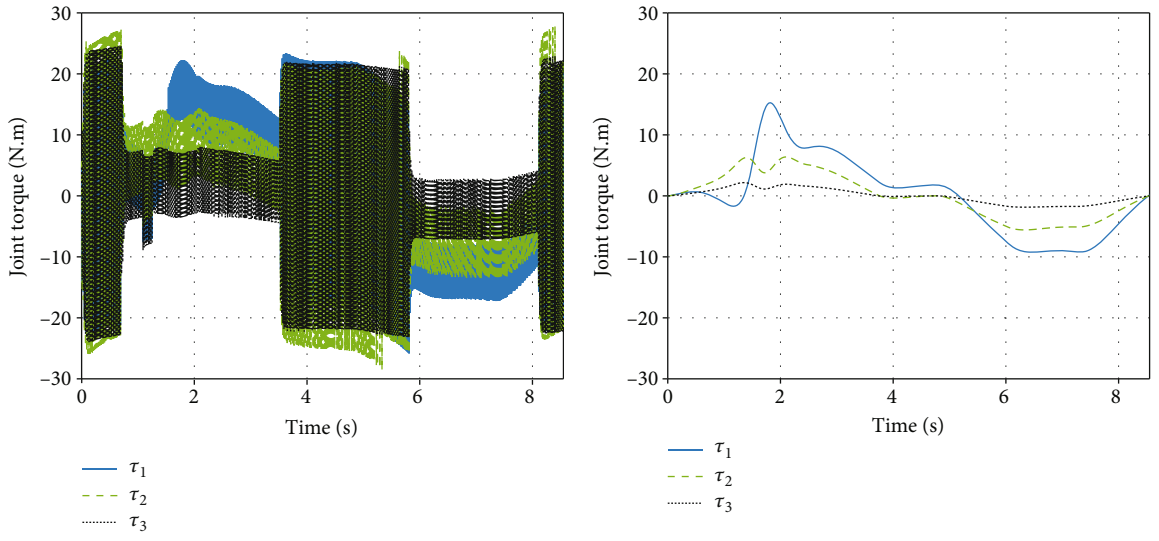
$$\mu_{A_k^7}(x) = \frac{1}{1 + e^{5(x-\pi/4)}}. \quad (60)$$

The robust sliding mode tracking control strategy and the robust fuzzy sliding mode tracking control strategy are simulated, respectively, and simulation results are obtained as follows. It can be seen in Figures 10(a) and 10(b) and



(a) Robust sliding mode controller (b) Robust fuzzy sliding mode controller

Figure 10: Comparison of tracking error of two control algorithms.



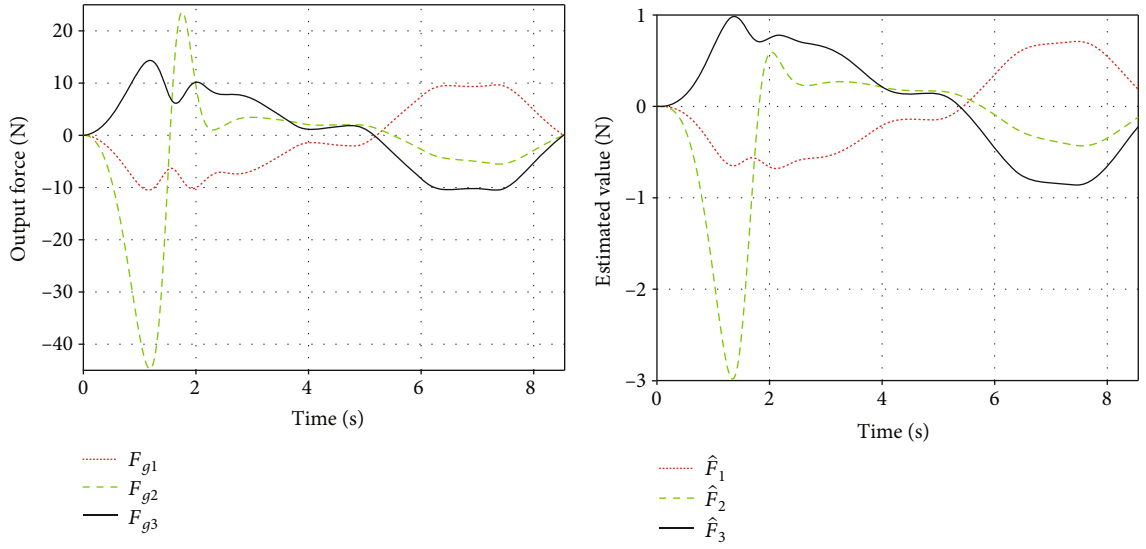
(a) Robust sliding mode controller (b) Robust fuzzy sliding mode controller

Figure 11: Comparison of joint control torque of the two control algorithms.

Figures 11(a) and 11(b) that, after increasing the gain and robust coefficient of the robust sliding mode control, the control accuracy is improved. However, the joint control torque has serious chattering. In contrast, fuzzy sliding mode control can effectively eliminate chattering while ensuring similar accuracy, and the peak value of joint control torque is also significantly better. Figure 12 shows the output force of the end effector and the corresponding estimated value obtained during the fuzzy sliding-mode control algorithm process.

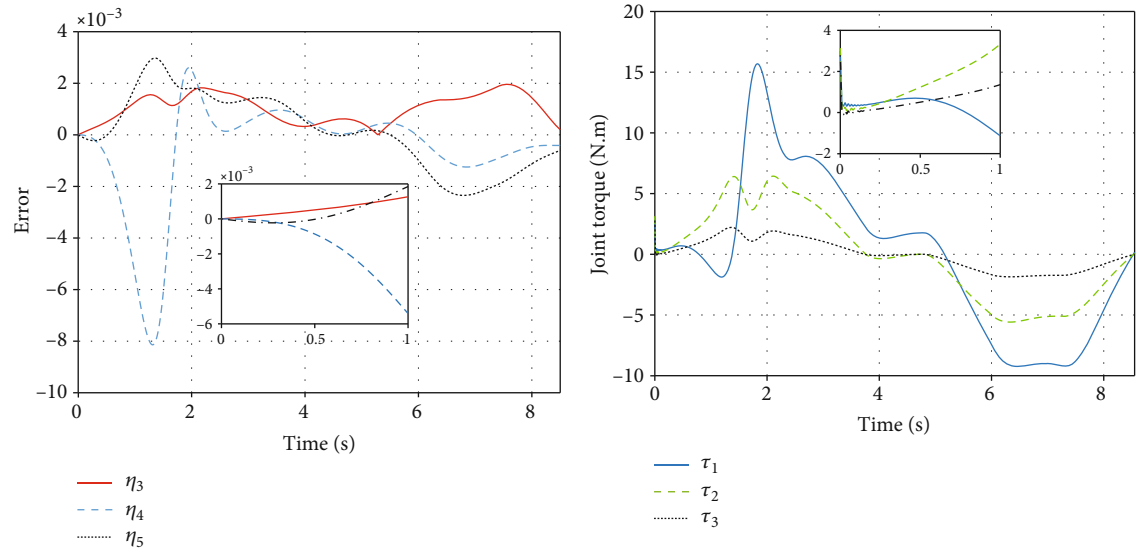
4.4. Simulation Results of Robust Fuzzy Sliding Mode Controller with Interference Force. In order to verify the robust performance of the fuzzy sliding mode controller, the interference force is set as follows:

$$\boldsymbol{F}_d = \begin{bmatrix} 0 \\ 0 \\ 0.6 \sin\left(2t + \frac{\pi}{3}\right) \\ 0.5 \sin\left(3t + \frac{\pi}{4}\right) \\ 0.5 \sin\left(3t + \frac{\pi}{4}\right) \\ 0 \end{bmatrix}. \tag{61}$$



(a) Output force of end-effector (b) Corresponding estimated value

Figure 12: Output force of end-effector of robust fuzzy sliding mode controller.



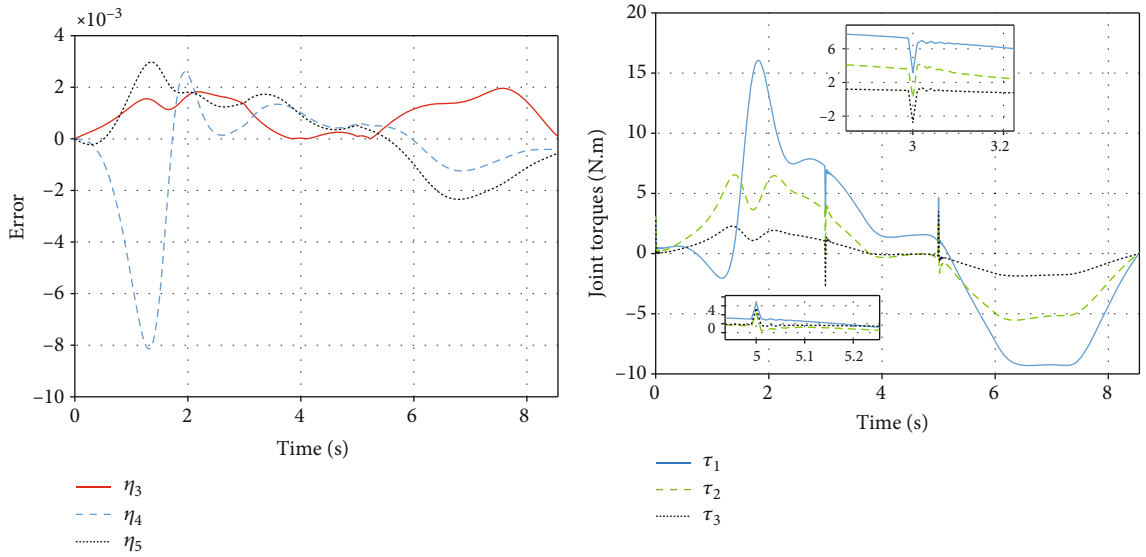
(a) Tracking error (b) Joint control torque

Figure 13: Simulation results of robust fuzzy sliding mode controller with interference force.

The simulation is performed under the same conditions as Section 4.3, and the following results are obtained. After the interference force is added, the state feedback error and joint control torque are shown in Figures 13(a) and 13(b). Comparing with Figures 10(b) and 11(b), respectively, it can be seen that the interference force only has an impact on the system within the first 1 s, which makes the error and joint torque amplify compared with when there is no disturbance, but keep it unanimous afterward. It demonstrates that the fuzzy sliding mode tracking controller has good robustness.

Furthermore, another simulation is performed with a pulse interference force set as follows:

$$\boldsymbol{F}_{d1} = \begin{cases} \begin{bmatrix} 0 \\ 0 \\ -2 + 0.6 \sin\left(2t + \dfrac{\pi}{3}\right) \\ -2 + 0.5 \sin\left(3t + \dfrac{\pi}{4}\right) \\ -2 + 0.5 \sin\left(3t + \dfrac{\pi}{4}\right) \\ 0 \end{bmatrix}, & 3 < t < 5, \\ \boldsymbol{F}_d, & \text{other.} \end{cases} \tag{62}$$



(a) Tracking error

(b) Joint control torque

Figure 14: Simulation results of robust fuzzy sliding mode controller with pulse interference force.

Following results are obtained. It can be seen in Figure 14(a) that the error keeps very similar, while the joint control torque fluctuates slightly in the third and fifth seconds in Figure 14(b); however, the system can still keep stable after the pulse, which demonstrates the robustness of the fuzzy sliding mode tracking controller again.

5. Conclusion

Firstly, based on the Lagrange principle and the generalized Jacobian matrix, the dynamic equations in the joint space and workspace are established separately. Then, a feedback calculation model is proposed based on the screw theory. Furthermore, the PD and robust fuzzy sliding mode control laws are designed separately. Finally, the closed-loop control strategies are constructed by integrating control laws, the GJM, a dynamic model, and a kinematic model. The effectiveness of each control law is verified by simulations, and the control accuracy and performance are compared and analyzed. It can be summarized that the proposed dynamic modelling method is feasible and efficient. The PD controller is proven to have relatively high control accuracy. Nevertheless, the accuracy reduces when the system state changes more dramatically. The robust sliding mode controller is able to overcome such disadvantages, and the fuzzy sliding mode controller could solve the chattering problem.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This work is funded by the National Science Foundation of China. Grant No.11902362.

References

[1] K. Cao, S. Li, Y. She, J. D. Biggs, Y. Liu, and L. Bian, "Dynamics and on-orbit assembly strategies for an orb-shaped solar array," *Acta Astronautica*, vol. 178, pp. 881–893, 2021.

[2] Y. Wang, X. Liang, K. Gong, and Y. Liao, "Kinematical research of free-floating space-robot system at position level based on screw theory," *International Journal of Aerospace Engineering*, vol. 2019, Article ID 6857106, 13 pages, 2019.

[3] A. Jain and G. Rodriguez, "Base-invariant symmetric dynamics of free-flying manipulators," *IEEE Transactions on Robotics and Automation*, vol. 11, no. 4, pp. 585–597, 1995.

[4] W. Xu, C. Li, X. Wang, Y. Liu, B. Liang, and Y. Xu, "Study on non-holonomic cartesian path planning of a free-floating space robotic system," *Advanced Robotics*, vol. 23, no. 1-2, pp. 113–143, 2009.

[5] R. Featherstone and D. Orin, "Robot dynamics: equations and algorithms," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, pp. 826–834, San Francisco, CA, USA, 2000.

[6] Y. Xu and H. Y. Shum, "Dynamic control and coupling of a free-flying space robot system," *Journal of Robotic Systems*, vol. 11, no. 7, pp. 573–589, 1994.

[7] P. J. From, V. Duindam, K. Y. Pettersen, J. T. Gravdahl, and S. Sastry, "Singularity-free dynamic equations of vehicle-manipulator systems," *Simulation Modelling Practice and Theory*, vol. 18, no. 6, pp. 712–731, 2010.

[8] P. Santini and P. Gasbarri, "Dynamics of multibody systems in space environment; Lagrangian vs. Eulerian approach," *Acta Astronautica*, vol. 54, no. 1, pp. 1–24, 2004.

[9] T. R. Kane and D. A. Levinson, *Dynamics, theory and applications*, McGraw–Hill Book Company, New York, 1985.

[10] C. M. Roithmayr and D. H. Hodges, "Dynamics: theory and application of Kane's method," *Journal of Computational and Nonlinear Dynamics*, vol. 11, no. 6, 2016.

[11] A. H. Bajodah, D. H. Hodges, and Y.-H. Chen, "New form of Kane's equations of motion for constrained systems," *Journal of Guidance, Control, and Dynamics*, vol. 26, no. 1, pp. 79–88, 2003.

[12] A. H. Bajodah, D. H. Hodges, and Y.-H. Chen, "Nonminimal Kane's equations of motion for multibody dynamical systems subject to nonlinear nonholonomic constraints," *Multibody System Dynamics*, vol. 14, no. 2, pp. 155–187, 2005.

[13] Y. Xu and M. Huang, "A new method of robotic dynamic analysis—Kane method," *Journal of Sichuan Institute of Light Industry and Chemical Technology*, vol. 8, no. 4, pp. 28–33, 1995.

[14] Z. Yin and X. Ge, "Dynamics analysis of a dual-arm space robot using Kane-Huston method," *Mechanical Science and Technology for Aerospace Engineering*, vol. 31, no. 8, pp. 1344–1348, 2012.

[15] F. C. Park, J. E. Bobrow, and S. R. Ploen, "A lie group formulation of robot dynamics," *The International Journal of Robotics Research*, vol. 14, no. 6, pp. 609–618, 1995.

[16] R. M. Murray, S. S. Sastry, and Z. Li, *A Mathematical Introduction to Robotic Manipulation*, CRC Press, London, 1994.

[17] F. H. Liu and H. T. Wu, "Research on modeling of robot manipulator dynamics based on screw theory," *Journal of Jiangsu University Science and Technology (Natural Science Edition)*, vol. 22, no. 2, pp. 52–55, 2008.

[18] C. Liu, J. Yuan, and M. Li, "Motion control algorithm of coordination manipulation for multi-arms free flying space robots," *Journal of Harbin Institute of Technology*, vol. 36, no. 10, pp. 1338–1340, 2004.

[19] H. Patolia, P. M. Pathak, and S. C. Jain, "Force control in single DOF dual arm cooperative space robot," in *Proceedings of the 2010 Spring Simulation Multiconference*, pp. 1–8, Orlando, Florida, 2010.

[20] M. W. Spong, "On the robust control of robot manipulators," *IEEE Transactions on Automatic Control*, vol. 37, no. 11, pp. 1782–1786, 1992.

[21] A. Zhu, H. P. Ai, and L. Chen, "A fuzzy logic reinforcement learning control with spring-damper device for space robot capturing satellite," *Applied Sciences*, vol. 12, no. 5, p. 2662, 2022.

[22] M. P. Polverini, A. M. Zanchettin, S. Castello, and P. Rocco, "Sensorless and constraint-based peg-in-hole task execution with a dual-arm robot," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 415–420, Stockholm, Sweden, 2016.

[23] H. Li and Z. Liang, "Continual motion control based on RGJM for space robot," in *2015 International Conference on Computer Science and Mechanical Automation (CSMA)*, pp. 292–296, Hangzhou, China, 2015.

[24] M. Galicki, "Finite-time trajectory tracking control in a task space of robotic manipulators," *Automatica*, vol. 67, pp. 165–170, 2016.

[25] Y. Su, C. Zheng, and P. Mercorelli, "Robust approximate fixed-time tracking control for uncertain robot manipulators," *Mechanical Systems and Signal Processing*, vol. 135, article 106379, 2020.

[26] B. Xian, M. S. deQueiroz, D. Dawson, and I. Walker, "Task-space tracking control of robot manipulators via quaternion feedback," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 1, pp. 160–167, 2004.

[27] H. L. Pham, V. Perdereau, B. V. Adorno, and P. Fraisse, "Position and orientation control of robot manipulators using dual quaternion feedback," in *Proceedings of 2010 IEEE International Conference on Intelligent Robots and Systems*, Taipei, Taiwan, 2010.

[28] L. Jiang, Y. Wang, and S. Xu, "Integrated 6-DOF orbit-attitude dynamical modeling and control using geometric mechanics," *International Journal of Aerospace Engineering*, vol. 2017, Article ID 4064328, 13 pages, 2017.

Hindawi
International Journal of Aerospace Engineering
Volume 2022, Article ID 7868142, 12 pages
https://doi.org/10.1155/2022/7868142

Research Article

Continuous Gesture Sequences Recognition Based on Few-Shot Learning

Zhe Liu, Cao Pan, and Hongyuan Wang

Changzhou University, Changzhou 213000, China

Correspondence should be addressed to Zhe Liu; 18416120@smail.cczu.edu.cn and Cao Pan; pccczu@cczu.edu.cn

Received 28 June 2022; Revised 23 August 2022; Accepted 20 September 2022; Published 11 October 2022

Academic Editor: Qing Gao

A large number of demands for space on-orbit services to ensure the on-orbit system completes its specified tasks are foreseeable, and the efficiency and the security are the most significant factors when we carry out an on-orbit mission. And it can improve human-computer interaction efficiency in operations with proper gesture recognition solutions. In actual situations, the operations are complex and changeable, so the gestures used in interaction are also difficult to predict in advance due to the compounding of multiple consecutive gestures. To recognize such gestures based on computer vision (CV) requires complex models trained by a large amount of datasets, it is often unable to obtain enough gesture samples for training a complex model in real tasks, and the cost of labeling the collected gesture samples is quite expensive. Aiming at the problems mentioned above, we propose a few-shot continuous gesture recognition scheme based on RGB video. The scheme uses Mediapipe to detect the key points of each frame in the video stream, decomposes the basic components of gesture features based on certain human palm structure, and then extracts and combines the above basic gesture features by a lightweight autoencoder network. Our scheme can achieve 89.73% recognition accuracy on the 5-way 1-shot gesture recognition task which randomly selected 142 gesture instances of 5 categories from the RWTH German fingerspelling dataset.

1. Introduction

Space on-orbit systems are play a crucial role in aerospace. In relative independent working environments, problems and instrument malfunctions are not always foreseeable. In order to make the on-orbit system completes the target tasks, it is necessary to maintain the services and eliminate the malfunctions. Space on-orbit servicing (OOS) can be regarded as a suitable mode to maintain the on-orbit system and prolong its service life. In the process of space on-orbit service, frequent human-machine interaction operations are required, especially in the tasks such as orbit cleaning, service maintenance, or the cargo resupply [1]. The gesture is kind of complex rigid body which carries abundant human posture information, and the challenge of its state detection coexists with its practice ability in such interaction scene. The gesture recognition mainly has two parts applications in those on-orbit service tasks. One is to use specific gestures as a medium for signal transmission to manipulate machines such as multidegree of freedom robotic arm [2] by different gestures, which can improve the working efficiency and safety, and the other is to encode gestures for mapping different operations. Relevant research in gesture recognition field started earlier and achieved fruitful achievements. One direction to recognize gestures is by the helping of external sensor equipment [3] to gather joints' relative depth assisted with vision information, using methods such as scale invariant feature transform (SIFT) [4] to manually extract gesture features and combine them with feature classifier for recognition, and the other direction is to regression from the RGB images directly. However, to detect RGB images directly based on visual schemes often requires the construction of complex models and large amounts of datasets for training. With the development of

machine learning and computing hardware, the deep learning models with strong capacity of image feature representation such as convolutional neural networks (CNNs) [5] or transformer [6] achieved more than 95% recognition accuracy on 2D or 3D gesture recognition tasks. Compared with gesture image recognition, continuous gesture recognition requires the model with stronger ability of feature extraction to recognize from gesture sequences due to the blur by the motions. Thus, we usually extract instances' features by modeling the whole gesture sequence, such as using representative long short-term memory (LSTM) or dynamic time warping (DTW) methods in the task with the help of external sensors.

When the above models are dealing with the single gesture or continuous gesture recognition tasks, it often needs to input a plenty of gesture datasets that contain various backgrounds and gesture shapes to obtain better performance and generalization ability. Especially when the problem is extended to the recognition of continuous gestures, the scale of the hypothetical space increases rapidly, the cost of labeling datasets is hard to accept, and it is difficult to collect enough instances fit for task in some cases. But some studies have shown that [7] even if the sample in the task is insufficient, the model can also achieve excellent generalization performance and competent the classification or regression task by conduct proper few-shot learning method.

There are many ways to realize few-shot learning schemes. In the early stage, the classification and recognition task of few-shot learning is based on the models with relatively simple structure, such as nearest neighbor or linear support vector machine (SVM). Later, with the development of deep learning fields, transfer learning can also be used to synchronize the model parameters learned from large number of samples to other similar tasks and then fine tune the model parameters based on a small number of samples [8], which are an effective way for few-shot learning, but still need a considerable dataset for previous learning.

Aiming at the above problems of continuous gesture recognition, we propose a gesture recognition scheme based on small samples. Our scheme fine tunes the parameters of the self-attention (SA) module according to the support set provided by the task, using Mediapipe framework to detect the gesture landmarks from the input images, then split and extract those points' feature information of continuous gestures with the help of a lightweight autoencoder network, and output the class of the most similar instance in the support set as the result. Section 3 introduces the Mediapipe, the architecture of autoencoder network, and how we deconstruct the gesture key point information. Section 4 lists the experiments related to the parameter evaluation and the performance of the model.

2. Question Definition and Related Work

2.1. Question Definition. The continuous gesture recognition scheme conducted in this paper mainly aims at the few-shot learning tasks of recognize gesture sequences, and this kind of task only provides the support set and the query set. The support set contains few instances in each class for model training, and the query set contains many instances for prediction. The prediction of gesture essentially is a process of calculating the feature states of the input query set D_q and support set D_s, respectively, by the gesture feature extractor $f_{\mathrm{AE}}(D)$, then output the class $\hat{y}$ of the instance with the maximum cosine similarity in the support set as the prediction results of query instance (1).

$$\hat{y} = \max \left\{ \cos \left(f_{\mathrm{AE}}(D_{s,i}), f_{\mathrm{AE}}\left(D_q\right) \right) \right\} D_{s,i} \in D_s. \quad (1)$$

In practical problems, there will be considerable differences between gesture actions of different instances even they belong to the same class, which is mainly because the gesture actions in the switching process between key gestures in an instance are not constrained. It will cause large prediction deviation if we treat all those frames as the feature frames of continuous gestures. Therefore, we need to eliminate these frames that carry few gesture features to reduce the deviation in the prediction task.

2.2. Related Work

2.2.1. Hand Landmark Detect. Mediapipe [9, 10] is an open-source vision based real-time on-device hand tracking solution and has made many improvements for the real-time recognition on the premise of ensuring the recognition accuracy. Mediapipe divides gesture key point landmark detect task into two parts: palm detection and key points landmark detection. The palm detector reduces the image size by cut out the gesture part from the origin image to improve the efficiency of key point detection. This paper uses this framework to realize the task of gesture key point detection. Details of the gesture key point landmark detection are introduced in Section 3.4, and the mapping coordinates of gesture key points are shown in Figure 1.

With the help of recognition targets' inherent structure information, references [6] extract the gesture feature information directly without further data augmentation, which enlightened the gesture recognition scheme conducted in this paper. They proposed a nonautoregressive coding mechanism by making full use of the internal structure information of 3D gestures such as the interdependence between joints, which provides complete gesture information for the decoder while maintains the parallel structure. By taking the average distance error of joint points as the performance evaluation index, the nonautoregressive transformer model obtained 6.47, 7.55, and 9.80 distance errors on Imperial College Vision Lab (ICVL), Microsoft Research Asia (MSRA), and New York University (NYU) datasets achieved state-of-the-art performance on the 3D gesture recognition tasks.

2.2.2. Continuous Gesture Recognition. The continuous gesture recognition tasks with the external sensors' data, Zhang [11] expressed the gesture feature information with vectors consist of hand coordinates, interfinger distance, fingertip angle, and hand moving speed with the help of leap motion sensor [12] and achieved 98.50% recognition accuracy on the dataset containing 16 dynamic gestures.

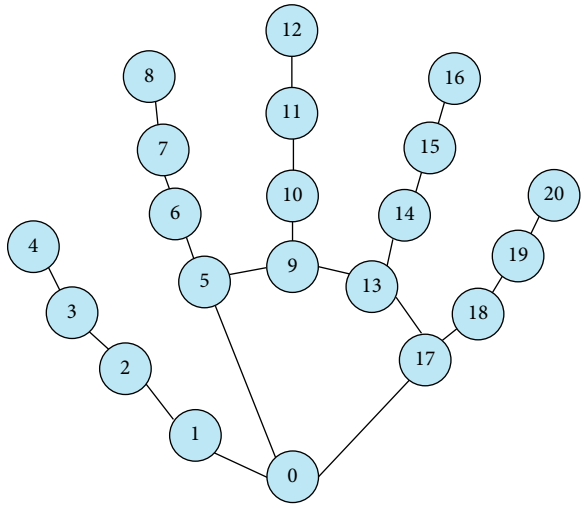


Figure 1: Gesture landmark mapping [9].

In reference [13], a small sample learning scheme for constructing convolutional neural network to analyze gesture sequence data returned by radar is proposed. Four basic gesture sequences were captured using frequency modulated continuous wave (FMCW) 60 GHz radar, and more than 94% recognition accuracy was achieved in the 5-way 1-shot task.

2.2.3. Image Superresolution. Due to the influence of recognition rate and motions, the resolution of the gesture part in the frame could reduce. For these changes in image resolution, we can recover information from those low-resolution images by some superresolution method before recognition. Deep convolutional neural networks (CNNs) have been adopted in superresolution widely. However, CNNs depend on its deep network structure to achieve better performance and often result any inconvenience in training such as instability or hard to converge. Coarse-to-fine super-resolution CNN (CFSRCNN) [14] are gathering complementary contextual information to overcome this issue and proposed a novel feature fusion scheme based on heterogeneous convolutions that achieved high efficiency of single image superresolution (SISR) without decrease the quality of reconstructed images.

Recent generative adversarial networks (GANs) also can help with those low-resolution image problems with small samples [15]. Enhanced superresolution group CNN (ESRGCNN) [16] is kind of flexible and efficient way in SISR. It balanced SISR performance and complexity by using group convolutional and residual learning techniques in group enhanced convolutional blocks and also used an adaptive up-sampling mechanism to make superresolution model more flexible in real tasks.

2.2.4. Transfer Learning. Transfer learning is mainly subsumed model transfer or data transfer, both of which have certain application value in small sample learning tasks. About model transfer scheme, a visual few-shot gesture recognition scheme is proposed in reference [8], which completes the recognition task by transferring the model parameters trained from a large number of datasets to another sampler but similar model. Specifically, train a GoogLeNet on Kungliga Tekniska Högskolan (KTH) dataset to extract gesture features, and use the probability network as the classifier to achieve 99.47% recognition accuracy on Keck gesture dataset. The few-shot continuous gesture recognition scheme in this paper uses similar strategies but uses a more lightweight feature extraction model pretrained on a small dataset. The few-shot recognition scheme in this paper also applies model transfer thoughts, pretrains the gesture feature extraction model from a self-made dataset, and then applies it to the new task support set and query set to extract the gesture feature information to realize the transfer of the model.

2.2.5. Fine Tuning. Reference [7] pointed out that using fine tuning strategy to build a small softmax network based on support set can improve the recognition accuracy of few-shot learning tasks by 2% to 7%. This paper also applies the fine tune strategy on the support set to improve the recognition accuracy of the model on specific task. The details of our fine tune method are described in Section 3.5.

3. Continuous Gesture Recognition

Our scheme mainly consists of two models to fulfill the work, the gesture key point landmark detector and the gesture feature extractor. To obtain the gesture key points' landmarks and relative depth information, we use Mediapipe gesture landmark detector to process the RGB-image sequences, and each frame outputs 63-dimensional vectors V_{lm}. Then decomposes the feature of vector V_{lm} into palm rotation feature and finger bending feature according to Section 3.1 and produces six 3-dimensional vectors V_{input}, the feature extraction model f_{AE} extracts the gesture feature data and outputs a 6-dimensional vector V_{output} as the gesture feature of the current frame. The model architecture is shown in Section 3.2. The vector is used to update the states of various gesture features in time sequence, and the mean value of each state feature is added to obtain a 6-dimensional vector V_{res} as the feature of this continuous gesture sequence instances. The detail of implementation process is described in Section 3.4, then output the instance's class with the highest cosine similarity in the support set as the result of prediction. The structure of the scheme is shown in Figure 2.

3.1. Gesture Information Decomposition. The gestures contain sufficient internal structure information such as joint attachment; thus, we decompose the gesture information before the recognition, so that the model can learn enough information even with few samples. The decomposition of gesture information is based on the 21 gesture key points' landmarks and relative depth information predicted by the gesture key point detection model. For any motions and postures of the palm, decompose them as follows:

(1) The curvature of five fingers is measured by the root of the same finger and the root of the palm at the distal, middle, and proximal phalanges of each finger

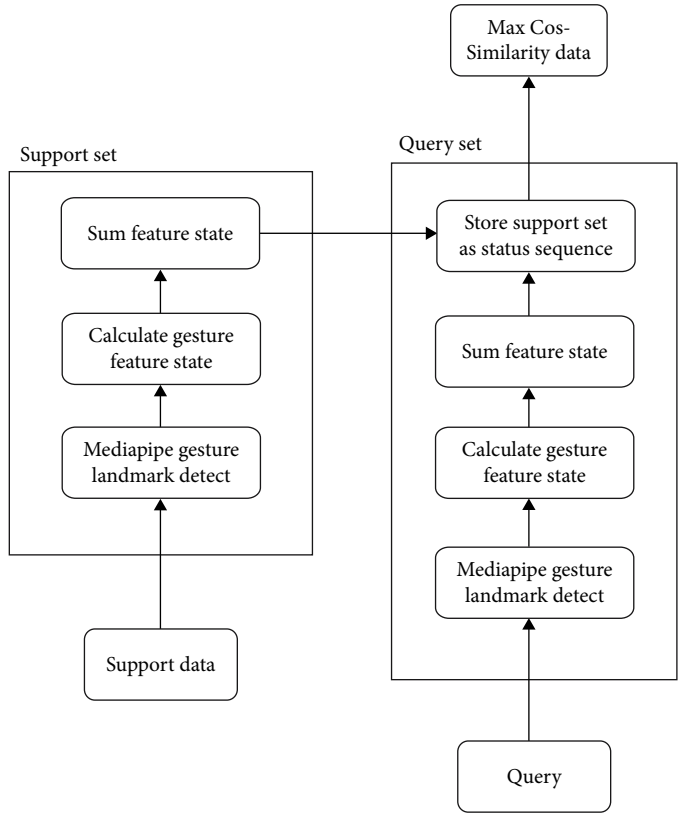


Figure 2: Structure of continuous gesture recognition scheme.

(2) The degree of palm turnover is measured by the distance between the keys at the base of the index and ring fingers

(3) The palm yaw degree is measured by the normalized vector from the palm root key to the middle finger root key

(4) The palm pitch degree is measured by the relative depth of the root of the middle finger to the palm root

The data in the gesture landmark vector V_{lm} are the fixed relative position; thus, there is no need to rescale the coordinates due to the different distance between the palm and the camera. The continuous gesture recognition method conducted in this paper divides the gesture feature into the above four basic parts.

3.1.1. Five Finger Joint Curvature Measurement. Due to the limitations of the palm joint, the fingers can only curl up and stretch in space. We use the five fingers bending degree information at any time, and the relative Euclidean distance between joint points is used to eliminate the influence of the palm rotation angle in space on the measurement of finger bending degree.

For any normal palm at any time, the included angle between the distal phalanx and the middle phalanx of the little thumb, ring finger, middle finger, and index finger is about 90-180 degrees; the included angle between the middle phalanx and the proximal phalanx is about 70-180 degrees; and the included angle between the proximal phalanx and the metacarpal bone is about 80-110 degrees. Although No. I metacarpal bone is more flexible than other four metacarpal bones, we assume that the angle between its proximal phalanx and metacarpal bone is roughly 90-180 degrees.

The gesture landmarks in this scheme are coordinates in three-dimensional space predicted by Mediapipe gesture landmark detector. Therefore, for the thumb, the Euclidean distance between the landmarks of the fingertip, middle segment, and No. I metacarpal bone and the landmarks at the root of the wrist is selected as the measurement of the bending degree (4). The other four fingers' bending degree consist of three parts' European distance, the top of distal phalanx and the top of proximal phalanx, the top of middle phalanx and the bottom of proximal phalanx, and the top of proximal phalanx and the wrist root. Using the embedded vector V_{lm} of five finger landmark data in space output by Mediapipe, calculate the five fingers' feature vector of the bending degree by (2).

3.1.2. Palm Turnover Measurement. In this scheme, the Euclidean distance between the II and IV metacarpal bones after rescaling is selected as the measurement of the degree of palm rotation. The landmarks of the gestures' key points corresponding to the II and IV metacarpal bones are 5 and 13, and the subscripts in the feature dataset are 13 and 37.

The Euclidean distance between them is calculated to get the result of the palm turnover's degree (2).

$$V_{\text{input}}[15] = \sqrt{\left(V_{lm}^{(15)} - V_{lm}^{(39)}\right)^2 + \left(V_{lm}^{(16)} - V_{lm}^{(40)}\right)^2}. \quad (2)$$

The formula that measures palm turnover's degree is continuous, monotonic, and differentiable in a rotation cycle; the degree range is between -0.06 and 0.06. The measurement value obtained by the counterclockwise rotation of the palm at the top angle increases from -0.06 to 0.06, and the measurement value obtained by the clockwise rotation decreases from 0.06 to -0.06, stripping the influence of the yaw and pitch angle on the measurement turnover.

3.1.3. Palm Pitch Measurement. The pitch degree of the palm in space is measured by the relative distance between the root of the middle finger and the root of the palm. The subscripts of those two points in the landmark map are 9 and 0. The measurement result of the pitch degree is shown in the following equation.

$$V_{\text{input}}[16] = \left(V_{lm}^{(38)} - V_{lm}^{(3)}\right). \quad (3)$$

3.1.4. Palm Yaw Measurement. The palm yaw degree is measured by the two-dimensional normalized vector of the palm root point to the middle finger root in space. This measurement method can eliminate the influence of the other two spatial angles on the measurement of palm yaw angle. Palm yaw angle is measured by the following equations.

$$\rho = \sqrt{V_{\text{input}}[17]^2 + V_{\text{input}}[18]^2}, \quad (4)$$

$$V_{\text{input}}[17 + i] = \frac{\left(V_{lm}^{(27+i)} - V_{lm}^{(i)}\right)}{\rho}\ i \in \{0, 1\}. \quad (5)$$

After the decomposition of gesture feature information, we get six vectors to preliminary describe the gesture in each frame. Then embedded them into a 17-dimensional vector V_{input} according to the subsequence of bending degree of thumb, index finger, middle finger, ring finger, little finger, and the palm pose vectors, the embedding map is shown in Figure 3. The embedded vector V_{input} cannot represent the gesture feature directly, so we need to further extract features from it.

3.2. Architecture. The model used to extract the decomposed gesture features in the few-shot learning task is a pretraining model combined multiple autoencoders with a self-attention module [17]. The goal of pretraining is to use an adequate training set including different actions and postures palm to get a lightweight model with strong generalization ability, so that we can fine tune the parameters by different tasks to improve model's performance a step further. Therefore, we added a self-attention mechanism in front of the encoder to extract content information in specific tasks [18]. Initially, we only train the autoencoder module to extract single gesture feature form the training set that independent of query set, when it is applied to the real tasks, then train the self-attention module's parameters by the support set to focus on the context information of continuous gestures, which is an effective way to improve the recognition accuracy of the model for the current task.

During prediction, the model is used to calculate the features of various samples in the support set, extract the feature information of instances in the query set, and then extract the feature state and eliminate the impact of gesture action change frames on the overall recognition task. Finally, calculate the cos similarity and output the class of the highest similarity instance in support set as the prediction result.

3.2.1. Autoencoder. The distribution of various gesture feature data obtained from the above decomposition in the original space is linear inseparable. It will loss a large amount of information if we use linear feature extraction method. Therefore, our model relies on the autoencoder to compress the vector dimension to extract gesture features. Section 4 shows relevant experiments. The input of the model is the decomposed 17-dimensional vector V_{input}. Since the coordinate value range of gesture key points recognized by Mediapipe is stored as a percentage value of the input image boundary according to the distance, it is easy to cause gradient disappearance if we use V_{input} for model training without any process. We use (6) to process the datasets with layer normalization firstly, where γ is the offset value and E is the mean value of V_{input}.

$$V_{\text{input}}[i] = \frac{V_{\text{input}}[i] - E\left(V_{\text{input}}\right)}{\sqrt{\text{Var}\left(V_{\text{input}}\right) + \gamma}}\ i \in N. \quad (6)$$

The standardized data is divided into six 3-dimensional vectors V_{di} according to the embedding method mentioned above, which is, respectively, transmitted to different autoencoders. Each encoder is composed of two hidden layers and independent of each other, and they have different parameters and connected to a decoder, respectively. The decoder is used to restore the output value of the encoder back to the input vector value. Calculate the loss function according to the decoders' results, and iteratively train the parameters of each batch. The strategy of training is introduced in Section 3.6. The bending degree feature of the thumb is extracted from the encoder subscript from 0 to 4, and the palm posture feature is extracted from the encoder subscript 5. When extracting features, the model only calculates the values of each encoder and embeds them into 6-dimensional vector as output. The feature extractor model structure is shown in Figure 4.

The 6-dimensional vector V_{output} indicates the gesture feature of the current frame. There are six elements in the embedded vector. The top five are the feature values of each finger's bending degree, and the last is the posture feature of the palm. The extracted gesture feature vectors of the whole frames are input into the state extraction module to eliminate the impact of gesture frames that did not carry adequate feature information on the gesture recognition task.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Thumb bending degree			Index finger bending degree			Middle finger bending degree			Ring finger bending degree			Pinky finger bending degree			Hand pose			

Figure 3: Feature vector embedding mapping.

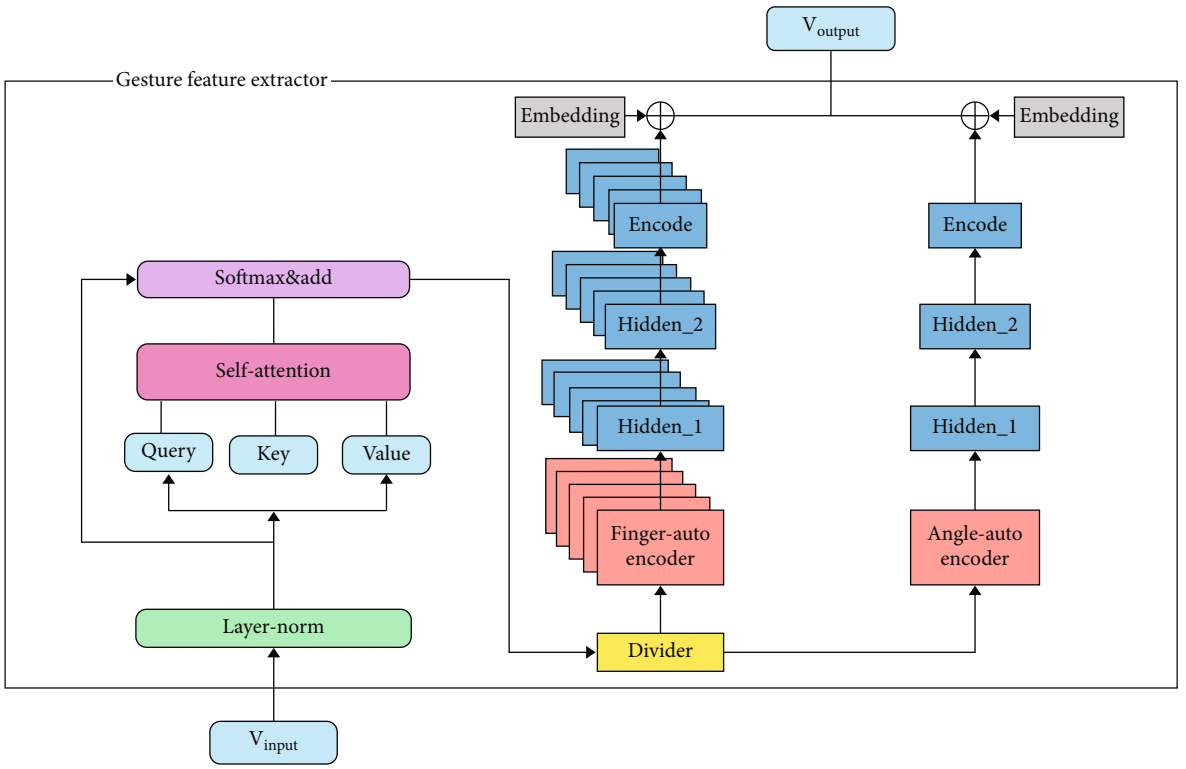


Figure 4: Architecture of the gesture feature extractor.

3.3. State Extraction of Sequences. For different video instances that belong to the same gesture sequence, the gesture states between instances may be misaligned due to the different duration time of a gesture state or the speed of the gesture transformation, and Figure 5 shows the results of using the pretraining model to extract two instances in the support set and the test set that belong to the same class. The test set instance lags behind the support set instance by about 20 frames. And there will be a large gap if we calculate the similarity between the two instances' feature sequences directly. Therefore, we propose a scheme of detecting and matching instances by extract the gesture feature state.

The feature states in this paper refer to the values of the vector V_{output} sequence that are maintained within a given threshold and stable for a certain frame length $\text{Frame}_{\text{len}}$. The method of state extraction receives the V_{output} vector and maintains the mean value avg of six current states. Whenever a new V_{output} is input, check whether the difference between the data in the vector and the mean value avg of their respective states is greater than the given threshold, and it is considered to be the end of a state when the avg is greater than the threshold value. Then check whether the

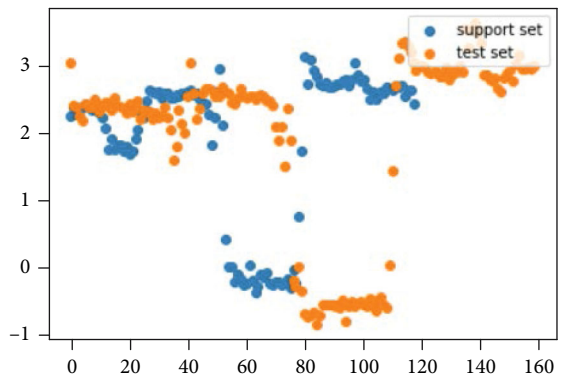


Figure 5: Example of gesture feature dislocation.

frame length $\text{Frame}_{\text{len}}$ reaches the minimum state frame length. Only when it is reached, record the mean value avg of the current state and reset the frame length. Otherwise, update the status mean value by the following equation.

$$\text{avg}_i = \frac{\text{avg}_i * \text{Frame}_{\text{len}} + V_{\text{input}}}{\text{Frame}_{\text{len}} + 1} \quad i \in \{0, 1, 2, 3, 4, 5\}. \tag{7}$$

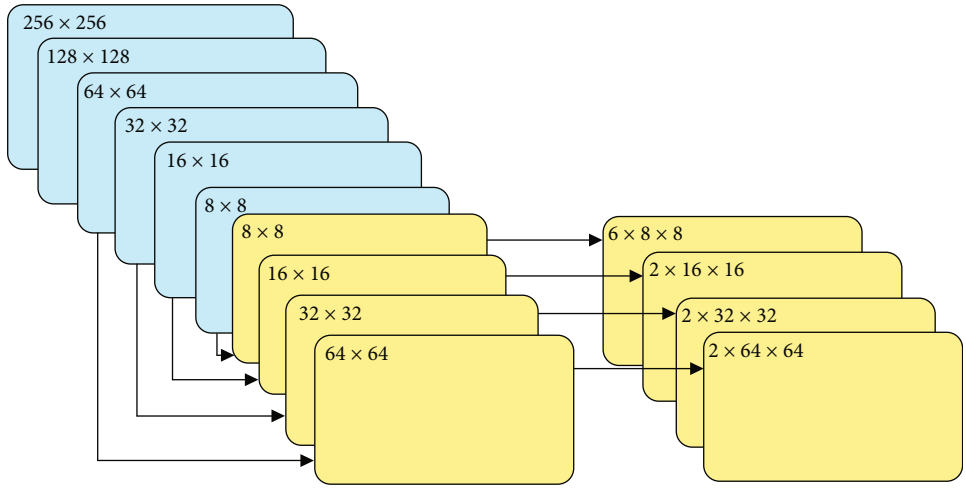


Figure 6: Structure of palm detector [9]. The blue squares are the encoders, and the yellow squares are the decoders.

Using above method to process feature vector of each instance, add the state feature of each dimension record of the instance to obtain a 6-dimensional vector. The matching between different instances is to calculate the similarity between vectors. In this paper, the cosine similarity between the two vectors is selected as the degree of similarity. For instances in the query set, the class of the instance with the greatest cosine similarity is in the support set as the prediction result.

3.4. Gesture Key Landmark Detect. In order to solve the large-scale problem of continuous gesture recognition, we first need to detect the key landmark of gestures in the image. In this paper, Mediapipe is used to detect the 21-key landmark of gestures in RGB images. The model inputs real-time RGB image data and returns the relative coordinates and depth of 21 hand key points in the image. The Mediapipe can achieve more than 30 frame rates in real-time processing tasks when the computing power is sufficient. The core framework of Mediapipe is implemented by C++. The model is mainly divided into two parts. First, the palm detector, which replaces the hand detector, uses a hand positioning frame with direction information to locate the position of the palm in the image. The second is the hand coordinate model, which obtains the coordinate data by detecting the gesture key landmark in the positioning frame processed by the palm detector.

3.4.1. Palm Detector. It is built by a relatively lightweight convolutional neural network with encoding and decoding structure. The palm part or the fist is a rigid object that is easy to recognize, thus to detect the palm or fist part from the input images instead of the whole hand. The palm detector receives the complete image as the input, cut out the palm part from the image when the palm is detected, and then input the result into landmark detector for regression, which greatly reduces the irrelevant contents from the origin image. The cut palm boundary box is square; compared with other shapes, it can reduce the proportion of useless

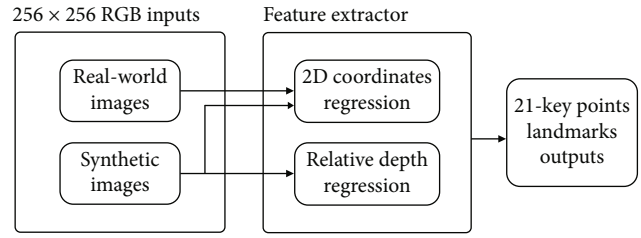


Figure 7: Structure of gesture landmark detector [9].

Table 1: Model performance on RWTH dataset.

Model	Similarity measure	Fine tuning	5-way 1-shot acc	5-way 3-shot acc
Full	MSE	N	77.40%	78.39%
Full	Cosine	N	86.67%	88.20%
Full	MSE	Y	79.45%	80.72%
Full	Cosine	Y	89.73%	90.16%

Table 2: Comparison with other models.

Model	5-way 1-shot accuracy
Nearest neighbor	75.81%
Nearest neighbor (SE)	77.45%
Matching network [20]	85.62%
$\text{CNN}(64)_{\times 4}$ [18]	80.95%
Autoencoder (ours)	89.73%

Table 3: Model performance on Sebastien Marcel dynamic dataset.

Model	Similarity measure	Fine tuning	4-way 1-shot acc	4-way 3-shot acc
Full	MSE	N	89.62%	90.41%
	Cosine	N	91.77%	91.73%
Full	MSE	Y	91.14%	91.11%
	Cosine	Y	92.17%	93.23%

Table 4: Influence of linear and nonlinear dimensionality reduction methods on model recognition accuracy.

Dataset	Similarity measure	PCA 4-way 1-shot	PCA 4-way 3-shot	Autoencoder 4-way 1-shot	Autoencoder 4-way 3-shot
RWTH	MSE	73.48%	74.82%	79.45%	80.72%
	Cosine	68.67%	70.16%	89.73%	90.16%
SMD	MSE	74.62%	74.47%	91.14%	91.11%
	Cosine	71.53%	74.13%	92.17%	93.23%

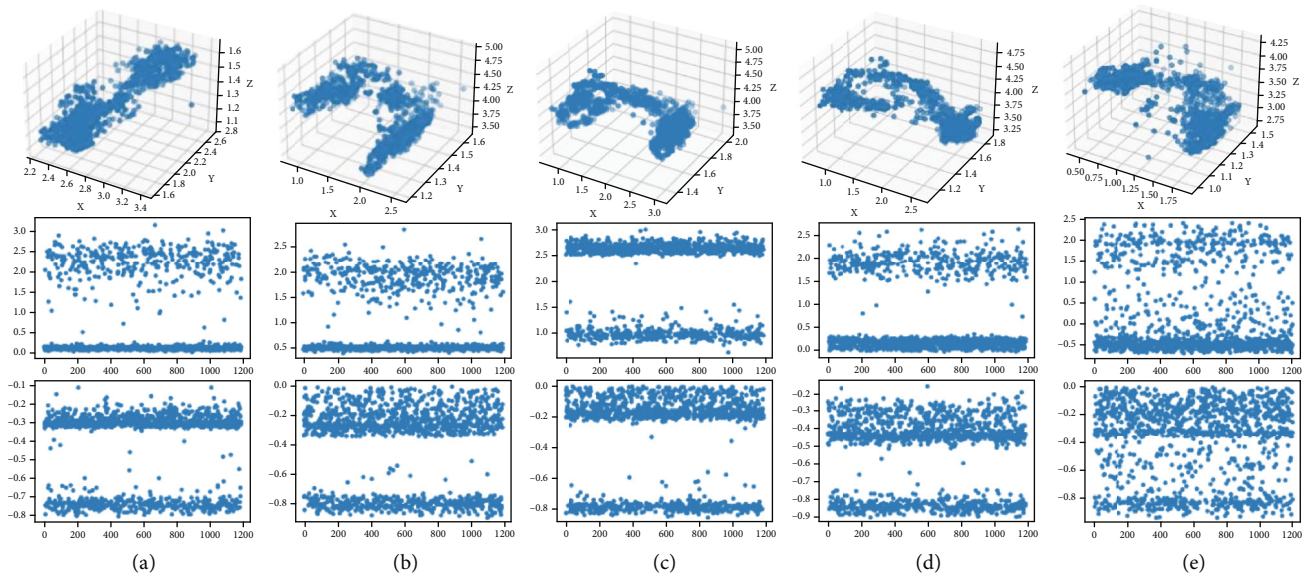


Figure 8: Effects of different dimensionality reduction methods on feature extraction.

information by 3 to 5 times and improve the efficiency of gesture key coordinate detection.

The encode-decode structure in the palm detector can extract the nonlinear features of sample instances through iterative training. The important thing is that the feature receptive field in large-scale images is relatively small, while the feature receptive field in small-scale images is relatively large. Such a structure can enable the model to perceive the feature information in images of different sizes. The focal loss [19] function is used in the model training. The ablation experiment designed in the paper found that the cross entropy loss function with encoder has a better accuracy on the dataset than that without encoder. The average accuracy of the cross entropy loss function without encoder is 86.22%, and that with encoder is 94.07%. The palm detector model structure is shown in Figure 6.

3.4.2. Gesture Landmark Detector. The input data in this part is a smaller image cut by the palm detector, which allows the model to focus on the 21-key point coordinate regression task. The dataset used in the model training includes not only the palm image data in different backgrounds from the real world but also the synthetic gesture data. On the one hand, the synthetic data provides the accurate position information of gesture key points in space, and landmark detector can learn additional relative depth information under the supervision of such data, hence, to improve the performance on the key points' coordinate regression task. On the other hand, in the actual gesture image, due to the high degree of freedom of the gesture itself and various occlusion problems, the accurate position information carried by such data can well enable the model to learn the gesture information of the occluded part. Reference [9] shows that model trained by datasets combined real-world and synthetic gestures performed better than only trained by real-world or synthetic gestures and achieved 13.4% mean-square error (MSE). The main structure of gesture key point detector is shown in Figure 7.

The 2.5D fidelity data obtained by the landmark detector include that the horizontal and vertical coordinates of the key landmark in the RGB images and the relative depth data between each key landmark at the root of the palm. Each time the model receives an image frame as input, cut it into a small-scale image that containing the complete hand by the palm detector and then input it into the key point coordinates detector to detect the three values of 21 key landmark points. Each frame of gesture image outputs 63 data as V_{lm} vector.

3.5. Fine Tuning. The fine-tuning strategy in few-shot learning often means to fine tune the model parameters according to the prediction results of few sample instances in the actual

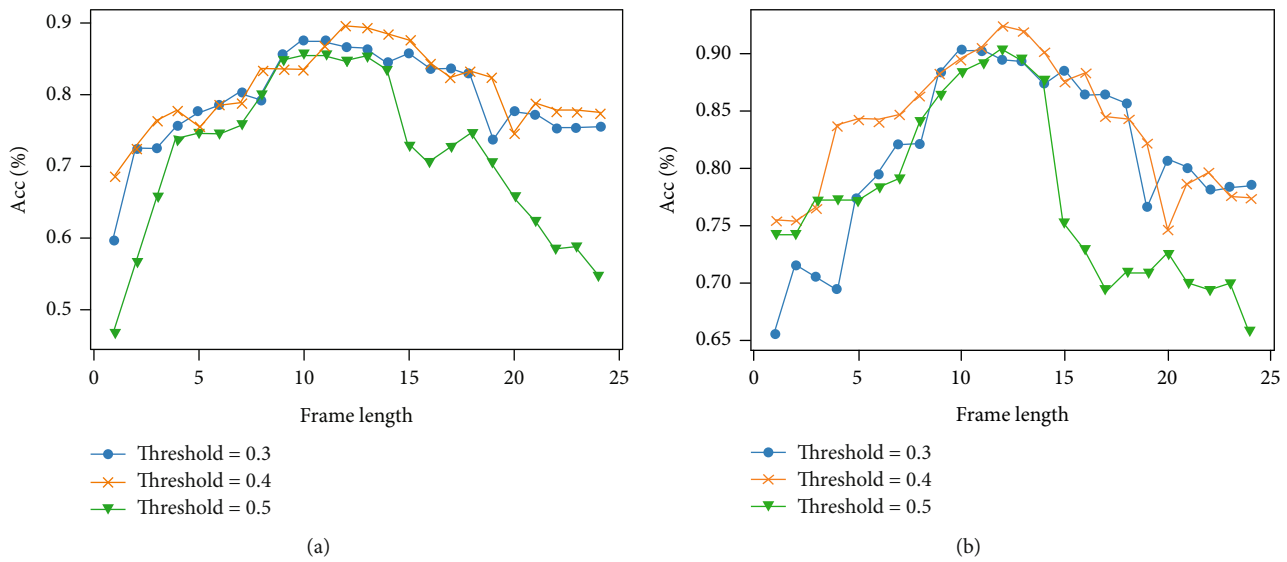


Figure 9: Effects of different threshold and $\text{Frame}_{\text{len}}$ on model recognition accuracy.

task after the pretraining. This strategy is often used to optimize the model of the transfer learning network. Experiments show that using this strategy to optimize the few-shot learning model can improve the prediction accuracy by 2-7% [7].

Because the number of instances in the support set is small, it is not suitable for training new classifiers, also the overall structure of the pretraining model is very light. Therefore, our fine-tuning strategy of the model is only to fine tune the parameters of the self-attention module based on the support set instances and introduce different attention scores for different feature vectors. In order to prevent the model from over fitting, a smaller iteration number and a larger iteration step are selected to fine tune the model. Experiments show that this fine-tuning strategy can improve the recognition accuracy of the model by 2-4%.

3.6. Training. The model is mainly training two parts: the autoencoder and the self-attention module. The dataset used for training autoencoders that extract bending degree contains the gesture key landmark information extracted from 11 RGB videos collected by 2 people who make palm grasping and opening movements under different backgrounds, a total of 1186 instances of training set, 20 pieces of which are randomly selected and divided into a batch. The dataset of the training palm pose feature extraction autoencoder contains the gesture key points extracted from 12 RGB videos collected by 2 people that make various gesture under different backgrounds.

The pretraining model uses empirical risk minimization strategy for training, in which six autoencoders are independent of each other, each learns feature information from the V_{input} set and select the cosine similarity between the decoder results and the instances as the loss function. We use Adam optimization algorithm and initial learning rate to 0.01. Each encoder iterates 20 batches of training sets.

This paper uses the one-head attention module of dot product attention mechanism. In specific tasks, the model uses the support set to train the W_q, W_k, and W_v linear transformation layers in the self-attention module as the fine-tuning strategy. Out of the residual structure in the self-attention module, the weights of the above linear are initialed to zero, which avoid to affect the training of the autoencoders. The stochastic gradient descent (SGD) optimizer is used to train the module parameters, the initial learning rate is set to 0.01, and the momentum is set to 0.9. The cosine similarity between the current frame and other frame features is used as the loss function for training. Each training set contains various randomly selected support set instances, and a total of 5 batches of data are iterated in the training process.

4. Experiment

This part shows some problems encountered in the process of model conduction, the selection of the recognition scheme, and the optimization of details and also includes the experiments to support the solutions we choice. Sections A and B mainly include the experiments about the performance of our method on different datasets, the selection of feature extraction methods for the deconstructed gesture information, different few-shot models' performance on the gesture recognition task, and the comparison of several linear and nonlinear feature extraction schemes. Section C introduces the experiment of feature state extraction of continuous gestures and specifically compared the recognition accuracy of different thresholds and continuous steps on the recognition task. Section D introduces the contribution of different autoencoders and the self-attention module to the accuracy of model recognition. We validated the result by an ablation experiment. The experiments above are mainly conducted on the RWTH German fingerspelling and the Sebastien Marcel dynamic public datasets.

Table 5: Ablation study of the gesture feature extractor model.

Model	Similarity measure	RWTH 5-way 1-shot acc	RWTH 5-way 3-shot acc	SMD 4-way 1-shot acc	SMD 4-way 3-shot acc
Full	MSE	79.45%	80.72%	91.14%	91.11%
	Cosine	89.73%	90.16%	92.17%	93.23%
None AE_5	MSE	79.41%	80.47%	90.36%	91.34%
	Cosine	85.62%	85.95%	91.69%	92.10%
None AE_4	MSE	75.71%	75.64%	89.97%	89.84%
	Cosine	82.88%	84.13%	90.38%	90.75%
None AE_3	MSE	75.34%	77.35%	86.79%	88.19%
	Cosine	84.25%	85.13%	90.20%	91.73%
None AE_2	MSE	79.42%	78.68%	77.69%	77.42%
	Cosine	88.36%	89.25%	81.99%	83.22%
None AE_1	MSE	72.60%	74.26%	76.40%	77.46%
	Cosine	85.62%	85.45%	80.40%	82.31%
None AE_0	MSE	74.66%	74.24%	81.18%	82.54%
	Cosine	86.30%	87.83%	86.27%	87.22%
None SA module	MSE	76.36%	77.69%	88.18%	88.78%
	Cosine	88.54%	88.83%	90.97%	91.02%

4.1. Results

(1) RWTH German fingerspelling dataset

The RWTH German fingerspelling dataset contains 35 different continuous gestures. Each gesture contains 88 video instances of stand 2 and stand 1, hereinafter, referred to as the RWTH dataset. Five continuous gestures are randomly selected in this experiment. Each gesture contains 22 video instances of stand 1 as the support set and query set. The model state threshold and continuous frame length are set to 0.4 and 14, and the average prediction accuracy of the model is obtained by repeatedly selecting 5 prediction categories. The prediction results are shown in Table 1. The table shows that the strategy of using the fine tune attention module in 1-shot or 3-shot recognition tasks can improve the recognition accuracy by 1.96% to 3.06% and achieve better performance by using cosine similarity measure the distance between feature vectors instead of mean square error (MSE).

Table 2 shows the results of various gesture feature extraction methods performing 5-way 1-shot tasks on the RWTH dataset. The nearest neighbor model selects a support instance nearest to the instance as the classification result and uses five gestures randomly selected in the dataset, one instance of each gesture as the support set for training. At the same time, the SE method is used to process the data to improve the model recognition accuracy by 1.64%. Matching network [20] is a model combining shallow CNN and bidirectional LSTM network. $\mathrm{CNN}(64)_{\times 4}$ in the table is a shallow network composed of 4 convolution layers with 64 channels, which is trained by LSTM meta-learner method [21, 22]. According to the prediction results of this task, the method proposed in this paper obtains extra information by dividing gesture feature, thus achieves higher recognition accuracy than other small sample learning methods.

(2) Sebastien Marcel dynamic (SMD) dataset

The dataset contains four consecutive gestures: clip, rotate, stop grasp OK and No, each gesture records the action of 10 people as samples. Table 3 shows the influence of different factors on the recognition accuracy such as similarity measurement, fine-tuning strategy, and the number of instances in each class. Without using fine-tuning strategy to training self-attention module, the accuracy of the model achieves 1% to 3% higher when we use cosine similarity as the similarity measurement than using MSE. The recognition accuracy increases after fine-tuned self-attention module's parameters, and the 1-shot prediction task using cosine similarity measurement method increased 2.54%, which improved most obviously. The recognition accuracy of model using cosine similarity was improved 2.33% by fine-tuning strategy in the 3-shot task, which was more significant than that using MSE as similarity measurement.

4.2. Gesture Feature Extraction. The extraction of gesture features is to reduce the dimension of the basic gesture feature vector V_{input} decomposed by the method above. The experiments in this section show the impact of linear and nonlinear dimensionality reduction methods on the performance of the model and calculate the recognition accuracy of the feature extractor under the principal component analysis (PCA) and autoencoder extraction methods on RWTH and Sebastien Marcel datasets. Table 4 shows that the recognition accuracy

obtained by using the autoencoder method is 10-20% higher than that obtained by PCA on the two datasets.

Figures 8(a)–8(e), respectively, show the feature extraction results of different dimensionality reduction methods on the original data. The first row in the figure shows the distribution of the original data in the three-dimensional space, and the second and the third rows are the distribution maps reduced to 1-dimensional using autoencoder and PCA. Although it is shown in the figure that the within class distance of each eigenvalue obtained by the two dimensionality reduction methods is also small, the distribution of the original data is nonlinear separable, so the line dimensionality reduction method loses more information in the process of dimensionality reduction, which get the inferior accuracy of the prediction.

4.3. State Extraction of Sequences. This section mainly shows the impact of different thresholds and $\text{Frame}_{\text{len}}$ parameters used in the state extraction method on the accuracy of model recognition. The threshold values of 0.3, 0.4, and 0.5 were selected, respectively, to test the model recognition accuracy of the minimum state frame length $\text{Frame}_{\text{len}}$ from 1 to 25 on the RWTH dataset and the SMD dataset. Figure 9(a) shows the experimental results on the RWTH dataset. When the threshold is 0.4 and the $\text{Frame}_{\text{len}}$ is 16, the model achieves the highest recognition accuracy of 91.33%. Figure 9(b) shows the experimental results on the SMD dataset. When the threshold is 0.4 and the frame length is 11, the model achieves the highest recognition accuracy of 92.13%.

4.4. Continuous Gesture Recognition. In order to show the influence of each component in the model on the overall recognition results, we constructed an ablation experiment to show the contribution of each autoencoders and the self-attention module to the model recognition accuracy. The experimental results are shown in Table 5.

Table 5 shows the recognition accuracy of the model on RWTH and SMD datasets with the removal of each autoencoder module or the self-attention module. The influence of each module on the test set's recognition accuracy sampled from RWTH dataset ranges from 1% to 7%, among which the ring finger bending encoder AE_3 contributing 6.85% accuracy of recognition. On the SMD dataset, the influence of each module on the overall recognition accuracy of the model ranges from 1% to 12%. Among them, AE_1 improved 11.77% accuracy in the recognition task, which has the greatest contribution.

5. Summary

This paper conducts a few-shot gesture recognition scheme combined a gesture landmark detector with a lightweight gesture feature extractor. The scheme uses Mediapipe gesture key landmark detection model to recognize gesture key landmark points on RGB image sequences. The recognition results are processed into a 6-dimensional feature vector by the gesture feature extractor. Various feature states in the sequence are extracted and added and then output instances' class with the highest cosine similarity in the support set as the result. In the 5-way 1-shot tasks on RWTH and Sebastien Marcel datasets, the recognition accuracy was achieved 89.73% and 92.17%, respectively.

Also, the gesture feature state extraction scheme in this paper is mainly applicable to the recognition task of a single palm in gesture sequences. We can use the multihead attention structure to extract different association features according to the different context associations in the gesture sequences, to improve the recognition performance of the model in those continuous gesture recognition tasks with strong context correlation, such as sign language recognition.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interests.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (61976028).

References

[1] J. P. Davis, J. P. Mayberry, and J. P. Penn, "On-orbit servicing: inspection repair refuel upgrade and assembly of satellites in space," *The Aerospace Corporation, Report*, vol. 60, 2019.

[2] J. Herron, D. Lopez, J. Jordan et al., "RGB-D robotic pose estimation for a servicing robotic arm," 2022, http://arxiv.org/abs/2207.11537.

[3] E. Rahimian, S. Zabihi, A. Asif, D. Farina, S. F. Atashzar, and A. Mohammadi, "FS-HGR: few-shot learning for hand gesture recognition via electromyography," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 29, pp. 1004–1015, 2021.

[4] L. Jing, "Image feature matching based on improved SIFT algorithm," *International Journal of Circuits, Systems and Signal Processing*, vol. 12, 2018.

[5] A. Mujahid, M. J. Awan, A. Yasin et al., "Real-time hand gesture recognition based on deep learning YOLOv3 model," *Applied Sciences*, vol. 11, no. 9, p. 4164, 2021.

[6] L. Huang, J. Tan, J. Liu, and J. Yuan, *Hand-Transformer: Non-Autoregressive Structured Modeling for 3D Hand Pose Estimation*, Springer, Cham, 2020.

[7] G. S. Dhillon, P. Chaudhari, A. Ravichandran, and S. Soatto, "A baseline for few-shot image classification," 2019, http://arxiv.org/abs/1909.02729.

[8] R. Cheng, *Research and Application of Small Sample Gesture Recognition Based on Convolutional Neural Network*, Taiyuan University of technology, 2020.

[9] F. Zhang, V. Bazarevsky, A. Vakunov et al., "Mediapipe hands: on-device real-time hand tracking," 2020, http://arxiv.org/abs/2006.10214.

[10] C. Lugaresi, J. Tang, H. Nash et al., "Mediapipe: a framework for building perception pipelines," 2019, http://arxiv.org/abs/1906.08172.

[11] P. Zhang, *Research on Recognition Based on Structural Features and Deep Learning*, Henan University of science and technology, 2019.

[12] J. Cheng, *Dynamic Gesture Recognition Based on the Leap Motion Controller*, Jiangsu University of Science and Technology, 2021.

[13] G. Mauro, M. Chmurski, M. Arsalan, M. Zubert, and V. Issakov, *One-Shot Meta-Learning for Radar-Based Gesture Sequences Recognition*, Springer, Cham, 2021.

[14] C. Tian, Y. Xu, W. Zuo, B. Zhang, L. Fei, and C. W. Lin, "Coarse-to-fine CNN for image super-resolution," *IEEE Transactions on Multimedia*, vol. 23, pp. 1489–1502, 2021.

[15] C. Tian, X. Zhang, J. C. W. Lin, W. Zuo, and Y. Zhang, "Generative adversarial networks for image super-resolution: a survey," 2022, http://arxiv.org/abs/2204.13620.

[16] C. Tian, Y. Yuan, S. Zhang, C. W. Lin, W. Zuo, and D. Zhang, "Image super-resolution with an enhanced group convolutional neural network," 2022, http://arxiv.org/abs/2205.14548.

[17] A. Vaswani, N. Shazeer, N. Parmar et al., "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[18] Z. Ji and X. Chai, "Few-shot learning based on self-attention and auto-encoder," *Journal of Tianjin University: Natural science and Engineering Technology Edition*, vol. 67, 2021.

[19] T.-Y. Lin, P. Goyal, R. B. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," 2017, http://arxiv.org/abs/1708.02002.

[20] O. Vinyals, C. Blundell, T. Lillicrap, and D. Wierstra, "Matching networks for one shot learning," 2017, http://arxiv.org/abs/1606.04080.

[21] Q. Cai, Y. Pan, T. Yao, C. Yan, and T. Mei, "Memory matching networks for one-shot image recognition," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 2018.

[22] S. Ravi and H. Larochelle, "Optimization as a model for few-shot learning," in *International Conference on Learning Representations*, Palais des Congrès Neptune, Toulon, France, 2017.

Hindawi
International Journal of Aerospace Engineering
Volume 2022, Article ID 3071244, 12 pages
https://doi.org/10.1155/2022/3071244

Research Article

Variable Structure Control and Its Ground Experimental Test for the Space Station Robot

Feilong Zhang,[1,2,3] Bi Zhang,[1,2] Bing Han,[4] Danyang Qu,[1,2,3] and Xingang Zhao[1,2]

[1]*State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China*
[2]*Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110016, China*
[3]*University of Chinese Academy of Sciences, Beijing 100049, China*
[4]*Hangzhou Innovation Institute, Beihang University, Hangzhou 310051, China*

Correspondence should be addressed to Bi Zhang; zhangbi@sia.cn and Xingang Zhao; zhaoxingang@sia.cn

Received 26 April 2022; Accepted 11 August 2022; Published 14 September 2022

Academic Editor: Chen Pengyun

Building a simulated weightless test system on the ground while making comprehensive comparisons of design controllers for a large and heavy multijointed space station robot is not an easy task. To save cost and improve the efficiency of the test, this paper develops a plan in which controllers undergo preliminary testing in a 6-DOF industrial robot. The key idea is gravity compensation included within the dynamic control algorithm of the robot to replace the function of the microgravity environment. It is generally difficult to build an accurate dynamic model for a serial-joint robot in a practical manner. Therefore, to guarantee the stability of the 6-DOF industrial robot in which the dynamic model is built inaccurately, we propose one of the simplest variable structure (VS) controllers, and the stability of the system is analyzed through the Lyapunov method. Last, experiments are carried out to provide preliminary comparisons among three potential algorithms for the space robot in a low-cost and efficient approach.

1. Introduction

Space robots have been playing an important role in orbital servicing missions, such as assisting the assembly of space stations, capturing or repairing faulty satellites, and cleaning orbital debris [1–5]. The primary characteristics of applied space equipment controllers are typically simple but reliable and effective. At present, there is a lack of confidence in launching equipment controlled by complex advanced control algorithms into space considering the risk and price of failure. Space-bound robots are used to serve the corresponding space station routines and tasks, after their performance and reliability in various types of working conditions are verified by trial and error, and under the simulated weightlessness or microgravity environment on the ground. Furthermore, without the weightlessness simulation system, the space robot designed for the space station may not even carry its own weight on the ground. To date, a variety of strategies have been published to compensate for the gravity of space robots for experimental tests on the ground. [6] emulates the zero-gravity environment for the space manipulator using an air-bearing platform, while the platform is mainly suitable for the planar mechanism whose motion is parallel to the air-bearing table [7]; A microgravity environment can also be created in a plane while in a free fall or a free-falling capsule at the microgravity center; however, the test time is too short, and available space for the robot is limited [8–10]; A microgravity test can be performed in the pool using neutral buoyancy. Nevertheless, the influence of fluid damping on test results cannot be negligible [11]; [12–15] emulate the process of capturing the space target by the free-floating robot mounted on satellites, using two industrial robots. However, the purpose is not to create the microgravity environment for the space robot but to generate the trajectory of the satellite; [16–19] design the suspension system to provide the zero gravity of the space robot for maintaining tension. However, it is difficult to remove the influences of the test results caused by the coupled vibration of the serial multijoint space robot and suspension system. Unfortunately, building a simulated weightless test system

for a large and heavy serial multijoint space station robot while choosing the most practical control algorithm from potential candidates is not a simple task and consumes a large amount of time, manpower, auxiliary facilities, and material resources [6–21].

The central difference between the controller designed for a space robot and that for an industrial robot on the ground is determined by considering gravity compensation or not. The gravity compensation included in the dynamic control algorithm of industrial robots can be regarded as a replacement for simulated weightless system functions. As a result, preliminary comparisons between various potential dynamic control algorithms on a 6-DOF industrial robot can be made to save cost and improve the efficiency of the test. This allows contrasting effects to be obvious and easily acquired, which provides prior knowledge and makes preliminary judgment regarding controllers designed for the space robot.

To achieve high precision and efficiency of robot tracking performance, it is necessary to introduce the robotic dynamics into the controller to reduce the effect of robotic nonlinear dynamic characteristics. The dynamics of serial-link robots have been well covered by almost all standard robotic textbooks [22–27]. Two main approaches are feedforward control and computed torque control [28], and both are based on robotic dynamics combined with one diagonal PD controller. Compared with computed torque control, the feedforward control is computationally less expensive at operating time and more easily realizes a high servo rate. This is all possible because the model-based dynamic compensation is "outside" the servo loop, and then, a fast inner servo loop is achieved. Furthermore, PD plus dynamic feedforward compensation in [22] is one of the simplest and most appealing dynamic controllers for robots [29]. It consists of one linear PD feedback and a nominal robot dynamic model computed as a function along the desired path only. Therefore, the advantage of the simple and efficient structure of this type of algorithm provides a reference to the controller design of the space robot.

Generally, the motion control of robots is not an easy task due to nonlinearity, strong coupling, uncertain modeling structure, or parameters. These factors may cause inaccurate joint motion and lead to poor robot performance. To solve these problems, various dynamic feedforward compensation plus PID-like or plus other single-loop feedback control algorithms and computed torque control-based methods have been proposed, including sliding-mode control (SMC) [30–36] or some other variable structure controllers [37–40], neural networks [41–46], fuzzy control [47, 48], and adaptive control [49, 50]. However, many of them are only tested in simulations or double-jointed robotic systems and may have more difficulties achieving the expected performance in the multijoint serial robotic system. One reason is that the current advanced control algorithms, such as neural networks, fuzzy control, and adaptive control, are characterized by complex structures and are time-consuming, and their computational time is possibly several times larger than the commercial robot servo period which is generally less than 10 ms. This may lead to poor real-time performance, let alone the current situation that most of those advanced control algorithms are mainly based on the computed torque control structure. Furthermore, reliable and practical controllers are always those simple ones in experience. At present, there is a lack of confidence to launch space robots controlled by those complex advanced control algorithms into space on account of the risk and price of failure. By comparison, a kind of variable structure controller is proposed in this paper to enhance the tracking performance of the robot under modeling uncertainty. It shows higher efficiency and a better control performance than [22] in the experiment because it combines the PD plus dynamic feedforward compensation and the Bang-Bang control to achieve compensation for the tracking error caused by the modeling uncertainty. One noticeable merit of the proposed method lies in the easier design and application in real robot systems than the controllers designed in [30–48] because of its simplicity in structure.

The main contributions of this paper are summarized as follows: (i) Since testing the different controllers and comparing them in a serial multijoint space station robot on the ground are difficult, this paper initially develops a strategy for controllers to be preliminarily tested in an industrial robot. The key idea is the gravity compensation included in the dynamic control algorithm of the robot to act in the function of a microgravity environment. It will be more efficient to obtain the superior controller from others; (ii) [22] points out that the dynamic model is always known to be inaccurate, and one reason is that it is severely challenging to acquire the structure of the friction model, let alone the corresponding parameters. To improve the control performance of the system, we propose the simplest variable structure control through the combination of the PD plus dynamic feedforward compensation and the Bang-Bang control together to compensate for tracking error caused by the modeling uncertainty, and the stability of the system is analyzed by Lyapunov theory. Then, we experiment to provide preliminary comparisons among several potential algorithms for the space robot in a low-cost and highly efficient way.

The rest of the paper is organized as follows. In Section 2, the dynamic parameters of the robot are identified by the least square method. In Section 3, to guarantee the stability of the system under the modeling uncertainty, a variable structure controller is designed based on the Lyapunov direct method. Then, we design the PD plus and the cascaded PD plus dynamic feedforward compensation to make an experimental comparison in Section 4. As a result, the cascaded PD plus dynamic feedforward compensation control more easily achieves the high-precision tracking performance than the proposed VS control or the PD plus, under the roughly built robotic dynamics. Finally, we present the conclusions in Section 5.

2. Identification of Dynamic Parameters

A typical 6-DOF serial industrial robot is shown in Figure 1.

Figure 1: 6-DOF serial robot.

The dynamic equation of the 6-link robot is written as

$$\tau = M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + f_c \operatorname{sign}(\dot{q}) + f_v\dot{q}, \quad (1)$$

where $q \in R^{6\times1}$ is the vector of joint displacements, $\tau \in R^{6\times1}$ is the vector of applied torque, $M(q) \in R^{6\times6}$ is the mass matrix of the manipulator, $C(q,\dot{q}) \in R^{6\times6}$ is the centrifugal/Coriolis force matrix, $G(q) \in R^{6\times1}$ is the vector of gravity terms, and $f_v \in R^{6\times6}$ and $f_c \in R^{6\times6}$ are diagonal matrices that consist of the viscous and Coulomb friction parameters, respectively.

Then, in accordance with [51], (1) can be rewritten into the following linear form with $n \times N_s = 6 \times 13$ identifiable base parameters:

$$\tau = Y(q,\dot{q},\ddot{q})p, \quad (2)$$

where

$$p_{78\times1} = [p_1,\cdots,p_6]^T,$$
$$p_i = \left[m_i, mx_i, my_i, mz_i, I_{ai}, I_{xxi}, I_{xyi}, I_{xzi}, I_{yyi}, I_{yzi}, I_{zzi}, f_{ci}, f_{vi}\right]^T. \quad (3)$$

Obviously, 6×13 unknown parameters constitute the dynamic parameters (3) of the robot. We generally reformulate (2) on account that not all parameters in p are independent. In this paper, the independent parameters are collected through QR decomposition.

We perform the robot under a random trajectory to obtain the multiple matrices $Y(\cdot)$ which form the matrix

$$[W]_{6M\times78} = \begin{bmatrix} Y(q(t_1),\dot{q}(t_1),\ddot{q}(t_1)) \\ \vdots \\ Y(q(t_M),\dot{q}(t_M),\ddot{q}(t_M)) \end{bmatrix}. \quad (4)$$

The number of matrices $Y(\cdot)$ or the sample number is $M = 10000$. The sample period is 0.1 s. We apply QR decomposition (5) to find the independent base dynamic parameters, with the orthonormal matrix $Q \in \mathbb{R}^{Mn\times nNs}$ and upper triangular matrix R:

$$W = QR. \quad (5)$$

Supposing that the main diagonal elements r_{ll} in the lth column of R are zero, the corresponding columns of R are collected in R_2, while the rest constitute R_1:

$$R = [R_1\ R_2]. \quad (6)$$

Similarly, the corresponding columns of W are collected consistent with R. According to (2), we have

$$W\pi = \left[[W_1]_{6M\times52}\ [W_2]_{6M\times26}\right]\begin{bmatrix}\pi_1\\ \pi_2\end{bmatrix} = T = \begin{bmatrix}\tau(1)\\ \vdots \\ \tau(M)\end{bmatrix}, \quad (7)$$

with independent π_1 and dependent π_2 which are collected from p in the same way as W.

According to (6) and (7), we have

$$T = W_1\pi^* \triangleq W_1(\pi_1 + k\pi_2), \quad (8)$$

where $k = R_2R_1^{-1}$.

Then, the unknown dynamic parameters are estimated by

$$[\pi^*]_{52\times1} = \left(W_1{}^T W_1\right)^{-1} W_1{}^T T. \quad (9)$$

The excitation reference trajectories for every joint are applied with the finite sum of 5 harmonic sine and cosine functions. The joint position, velocity, and acceleration of the ith joint are

$$q_i(t) = \sum_{l=1}^{5}\left[\frac{a_l}{w_f l}\sin\left(w_f lt\right) - \frac{b_l}{w_f l}\cos\left(w_f lt\right) + q_{i0}\right],$$
$$\dot{q}_i(t) = \sum_{l=1}^{5}\left[a_l\cos\left(w_f lt\right) + b_l\sin\left(w_f lt\right)\right],$$
$$\ddot{q}_i(t) = w_f\sum_{l=1}^{5}\left[a_l l\cos\left(w_f lt\right) + b_l l\sin\left(w_f lt\right)\right], \quad (10)$$

where the fundamental frequency is $w_f = 0.05$ and q_{i0} is the offset of the joint position of the trajectory. The parameters a_l and b_l are determined by trial and error or the following optimization process. The excitation trajectory $q^*(t)$ is determined by the optimization issue of (11) which is directly equivalent to optimizing the condition number for less estimation error while having less complexity and

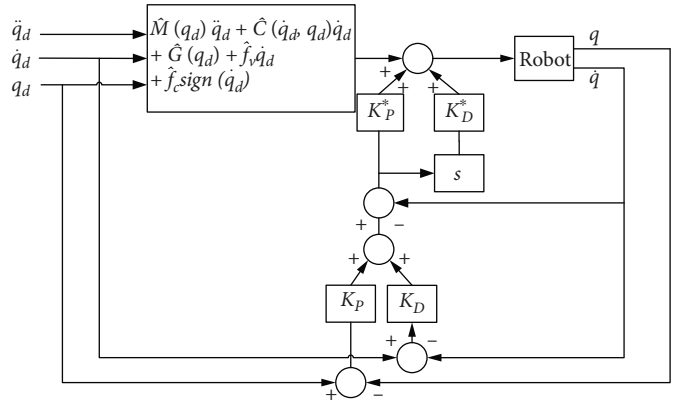


Figure 2: Control scheme of the robot.

Table 1: Controller parameters of each joint.

Joint	K_P ; K_D	r_i
1	962260; 80000	1000
2	886260; 209500	100
3	209500; 69000	100
4	1852500; 42000	1000
5	489500; 26700	1000
6	87000; 7500	7000

calculation time:

$$q^*(t) = \arg\min(W_1) \tag{11}$$

subject to

$$-q_{\max\, i} \le q_i(t) = \sum_{l=1}^{5} \frac{a_l}{w_f l} \sin(w_f l t) - \frac{b_l}{w_f l} \cos(w_f l t) \le q_{\max\, i}, \tag{12}$$

$$-v_{\max\, i} \le \dot{q}_i(t) = \sum_{l=1}^{5} a_l \cos(w_f l t) + b_l \sin(w_f l t) \le v_{\max\, i}, \tag{13}$$

$$-a_{\max\, i} \le \ddot{q}_i(t) = w_f \sum_{l=1}^{5} a_l l \cos(w_f l t) + b_l l \sin(w_f l t) \le a_{\max\, i}, \tag{14}$$

$$q_i(t_0) = q_i(0) = \sum_{i=1}^{5} \frac{b_l}{w_f l} + q_{i0} = 0, \tag{15}$$

$$\dot{q}_i(t_0) = \dot{q}_i(0) = \sum_{i=1}^{5} a_l = 0, \tag{16}$$

$$\ddot{q}_i(t_0) = \ddot{q}_i(0) = \sum_{i=1}^{5} w_f l b_l = 0, \tag{17}$$

where $q_{\max} = [q_{\max\,1}, \cdots, q_{\max\,6}]^T = [153, 24, 40, 153, 90, 180]^T/180\pi$, $v_{\max} = [v_{\max\,1}, \cdots, v_{\max\,6}]^T = [296, 240, 310, 355, 267, 500]^T/180\pi$, and $a_{\max} = [a_{\max\,1}, \cdots, a_{\max\,6}]^T = [1800, 1500, 1800, 1800, 1300, 3000]^T/180\pi$ are the constraint vectors for position, velocity, and acceleration, respectively.

It is possible to use any optimization method to solve the above optimization problem. In this paper, the parameters a_l and b_l are solved by the interior-point method. The maximum number of iterations is 20000. The initial values of a_l and b_l are chosen as random numbers within $[-0.5, 0.5]$.

The dynamic model is often not known accurately. For example, the structure of the friction model is generally difficult to know, let alone the parameter values. Furthermore, it is unrealistic to have precise parameter values in the model at all times through applying the least square method [51, 52] in one time since the dynamic parameters always change as the robot moves. Therefore, the following controller is inevitably designed in accordance with the inaccurate dynamic parameters.

3. Controller Design

The robotic dynamic model has the following properties [23, 27]:

(P1)The matrix $M(q)$ is positive definite symmetric and satisfies $\underline{M}\|x\|^2 \le x^T M(q) x \le \bar{M}\|x\|^2$ for positive constants $\bar{M}$, $\underline{M} > 0$

(P2)$\dot{M}(q) - 2C(q, \dot{q})$ is skew-symmetric

(P3)The matrix $C(q, \dot{q})$ satisfies $\|C(q, \dot{q})\| \le \mu_c$ for positive constant $\mu_c > 0$

(P4) The vector $G(q)$ satisfies $\|G(q)\| \le \mu_g$ for positive constant $\mu_g > 0$

From (9), we can obtain $\hat{M}(q)$, $\hat{C}(q, \dot{q})$, $\hat{G}(q)$, $\hat{f}_c$, and $\hat{f}_v$ which represent the estimation of $M(q)$, $C(q, \dot{q})$, $G(q)$, f_c, and f_v, respectively. Define $\tilde{M}(q) = \hat{M}(q_d) - M(q)$, $\tilde{C}(q, \dot{q}) = \hat{C}(q_d, \dot{q}_d) - C(q, \dot{q})$, $\tilde{G}(q) = G(q_d) - G(q)$, and the nonnegative definite diagonal gain matrices K_P, K_D, and $\Upsilon = \text{diag}(r_1, \cdots, r_6)$; $q_d = [q_1^*, \cdots, q_6^*]^T$ represents the desired trajectory for the robot joints.

Table 2: Estimated dynamic parameters of the robot.

Parameters	Joint 1	Joint 2	Joint 3	Joint 4	Joint 5	Joint 6
m_i (kg)	10	10	10	10	10	10
mx_i (kg·m)	0.000000	1467.621893	120.973665	0.877570	-5.497015	-3.474394
my_i (kg·m)	0.000000	-485.168899	512.922191	-4.105343	49.201786	9.947225
mz_i (kg·m)	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
I_{xxi} (kg·m^2)	0.000000	-3105.984543	-429.633816	139.535980	-35.190459	-5.898670
I_{xyi} (kg·m^2)	0.000000	-2279.068182	175.632270	-37.783294	-5.612843	4.712513
I_{xzi} (kg·m^2)	0.000000	-168.579890	137.144088	1.357674	-25.300536	-3.631684
I_{yyi} (kg·m^2)	0.000000	-2153.914022	-14.634628	-0.000770	-0.030217	-0.012071
I_{yzi} (kg·m^2)	0.000000	-467.555176	158.191065	-34.563860	-15.506560	-19.074189
I_{zzi} (kg·m^2)	1355.845148	-4625.319065	315.581754	123.918230	39.386756	-10.111644
I_{ai} (kg·m^2)	0.000000	0.000000	-1198.010476	81.708986	-14.478101	-27.911115
f_{ci} (Nm·s/rad)	5469.258178	4450.476442	8730.326694	2888.554652	1857.202169	-2029.910574
f_{vi} (Nm·s/rad)	7102.364470	31212.701099	16441.785926	3909.460084	1322.947888	1322.947888

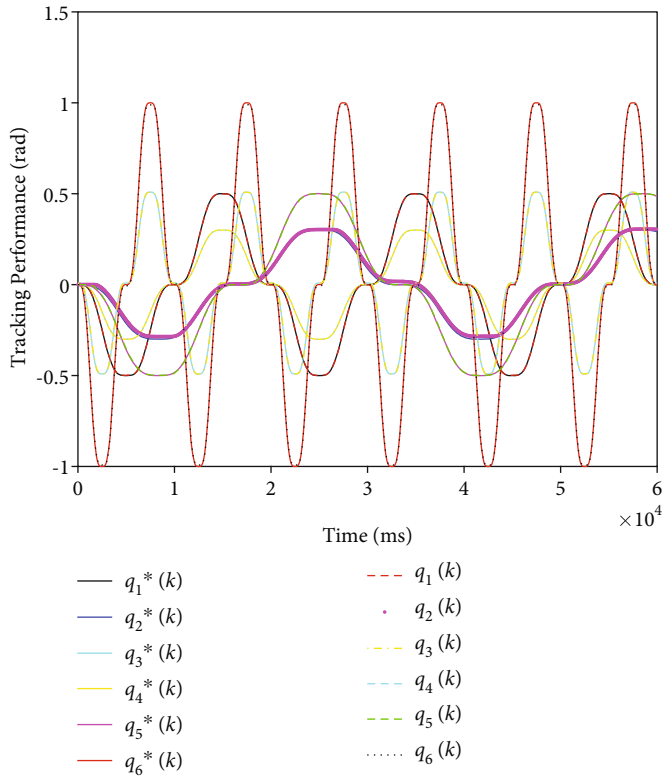


Figure 3: Tracking performance of each robotic joint controlled by variable structure control.

Define $e = q_d - q$. We choose the Lyapunov function

$$V = \frac{1}{2}\dot{e}^T M(q)\dot{e} + \frac{1}{2}e^T K_P e, \tag{18}$$

which is continuous and nonnegative. According to the properties of the robotic dynamic model, we can obtain (19) by differentiating (18) as follows:

$$\begin{aligned}\dot{V} &= \dot{e}^T M(q)\ddot{e} + \frac{1}{2}\dot{e}^T \dot{M}(q)\dot{e} + \dot{e}^T K_P e \\ &= \dot{e}^T (M(q)\ddot{e} + C(q,\dot{q})\dot{e} + K_P e) \\ &= \dot{e}^T (M(q)\ddot{q}_d + C(q,\dot{q})\dot{q}_d + K_P e \\ &\quad + G(q) + f_c \operatorname{sign}(\dot{q}) + f_v \dot{q} - \tau).\end{aligned} \tag{19}$$

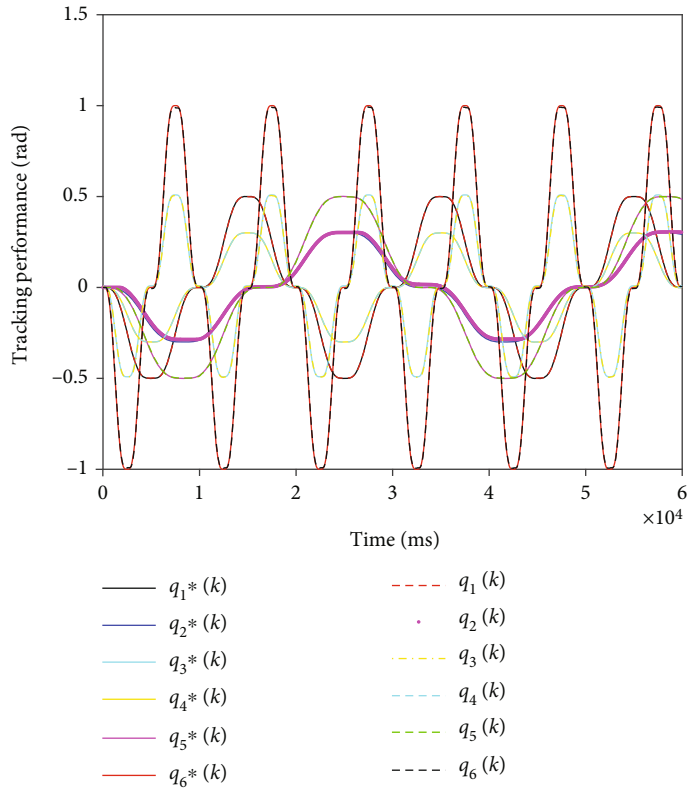


Figure 4: Tracking performance of each robotic joint controlled by PD plus.

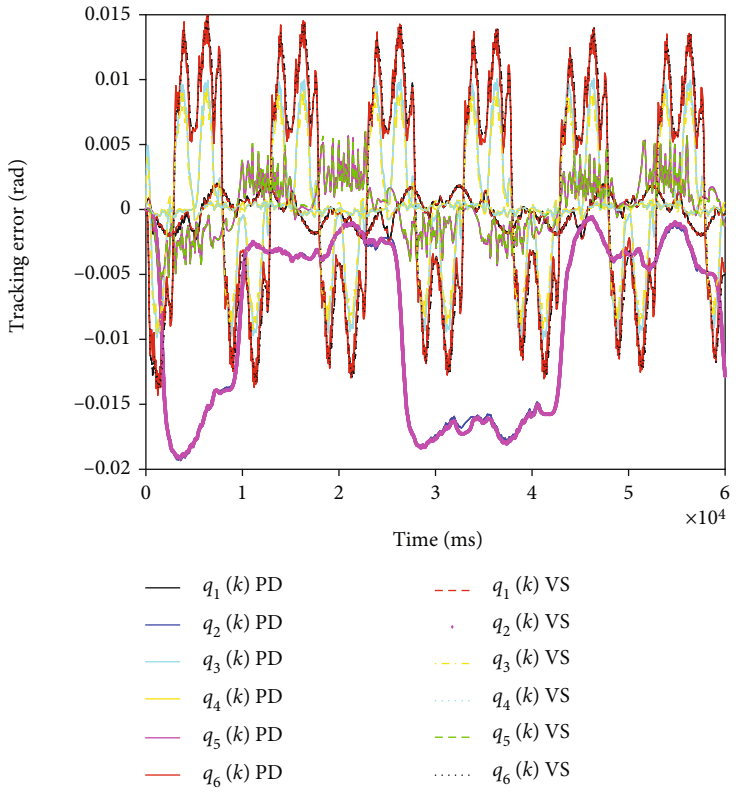


Figure 5: Tracking error comparison between variable structure control and PD plus.

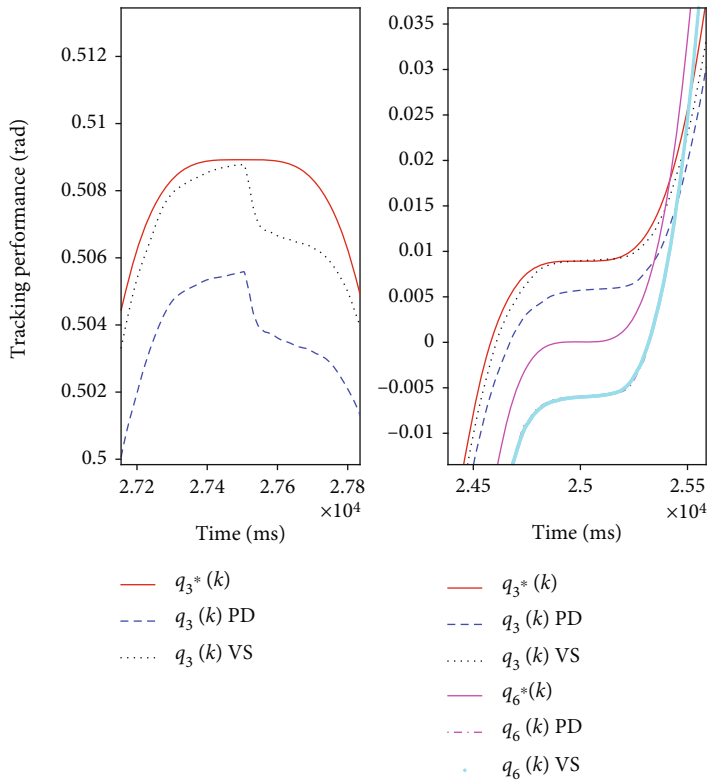


Figure 6: Tracking performance of the 3rd joint and the 6th joint of the robot controlled by variable structure control and PD plus.

Table 3: Controller parameters of each joint.

Joint	K_P; K_D	K_P^*; K_D^*
1	800; 0.1	800; 70
2	650; 0.1	6000; 15
3	1000; 0.1	2500; 60
4	650; 0.1	250; 40
5	800; 0	600; 15
6	650; 0.1	600; 15

Then, we design the controller

$$\tau = \tau_{\text{dynamic}} + \tau_{\text{PD}}, \tag{20}$$

$$\begin{aligned}\tau_{\text{dynamic}} = {} & \widehat{M}(q_d)\ddot{q}_d + \widehat{C}(q_d, \dot{q}_d)\dot{q}_d + \widehat{G}(q_d) \\ & + \widehat{f}_c \text{ sign } (\dot{q}_d) + \widehat{f}_v \dot{q}_d,\end{aligned} \tag{21}$$

$$\tau_{\text{PD}} = K_P e + K_D \dot{e} + \Upsilon \text{ sign } (\dot{e}) \tag{22}$$

to have

$$\begin{aligned}\dot{V} &= -\dot{e}^T (K_D \dot{e} + \bar{\sigma} + \Upsilon \text{ sign } (\dot{e})) \\ &= -\dot{e}^T K_D \dot{e} - \dot{e}^T \bar{\sigma} - \dot{e}^T \Upsilon \text{ sign } (\dot{e}),\end{aligned} \tag{23}$$

where

$$\begin{aligned}\bar{\sigma} &= \tilde{M}(q)\ddot{q}_d + \tilde{C}(q, \dot{q})\dot{q}_d + \tilde{G}(q) + \tilde{F}_f, \\ \tilde{F}_f &= \widehat{f}_c \text{ sign } (\dot{q}_d) - f_c \text{ sign } (\dot{q}) + \widehat{f}_v \dot{q}_d - f_v \dot{q}.\end{aligned} \tag{24}$$

Herein, we choose

$$r_i > \|\bar{\sigma}\|_{\max}, \tag{25}$$

to have

$$\dot{V} < 0. \tag{26}$$

For the case $n = 6$, the iterative Newton-Euler scheme is approximately 100 times more efficient than the Lagrangian approach. Consequently, we normally realize (21) by the iterative Newton-Euler dynamics shown as follows.

(a) Outward Iterations $i : 0 \longrightarrow 5$

The joint $i + 1$ rotational velocity is

$$w_{i+1} = R_i^{i+1} w_i + \dot{q}_{i+1}^* \widehat{Z}_{i+1}, \tag{27}$$

where $\widehat{Z}_{i+1}$ represents the axis pointing along the $i + 1$th joint axis and R_i^{i+1} is the rotation matrix.

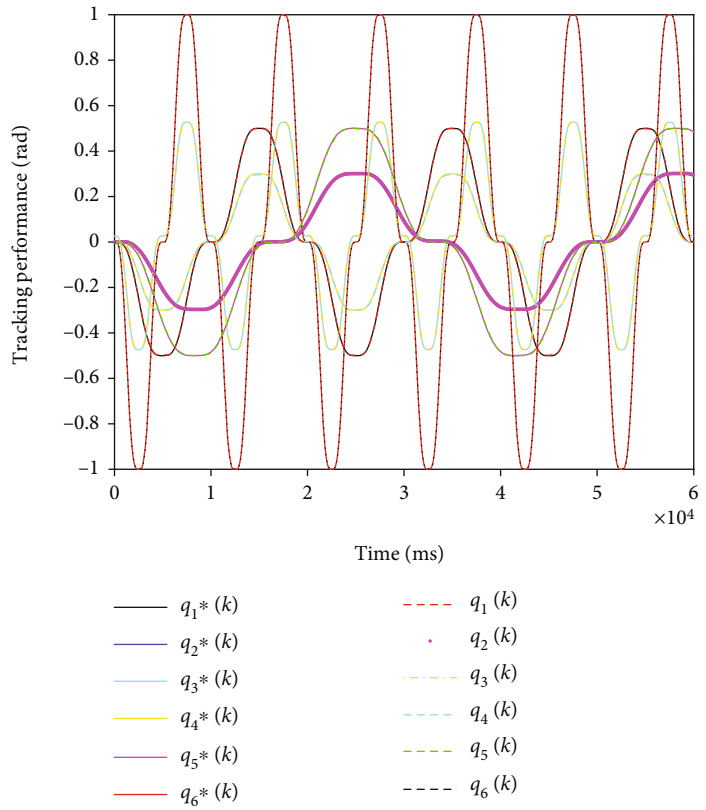


Figure 7: Tracking performance of each robotic joint controlled by cascaded PD plus.

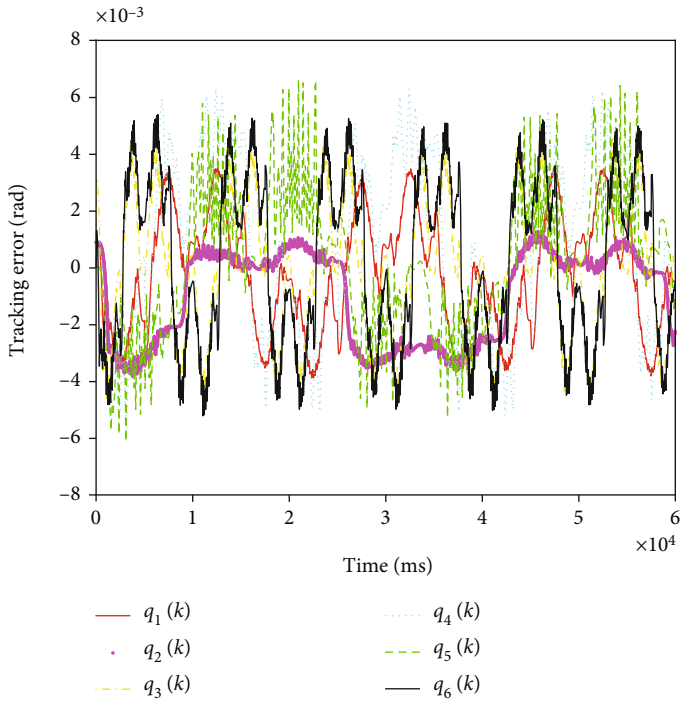


Figure 8: Tracking error of each robotic joint controlled by cascaded PD plus.

The angular acceleration from one link to the next is described by

$$\dot{w}_{i+1} = R_i^{i+1}\dot{w}_i + R_i^{i+1}w_i \times \dot{q}_{i+1}\widehat{Z}_{i+1} + \ddot{q}_{i+1}^{*}\widehat{Z}_{i+1}. \quad (28)$$

Noticeably, (27) and (28) are different from the iterations in the textbook [22], since the velocity in (27) and acceleration in (28) of each joint are the desired velocity and acceleration, respectively.

Then, we obtain the linear acceleration of each link-frame origin through

$$\dot{v}_{i+1} = R_i^{i+1}\left(\dot{w}_i \times P_i^{i+1} + w_i \times \left(w_i \times P_i^{i+1}\right) + \dot{v}_i\right) \quad (29)$$

and obtain the linear acceleration of the center of mass of each link

$$\dot{v}_{C_{i+1}}^{i+1} = \dot{w}_{i+1} \times P_{C_{i+1}}^{i+1} + w_{i+1} \times \left(w_{i+1} \times P_{C_{i+1}}^{i+1}\right) + \dot{v}_{i+1}, \quad (30)$$

where $\{C_i\}$ represents a frame attached to each link and has its origin located at the center of mass of the link with the same orientation as the link frame $\{i\}$; $P_{C_{i+1}}^{i+1}$ locates the position of $\{C_{i+1}\}$ relative to the $i+1$th joint.

Then, we can obtain the inertial force and torque acting at the center of the mass of each link (31) and (32) through Newton-Euler equations as follows:

$$F_{i+1} = m_{i+1}\dot{v}_{i+1}^{C_{i+1}}, \quad (31)$$

$$N_{i+1} = I_{i+1}^{C_{i+1}}\dot{w}_{i+1} + w_{i+1} \times I_{i+1}^{C_{i+1}}w_{i+1}, \quad (32)$$

where $I_{i+1}^{C_{i+1}}$ determines the mass moments of inertia relative to the center of mass in the $i+1$th link.

(b) Inward Iterations $i : 6 \longrightarrow 1$

The force exerted on link i by link $i-1$ is

$$f_i = R_{i+1}^{i}f_{i+1} + F_i. \quad (33)$$

12The torque exerted on link i by link $i-1$ is

$$n_i = N_i + R_{i+1}^{i}n_{i+1} + P_{C_i}^{i} \times F_i + P_{i+1}^{i} \times R_{i+1}^{i}f_{i+1}. \quad (34)$$

Finally, we obtain the required joint torques

$$\tau_i = n_i^T\widehat{Z}_i, \quad (35)$$

which constitute $\tau_{\text{dynamic}} = [\tau_1,\cdots,\tau_6]^T$ in (21).

Controller (20) guarantees the stability of the system in theory. Furthermore, if we let $r_i = 0, i = 1,\cdots,6$, the controller is the feedforward nonlinear control shown in Figure 10.6 in [22]. Furthermore, if we correct τ_{PD} (22) into the following cascaded PD controller

$$\tau_{\text{PD}} = K_P^{*}[(K_Pe + K_D\dot{e}) - \dot{q}] + K_D^{*}\frac{d}{dt}[(K_Pe + K_D\dot{e}) - \dot{q}], \quad (36)$$

which uses PD in both the outer loop and inner loop, and then, the cascaded PD plus dynamic feedforward control is shown in Figure 2.

The position control loop (outer loop) is to maintain the positional trajectory tracking. The error in the position control loop provides the desired velocity for the velocity loop (inner loop). We apply a PD controller $(K_Pe + K_D\dot{e})$ on the basis of the error between the desired and actual positions to generate the desired speed of the motor so that the actual position tracks the desired position closely. The velocity loop uses the PD controller plus dynamic feedforward compensation to generate the desired torque for the robot joint (motor) so that the actual velocity tracks the desired velocity closely. We normally adjust the gains of the joint controller: K_P, K_D, K_P^{*}, and K_D^{*} to change the behaviors of the robot subject to variation in the dynamic model caused by various poses and payloads, variation in friction with temperature and time, and some disturbance torques owing to complicated velocity and acceleration coupling.

By removing the gravity compensation item of the controller (20), we can obtain the practical controller for the on-orbit space robot:

$$\tau = \widehat{M}(q_d)\ddot{q}_d + \widehat{C}(q_d,\dot{q}_d)\dot{q}_d + \widehat{f}_c\ \text{sign}\ (\dot{q}_d) + \widehat{f}_v\dot{q}_d + \tau_{\text{PD}}. \quad (37)$$

4. Experiment

The proposed variable structure method is applied to the robot, and the controller parameters of each joint are listed in Table 1. The estimated dynamic parameters of the robot are shown in Table 2. The tracking performance of each robotic joint is shown in Figure 3.

When we let $r_i = 0\,(i = 0,\cdots,6)$, the proposed variable structure controller degenerates into the PD plus. The tracking performance of each robotic joint controlled by PD plus is shown in Figure 4. The tracking error comparison between VS control and PD plus is shown in Figure 5.

Figures 3 and 4 show that both outputs of the robot controlled by the proposed VS control and PD plus are able to track the desired trajectories. Figure 5 shows that the tracking error curves of the robot controlled by the proposed VS method and PD plus are close. Hence, the drawing of partial enlargement of the output of the 3rd joint and the output of the 6th joint of the robot controlled by variable structure control and PD plus are shown in Figure 6 to make a further comparison.

Figure 6 shows that the tracking performance of the 3rd joint of the robot controlled by the proposed VS control is better than that controlled by PD plus for less tracking error from 27200 ms to 27800 ms and from 24500 ms to 25500 ms, while there is no obvious difference between the two methods acting at the 6th joint. These findings are possibly because the proposed VS control combines the PD plus dynamic feedforward compensation and the Bang-Bang control together, and the Bang-Bang control item Υ sign $(\dot{e}$) can be used to suppress all matching uncertainties and unpredictable system dynamics. In this experiment, the

Bang-Bang control item can provide timely compensation for more tracking errors caused by the modeling uncertainty and consequently improves both the tracking accuracy and tracking speed of the robot.

Then, we apply cascaded PD plus dynamic feedforward compensation control to the robotic manipulator for comparison with the aforementioned two methods. The cascaded PD controller parameters of each joint are listed in Table 3.

The tracking performance of each robotic joint is shown in Figure 7. The tracking error of each joint controlled by cascaded PD plus is shown in Figure 8.

With less effort spent on the adjustment of controller parameters in the experiment, the cascaded PD plus easily guarantees the tracking error of each joint of the robot within $\pm 8 \times 10^{-3}$ rad which is shown in Figure 8. The reason is summarized as follows: As demanded by the outer loop, the inner loop is adopted to generate the desired torque for the robot joint so that the actual velocity tracks the desired velocity closely. The outer loop is used to determine the velocity of the joint that minimizes position error. By comparison, we have tested the proposed VS control and PD plus by trial and error in our experiment to guarantee the tracking error of each joint of the robot within $\pm 2 \times 10^{-2}$ rad which is shown in Figure 5. By comparing Figures 8 and 5, we find that the cascaded PD plus dynamic feedforward compensation control more easily guarantees the high-precision tracking performance than the proposed VS control or the PD plus in the experiment, under the roughly built robotic dynamics. Actually, the PD plus typically represents a class of single-loop feedback control. Furthermore, this also implies that many SMC or other variable structure controls based on a single-loop feedback strategy may not more easily achieve better performance than the cascaded PD method. Based on this, we recommend the cascaded PID plus feedforward dynamic model for the calculation of the desired torque in the 6-DOF serial robot rather than the single-loop feedback strategy. The controller structure may be more important than the adjustment of the controller parameters in the robotic system.

The controller design or the choice of the controller from different candidates for the space robot remains prudent work. One purpose of the above analyses and experimental results is to provide a source of prior knowledge and practice experience for the controller design and the test of the space robot. Furthermore, by removing gravity compensation, we can design the controller of the space station robot in the same way as the above controllers of the industrial robot.

5. Conclusions

To suggest simple yet effective controllers for the large and heavy serial multijoint space station robot, we made comparisons among our proposed variable structure method, PD plus and the cascaded PD plus on the industrial robot in this paper. First, the dynamic parameters of the robot are identified by the least square method. Second, to guarantee the stability of the system under modeling uncertainty in theory, a variable structure controller is designed based on the Lyapunov direct method. It can be separated into two parts: the PD plus and the sign function. Third, experiments show that the cascaded PD plus dynamic feedforward compensation control more easily guarantees the high-precision tracking performance than the proposed variable structure control or the PD plus, under the roughly built robotic dynamics.

Data Availability

The experimental data used to support the findings in this paper can be available from the first author and corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grant Number 62103406 and is also supported by the Natural Science Foundation of Liaoning Province under Grant Number 2021-MS-032.

References

[1] X. Zhang, J. Liu, J. Feng, Y. Liu, and Z. Ju, "Effective capture of nongraspable objects for space robots using geometric cage pairs," *IEEE/ASME Transactions on Mechatronics*, vol. 25, no. 1, pp. 95–107, 2020.

[2] X. Zhang and J. Liu, "Effective motion planning strategy for space robot capturing targets under consideration of the berth position," *Acta Astronautica*, vol. 148, pp. 403–416, 2018.

[3] P. Huang, L. Chen, B. Zhang, Z. Meng, and Z. Liu, "Autonomous rendezvous and docking with nonfull field of view for tethered space robot," *International Journal of Aerospace Engineering*, vol. 2017, Article ID 3162349, 11 pages, 2017.

[4] Y. Wang, X. Liang, K. Gong, and Y. Liao, "Kinematical research of free-floating space-robot system at position level based on screw theory," *International Journal of Aerospace Engineering*, vol. 2019, Article ID 6857106, 13 pages, 2019.

[5] S. Sun, C. Wei, Z. Huang et al., "Adaptive control of space robot despinning tumbling target using flexible brushes," *International Journal of Aerospace Engineering*, vol. 2021, Article ID 6196556, 9 pages, 2021.

[6] M. Sabatini, P. Gasbarri, and G. B. Palmerini, "Coordinated control of a space manipulator tested by means of an air bearing free floating platform," *Acta Astronautica*, vol. 139, pp. 296–305, 2017.

[7] Z. Chen, Z. Luo, Y. Wu, W. Xue, and W. Li, "Research on high-precision attitude control of joint actuator of three-axis air-bearing test bed," *Journal of Control Science and Engineering*, vol. 2021, Article ID 5582541, 11 pages, 2021.

[8] Y. Watanabe and Y. Nakamura, "Experiments of a space robot in the free-fall environment," *Artificial Intelligence, Robotics and Automation in Space*, vol. 440, article 601, 1999.

[9] C. Menon, A. Aboudan, S. Cocuzza, A. Bulgarelli, and F. Angrilli, "Free-flying robot tested on parabolic flights:

kinematic control," *Journal of Guidance, Control, and Dynamics*, vol. 28, no. 4, pp. 623–630, 2005.

[10] C. Menon, S. Busolo, S. Cocuzza et al., "Issues and solutions for testing free-flying robots," *Acta Astronautica*, vol. 60, no. 12, pp. 957–965, 2007.

[11] Y. Yuan, P. Zhang, Z. Wang, L. Guo, and H. Yang, "Active disturbance rejection control for the ranger neutral buoyancy vehicle: a delta operator approach," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 12, pp. 9410–9420, 2017.

[12] I. Rekleitis, E. Martin, G. Rouleau, R. L'Archevêque, K. Parsa, and E. Dupuis, "Autonomous capture of a tumbling satellite," *Journal of Field Robotics*, vol. 24, no. 4, pp. 275–296, 2007.

[13] F. Aghili, "A prediction and motion-planning scheme for visually guided robotic capturing of free-floating tumbling objects with uncertain dynamics," *IEEE Transactions on Robotics*, vol. 28, no. 3, pp. 634–649, 2012.

[14] H. Liu, B. Liang, W. Xu, D. Zhang, and X. Wang, "A ground experiment system of a free-floating robot for fine manipulation," *International Journal of Advanced Robotic Systems*, vol. 9, no. 5, p. 183, 2012.

[15] R. Lampariello, H. Mishra, N. Oumer, P. Schmidt, M. de Stefano, and A. Albu-Schaffer, "Tracking control for the grasping of a tumbling satellite with a free-floating robot," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3638–3645, 2018.

[16] H. Fujii, K. Uchiyama, H. Yoneoka, and T. Maruyama, "Ground-based simulation of space manipulators using test bed with suspension system," *Journal of Guidance, Control, and Dynamics*, vol. 19, no. 5, pp. 985–991, 1996.

[17] H. Shi, S. Ma, M. Huo, and N. Qi, "Design and control of a position servo system in the zero gravity simulation of space manipulators," in *2015 International Conference on Fluid Power and Mechatronics (FPM)*, pp. 501–504, Harbin, China, August 2015.

[18] H. B. Brown and J. M. Dolan, "A novel gravity compensation system for space robots," 1994.

[19] Y. He, F. Zhang, M. Yang, and Z. J. J. R. Xu, "Design of tracking suspension gravity compensation system for satellite antenna deployable manipulator," *Robot*, vol. 40, no. 3, pp. 377–384, 2018.

[20] Z. Xu, X. Bai, J. Wang, M. Liu, and X. Zhang, "The control method of force loading of robot on load transfer mechanism of space station," in *2019 IEEE 15th International Conference on Control and Automation (ICCA)*, pp. 518–523, Edinburgh, UK, July 2018.

[21] Y. He, F. L. Zhang, M. Y. Liu, and Z. Xu, "Controller design and accuracy evaluations for the space station sun directional device test system," *Journal of Astronautics*, vol. 39, no. 6, pp. 674–682, 2018.

[22] J. J. Craig, *Introduction to Robotics: Mechanics and Control*, Prentice Hall, Boston, USA, 2004.

[23] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *Robot Modeling and Control*, Wiley, NJ, USA, 2005.

[24] R. P. Paul, "Robot manipulators: mathematics, programming, and control: the computer control of robot manipulators," 1981.

[25] P. I. Corke and O. Khatib, *Robotics, Vision and Control: Fundamental Algorithms in MATLAB*, Springer, Berlin, 2017.

[26] B. Siciliano, O. Khatib, and T. Kröger, *Springer Handbook of Robotics*, Springer, Berlin, 2007.

[27] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: Modelling, Planning and Control*, Springer Science & Business, Berlin, 2010.

[28] S. Moberg and S. Hanssen, "A DAE approach to feedforward control of flexible manipulators," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pp. 3439–3444, Rome, Italy, April 2007.

[29] V. Santibañez and R. Kelly, "PD control with feedforward compensation for robot manipulators: analysis and experimentation," *Robotica*, vol. 19, no. 1, pp. 11–19, 2001.

[30] V. Parra-Vega, S. Arimoto, Y. H. Liu, G. Hirzinger, and P. Akella, "Dynamic sliding PID control for tracking of robot manipulators: theory and experiments," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 6, pp. 967–976, 2003.

[31] P. R. Ouyang, J. Acob, and V. Pano, "PD with sliding mode control for trajectory tracking of robotic system," *Robotics and Computer-Integrated Manufacturing*, vol. 30, no. 2, pp. 189–200, 2014.

[32] T. Kara and A. H. Mary, "Adaptive PD-SMC for nonlinear robotic manipulator tracking control," *Studies in Informatics and Control*, vol. 26, no. 1, pp. 49–58, 2017.

[33] S. I. Han and J. Lee, "Finite-time sliding surface constrained control for a robot manipulator with an unknown deadzone and disturbance," *ISA Transactions*, vol. 65, pp. 307–318, 2016.

[34] N. M. H. Norsahperi and K. A. Danapalasingam, "An improved optimal integral sliding mode control for uncertain robotic manipulators with reduced tracking error, chattering, and energy consumption," *Mechanical Systems and Signal Processing*, vol. 142, article 106747, 2020.

[35] X. Zhang, J. G. Liu, Q. Gao, and Z. Ju, "Adaptive robust decoupling control of multi-arm space robots using time-delay estimation technique," *Nonlinear Dynamics*, vol. 100, no. 3, pp. 2449–2467, 2020.

[36] X. Zhang, J. Liu, Y. Tong, Y. Liu, and Z. Ju, "Attitude decoupling control of semifloating space robots using time-delay estimation and supertwisting control," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 57, no. 6, pp. 4280–4295, 2021.

[37] C. Zheng, Y. Su, and P. Mercorelli, "A simple nonlinear PD control for faster and high-precision positioning of servomechanisms with actuator saturation," *Mechanical Systems and Signal Processing*, vol. 121, pp. 215–226, 2019.

[38] Y. Su, C. Zheng, and P. Mercorelli, "Robust approximate fixed-time tracking control for uncertain robot manipulators," *Mechanical Systems and Signal Processing*, vol. 135, article 106379, 2020.

[39] Y. Wang, M. Chen, and Y. Song, "Robust fixed-time inverse dynamic control for uncertain robot manipulator system," *Complexity*, vol. 2021, Article ID 6664750, 12 pages, 2021.

[40] K. Huang, Y. Xian, S. Zhen, and H. Sun, "Robust control design for a planar humanoid robot arm with high strength composite gear and experimental validation," *Mechanical Systems and Signal Processing*, vol. 155, article 107442, 2021.

[41] W. Yu and J. Rosen, "Neural PID control of robot manipulators with application to an upper limb exoskeleton," *IEEE Transactions on cybernetics*, vol. 43, no. 2, pp. 673–684, 2013.

[42] W. He, B. Huang, Y. Dong, Z. Li, and C. Y. Su, "Adaptive neural network control for robotic manipulators with unknown deadzone," *IEEE Transactions on Cybernetics*, vol. 48, no. 9, pp. 2670–2682, 2018.

[43] E. O. Freire, F. G. Rossomando, and C. M. Soria, "Self-tuning of a neuro-adaptive PID controller for a SCARA robot based on neural network," *IEEE Latin America Transactions*, vol. 16, no. 5, pp. 1364–1374, 2018.

[44] D. C. Gandolfo, F. G. Rossomando, C. M. Soria, and R. O. Carelli, "Adaptive neural compensator for robotic systems control," *IEEE Latin America Transactions*, vol. 17, no. 4, pp. 670–676, 2019.

[45] L. A. Soriano, E. Zamora, J. M. Vazquez-Nicolas, G. Hernández, J. A. Barraza Madrigal, and D. Balderas, "PD control compensation based on a cascade neural network applied to a robot manipulator," *Frontiers in Neurorobotics*, vol. 14, 2020.

[46] F. G. Rossomando, E. Serrano, C. M. Soria, and G. Scaglia, "Neural dynamics variations observer designed for robot manipulator control using a novel saturated control technique," *Mathematical Problems in Engineering*, vol. 2020, Article ID 3240210, 14 pages, 2020.

[47] J. Armendariz, V. Parra-Vega, R. García-Rodríguez, and S. Rosales, "Neuro-fuzzy self-tuning of PID control for semiglobal exponential tracking of robot arms," *Applied Soft Computing*, vol. 25, pp. 139–148, 2014.

[48] J. L. Meza, V. Santibáñez, R. Soto, and M. A. Llama, "Fuzzy self-tuning PID semiglobal regulator for robot manipulators," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 6, pp. 2709–2717, 2012.

[49] M. Bagheri, I. Karafyllis, P. Naseradinmousavi, and M. Krstic, "Adaptive control of a two-link robot using batch least-square identifier," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 1, pp. 86–93, 2021.

[50] H. S. Jayakody, L. Shi, J. Katupitiya, and N. Kinkaid, "Robust adaptive coordination controller for a spacecraft equipped with a robotic manipulator," *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 12, pp. 2699–2711, 2016.

[51] J. Jin and N. Gans, "Parameter identification for industrial robots with a fast and robust trajectory design approach," *Robotics and Computer-Integrated Manufacturing*, vol. 31, pp. 21–29, 2015.

[52] M. Neubauer, H. Gattringer, and H. Bremer, "A persistent method for parameter identification of a seven-axes manipulator," *Robotica*, vol. 33, no. 5, pp. 1099–1112, 2015.

Hindawi
International Journal of Aerospace Engineering
Volume 2022, Article ID 3641051, 13 pages
https://doi.org/10.1155/2022/3641051

Research Article

Vibration Behaviour Analysis and Vibration Suppression Studies of the Space Robot

Tian Yong,[1] **Yue Xiang**,[2,3] **Feng Yan**,[2] **and Jiang Binzhang**[2]

[1]*College of Mechanical Electrical and Vehicle Engineering, Weifang University, Weifang, China*
[2]*College of Engineering, Shenyang Agricultural University, Shenyang 110866, China*
[3]*State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China*

Correspondence should be addressed to Yue Xiang; yuexiang@syau.edu.cn

Received 30 April 2022; Revised 12 June 2022; Accepted 21 August 2022; Published 9 September 2022

Academic Editor: Qing Gao

Aiming at the problem that the vibration of space robot will reduce the dynamic accuracy of robot, a method of vibration suppression of space robot through trajectory planning is proposed. First, the joint was simplified to a linear torsion spring, and the flexible rod was modelled using the finite element method, resulting in a flexible motion model of the robot. Then, a rigid-soft coupled dynamic model that combined the flexible motion model with the Lagrange method was established. Using the dynamic model, the factors influencing the vibration behaviour of space robots were analysed. Finally, the space robot was subjected to vibration suppression through trajectory planning. As verified by experimental analysis, different trajectories and loads affect the excitation force and inherent characteristics of the robot. In the design process, it is necessary to consider the relationship between the excitation force and natural frequency. The trajectory planning method has apparent effects on vibration suppression. The vibration amplitude was significantly reduced, which can improve the positioning accuracy and work efficiency of the robot.

1. Introduction

Space robots can be used to realise space control automation and intelligence, which can replace or assist astronauts in completing on-orbit assembly, on-orbit maintenance, fuel refuelling, orbit cleaning, spacecraft inspection, and other tasks, improve the efficiency of space exploration activities, and have broad application prospects. Space operation tasks, represented by on-orbit services, are becoming increasingly complex. Studying the precise and dexterous control technology of space robotic arms will be the key to human space exploration [1–5].

As an essential category of space robots, humanoid space robotic arms are mainly used to complete complex assembly, maintenance, and high-precision tasks. Compared with conventional robotic arms, space flexible robotic arms have the following characteristics: (1) the space robotic arm is made of lightweight materials, and the structure is a slender member, which is lightweight and fast in response; (2) the space manipulator develops in the direction of joint lightweight and integration, and adopts harmonic transmission, making the entire system a complex rigid-soft coupling nonlinear system [6]. Therefore, it is challenging to model the dynamics of spatially flexible robotic arms. Hu et al. presented an optimal configuration selection method for calibration of robots which is researched by the influencing factor separation method [7, 8]. When performing the grasping task, the space robot arm will inevitably generate vibrations owing to its flexibility, which will affect the accuracy of the operation. If the vibration is considerable, the entire system may fail. Therefore, to ensure that the space flexible robotic arm better serve space, an analysis of the vibration characteristics of the adjustable arm rod and the suppression of its vibration are necessary [9, 10].

There are three main methods for suppressing the low-frequency vibrations of flexible robots: passive control, active control, and trajectory planning. The effect of passive management is relatively simple; however, in practical applications, it can only be used for single linkages and components with relatively simple structures [11, 12]. The active control method

uses a controller design to monitor vibration and suppress vibration in real time [13]. Systematic dynamical equations were established using the conservation of linear momentum, conservation of angular momentum, theoretical modal methods, and Lagrange equations. For the fast subsystem, the speed difference feedback control and the linear secondary optimal control are designed to suppress the vibration of the flexible joint and flexible connecting rod, respectively [14]. The vibration suppression problem of flexible robots has been studied using optimal control methods. In these methods, the robot is suppressed by the dynamic model of the robot and state feedback equation [15].

The kinetic equations of the flexible robotic arm are decoupled using the singular perturbation method. Vibration is effectively suppressed by designing an observer to detect the vibration status of the system in real time, quickly predict the beat of the system, and control the robot system in real time through a feedback control system [16]. When the Kalman filter was used as the state trajectory quantity, the flex arm vibrated. A dynamic model of flexible manipulators considering hydrodynamic force was established by combining the Lagrange equation and Morison formula [17]. An adaptive fuzzy sliding mode control scheme to compensate for uncertain factors was designed to track the joint trajectory and suppress vibration. The rigid-soft coupling dynamics of a space robot system with flexible attachments were established. A vibration suppression method for autonomously capturing a target in the precollision stage of the base was proposed using a model prediction algorithm [18]. The finite-element method was used to establish a dynamic model of the system [19]. On this basis, vibration suppression experiments of six different cable-driven parallel robots were conducted using fuzzy PID control and active control methods. Active control methods place high demands on the accuracy of the dynamic models. Compared with operational management, trajectory planning has the advantages of low-accuracy requirements for dynamic models and relatively simple control. The method of trajectory planning is divided into two types: planning only at the kinematic level, such as planning the trajectory of the robot, taking the integral of the square of the acceleration as the optimization goal, reducing the impact, and reducing the vibration by smoothing the trajectory of the robot [20].

Using a five-time polynomial and the acceleration limit of each joint, a method of vibration suppression of a spatial flexible robotic arm based on trajectory planning that effectively suppresses the vibration of the robotic arm was proposed [21]. Planning at the kinematic level poorly affects the dampening vibrations and the other types of trajectory planning at the kinetic level. By establishing a flexible dynamic model of the robot, trajectory planning can be carried out based on spline curves and polynomial curves with the excitation force as the optimization goal; however, such a model is appropriate for single-degree-of-freedom or planar robots [22–25]. A vision-based noncontact vibration suppression method was proposed based on the vibration amplitude error prediction model, and an error estimation controller was designed so that the vibration could be controlled in real time [26]. However, this study only focused on vibration suppression, and the factors influencing the vibration behaviour were not revealed.

By analysing the influence of the motion law of triangles and trapezoids on the vibration of the system, it has been revealed that the residual vibration of the system is suppressed well when the deceleration time of the system is equal to the first-order vibration cycle of the system [27]. Zhao [28] responded to the dynamic response of a flexible robotic arm with a moving end effector in different motion states and analysed the effect of actuator movement on the robot vibration. Yu et al. [29, 30] proposed the design idea of robot path planning and the path parameter selection method from the perspective of the spectrum. These approaches effectively reduce the flexible vibration of rotational motion.

In this study, the factors influencing the vibration behaviour of space robots were analysed, and the vibration of space robots was suppressed using the analysis results. First, the mechanical structure of the space robot was diagnosed, the integrated joint was assumed to be equivalent to a linear torsion spring, the flexible connecting rod was modelled by the finite element method, and the rigid-soft coupling dynamic model of the robot was established by combining the Lagrange method. The vibration model of the robot was obtained based on the dynamic model, and the vibration model was used to analyse the influence of the vibration force and frequency generated by the trajectory on the vibration, as well as the influence of different trajectories and loads on the natural frequency. Finally, vibration suppression was carried out using the trajectory planning method.

2. Rigid-Flexible Coupling Dynamic Model of Space Robots

2.1. Basic Composition. To improve the flexibility of the space robot, it had seven DOFs. The main characteristics of the robot are that it is lightweight and has a large load-weight ratio. The robot is equipped with a flexible integrated joint and lightweight connecting rod to satisfy the aforementioned characteristics. Figure 1 shows the configuration of the space robot SHIR5. The system mainly includes a mechanical system, control system framework, and operating system. The robot body is composed of an integrated joint, lightweight connecting rod, control box, and an external instructor, as shown in Figure 2.

An integrated joint is an essential part of a robot that integrates servomotors, reducers, torque sensors, encoders, and actuators. The composition of the integrated joints is illustrated in Figure 3. A high-performance permanent-magnet torque motor and harmonic reducer with a high transmission ratio were adopted to achieve lightness and a large load-weight ratio. A position encoder was installed at the motor and output shaft ends to improve the positioning accuracy. A torque sensor was installed at the output shaft of the joint module to realise compliant motion control of the manipulator and safety of the human fusion operation. Other components include brakes and drive control boards.

Owing to the immense flexibility of the harmonic reducer and torque sensor in the lightweight connecting rod and

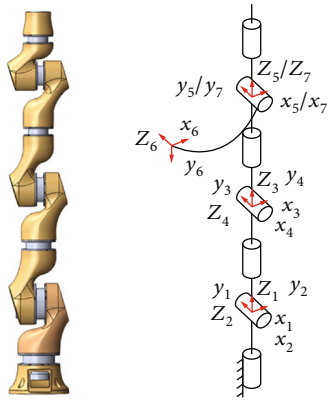


Figure 1: Configuration of space robot SHIR5.

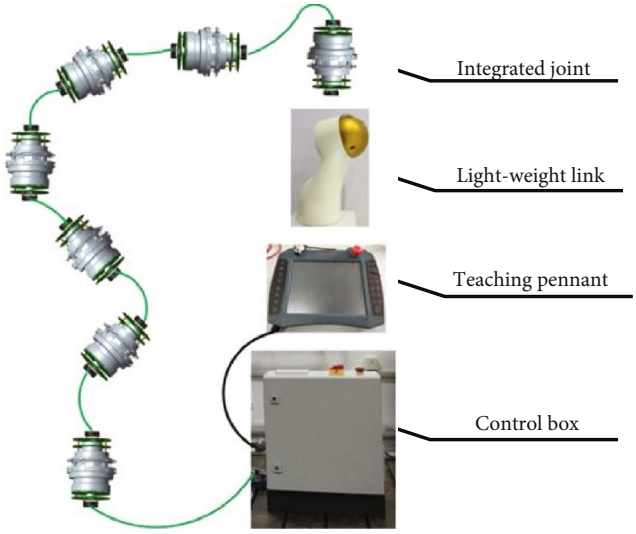


Figure 2: Composition of the robot system.

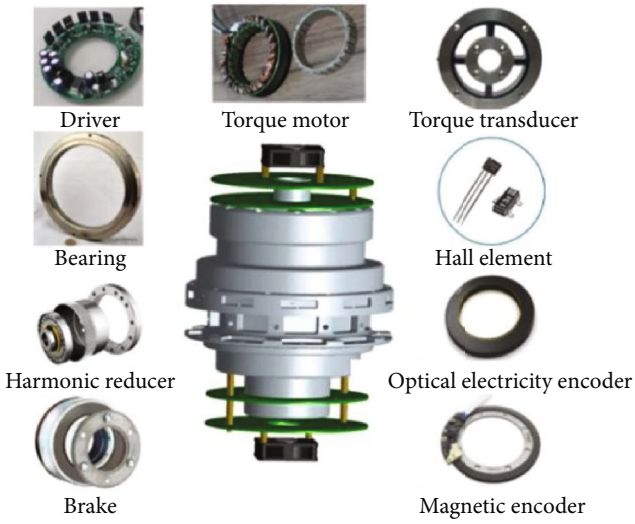


Figure 3: Composition of the integrative joints of robot.

integrated joint, the rigidity of the robot system is poor, which causes the robot to vibrate during operation and reduces its dynamic accuracy and stability. Hence, it was necessary to perform flexible coupling dynamic modelling and dynamic characteristic analysis of the robot.

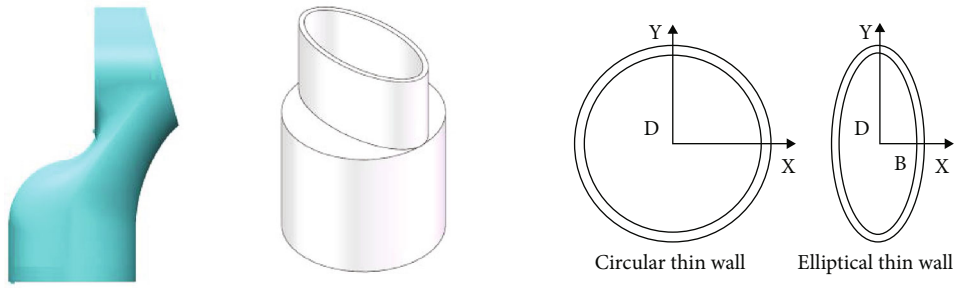
2.2. Rigid-Flexible Coupling Dynamic Modelling. Figure 4(a) shows that the connecting rod of the robot was irregular in shape. First, the connecting rod was simplified and divided into circular, noncircular, and thin-walled parts. The noncircular thin-walled part was simplified through an elliptical section, which was conducive to establishing the solution model. A schematic of the thin circular and elliptical walls is shown in Figure 4(b). The diameter of the circular section was defined as D, and the major and minor axes of the elliptical section were defined as D and B, respectively. The minor axis of the ellipse was along the x-direction, and the thickness of the connecting rod was d.

The space robot is equipped with an integrated joint that uses a harmonic reducer and a torque sensor to increase joint flexibility. The tilting stiffness of the joint is significantly greater than its torsional stiffness. Therefore, this study simplified the joint using a linear torsional spring, as shown in Figure 5. θ is the rotation angle of the motor rotor obtained after conversion of the reduction ratio of the harmonic reducer. The motor rotor rotates the flexible rod using a harmonic reducer and torque sensor. k represents the stiffness of the joint and δ represents the elastic deformation of the joint.

An overall simplified model that considers the flexibility of the connecting rod and joint was obtained using the above simplification of the connecting rod and joint, and the robot was assembled. As shown in Figure 6, the connecting rod was simplified into two elements: a circular section and an elliptical section. The serial number of element j on connecting rod L_i was denoted by ij, and the length and mass per unit length of the element were denoted by l_{ij} and m_{ij}, respectively. The mass of joint J_i and the torsional stiffness were denoted by m_{pi} and k_i, respectively. Connecting rods L_7 and joint J_7 had little effect on the system. To simplify the model, L_7, J_7, and P were considered equivalent to a rigid connecting rod, a concentrated mass, and the load, respectively.

A spatial beam element was used to model the flexible link of the robot. The transverse bending deformation of the link was expressed by the cubic Hermite function. The axial tensile deformation and bending deformation around the axis were expressed by a linear function. The deformation displacement and deformation angle of the element were denoted as u and v, where $u = [u_x, u_y, u_z]^T$ and $v = [v_x, v_y, v_z]^T$, respectively. The flexible joint was simplified as a linear spring, and the flexible motion relationship of the robot was obtained.

A right-handed coordinate system was established for the robot. The base frame of the robot was $O - x_0y_0z_0$, and the local frame $O - x_iy_iz_i$ was fixed to the flexible link. The z-axis of the local frame always coincided with the axis of the i_{th} link, as shown in Figure 7. θ is the angle of the joint



(a) Three-dimensional model and simplified model of the connecting rod (b) Thin-skin model

Figure 4: Simplified model of the connecting rod.

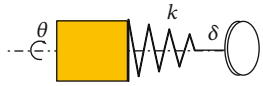


Figure 5: Simplified model of joints.

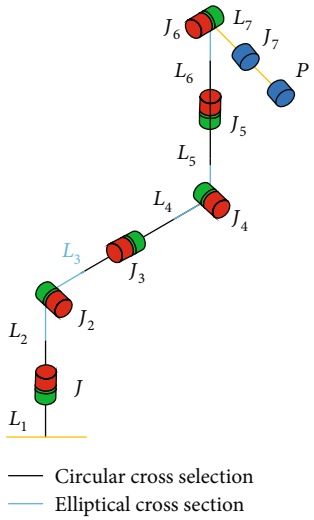


Figure 6: Simplified model of the joints.

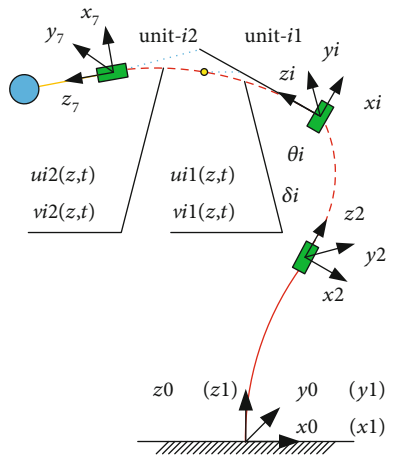


Figure 7: Flexible motion relationship of robot.

and δ is the flexible deformation of the joint. In a locally coordinated system, two units exist on the i_{th} link. In this study, the deflection and angle of unit deformation were represented by $u_{ij}(z,t)$ and $v_{ij}(z,t)$, respectively, where $i(i=1,2,\cdots,6)$ is the number of links, and $j(j=1,2)$ is the number of units, z is a point on the link unit.

It is assumed that $r_i{}'$ is the position of a point on the i_{th} link. When a point is located in the first unit of link, that is, when $j=1$, $r_i{}' = [u_{i1}(1,1), u_{i1}(2,1), \text{and}\, z + u_{i1}(3,1)]$ (1). When point is located in the second unit of link $(j=2)$, $r_i{}' = [u_{i2}(1,1), u_{i2}(2,1), \text{and}\, l_{i1} + z + u_{i2}(3,1)]$ (1). Thus, $r_i{}'$ can be described as

$$r_i' = \begin{bmatrix} 0 \\ 0 \\ (j-1)l_{i1} + z \\ 1 \end{bmatrix} + \begin{bmatrix} u_{ij} \\ 0 \end{bmatrix}, \tag{1}$$

where l_{i1} is the length of the first unit, $i=1,2,\cdots,6$, and $j=1,2$.

The transformation matrix of the flexible link can be obtained by multiplying the transformation matrix of the former link by that of the former joint. It is assumed that $T_{i1}=f_1(\theta)$, $T_{i2}=f_2(\delta)$, and $T_{i3}=f_3(u,v)$ are the transformation matrices of the joint revolution, flexible deformation of the joint, and the link, respectively.

When i is odd,

$$T_{i1} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & 0 \\ \sin\theta_i & \cos\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$
$$T_{i2} = \begin{bmatrix} 1 & -\delta_i & 0 & 0 \\ \delta_i & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2}$$

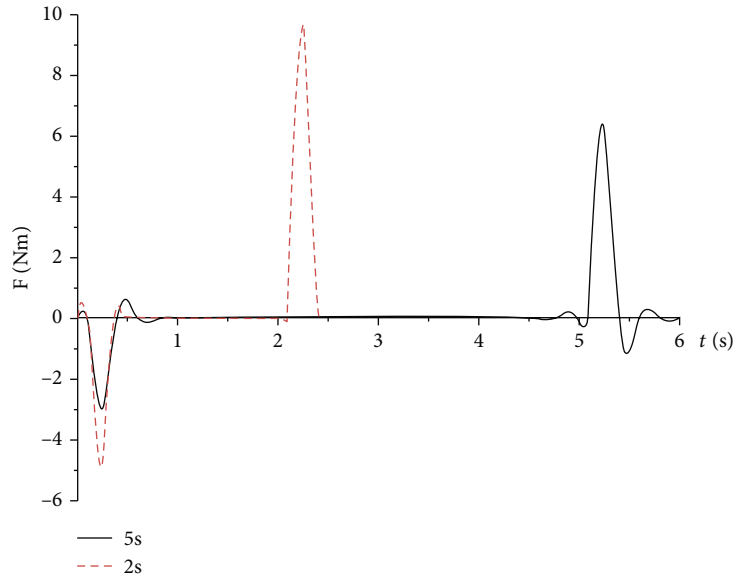


Figure 8: Excitation forces when task was completed in 5 s and 2 s.

When i is even,

$$T_{i1} = \begin{bmatrix} 1 & 0 & -\sin\theta_i & 0 \\ 0 & \cos\theta_i & 0 & 0 \\ 0 & \sin\theta_i & \cos\theta_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$T_{i2} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & -\delta_i & 0 \\ 0 & \delta_i & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (3)$$

$$T_{i3} = \begin{bmatrix} 1 & -v_x & v_y & u_x \\ v_x & 1 & -v_z & u_y \\ -v_y & v_z & 1 & z+u_z \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

The position of any point on link i with respect to the base frame can be expressed as

$$r_i = T_{13} * T_{11} * T_{12} \cdots * T_{i3} * T_{i1} * T_{i2} * r_i'. \quad (4)$$

The velocity at any point on link can be obtained by calculating the derivative of r_i with respect to time. Then, the equations of the rigid-flexible coupled dynamics of the robot can be constructed based on the Lagrange formulation. In this study, the effects of the elastic deformation φ on the mass, damping, stiffness, and gravity matrices were ignored to simplify the dynamic model of the robot. Moreover, the H matrix was omitted because the centripetal and Coriolis forces have little influence on the dynamics. Thus, the dynamics of the robot can be expressed as

$$\begin{bmatrix} M_{\theta\theta}(\theta) & M_{\theta\varphi}(\theta) \\ M_{\theta\varphi}^T(\theta) & M_{\varphi\varphi}(\theta) \end{bmatrix} \begin{bmatrix} \ddot{\theta} \\ \ddot{\varphi} \end{bmatrix} + \begin{bmatrix} C_{\theta\theta}(\theta) & C_{\theta\varphi}(\theta) \\ C_{\theta\varphi}(\theta) & C_{\varphi\varphi}(\theta) \end{bmatrix} \begin{bmatrix} \dot{\theta} \\ \dot{\varphi} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & K(\theta) \end{bmatrix} \begin{bmatrix} \theta \\ \varphi \end{bmatrix} + \begin{bmatrix} G(\theta) \\ 0 \end{bmatrix} = \begin{bmatrix} \tau \\ 0 \end{bmatrix}, \quad (5)$$

where M, C, K, and G are the mass matrix, centripetal force, Coriolis force, stiffness matrix, and gravity of the system, respectively, and $\varphi = [\delta, u, v]$, τ is the torque required for the joint.

There are many ways to express the dynamic characteristics of robots. Natural frequency, as an essential indicator, can reflect the dynamic characteristics of a robot and its overall flexibility. The joint stiffness of the robot, structural parameters of the connecting rod, and load significantly influence the natural frequency of the robot, thereby affecting its dynamic characteristics. The natural frequency equation of the robot can be obtained using its dynamic model.

$$\det\left(\omega[I] - M^{-1}K\right) = 0. \quad (6)$$

3. Vibration Characteristics Analysis

Equation (5) can be rewritten as

$$M_{\varphi\varphi}(\theta)\ddot{\varphi} + C_{\varphi\varphi}(\theta)\dot{\varphi} + K(\theta)\varphi = -M_{\theta\varphi}^T(\theta)\ddot{\theta} - C_{\theta\varphi}^T(\theta)\dot{\theta}. \quad (7)$$

Equation (6) can be regarded as the vibration equation for the flexible deformation of a robot. The right side of the equation represents the excitation force. Thus, the equation of motion is obtained as follows:

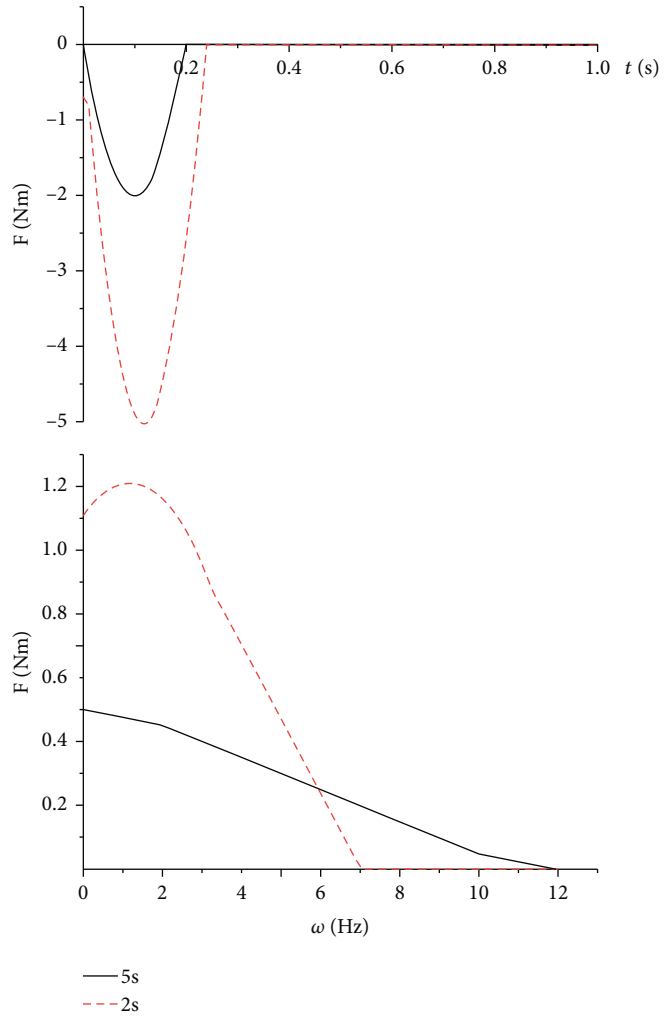


Figure 9: Frequency analysis of the excitation forces for tasks with different durations.

$$M_{\varphi\varphi}(\theta)\ddot{\varphi} + C_{\varphi\varphi}(\theta)\dot{\varphi} + K_{\varphi\varphi}\varphi = F\left(\theta, \dot{\theta}, \ddot{\theta}\right), \tag{8}$$

where $M_{\varphi\varphi}$, $C_{\varphi\varphi}$, $K_{\varphi\varphi}$, θ, φ, and F are the mass matrix, damping matrix, stiffness matrix, desired trajectory of the joint, flexible deformation of the link and joint, and excitation force, respectively, which are determined by the robot's desired motion.

The vibration response of a system can be classified as a transient or steady-state response. An effective method for suppressing the vibration of the robot is reducing the excitation force, because the steady response depends on the excitation force, which suppresses the vibration of the robot.

There are two types of vibrations in a robot: vibration during movement and residual vibration after movement. The steady-state response is the vibration generated by an external excitation force in the robot. The amplitude of the vibration was related to the magnitude and frequency of the excitation force. Resonance occurs when the frequency of the excitation force approaches the natural frequency.

Using a Fourier transform, the excitation force was decomposed into a series of simple harmonic excitation forces of different frequencies, and then a response superposition was performed. The steady-state response can be expressed as follows:

$$\begin{gathered} x(t) = B\sin(\omega t - \varphi), \\ B = \frac{F_0}{\sqrt{(k - m\omega^2)^2 + c^2\omega^2}} = \frac{F_0/k}{\sqrt{(1 - \nu^2)^2 + (2\xi\nu)^2}}, \\ \varphi = tg^{-1}\frac{c\omega}{k - m\omega^2} = tg^{-1}\frac{2\xi\nu}{1 - \nu^2}, \end{gathered} \tag{9}$$

where $\nu = \omega/\omega_n$ is the frequency ratio and $\xi = c/2\omega_n m$ is the damping ratio.

Therefore, when analysing the motion characteristics of a robot, the magnitude of the excitation force cannot be analysed, and the frequency must be analysed. The vibration equation of the robot shows that its elastic deformation varies with the rigid joint motion θ. This section discusses the dynamic characteristics of robots covering the same trajectories with durations of 2 and 5 s.

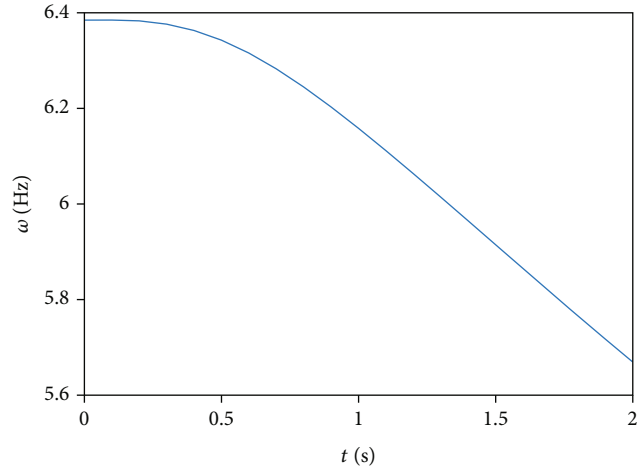


Figure 10: Change in the first-order natural frequency of the robot with respect to time.

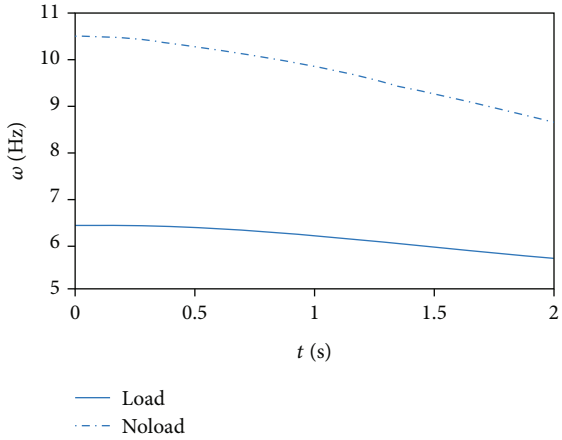


Figure 11: Effect of a load on the first-order natural frequency of the robot.

3.1. Influence of the Excitation Force on the Dynamic Characteristics of the Robot. When tasks with the same trajectory and different durations were completed, the excitation force F, which causes robot vibration, was found to differ with the duration of the trajectory. As shown in Figure 8, the excitation force F for the task completed in 5 s was smaller than that for the task completed in 2 s. Therefore, at the start and stop stages, the vibration of the robot when the task was completed in 2 s was more evident because of the larger excitation force F.

3.2. Influence of the Frequency on the Dynamic Characteristics of the Robot. The vibration amplitude of the robot is related to the magnitude of the excitation force, F, and its frequency. When the frequency of the excitation force was closer to the natural frequency of the system, the vibration became more intense. Therefore, it was necessary to analyse the frequencies of the excitation forces F_5 and F_2 for tasks having durations of 5 and 2 s. The results of the analysis are shown in Figure 9. The excitation frequency generated by the excitation force F_5 in the 5 s process is mainly distributed in the range of 0-10 Hz, and the excitation frequency generated by the excitation force F_2 in the 2 s process is primarily distributed in the range of 0-6 Hz. The excitation frequency generated by the excitation force F_2 in the process of 2 s is relatively concentrated. Therefore, the frequency of excitation force generated by different trajectories is different, thus affecting the excitation effect of the system.

As shown in Figure 10, the first-order natural frequency was between 5.7 and 6.4 Hz. Spectrum analysis of the excitation force (Figure 9) revealed that the amplitude of the frequency components in F_5 was higher than that in F_2. Therefore, during the subsequent relatively stable motion, the vibration generated when the task was completed in 5 s was more evident than that generated by the task with a 2 s duration when the excitation force was not significantly different.

3.3. Effect of a Load on the Dynamic Characteristics of the Robot. Figure 11 shows the change in the first-order natural frequency when the robot moved with and without a load. It can be observed that a load has a significant influence on the natural frequency. After adding the load, the natural frequency of the robot reduced, and the amplitude of the frequency components with similar excitation and natural frequencies increased; hence, the vibration intensified significantly. Therefore, the load cannot exceed a specific value to ensure dynamic accuracy and prevent severe vibration.

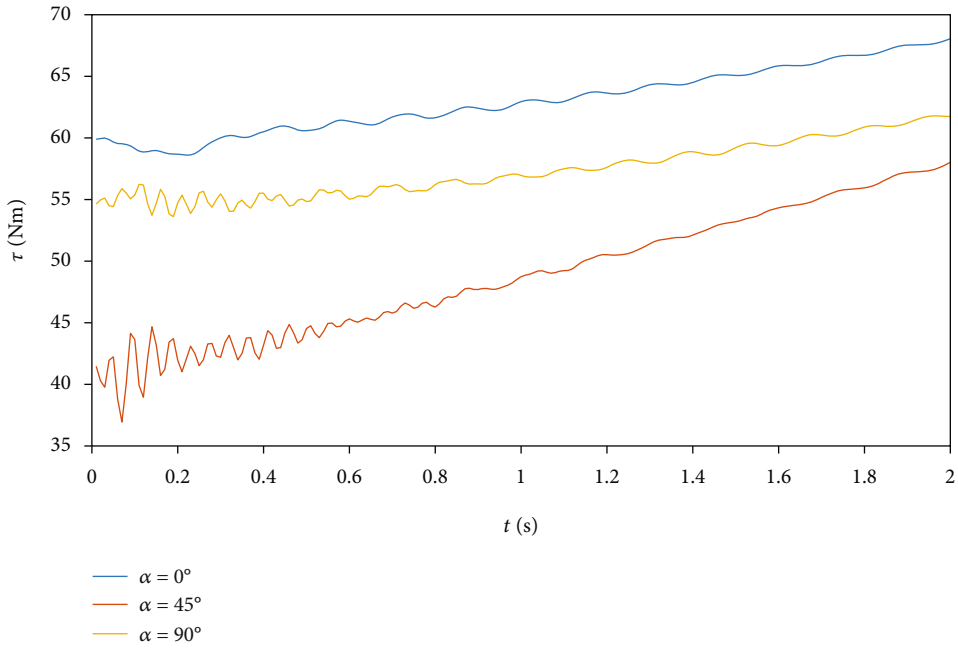


Figure 12: Effect of different initial arm angles on vibration of the robot.

Table 1: Initial value of arm angle corresponding to the angle of each joint.

α	0°	45°	90°
Joint angles	[-0.1484, 0.2176, 0, 1.7526, -0.3814, -0.4852, -0.1630]	[-1.1881, 0.6817, 1.4924, 1.7526, 0.3604, -0.4772, -1.5397]	[-1.0822, 1.2726, 1.9487, 1.7526, 0.6073, -0.7833, -2.4546]

3.4. Influence of the Initial Pose on Vibration. During the optimization process, the initial value of the arm angle α had a significant influence on the optimization results. For the same end position, different α values led to different initial joint angles and vibration effects. In this section, the effect of the initial arm angles on the end vibrations under the same trajectory is reported. The considered arm angles were 0°, 45°, and 90°. As shown in Figure 12, when the initial arm angle α was 0°, the torque of the joint was the largest, whereas the corresponding vibration was the smallest. When $\alpha = 45°$, the initial torque of the joint was the smallest; however, the corresponding vibration was significant. The angles of each joint of the robot corresponding to the initial arm angle α are listed in Table 1. In this study, when the initial conditions were relatively poor (i.e., $\alpha = 45°$), the initial vibration of the joint was significant.

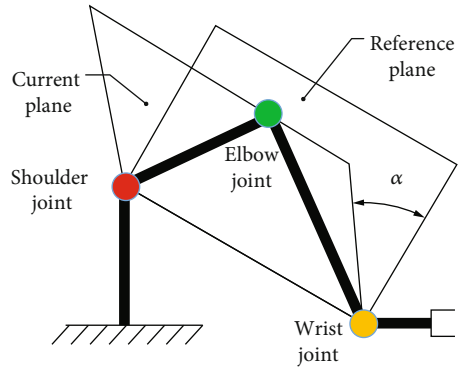


Figure 13: Principle diagram of the inverse kinematics solution for 7-DOF redundant robot.

4. Vibration Suppression

To ensure end accuracy, the arm angle of the robot was optimised in zero space using the analytical solution method, and a trajectory with a vibration suppression effect was obtained under the premise of satisfactory joint angle, angular velocity, and angular acceleration of the robot. The analytical solution method for the inverse solution of 7-DOF redundant robots is based on the method introduced [31], and its principle is illustrated in Figure 13. The shoulder and wrist joints are equivalent joints, and α is the arm angle, i.e., the angle between the current plane formed by the manipulator and the specified reference plane. Each joint angle value as a function of pose x and arm angle α is expressed as follows:

$$\theta = f^{-1}(x, \cdot \alpha), \tag{10}$$

where f denotes the kinematic function of the robot.

(a) Position and posture of P_1 (b) Position and posture of P_2

Figure 14: Initial posture and end posture of the robot.

The inverse kinematics solution method revealed that the robot's inverse kinematics were greatly affected by the angle α, which needs to be optimised in the solution process. The angular velocity of the joint and the angular velocity of the robot can be obtained by differentiating the joint angle of the robot as follows:

$$\begin{aligned} \dot{\theta}_{t+1} &= (\theta_{t+1} - \theta_t)/\Delta t, \\ \ddot{\theta}_{t+1} &= \left(\dot{\theta}_{t+1} - \dot{\theta}_t\right)/\Delta t, \end{aligned} \tag{11}$$

where Δt is one control cycle of the robot.

Using the above equations, the planning problem of the positioning layer was transformed into a problem of optimising the arm angle α for the minimum excitation force under the premise of a satisfactory joint angle, angular velocity, and angular acceleration.

$$\begin{aligned} \min \quad & \zeta = \left(M_{\theta\varphi}^T \ddot{\theta} + C_{\theta\varphi}^T \dot{\theta}\right)^T \left(M_{\theta\varphi}^T \ddot{\theta} + C_{\theta\varphi}^T \dot{\theta}\right), \\ \text{s.t.} \quad & \theta_{\min} \le \theta \le \theta_{\max}, \\ & V_{\min} \le \dot{\theta} \le V_{\max}, \\ & A_{\min} \le \ddot{\theta} \le A_{\max}. \end{aligned} \tag{12}$$

5. Experimental Study

An experimental study was conducted to validate the proposed vibration suppression method based on trajectory generation. Figure 14 shows a self-developed collaborative robot with seven DOFs and anthropomorphic configuration. The trajectory of the robot end was generated to study the impact of joint angle θ on the vibration of the robot. The robot end moved in a straight line from point P_1 to point P_2 and its attitude remained unchanged. The trajectory generation method is based on S-curve. The results for position, speed, maximum acceleration, and acceleration are shown in Figure 15. The details of the model parameters are listed in Table 2. The initial and target attitudes of the robot are shown in Figure 14. Planning parameters of the robot end are listed in Table 3.

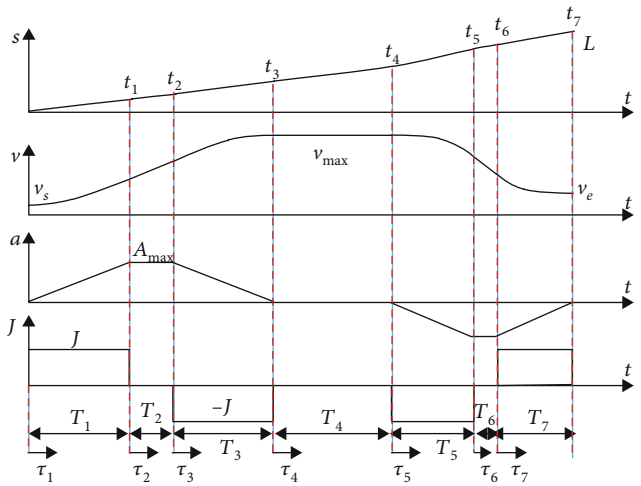


Figure 15: S-curve planning.

A torque sensor was mounted on the joint to measure the vibration of the robot. A low-pass filter was used to remove unwanted noise from the output of the torque sensor, and the amplitude of the signal was regarded as the amplitude of the vibration. The torque at the joint was calculated by substituting the actual angular velocity of the joint into the theoretical model.

$v_{\max}$ is the maximum speed of the robot end, T_k is the running time for each stage where $k = 1, 2, \cdots, 7$, L is the displacement of the robot throughout the operation, $A_{\max}$ is the maximum acceleration in the acceleration stage, and J is the acceleration of each stage.

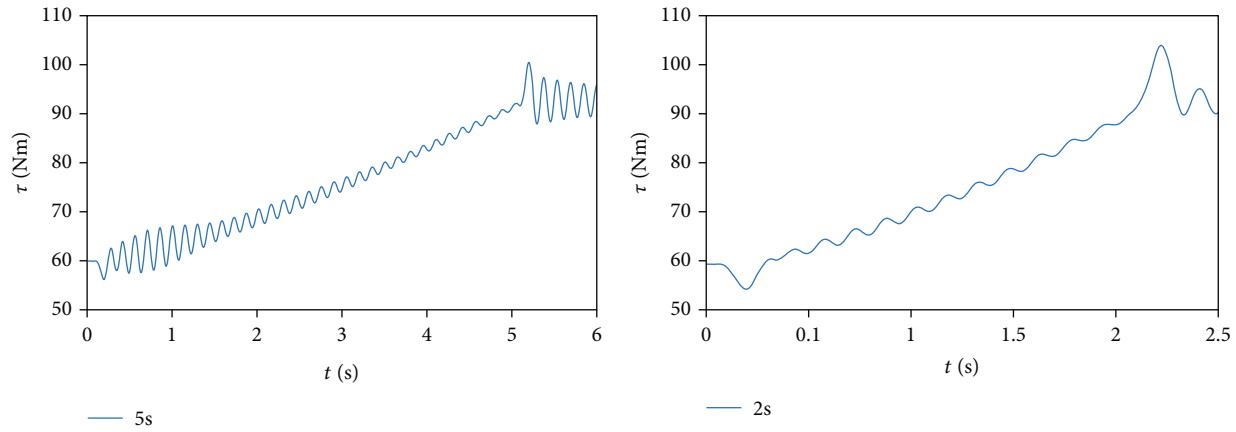
5.1. Effects of Different Trajectories and Loads on Vibration. Torque sensors were used on the joints to observe the vibrations of the robot. Figure 16 shows the vibration behaviour of the robot that covers the same trajectory in 2 and 5 s. Initially, the vibration corresponding to the 2 s motion was significantly more intense than that corresponding to the 5 s motion. For subsequent motions, the vibration for the 5 s

Table 2: Initial posture and joint angle of the robot.

Points	P_1	P_2
Position/mm (x, y, z)	[506.102, -75.6716, 264.402]	[661.72, 370.513, 138.943]
Attitude/°	[1.5096, 86.8577, -29.4258]	[1.5096, 86.8577, -29.4258]
Joint angle/rad	[-0.1484, 0.2176, 0, 1.7526, -0.3814, -0.4852, -0.1630]	[0.5087, 0.8476, 0.0033, 1.0840, 0.8939, -0.6359, -1.3373]

Table 3: Planning parameters of the robot end.

Parameters	2 s	5 s
T_i/s	[0.08, 0, 0.08, 1.84, 0.08, 0, 0.08]	[0.18, 0, 0.18, 4.84, 0.18, 0, 0.18]
J/(mm/s^3)	[12260, 80460, -9930]	[1000, 2870, -920]
A/(mm/s^2)	[980, 6440, -800]	[180, 1150, -324]
v/(mm/s)	[78.5, 225.5, -64]	[31.5, 90.5, -25.5]
L/mm	[158, 454, -128]	[158, 454, -128]



(a) Joint torque for the 5 s motion

(b) Joint torque for the 2 s motion

Figure 16: Influence of different trajectories on robot vibration.

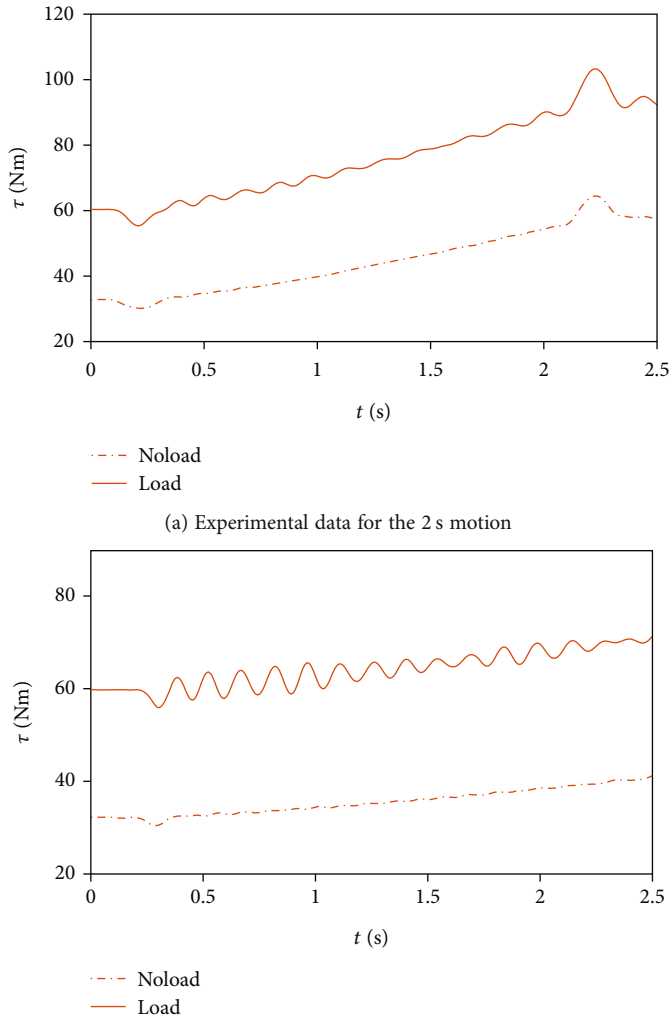
motion was more intense than that for the 2 s motion. The excitation force F_2 was larger than F_5, indicating that the size of the excitation force alone did not determine the vibration of the robot.

It can be observed from the analysis results in Figure 17 that the load has a significant influence on the vibration of the robot. Its influence on the 2 s motion was smaller than that on the 5 s motion. With a load, the vibration for the 5 s motion was more pronounced; however, without a load, the vibration for the 5 s motion was significantly reduced. This reduction was more evident than that for the 2 s motion.

It can be observed from the vibration equation and vibration phenomenon of the robot that the trajectory of the robot joint has a significant influence on the excitation force. Therefore, the excitation force can be optimised by optimising the robot joint trajectory. The smaller the excitation force, the better the vibration reduction of the robot. For a task with a required end trajectory, the trajectory of the joint cannot be represented by the existing trajectory equation. Under the premise of a limited end trajectory, the redundant characteristics of the robot were used to optimise the joint trajectory through self-motion. Finally, vibration suppression of the robot was realised.

5.2. Verification of the Vibration Suppression Algorithm. For the results presented in this section, the initial arm angle α of the robot was set to 45°, the planning period of the robot was 0.01 s, and the robot load was 5 kg. The trajectory of the robot was optimised by the method described in this article. Then, the optimised robot joint trajectory was substituted into the dynamic model of the robot, and the torque of the second joint of the robot was obtained using the Newmark method.

Figure 18 shows the joint torque obtained by the position layer trajectory planning compared with that obtained before planning. Before vibration suppression, the robot experiences a slight vibration owing to the influence of



(a) Experimental data for the 2 s motion

(b) Experimental data for the 5 s motion

Figure 17: Effect of the load on vibration.

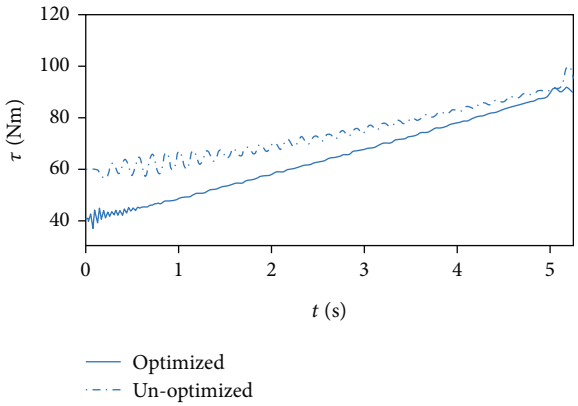


Figure 18: Comparison of joint torques before and after optimization.

acceleration during the initial movement, and the joint torque vibration amplitude is approximately 5 Nm. Subsequently, the vibration gradually becomes intense during smooth operation (0.5 s–1.5 s). The initial motion generates a shock force through the optimised trajectory owing to acceleration, resulting in a slight vibration, which is significantly weakened during smooth operation. Through comparison, the vibration of the joint torque after planning was considerably lower than that before planning, which illustrates the effectiveness of the trajectory planning method for vibration suppression of the robot.

Figure 19 compares the joint angles before and after the optimization of robot trajectory planning. It can be seen from the figure that the trajectory of optimised anterior and posterior joints 4 does not change. The trajectory of the other joints changes owing to the self-motion characteristics of the redundant robot.

Figure 20 shows the x-direction trajectory of the optimised front and rear ends. It can be seen from the figure that the trajectories of the front and back ends of the optimization are the same, indicating that the position layer trajectory planning can effectively suppress vibrations to ensure the end trajectory.

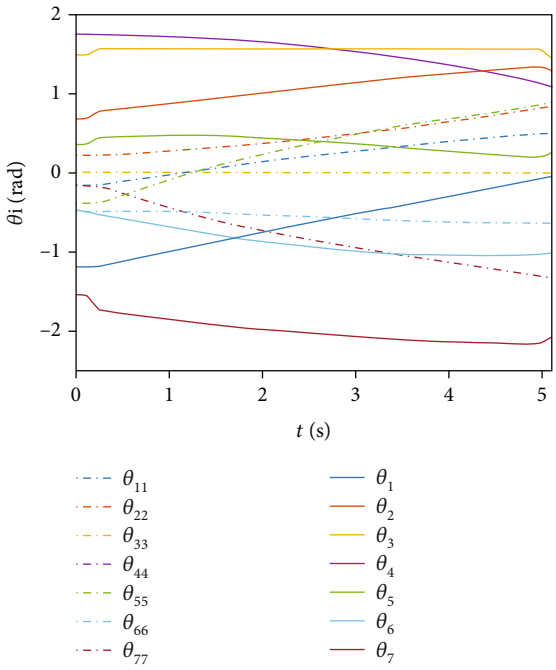


Figure 19: Contrast of joint trajectories. Note that θ_{ii} represents the unoptimised joint trajectory, and θ_i represents the optimised joint trajectory.

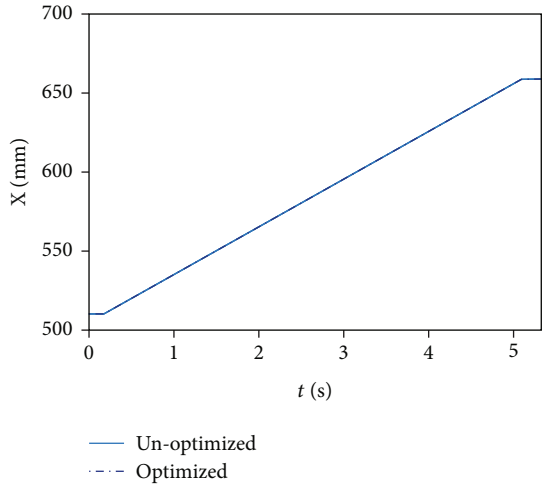


Figure 20: Trajectory of the robot end based on location planning.

6. Conclusions

In this study, the factors influencing the vibration behaviour of a robot were analysed by establishing a rigid-flexible coupling dynamic model of a flexible space robot. The following conclusions can be drawn: (1) the same path and different motion times lead to different vibration behaviours of the robot. In the trajectory planning process, the size and frequency of the excitation force should be considered comprehensively. The excitation force must be reduced and monitored to avoid the natural frequency of the system. (2) The loads affected the dynamic parameters of the robot and changed its natural frequency. Thus, the load capacity should be effectively limited during robot design. The above conclusions reveal the factors influencing robot vibration behaviour and provide support for the design and development of robots.

Subsequently, through the above analysis, the trajectory planning method was used to suppress the vibration of the robot, and the joint torque before and after the vibration suppression was compared experimentally. The vibration of the joint torque after planning was significantly lower than that before planning. The vibration amplitude reduced by 80%, which can improve the positioning accuracy and work efficiency of the robot, and illustrates the effectiveness of the trajectory planning method for the vibration suppression of a robot.

Data Availability

Data is available at http://www.re3data.org.

Conflicts of Interest

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

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under grant 51605311 and the Youth Program of Liaoning Education Department under grant LSNQN202025.

References

[1] H. Liu, D. Y. Liu, and Z. N. Jiang, "Space manipulator technology: review and prospect," *Acta Aeronauticaet Astronautica Sinica*, vol. 4, no. 1, pp. 524164–524164, 2021.

[2] G. Meng, L. L. Han, and C. F. Zhang, "Research progress and technical challenges of space robot," *Acta Aeronauticaet Astronautica Sinica*, vol. 42, no. 1, pp. 523963–523963, 2021.

[3] X. Zhang, J. G. Liu, J. Feng, Y. W. Liu, and Z. Ju, "Effective capture of nongraspable objects for space robots using geometric cage pairs," *IEEE/ASME Transactions on Mechatronics*, vol. 25, no. 1, pp. 95–107, 2019.

[4] X. Zhang, J. Liu, Y. Tong, Y. Liu, and Z. Ju, "Attitude decoupling control of semifloating space robots using time-delay estimation and supertwisting control," *IEEE Transactions on Aerospace Electronic Systems*, vol. 57, no. 6, pp. 4280–4295, 2021.

[5] X. Zhang and J. Liu, "Effective motion planning strategy for space robot capturing targets under consideration of the berth position," *Acta Astronautica*, vol. 148, pp. 403–416, 2018.

[6] H. Liu, D.-Y. Liu, and Z.-N. Jiang, "Review of space manipulator control technologies," *Robot*, vol. 44, no. 3, p. 22, 2022.

[7] M. Hu, H. Wang, and X. Pan, "Optimal configuration selection for stiffness identification of 7-Dof collaborative robots," *Intelligent Service Robotics*, vol. 13, no. 3, pp. 379–391, 2020.

[8] M. Hu, H. Wang, and X. Pan, "Multi-objective global optimum design of collaborative robots," *Structural Multidisciplinary Optimization*, vol. 62, no. 3, pp. 1547–1561, 2020.

[9] H. Liu, Z.-Q. Li, Y.-W. Liu et al., "Key technologies of TianGong-2 robotic hand and its on-orbit experiments," *Scientia Sinica Technologica*, vol. 48, no. 12, pp. 1313–1320, 2018.

[10] Z.-C. Qiu, "Review on research progress in vibration measurement and control of flexible manipulators," *Information and Control*, vol. 50, no. 2, p. 141, 2021.

[11] D. Bandopadhya, D. K. Bhogadi, B. Bhattacharya, and A. Dutta, "Active vibration suppression of a flexible link using ionic polymer metal composite," in *IEEE Conference on Robotics, Automation and Mechatronics*, pp. 1–6, Bangkok, Thailand, June 2006.

[12] A. Warkentin and S. E. Semercigil, "Variable stiffness control of a single-link flexible robotic arm," *Journal of Sound Vibration*, vol. 187, no. 1, pp. 1–21, 1995.

[13] L. Xie, X. Yu, and L. Chen, "Robust fuzzy sliding mode control and vibration suppression of free-floating flexible-link and flexible-joints space manipulator with external interference and uncertain parameter," *Robotica*, vol. 40, no. 4, pp. 997–1019, 2022.

[14] L. Sun, W. Yin, M. Wang, and J. Liu, "Position control for flexible joint robot based on online gravity compensation with vibration suppression," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 6, pp. 4840–4848, 2017.

[15] K. Li, H. Wang, X. Liang, and Y. Miao, "Visual servoing of flexible-link manipulators by considering vibration suppression without deformation measurements," *IEEE Transactions on Cybernetics*, vol. 2021, pp. 1–10, 2021.

[16] M. P. P. Reddy and J. Jacob, "Vibration control of flexible link manipulator using SDRE controller and Kalman filtering," *Studies in Informatics and Control*, vol. 26, no. 2, pp. 143–150, 2017.

[17] H. Huang, G. Tang, H. Chen, L. Han, and D. Xie, "Dynamic modeling and vibration suppression for two-link underwater flexible manipulators," *IEEE Access*, vol. 10, pp. 40181–40196, 2022.

[18] D. Meng, W. Lu, W. Xu et al., "Vibration suppression control of free-floating space robots with flexible appendages for autonomous target capturing," *Acta Astronautica*, vol. 151, pp. 904–918, 2018.

[19] H. Sun, S. Hou, Q. Li, and X. Tang, "Research on the configuration of cable-driven parallel robots for vibration suppression of spatial flexible structures," *Aerospace Science and Technology*, vol. 109, article 106434, 2021.

[20] H. Liu, X. Lai, and W. Wu, "Time-optimal and jerk-continuous trajectory planning for robot manipulators with kinematic constraints," *Robotics Computer-Integrated Manufacturing*, vol. 29, no. 2, pp. 309–317, 2013.

[21] L. Cui, H. Wang, and W. Chen, "Trajectory planning of a spatial flexible manipulator for vibration suppression," *Robotics Autonomous Systems*, vol. 123, article 103316, 2020.

[22] T. Zhang, M. Zhang, and Y. Zou, "Time-optimal and smooth trajectory planning for robot manipulators," *International Journal of Control, Automation, Systems*, vol. 19, no. 1, pp. 521–531, 2021.

[23] B. Cao, K. Sun, T. Li, Y. Gu, M. Jin, and H. Liu, "Trajectory modified in joint space for vibration suppression of manipulator," *IEEE Access*, vol. 6, pp. 57969–57980, 2018.

[24] M. Wu, J. Mei, Y. Zhao, and W. Niu, "Vibration reduction of delta robot based on trajectory planning," *Mechanism Machine Theory*, vol. 153, article 104004, 2020.

[25] R. H. Lei and L. Chen, "Finite-time tracking control and vibration suppression based on the concept of virtual control force for flexible two-link space robot," *Defence Technology*, vol. 17, no. 3, pp. 874–883, 2021.

[26] C. Jalendra, B. Rout, and A. Marathe, "Vision sensor based residual vibration suppression strategy of non-deformable object for robot-assisted assembly operation with gripper flexibility," *Industrial Robot: the international journal of robotics research and application*, vol. 49, no. 5, pp. 851–864, 2022.

[27] Y.-F. Du and C. Wang, "Residual vibration control of flexible manipulator," *Journal of Vibration and Shock*, vol. 38, no. 7, pp. 165–171, 2019.

[28] L. Zhao, "Dynamic analysis of a flexible robotic arm carrying a moving end effector," *Journal of Vibration and Shock*, vol. 39, no. 8, pp. 112–117, 2020.

[29] Z. Yu, Y. Guo, W. Yao, Y. F. Wu, and J. Guo, "Experiments on path planning for rotational motions of a flexible manipulator using spectral analysis based approach," *Journal of Vibration Engineering*, vol. 33, no. 4, pp. 717–723, 2020.

[30] Z. Yu, C. Zhong, and Y. Guo, "Spectral analysis and parameter selection for BCB attitude maneuver path of flexible spacecraft," in *IEEE International Conference on Information & Automation*, pp. 729–734, Yinchuan, China, August, 2013.

[31] M. Shimizu, H. Kakuya, W.-K. Yoon, K. Kitagaki, and K. Kosuge, "Analytical inverse kinematic computation for 7-DOF redundant manipulators with joint limits and its application to redundancy resolution," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1131–1142, 2008.

Hindawi
International Journal of Aerospace Engineering
Volume 2022, Article ID 6523234, 15 pages
https://doi.org/10.1155/2022/6523234

Research Article

Multiscale Feature Fusion Attention Lightweight Facial Expression Recognition

Jinyuan Ni,[1] **Xinyue Zhang**,[2] **and Jianxun Zhang**[1]

[1]*College of Computer Science and Engineering, Chongqing University of Technology, 400054, China*
[2]*Sydney Smart Technology College, Northeastern University, 066004, China*

Correspondence should be addressed to Jianxun Zhang; zjx@cqut.edu.cn

Received 27 June 2022; Revised 19 July 2022; Accepted 2 August 2022; Published 24 August 2022

Academic Editor: Qing Gao

Facial expression recognition based on residual networks is important for technologies related to space human-robot interaction and collaboration but suffers from low accuracy and slow computation in complex network structures. To solve these problems, this paper proposes a multiscale feature fusion attention lightweight wide residual network. The network first uses an improved random erasing method to preprocess facial expression images, which improves the generalizability of the model. The use of a modified depthwise separable convolution in the feature extraction network reduces the computational effort associated with the network parameters and enhances the characterization of the extracted features through a channel shuffle operation. Then, an improved bottleneck block is used to reduce the dimensionality of the upper layer network feature map to further reduce the number of network parameters while enhancing the network feature extraction capability. Finally, an optimized multiscale feature lightweight attention mechanism module is embedded to further improve the feature extractability of the network for human facial expressions. The experimental results show that the accuracy of the model is 73.21%, 98.72%, and 95.21% on FER2013, CK+ and JAFFE, respectively, with a covariance of 10.14 M. Compared with other networks, the model proposed in this paper has faster computing speed and better accuracy at the same time.

1. Introduction

In recent years, with the rapid development of space technology, human-robot interaction in on-orbit service (OOS) space robots has become an important research area in space technology [1–3]. Although the intelligence of space robots is limited, space human-computer interaction plays an important role in space mission applications. Space robots can replace or assist astronauts in various on-board/off-board activities, and it is particularly important for space robots to recognise astronaut commands [4]. Facial expression recognition by astronauts is a widely used method of human-robot interaction in space that does not rely on the highly intelligent capabilities of space robots, and it can effectively combine the decision-making capabilities of humans with the precise operational capabilities of space robots to improve their operational capabilities [5–7]. The accuracy of the astronaut's facial expression recognition and the size of the expression recognition model are important indications of the increased efficiency of space robots.

Early on in the research process, the features of facial expressions were basically extracted manually, but the recognition accuracy was not high because the facial expressions in the natural environment were easily affected by many factors, such as occlusion, background, and pose [8]. In recent years, deep learning has achieved major breakthrough results in image recognition. Sun et al. [9] designed a facial expression recognition system combining shallow and deep features with an attention mechanism and proposed an attention mechanism model based on the relative positions of facial feature points and textural features of local regions of faces for better extraction of shallow features. Wenmeng and Hua [10] proposed a new end-to-end coattentive multitasking convolutional neural network that consists of a channel coattentive module and a spatial coattentive module. Their approach demonstrates better performance relative to single tasking and multitasking. Shi et al. [11]

proposed a facial expression recognition method based on a multibranch cross-connected convolutional neural network, which was built based on residual connections, network-in-network, and tree structure combined; it also added fast cross-connections for the summation of the convolutional output layer, which makes the data flow between networks smoother and improves the feature extractability of each sensory domain. Kong et al. [12] proposed a lightweight facial expression recognition method based on an attention mechanism and key region fusion, and to reduce the computational complexity, a lightweight convolutional neural network was used as the basic recognition model for expression classification, which reduces the computational effort of the network to some extent. Zhou et al. [13] designed a lightweight convolutional neural network that uses a multitask cascaded convolutional network to accomplish face detection and combines a residual module and a depthwise separable convolutional module to reduce a large number of parameters of the network and make the model more portable.

Although most of the above studies were able to extract features and lighten the model to some extent, there are still shortcomings. For example, the face acquisition process is susceptible to factors, such as lighting, background, and pose, resulting in a reduced learning ability of the model when training the face sample set and insufficient feature extractability. The number of network layers of the deep learning model also affects the accuracy of classification recognition to a certain extent, i.e., as the number of network layers increases, the phenomenon of gradient disappearance occurs, causing a decrease in recognition accuracy. To solve the above problems, this paper proposes a multiscale feature fusion attention lightweight network, making the following main contributions.

First, during the image preprocessing stage, a random erasing method based on data labels is used to mask the facial expression images to expand the training set samples and improve the robustness of the model.

Second, to further extract the deep features of facial expressions, an improved convolutional block attention module (CBAM) is embedded in the model, which rerepresents the features of facial expressions in both channel and spatial dimensions.

Third, to solve the problem of model redundancy caused by too many convolutional layers, the improved bottleneck layer is used to reduce the dimensionality of the network, which saves the computation of the network and increases the nonlinear expression capability of the model.

Fourth, to lighten the model, an improved depthwise separable convolution module is added to reduce the number of parameters computed by the network while speeding up the network operations.

Finally, through comparison with different network models, it can be verified that the model proposed in this paper has higher accuracy and lightness.

2. Related Work

2.1. Spatial/Channel Attention Mechanism [14, 15]. CBAM is a lightweight module that combines channel attention and spatial attention to dramatically improve model performance while requiring a small amount of computation and a small number of parameters. The channel attention mechanism [16–18] focuses on which channel features are meaningful using global average pooling and global maximum pooling to obtain two feature maps and then feeds them sequentially into a weight-sharing multilayer perceptron with a 1×1 convolution to better fuse channel information. The spatial attention mechanism [19, 20] focuses on spatial features, mainly on the part of the input image that is richer in effective information. One of the pooling operations is performed along the channel axis, i.e., each pooling compares values between different channels rather than values in different regions of the same channel.

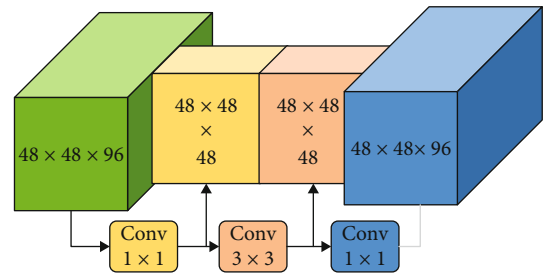


Figure 1: Structure of the bottleneck.

2.2. BottleNeck Layer. The bottleneck layer [21] is the core structure of the residual network [22], which mainly contains three convolutional layers, as shown in Figure 1. The size of the convolution kernel in the first layer is 1×1, which is mainly aimed at reducing the dimensionality of the feature map and thus the number of network parameters. The size of the convolution kernel in the second layer is 3×3, and the main purpose is to extract deeper semantic information without enhancing the number of network parameters. The convolutional kernel size of the third layer is 1×1, and the main purpose is to updimension the feature map to obtain the desired dimension size.

The formula for calculating the number of parameters during the conventional convolution operation is shown in

$$\text{Params} = C_{\text{in}} \times k \times k \times C_{\text{out}}, \tag{1}$$

where Params represents the number of parameters in the convolution process, C_{in} and C_{out} represent the number of channels of the input and the output feature map, respectively, and k represents the size of the convolutional kernel.

Assuming that the size of the intermediate feature map channels is C_{mid}, the number of parameters during the bottleneck operation is shown in

$$P_{\text{BottleNeck}} = C_{\text{in}} \times 1 \times 1 \times C_{\text{mid}} + C_{\text{mid}} \times 3 \times 3 \times C_{\text{mid}} + C_{\text{mid}} \times 1 \times 1 \times C_{\text{out}}. \tag{2}$$

From the calculation of the input and the output feature map sizes in Figure 1, the number of parameters

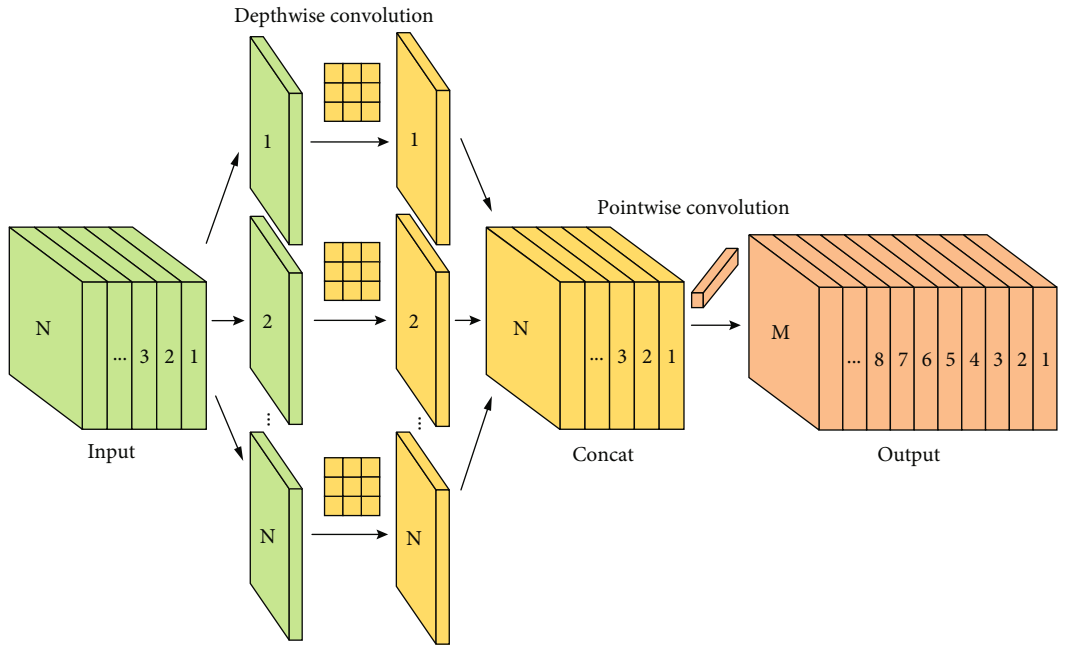


Figure 2: Structure of depthwise separable convolution.

generated by the regular convolution operation process can be obtained:

$$P_{\text{Conv}} = 96 \times 3 \times 3 \times 96. \tag{3}$$

The number of parameters generated by the bottleneck layer is:

$$P_{\text{BottleNeck}} = 96 \times 48 + 48 \times 9 \times 48 + 48 \times 96. \tag{4}$$

By comparing the two, the number of parameters generated during the bottleneck operation is greatly reduced.

2.3. Depthwise Separable Convolution. Depthwise separable convolution [23] is the core structure of the lightweight network MobileNet [24, 25], which is a combination of two parts: depthwise convolution and pointwise convolution. The specific structure is shown in Figure 2. Depthwise separable convolution contains a lower number of parameters and lower computational cost than the conventional convolution operation process. The number of convolution kernels in depthwise convolution is the same as the number of channels in the previous layer, and one convolution kernel is responsible for one channel. The number of channels in the feature map generated by this process is the same as the number of input channels, which cannot extend the dimensionality of the feature map, and the convolution operation for each channel independently cannot effectively use the feature information of different channels at the same spatial location. Pointwise convolution [26] mainly uses a 1×1 convolution to combine the feature maps obtained in the previous step in a weighted manner in the depth direction.

We assume that the input feature map size is $W_i \times H_i \times C$; W_i, H_i, and C represent the width, height, and the number of channels of the input feature map, respectively. The standard convolution size is $W_c \times H_c \times C \times H$, which denotes the width, height, the number of channels, and the number of convolution kernels of the conventional convolution, respectively, and the size of the output feature map after the conventional convolution operation is $W_o \times H_o \times N$. Then, the computation of the regular convolution is:

$$F_{\text{Conv}} = W_o \times H_o \times N \times W_c \times H_c \times C. \tag{5}$$

The depthwise separable convolution first uses a convolution size of $W_c \times H_c \times C \times 1$ convolution for depthwise convolution and then uses $1 \times 1 \times C \times N$ of the convolution for pointwise convolution. The depthwise separable convolution is computed as:

$$F_{\text{DSC}} = W_o \times H_o \times 1 \times W_c \times H_c \times C + W_o \times H_o \times N \times 1 \times 1 \times C. \tag{6}$$

The ratio of the two is:

$$\frac{F_{\text{DSC}}}{F_{\text{Conv}}} = \frac{1}{N} + \frac{1}{W_c \times H_c}. \tag{7}$$

If the size of the input feature map is $48 \times 48 \times 96$ and the size of the convolution kernel is $3 \times 3 \times 96$, then the ratio of the parameter computation is $(1/96) + (1/9)$. Therefore, if the depthwise separable convolution is used instead of the regular convolution, the computation is reduced by a factor of nearly 9.

2.4. Wide Residual Neural Network Model. To resolve the problem of gradient disappearance caused by increasing depth in deep neural networks, a residual learning unit is

introduced to more easily optimize deep networks by adjusting the relationship between the input and output through constant mapping. In the ResNet residual learning unit, the neural network input is x, while the best mapping is $H(x)$, $F(x)$ denotes ResNet Function, after the nonlinear convolution layer to achieve $F(x) = H(x) - x$, the constant mapping of itself is expressed as $H(x) = F(x) + x$. This constant mapping can then reduce the complexity and the computation of the model and, to a certain extent, mitigate problems such as gradient disappearance caused by stacking with the number of layers. However, the deep residual network pursues network depth too much, and the performance of the model does not improve considerably as the number of modules increases. The Wide ResNet residual learning module [27] adds a factor k to the original residual module to widen the number of convolution kernels [28], which reduces the number of layers, where k denotes the number of multiples of filters in the convolution layer. However, it does not reduce the model parameters, and it speeds up the computation, making it easier for the stacking layer to learn new features from the input image features.

The residual learning unit is shown in Figure 3, where dropout regularization prevents overfitting of the model and ReLU denotes the activation function; a is the ResNet residual learning module, and b is the Wide ResNet residual learning module.

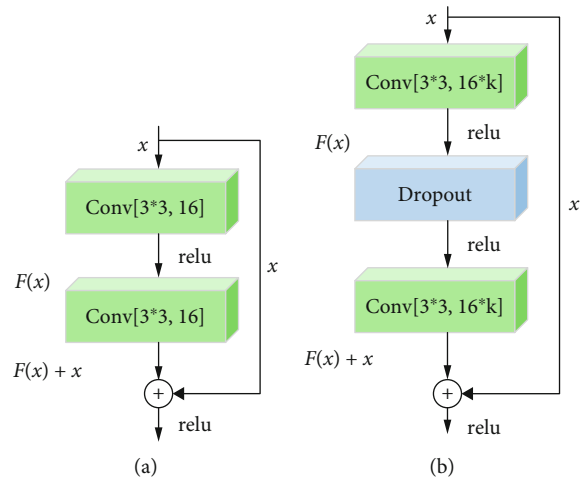


Figure 3: Learning module.

3. Methods

3.1. Overall Architecture. Since too few layers of a fully connected neural network will lead to insufficient feature representation of facial expressions in the model, too many layers will increase the computation of the network and cause the problem of network redundancy. This paper combines the above problems and designs a multiscale feature fusion attention lightweight facial expression recognition network. In the image preprocessing stage, noise is added to the training set by an improved random erasing method, which enhances the robustness of the model while enriching the entire dataset.

Then, the preprocessed facial expression images are passed into the network. First, the number of parameters of the model is reduced by the depthwise separable shuffle module to speed up the computing speed of the network. The SCAM is embedded in the middle, and then, the network is characterized by the grouping bottleneck module to reduce the dimensionality of the network, which saves the computation of the network and increases the nonlinear expression capability of the model. Then, it passes through the depthwise separable shuffle module and finally enters the Softmax layer to classify the output results. The overall architecture of the model is shown in Figure 4.

The input object image size of this network is 48×48, and the number of channels is 3. Each convolutional layer is followed by a BN layer and a ReLU activation function layer. The BN layer accelerates the training and convergence of the network and prevents the gradient from disappearing to a certain extent. To improve the feature representability of the network, a SCAM is provided behind each packet bottleneck module and depthwise separable shuffle module. After entering the grouping bottleneck module, the dimensionality of the output is halved, the number of channels is doubled, the risk of overfitting is reduced, and the parameters of the computation are reduced by the Global-Ave-Pooling layer. Finally, the pictures are classified by the Softmax layer, and the categories contain a total of 7 categories: angry, disgusted, scared, happy, sad, surprised, and neutral. The model parameters are shown in Table 1.

3.2. Image Preprocessing. Data enhancement is a common method in the image preprocessing stage, which mitigates the overfitting of the model and improves its generalizability to a certain extent. This paper expands the training set samples and enhances the robustness of the model by adding a small amount of noise to the images through an improved random erasing method [29].

First, in the preprocessing stage, the probability of random erasing of the object image is set as p, the area of the original image is set as S, the minimum and the maximum thresholds of the random erasing image are set as S_l and S_h, respectively, the aspect ratio of the occlusion matrix is set as r_e, the area of random erasing is set as S_e, the height of the area of the random erasing matrix is set as H_e, and the width of the area of the random erasing matrix is set as W_e. An example of the random erasing formula is as follows:

$$S_e = S \times \text{Random}(S_l, S_h), \tag{8}$$

$$H_e = \sqrt{S_e \times r_e}, \tag{9}$$

$$W_e = \sqrt{\frac{S_e}{r_e}}. \tag{10}$$

Among them, the specific parameters of random erasing are set, as shown in Table 2.

A randomly selected point $P_e = (x_e, y_e)$ on the image, x_e and y_e, is bounded by the following example, where W

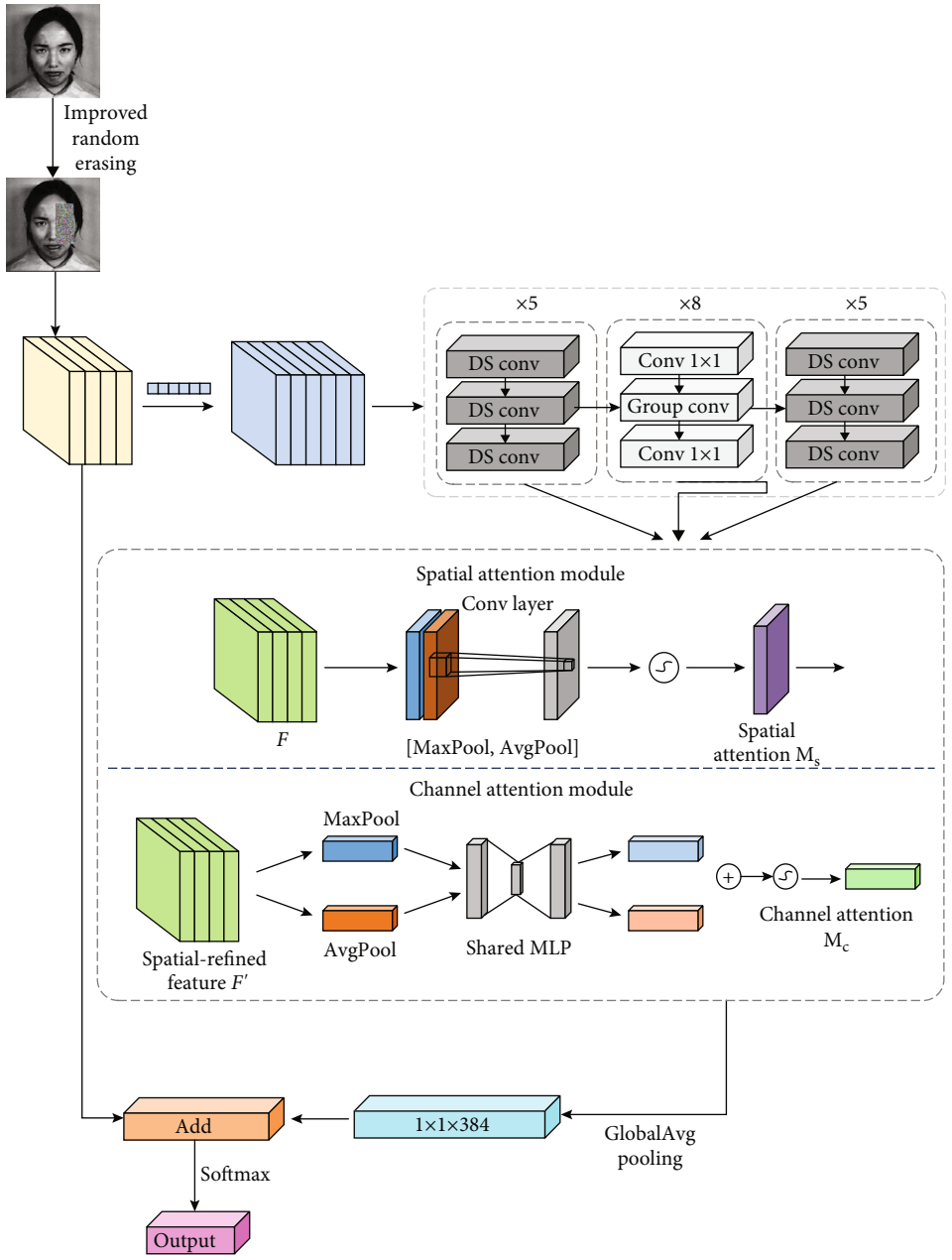


Figure 4: Overall architecture.

indicates the width of the image and H indicates the height of the image.

$$\begin{Bmatrix} x_e = \text{random}(0, W) \\ y_e = \text{random}(0, H) \end{Bmatrix}. \tag{11}$$

Since the background noise of the facial expression pictures affects the accuracy of recognition and the random erasing processing does not necessarily cover the facial expression region, causing redundancy in the original dataset, the random erasing method is improved to ensure that the random erasing region must be at the face location, and the coordinate values of x_e and y_e are requalified, for example, as follows:

$$\begin{Bmatrix} x_e = \text{random}(R_x, R_{xl}) \\ y_e = \text{random}(R_y, R_{yl}) \end{Bmatrix}, \tag{12}$$

where R_x, R_y, R_{xl}, and R_{yl} denote the true coordinate values of the upper left vertex and the upper right vertex of the face image range, respectively. By limiting the selection range of the random point P_e points so that each random erasing can cover the facial expression range, the random erasing

Table 1: Model parameters.

Light-NTWRN				
Type	Filters	Size	Output	Repetition
Input	—	—	48 × 48 × 3	—
Conv	3 × 3	16	48 × 48 × 16	—
BN+ReLU	—	—	48 × 48 × 16	—
DS-1	3 × 3	96	48 × 48 × 96	5
BN+ReLU			48 × 48 × 96	
DS-2	3 × 3	96	48 × 48 × 96	
BN+ReLU+dropout			48 × 48 × 96	
DS-3	3 × 3	96	48 × 48 × 96	
BN+ReLU+SCAM			48 × 48 × 96	
Conv-1	1 × 1	192	24 × 24 × 192	8
BN+ReLU			24 × 24 × 192	
GConv-1	3 × 3	192	24 × 24 × 192	
BN+ReLU+dropout			24 × 24 × 192	
Conv-2	1 × 1	192	24 × 24 × 192	
BN+ReLU+SCAM			24 × 24 × 192	
DS-4	3 × 3	384	12 × 12 × 384	5
BN+ReLU			12 × 12 × 384	
DS-5	3 × 3	384	12 × 12 × 384	
BN+ReLU+dropout			12 × 12 × 384	
DS-6	3 × 3	384	12 × 12 × 384	
BN+ReLU+SCAM			12 × 12 × 384	
GlobalAvg pooling	—	—	1 × 1 × 384	—
Softmax	—	—	1 × 1 × 7	—

Table 2: Random erasing parameters.

Parameter	Value
p	0.5
S_l	0.05
S_h	0.3
r_e	0.3

method and the improved method are compared, as shown in Figure 5.

As seen from Figure 5, the improved method can ensure that each random erasing is within the range of facial expressions, artificially extends the dataset of training samples, improves the robustness of the model, and effectively reduces the risk of model overfitting.

3.3. Spatial Channel Attention Module (SCAM). To further extract the deep features of different facial expressions and improve the accuracy of facial expression recognition, this paper improves the lightweight attention module (convolutional block attention module) proposed by Woo et al. [30]. This is a simple and effective attention module for convolutional neural networks. Given an intermediate feature map, our module sequentially generates attention maps along two separate dimensions, channel and space, and then multiplies the attention map into the input feature map for adaptive feature refinement. Because SCAM is a lightweight, general-purpose module, it can be seamlessly integrated into any CNN architecture with negligible computational cost. Since convolutional operations extract information features by mixing cross-channel and spatial information, we use our modules to emphasise features that are meaningful in these two main dimensions: the channel and the spatial axis. To achieve this, we apply the channel and spatial attention modules in turn, so that each branch can learn what and where to pay attention to on the channel and spatial axes, respectively. Our modules thus effectively aid the flow of information in the network by learning which information needs to be emphasised or suppressed. The features of the object image are represented in two dimensions, spatial and channel, first by the spatial attention module and then by the channel attention module, and finally, the generated features are obtained. The structure of the SCAM proposed in this paper is shown in Figure 6.

The proposed SCAM contains two independent submodules, the spatial attention module and the channel attention module, which perform feature extraction on space and channels. The input feature map F is passed through the two attention modules first, and then, the final features are output F'', $M_s(F)$ indicates that the feature map F has passed the spatial attention mechanism, $\otimes$ is multiplied by the corresponding element, and F' indicates the output feature map after passing the spatial attention mechanism; $M_c(F')$ indicates that the feature map F has passed the channel attention mechanism, and F'' indicates the output feature map after passing the SCAM attention mechanism, as shown in the following example:

$$F' = M_s(F) \otimes F, \tag{13}$$

$$F'' = M_c\left(F'\right) \otimes F'. \tag{14}$$

(1) Spatial attention module

In the process of facial expression recognition, different expressions are associated with specific regions. Moreover, an overall facial expression consists of several regions, and more attention needs to be paid to the local features with the highest expression relevance. The SCAM is shown in Figure 7.

First, the input feature map will perform global max pooling and global average pooling, followed by a CONCAT operation based on the channel and a 7×7 convolutional dimensionality reduction, and finally, it will generate the spatial attention feature by sigmoid normalization, where $\mathrm{MaxPool}(F)$ denotes the global max pooling, $\mathrm{AvgPool}(F)$ denotes the global average pooling, $f^{7\times7}$ denotes the convolution kernel for 7×7 size, σ is the sigmoid function, and

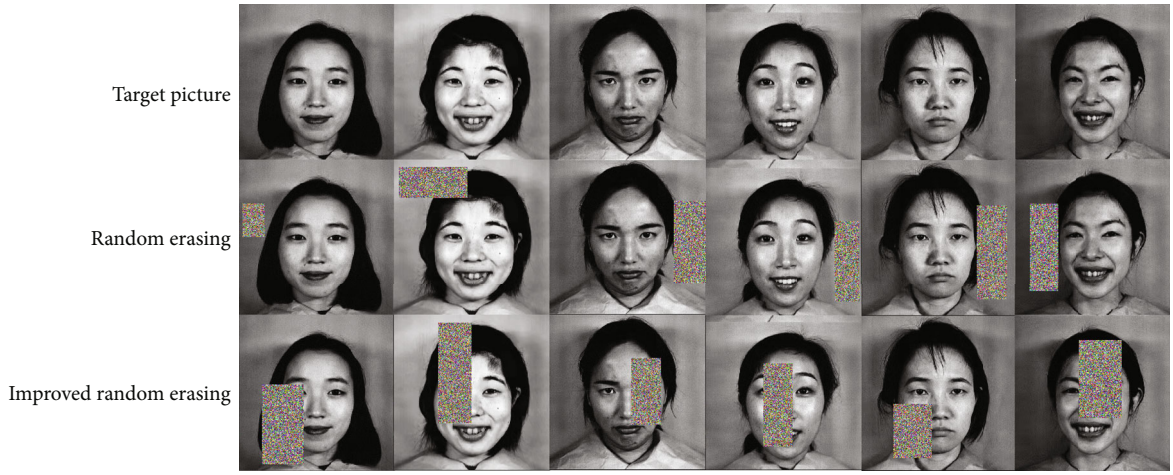


Figure 5: Comparison chart of the random erasing experiment.

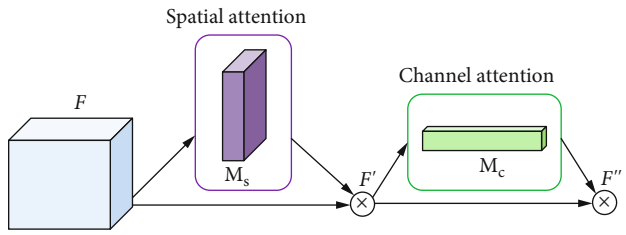


Figure 6: Scam overall attention module.

$M_s(F)$ is the output feature map after passing the spatial attention mechanism. The example is as follows:

$$M_s(F) = \sigma\left(f^{7\times7}([\text{AvgPool}(F); \text{MaxPool}(F)])\right). \quad (15)$$

(2) Channel attention module

To represent the feature information of facial expressions in multiple dimensions, the feature maps output by the spatial attention module are used as the input of this module, based on global max pooling and global average pooling of width and height, respectively, and the two obtained features are fed into a neural network composed of hidden layers and a multilayer perceptron (MLP). Then, the final features are merged and output using element-by-element summation, as follows:

$$M_c(F) = \sigma(\text{MLP}(\text{AvgPool}(F)) + \text{MLP}(\text{MaxPool}(F))). \quad (16)$$

3.4. Grouping Bottleneck Method. In this paper, the grouping bottleneck is improved based on the group convolution method, and its specific structure is shown in Figure 8, which consists of 1×1 and 3×3 convolutions, where the number of convolution kernels in the first layer of 1×1 is half of the number of input feature map channels, and the reduction in the number of convolution kernels can reduce the number of network parameters. The size of the input feature map of the bottleneck block is $48\times48\times96$, and the size of the feature map is $24\times24\times192$ after the $1\times1\times192$ convolution. A 1×1 convolution reduces the number of parameters of the network by half while deepening the network to extract deep semantic information, which substantially reduces the subsequent convolution computation. The second layer of the bottleneck block is a 3×3 convolution as a group convolution layer, the number of convolution kernels of group convolution is the number of channels of the input feature map, the feature map of $24\times24\times192$ is divided into 192 feature maps of $24\times24\times1$ by channel, the features are extracted using 192 convolution kernels of 3×3, the corresponding element positions of the input and output feature maps are summed in pairs to obtain the final feature map, and the method of summing corresponding elements can solve the network degradation problem to some extent. Since the second layer of the original structure is a 3×3 convolutional structure changed to a 3×3 grouped convolutional structure, it can reduce the number of parameters of the network, reduce the complexity of the model, and improve the computational speed of the network, because when the ordinary convolutional operation is performed, the input feature map size is $C\times H\times W$ and there are N convolutional kernels, then the output feature map, and the number of convolutional kernels. The size of each convolutional kernel is $C\times K\times K$, and the total number of parameters of N convolutional kernels is $N\times C\times K\times K$.

Grouped convolution groups the input feature maps and then convolves each group separately, if the input feature map size is $C\times H\times W$ and the number of output feature maps is N; if we set to divide into G groups, the number of input feature maps of each group is C/G, then the number of output feature maps of each group map is N/G, the size of each convolutional kernel is $(C/G)\times K\times K$, the total number of convolutional kernels is still N, the number of convolutional kernels in each group is N/G, the convolutional kernels only convolve with the input map of the same

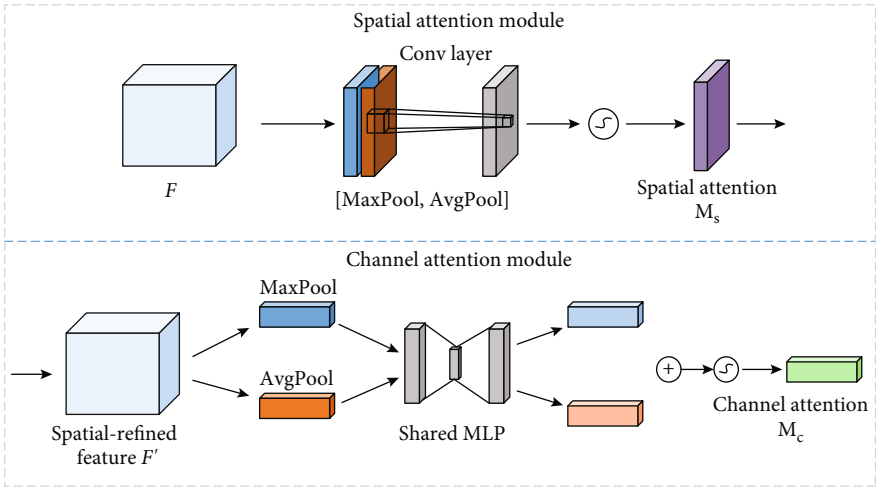


Figure 7: SCAM.

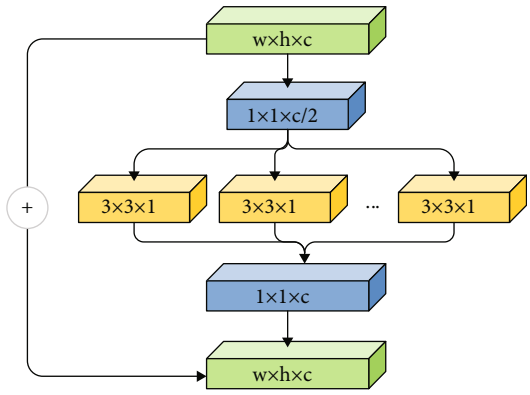


Figure 8: Grouping bottleneck.

group with them, and the total number of convolutional kernels is $N \times (C/G) \times K \times K$, so the total number of parameters is the original $1/G$.

The number of parameters of the bottleneck and the grouping bottleneck is shown in Table 3, and it was found that the grouping bottleneck block has a substantial decrease in the number of parameters compared with the original bottleneck block, with a ratio of nearly 1/10. The nonlinear expression capability of the model is increased.

3.5. Depthwise Separable Shuffle Method. In this paper, channel shuffling is used to improve the depthwise separable convolution [31, 32], and its structure is shown in Figure 9. The depthwise separable convolution first uses depthwise convolution to process the input feature map, and different channels use different convolution operations and then use the CONCAT method for channel stitching. Thus, the final output features are derived from only part of the input channel features, and there is no information exchange between the different channels, which leads to the limited characterizability of the extracted features. Although the depthwise separable convolution uses pointwise convolution to further

Table 3: Comparison of the number of participants before and after bottleneck improvement.

Module	Number of participants (pcs)	Ratio
3 × 3Conv	580327	1.78
Bottleneck	324873	1
Grouping bottleneck	33257	0.10

increase the dimensionality of features, which can enhance the communication of spatial feature information to a certain extent, the increase in dimensionality leads to an increase in the number of network parameters. Deep separable convolution is divided into deep convolution operation and point-by-point convolution operation. In the deep convolution operation, if the input feature dimension is $D_F \times D_F \times M$, M is the number of channels, and the parameter of the convolution kernel is $D_k \times D_k \times 1 \times M$, the output feature dimension after deep convolution is $D_F \times D_F \times M$. Each channel only corresponds to one convolution kernel when convolving, so the FLOPs are $M \times D_F \times D_F \times D_k \times D_k$. In the point-by-point convolution operation, the input is the feature after deep convolution, the dimension is $D_F \times D_F \times M$, the parameter of convolution kernel is $1 \times 1 \times M \times N$, the output dimension is $D_F \times D_F \times M$, the convolution process does 1×1 standard convolution for each feature, and the FLOPs are $N \times D_F \times D_F \times M$. In this paper, the point-by-point convolution operation is replaced by the channel shuffle method, which reduces the number of parameters of the point-by-point convolution operation. The channel shuffle method has the same function as the point-by-point convolution method; however, the number of parameters of the whole network does not increase because its dimensionality does not change, while the characterizability of the features is enhanced, which reduces the complexity of the network to some extent improves the training speed of the model, which can improve the whole network's face recognition accuracy.

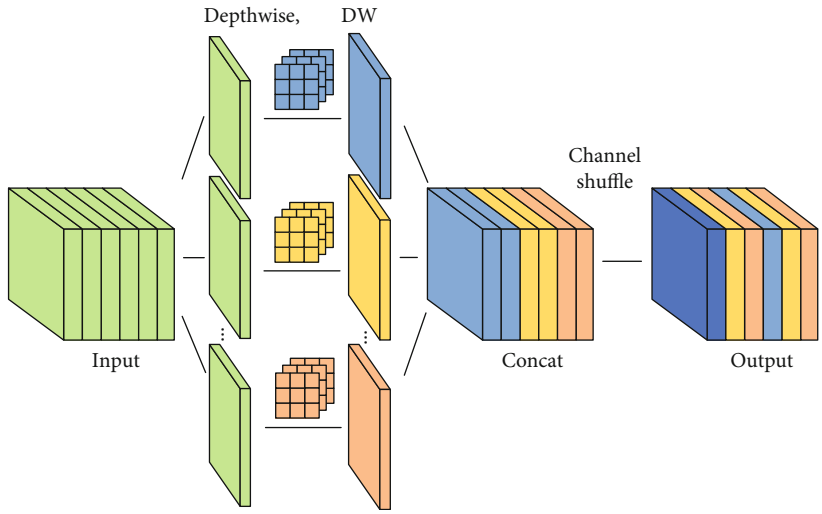


Figure 9: Depthwise separable shuffle method.

4. Experiment and Analysis

4.1. Experiment Preparation. To verify the accuracy and the effectiveness of the Light-NTWRN network model proposed in this paper, the light-NTWRN network model is subjected to comparative ablation experiments on the FER2013, CK+ dataset, and JAFFE dataset. The experiment is based on the TensorFlow deep learning framework for training, and testing is conducted on Pycharm with the following hardware environment configuration: Win10 operating system, Intel Core i7-10700F with 2.9 GHz CPU and 16 G RAM and NVIDIA GeForce RTX 3070 (8 GB) graphics card. During the experiments, 70% of the facial expression images are randomly selected as the training set, and 30% of the facial expression images are randomly selected as the test set. Additionally, the experimental parameters are set as shown in Table 4.

Table 4: Experimental parameter settings.

Parameter	FER2013	CK+	JAFFE
Optimizer	SGD	SGD	SGD
Momentum	0.9	0.9	0.9
Batch size	30	20	40
Learning rate	0.01	0.01	0.01
Learning rate decay	0.5/50	0.5/50	0.5/50
Loss function	Cross entropy	Cross entropy	Cross entropy
Epochs	300	300	300

4.2. Facial Expression Dataset. The FER2013 facial expression dataset consists of 35,886 facial expressions, and the dataset is expanded to 80,000 by an improved random erasing method where the training set contains 56,000 and the test set contains 24,000, and each image is composed of a grayscale image with a fixed size of 48×48, which contains a total of 7 expressions, namely, angry, disgusted, fear, happy, sad, surprised, and neutral. The facial expression images of FER2013 are more difficult to recognize because of the interference of occlusion, pose, low contrast, and background.

CK+ is expanded from the Cohn-Kanda dataset, which contains a total of 123 participants, 593 image sequences, and a total of 7 expressions. The CK+ dataset acquisitions are all collected under the same lighting background, the acquisition environment is better, and the dataset is expanded to 1500 images through an improved random erasing method, with 70% of the training set and 30% of the test set.

The JAFFE dataset was selected from 10 Japanese female students who each made 7 different expressions, consisting of a total of 213 photos, which were expanded to 3408 photos by rotation, flip, contrast enhancement, panning, cropping, scaling, and improved random erasing methods.

4.3. Ablation Experiment. To verify the effectiveness of the Light-NTWRN network model proposed in this paper, ablation experiments are conducted for each module, and the experimental results are shown in Table 5. WRN denotes the improved wide residual network, RE denotes the improved random erasing method, SCAM denotes the improved attention mechanism module, GBN denotes the grouping bottleneck method, and DS denotes the depthwise separable shuffle method, where WRN+RE+SCAM+GBN+DS denotes the Light-NTWRN network proposed in this paper.

First, the facial expression images are input into the model after the improved random erasing operation, and for the model to acquire more local features of facial expressions, the improved SCAM is embedded into the network to reassign the feature weights of facial expressions from both the channel and space dimensions. The grouping bottleneck method is improved to solve the problem of model redundancy caused by too many convolutional layers. To reduce the number of parameters computed by the network and speed up the network operation, an improved depthwise

Table 5: Light-NTWRN network ablation experiments.

Model	FER2013 (%)	CK+ (%)	JAFFE (%)	Parameter (M)
WRN	69.60%	95.38%	91.42%	18.35
WRN+RE	71.05%	97.54%	93.88%	18.35
WRN+RE+SCAM	72.27%	98.35%	94.28%	23.21
WRN+RE+SCAM+GBN	72.58%	98.60%	94.87%	15.72
Light-NTWRN (ours)	73.21%	98.72%	95.21%	10.14

separable shuffle method is added. To verify the effectiveness of each improved module, the Light-NTWRN network ablation experiments are shown in Table 5.

The ablation experiments are shown in Figure 10, where part a represents the FER2013 ablation experiment, part b represents the CK+ ablation experiment, and part c represents the JAFFE ablation experiment. According to the ablation experiments of the FER2013 dataset in part a, we can see that Light-NTWRN has the fastest convergence rate, and the model recognition accuracy grows slowly when trained to 100 epochs and gradually levels off when trained to 210 epochs. The accuracy gradually levels off, and the highest accuracy reaches 73.21%.

From the ablation experiments of the CK+ dataset in part b of Figure 10, it can be seen that the accuracy of the model increases rapidly at the beginning of training, and the accuracy of the model recognition oscillates up and down from the 50th epoch to the 100th epoch. When the training reaches 150 epochs, the accuracy tends to be stable, and the highest accuracy can reach 98.72%. From the ablation experiments of the JAFFE dataset in part c, we can see that the accuracy of the model also grows faster at the beginning of training, and when, the training reaches 180 epochs, the accuracy tends to be stable, and the highest accuracy can reach 95.21%. From the dataset, it was found that the accuracy of the model is improved after adding SCAM, but there is a slight loss of its network operation speed. GBN and DS can effectively reduce the number of network parameters and improve the accuracy of the model. Furthermore, the accuracy of the model proposed in this paper on the three datasets FER2013, CK+, and JAFFE is improved by 3.61%, 3.34%, and 3.79%, respectively, compared with that of the original model, and the number of parameters is reduced by 44.74% compared to that for the original network, which proves that the proposed model has better effectiveness and faster computing speed.

To further verify the effectiveness and the robustness of the proposed model in this paper, the confusion matrix experiments are shown in Figure 11, where part a represents the confusion matrix on the FER2013 dataset, part b represents the confusion matrix on the CK+ dataset, and part c represents the confusion matrix on the JAFFE dataset.

From the confusion matrix on the FER2013 dataset in part a, we can see that the recognition accuracy of the three categories of anger, fear, and sadness is low because the activities of these three categories of facial expressions are less obvious, and the feature points are difficult to extract. The recognition performance of each category on the CK+ dataset is better, and the accuracy is higher. On the JAFFE dataset, the recognition accuracy of the anger and disgust categories is lower because the misidentified samples all belong to the negative category of emotions, which are more similar, facial features are difficult to extract, so recognition is more challenging.

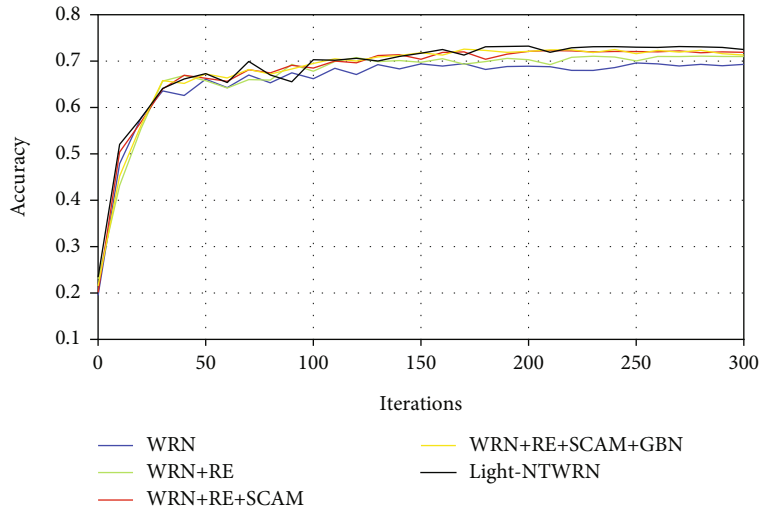
4.4. Mainstream Algorithm Comparison Experiment. To verify the effectiveness of the Light-NTWRN algorithm proposed in this paper for facial expression recognition, comparison experiments are conducted with five mainstream algorithms, mainly AlexNet, VGG16, VGG19, ResNet18, and ResNet50, to compare the size of the number of parameters and the specific recognition accuracy on the three datasets, and the specific results are shown in Table 6.

The Light-NTWRN algorithm proposed in this paper has the highest accuracy for facial expression recognition on the FER2013 dataset, with an improvement of nearly 2% compared to the ResNet50 model in the mainstream algorithm. From the experimental results on the CK+ dataset, the recognition accuracy of the VGG16 model is the highest among the mainstream networks, while the recognition accuracy of the model proposed in this paper is improved by 3.26% compared to the VGG16. The recognition accuracy on the JAFFE dataset is as high as 95.21%.

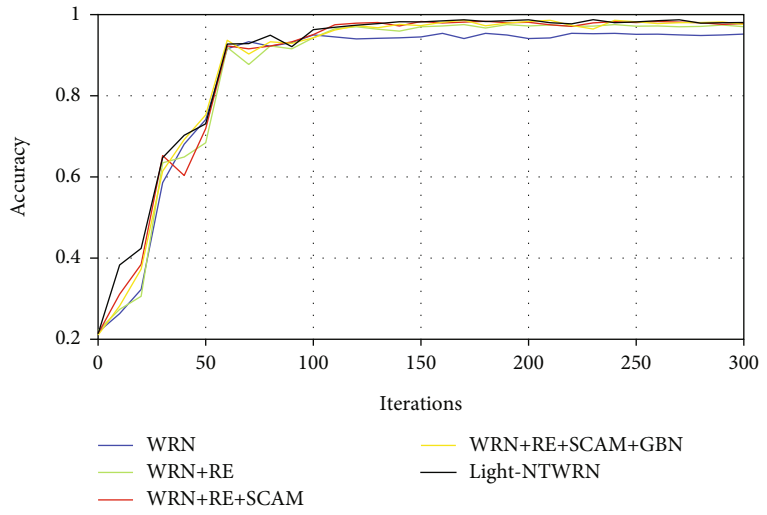
This can further verify the effectiveness of the three improved methods proposed in this paper, which can improve the recognition accuracy. Compared with the other five mainstream algorithms, the Light-NTWRN algorithm proposed in this paper has the highest accuracy and the best algorithm performance in terms of facial expression recognition and has strong generalization.

In terms of the number of model parameters, the number of model parameters of the network proposed in this paper is 10.14 M, which is the lowest compared with the other five mainstream algorithms and can maintain high recognition accuracy, which verifies the advanced and excellent model. It also further verifies the effectiveness of the three improvement methods proposed in this paper for model lightweighting.

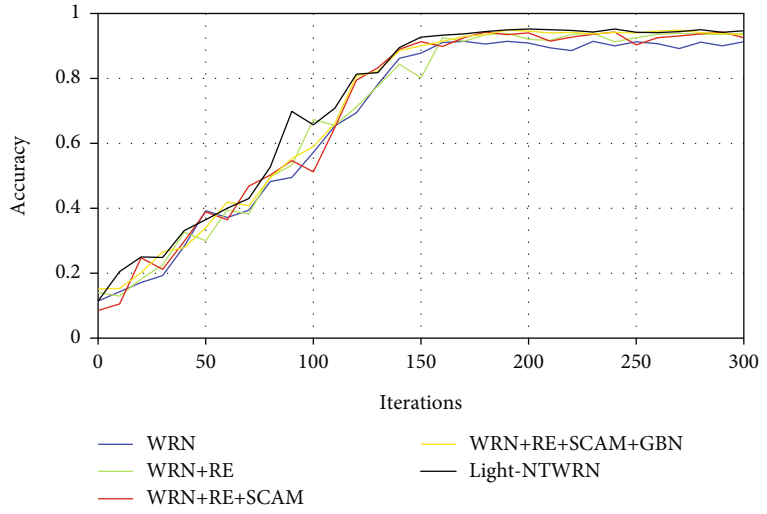
We compare the proposed method in this paper with other existing methods. The existing more advanced methods are MANet [33], a model that obtains key region features by adaptively learning weights; Minaee [34], a model that assigns residual blocks to spatial mask information; WMDCNN [35], a model that mixes two-channel weighting of static images; and APRNET50 [36], a model that uses multiscale feature extraction blocks instead of residual units. The



(a) FER2013 ablation experiment

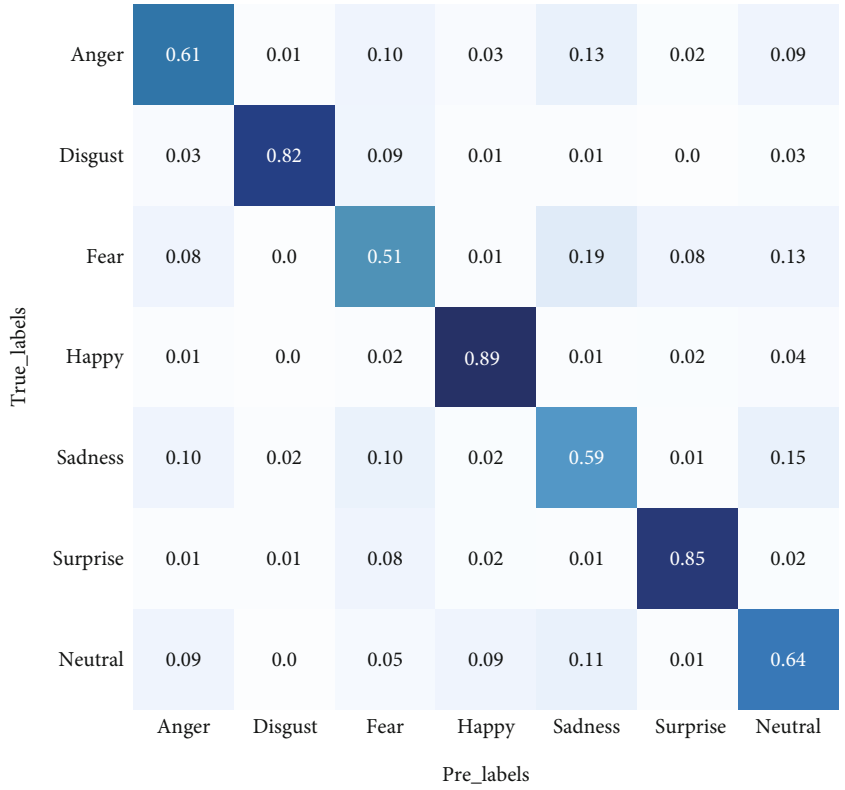


(b) CK+ ablation experiment



(c) JAFFE ablation experiment

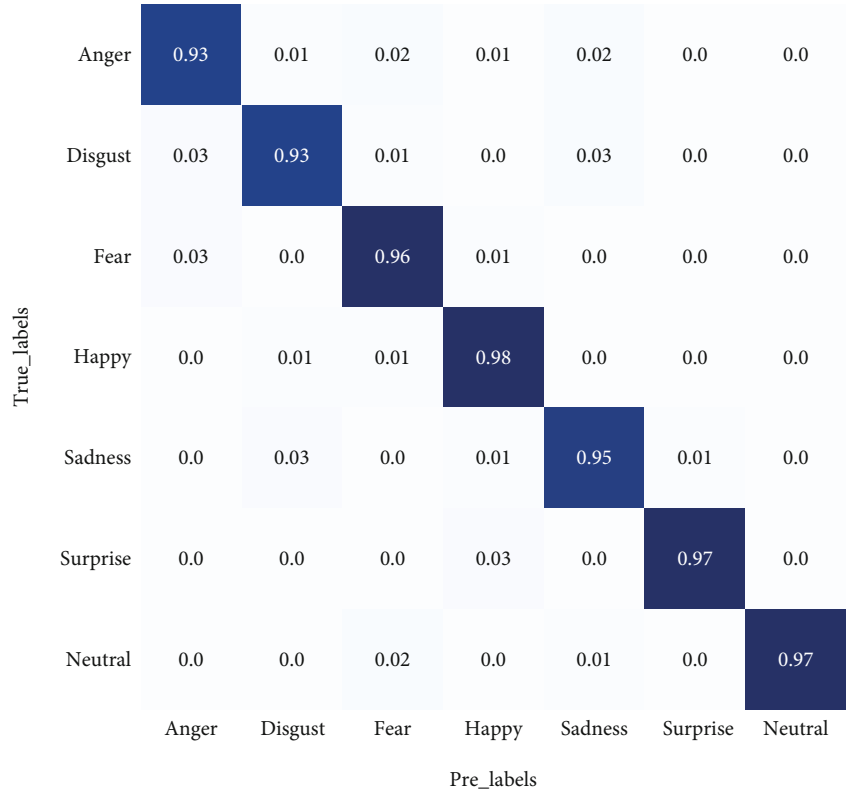
Figure 10: Ablation experiment.



(a) Confusion matrix on the Fer2013 dataset

(b) Confusion matrix on the CK+ dataset

Figure 11: Continued.



(c) Confusion matrix on the JAFFE dataset

Figure 11: Confusion matrix.

Table 6: Comparison experiments of mainstream algorithms.

Model	FER2013 (%)	CK+ (%)	JAFFE (%)	Parameter (M)
AlexNet	67.51	87.59	89.83	60.92
VGG16	68.89	95.46	91.04	14.75
VGG19	68.53	92.18	90.37	20.06
ResNet18	70.09	89.39	92,55	11.69
ResNet50	71.26	92.46	93.08	25.56
Light-NTWRN (ours)	73.21	98.72	95.21	10.14

Table 7: Recognition rates of various algorithms on the facial expression dataset.

Model	FER2013 (%)	CK+ (%)	JAFFE (%)
MANet [33]	69.46	96.28	—
Minaee [34]	70.20	98.00	92.80
WMDCNN [35]	—	98.50	92.30
APRNET50 [36]	73.00	94.95	94.80
Light-NTWRN (ours)	73.21	98.72	95.21

comparison is performed on the FER2013, CK+, and JAFFE datasets. It can be seen in Table 7 that the model proposed in this paper has the highest accuracy, and the effectiveness of the model proposed in this paper can be proven by the above experiments.

5. Conclusion

This paper proposes a multiscale feature fusion attention lightweight facial expression recognition method that effectively suppresses the influence of irrelevant feature information on the model while slowing the gradient disappearance caused by too many layers of the neural network, thus reducing the number of parameters computed by the network and improving the computational speed of the model. The improved SCAM module focuses more on feature information to speed up the convergence of the model and improve its performance. The improved random erasing method expands the training set while enhancing the robustness of the model to noise. The grouping bottleneck method reduces the dimensionality of the target image while increasing the nonlinear expression capability of the model. In addition, the depthwise separable shuffle method reduces the number

of parameters computed by the network while speeding up the computational speed of the network. The accuracy of the proposed model (Light-NTWRN) is 73.21% on the FER2013 dataset, 98.72% on the CK+ dataset, and 95.21% on the JAFFE dataset, while having a lower number of parameters, and the experimental results are better than many current mainstream algorithms, showing better effectiveness and robustness. However, the recognition accuracy is still not high enough in the case of obscured facial expressions, and more attention should be given to the recognition performance of these datasets in the future.

Data Availability

The raw/processed data required to reproduce these findings cannot be shared at this time as the data also forms part of an ongoing study.

Conflicts of Interest

There is no conflict of interest regarding the publication of this paper.

Acknowledgments

The work was supported by the Key Research Project of Science and Technology of Chongqing Education Commission (no. kjzd-k201801901) and Chongqing Postgraduate Research and Innovation Project (CYS22663).

References

[1] P. E. Jianqing, W. U. Haoxuan, L. I. Tianliang, and H. A. Yu, "Workspace, stiffness analysis and design optimization of coupled active-passive multilink cable-driven space robots for on-orbit services," *Chinese Journal of Aeronautics*, 2022.

[2] K. Hambuchen, J. Marquez, and T. Fong, "A review of NASA human-robot interaction in space," *Current Robotics Reports*, vol. 2, no. 3, pp. 265–272, 2021.

[3] Q. Gao, X. Zhang, and W. Pang, "Fast and accurate hand visual detection by using a spatial-channel attention SSD for hand-based space robot teleoperation," *International Journal of Aerospace Engineering*, vol. 2022, Article ID 3396811, 11 pages, 2022.

[4] L. Yingxiao, H. Ju, M. Ping, and R. Jiang, "Target localization method of non-cooperative spacecraft on on-orbit service," *Chinese Journal of Aeronautics*, 2022.

[5] X. L. Ding, Y. C. Wang, Y. B. Wang, and K. Xu, "A review of structures, verification, and calibration technologies of space robotic systems for on-orbit servicing," *SCIENCE CHINA Technological Sciences*, vol. 64, no. 3, pp. 462–480, 2021.

[6] R. R. Santos, D. A. Rade, and I. M. da Fonseca, "A machine learning strategy for optimal path planning of space robotic manipulator in on-orbit servicing," *Acta Astronautica*, vol. 191, pp. 41–54, 2022.

[7] P. Rousso, S. Samsam, and R. Chhabra, "A mission architecture for on-orbit servicing industrialization," in *2021 IEEE Aerospace Conference (50100)*, pp. 1–14, Big Sky, MT, USA, 2021.

[8] J. Xing and J. Zhong, "MiniExpNet: a small and effective facial expression recognition network based on facial local regions," *Neurocomputing*, vol. 462, pp. 353–364, 2021.

[9] X. Sun, P. Xia, and F. Ren, "Multi-attention based deep neural network with hybrid features for dynamic sequential facial expression recognition," *Neurocomputing*, vol. 444, pp. 378–389, 2021.

[10] Y. Wenmeng and X. Hua, "Co-attentive multi-task convolutional neural network for facial expression recognition," *Pattern Recognition*, vol. 123, p. 108401, 2022.

[11] S. Cuiping, T. Cong, and W. Liguo, "A facial expression recognition method based on a multibranch cross-connection convolutional neural network," *IEEE ACCESS*, vol. 9, pp. 39255–39274, 2021.

[12] Y. Kong, Z. Ren, K. Zhang, S. Zhang, Q. Ni, and J. Han, "Lightweight facial expression recognition method based on attention mechanism and key region fusion," *Journal of Electronic Imaging*, vol. 30, no. 6, article 063002, 2021.

[13] N. Zhou, R. Liang, and W. Shi, "A lightweight convolutional neural network for real-time facial expression detection," *IEEE Access*, vol. 9, pp. 5573–5584, 2020.

[14] Z. Niu, G. Zhong, and H. Yu, "A review on the attention mechanism of deep learning," *Neurocomputing*, vol. 452, pp. 48–62, 2021.

[15] Y. Chen, L. Liu, V. Phonevilay et al., "Image super-resolution reconstruction based on feature map attention mechanism," *Applied Intelligence*, vol. 51, no. 7, pp. 4367–4380, 2021.

[16] H. Zhang, G. Peng, Z. Wu, J. Gong, D. Xu, and H. Shi, "MAM: a multipath attention mechanism for image recognition," *IET Image Processing*, vol. 16, no. 3, pp. 691–702, 2022.

[17] L. Yao, S. He, K. Su, and Q. Shao, "Facial expression recognition based on spatial and channel attention mechanisms," *Wireless Personal Communications*, vol. 125, no. 2, pp. 1483–1500, 2022.

[18] H. Wang and H. Zhang, "Adaptive target tracking based on channel attention and multi-hierarchical convolutional features," *Pattern Analysis and Applications*, vol. 25, no. 2, pp. 305–313, 2022.

[19] Z. Qiu, S. I. Becker, and A. J. Pegna, "Spatial attention shifting to emotional faces is contingent on awareness and task relevancy," *Cortex*, vol. 151, pp. 30–48, 2022.

[20] C. Chen, D. Gong, H. Wang, Z. Li, and K. Y. K. Wong, "Learning spatial attention for face super-resolution," *IEEE Transactions on Image Processing*, vol. 30, pp. 1219–1231, 2021.

[21] Z. Xue, T. Li, S. T. Peng, C. Y. Zhang, and H. C. Zhang, "A data-driven method to predict future bottlenecks in a remanufacturing system with multi-variant uncertainties," *Journal of Central South University*, vol. 29, no. 1, pp. 129–145, 2022.

[22] S. Panigrahi and U. S. N. Raju, "Pedestrian detection based on hand-crafted features and multi-layer feature fused-Res Net model," *International Journal on Artificial Intelligence Tools*, vol. 30, no. 5, article 2150028, 2021.

[23] C. Sekhar Vorugunti, V. Pulabaigari, P. Mukherjee, and A. Sharma, "DeepFuseOSV: online signature verification using hybrid feature fusion and depthwise separable convolution neural network architecture," *IET Biometrics*, vol. 9, no. 6, pp. 259–268, 2020.

[24] R. F. Rachmadi, S. Nugroho, and I. Purnama, "Lightweight residual network for person re-identification," *IOP Conference Series Materials Science and Engineering*, vol. 1077, no. 1, article 012046, 2021.

[25] Y. Nan, J. Ju, Q. Hua, H. Zhang, and B. Wang, "A-MobileNet: an approach of facial expression recognition," *Alexandria Engineering Journal*, vol. 61, no. 6, pp. 4435–4444, 2022.

[26] C. F. Xception, "Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1251–1258, America Hawaii, 2017.

[27] A. I. Mohammed and A. A. Tahir, "A new optimizer for image classification using wide ResNet (WRN)," *Academic Journal of Nawroz University*, vol. 9, no. 4, pp. 1–13, 2020.

[28] S. Zagoruyko and N. Komodakis, "Wide residual networks," 2017, https://arxiv.org/abs/1605.07146.

[29] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 7, 2017.

[30] S. Woo, J. Park, J. Y. Lee, and I. S. Kweon, "Cbam: convolutional block attention module," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 3–19, Munich Germany, 2018.

[31] L. Wang and D. He, "Image super-resolution reconstruction algorithm based on channel shuffle," in *2021 Asia-Pacific Conference on Communications Technology and Computer Science (ACCTCS)*, pp. 225–229, Shenyang, China, 2021.

[32] X. Y. Zhang, K. Zhao, T. Xiao, M. M. Cheng, and M. H. Yang, "Structured sparsification with joint optimization of group convolution and channel shuffle," in *Uncertainty in Artificial Intelligence. PMLR*, pp. 440–450, America New York, 2021.

[33] Y. Gan, J. Chen, Z. Yang, and L. Xu, "Multiple attention network for facial expression recognition," *IEEE Access*, vol. 8, pp. 7383–7393, 2020.

[34] A. Abdolrashidi, "Deep-emotion: facial expression recognition using attentional convolutional network," *Sensors*, vol. 21, no. 9, p. 3046, 2021.

[35] H. Zhang, B. Huang, and G. Tian, "Facial expression recognition based on deep convolution long short-term memory networks of double-channel weighted mixture," *Pattern Recognition Letters*, vol. 131, pp. 128–134, 2020.

[36] C. Jiamin and X. Yang, "Expression recognition based on attention pyramid convolution residual network," Computer engineering and application: 1-11 [2022-04-26] http://kns.cnki.net/kcms/detail/11.2127.TP.20210702.1749.004.html.

Hindawi
International Journal of Aerospace Engineering
Volume 2022, Article ID 9322534, 18 pages
https://doi.org/10.1155/2022/9322534

Research Article

3-DOF Position and Orientation Control of an Air Flotation Platform for Spacecraft Ground Microgravity Simulation by Using Double Closed-Loop Cascade PIDnn

Siyue Wang,[1,2] **Qing Gao**,[3] **Jinguo Liu**,[2,4] **Jingkai Feng**,[2,4,5] **and Jingshu Liufu**[2,6]

[1]*Shenyang Ligong University, Shenyang 110158, China*
[2]*State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China*
[3]*Institute of Robotics and Intelligent Manufacturing & School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen 518172, China*
[4]*Institute of Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110169, China*
[5]*University of Chinese Academy of Sciences, Beijing 100049, China*
[6]*University of Science and Technology Liaoning, Anshan 110325, China*

Correspondence should be addressed to Jinguo Liu; liujinguo@sia.cn

Received 24 April 2022; Revised 5 July 2022; Accepted 1 August 2022; Published 22 August 2022

Academic Editor: Adel Ghenaiet

Space assistant robots can help astronauts or their assistants perform certain tasks. A ground microgravity simulation environment is built for the space assistant robot AAR-2. The hardware requirements of the ground simulation by the 3-DOF microgravity air flotation platform. An algorithm is designed for this simulation system. By using momentum and RMSprop methods to improve the PID neural network, the challenging problem of strong coupling between system nonlinearity and variables is solved. Firstly, the paper introduces the hardware system and deduces the dynamic model of the system. Then, the algorithm is calculated and simulated. Through simulation, the effectiveness and feasibility of the algorithm are compared and proved. Finally, the control system is simulated by MATLAB/Simulink and compared with other advanced algorithms. The simulation results show that the designed neural network controller can quickly and accurately control the 3-DOF of freedom motion of AAR-2.

1. Introduction

With the continuous development of aerospace, many countries have carried out the research and development of space robots to replace astronauts to perform some specific tasks, for example, space rendezvous and docking, space debris avoidance [1], fuel supply, on orbit maintenance [2], on orbit component reconstruction, etc. In the past, most of these tasks were completed by astronauts, mainly due to the complexity of the space environment and the arduousness of the tasks. The characteristics of space greatly limit the astronauts' activities inside and outside the capsule, and the complex and cumbersome operations greatly increase the astronauts' work and psychological pressure. Therefore, in recent years, more and more space operation tasks gradually turn to space robots.

For the design, manufacture, and operation of space robots, simulation experiments are required in the ground microgravity environment for overall performance evaluation, component testing, key parameter determination, and various system verifications [3]. Therefore, it directly promotes the research on building a space microgravity environment. If you want to conduct a space robot simulation experiment, you need to build a microgravity experiment environment on the ground [4].

The methods of constructing space microgravity environment mainly include tower dropping method [5, 6], water floating method, and air floating method [7, 8]. The falling tower method is a method to generate microgravity experimental environment by performing free falling motion in microgravity tower (well). It is usually carried out by

building high towers or digging deep wells on the ground. Because of the special experimental environment and the limited weightlessness time, this method has great limitations, so it is not commonly used as a method to build a microgravity environment in the laboratory. The water floating method refers to a state in which when the density of the object is the same as that of the liquid. The object can be suspended at any point in the liquid. However, this method does not have real-time operability in the experimental process and needs to constantly adjust the counterweight to ensure balance and stability. On the other hand, it also requires that the experimental equipment can work normally under the liquid without affecting the accuracy. Therefore, this method will bring higher cost and experimental operation requirements to the test. However, the air flotation platform has the advantages of low cost, low operation difficulty, and sustainable real-time operation, so most experiments use the air flotation platform to simulate the microgravity environment [9, 10]. The core component of air flotation platform is gas bearing, which can be divided into plane gas bearing and spherical gas bearing. The compressed air flows out through the small hole or slit of the gas bearing and forms an air film between the workbench and the gas bearing, so as to achieve the purpose of frictionless movement. The air flotation platform has no contact with the worktable and produces a suspended effect, so it overcomes the influence of gravity to a great extent. Therefore, this paper selects the air flotation platform as the ground microgravity simulation platform of the space assistant robot and simulates the motion of the space assistant robot through the 3-DOF of air flotation platform.

Many scholars have different views and studies on the position and attitude control of spacecraft in microgravity environment. Terui et al. proposed a sliding mode control and state-dependent Riccati equation method to achieve 6-DOF position and attitude maneuverability without uncertainty and disturbance [11, 12]. In the follow-up, high-pass filter and output feedback control law are used to solve the problem of translational and attitude coupling, but the stability and convergence time are unconstrained [13]. In order to solve the problem of model uncertainty and environmental disturbance, Wu et al. proposed a finite time controller based on nonsingular terminal sliding mode control technology, which can control the translation and attitude change of spacecraft [14]. Liu et al. realized the 3-DOF free attitude control of AAR-1. In order to meet the characteristics of multivariable, strong coupling, and nonlinear systems, the traditional PIDnn controller is used to quickly achieve stability and control effect [15]. Gao et al. upgraded the functions of AAR-1, designed AAR-2, and realized high-precision control of AAR-2 by combining synovial control and fuzzy control [16]. Malladi et al. adopted the nonlinear model predictive control (NMPC) method to solve the highly coupled and nonlinear problem of position and attitude [17]. Aiming at the fast attitude dynamics and slow position dynamics caused by time scale separation, a nonlinear hierarchical control law was proposed [18].

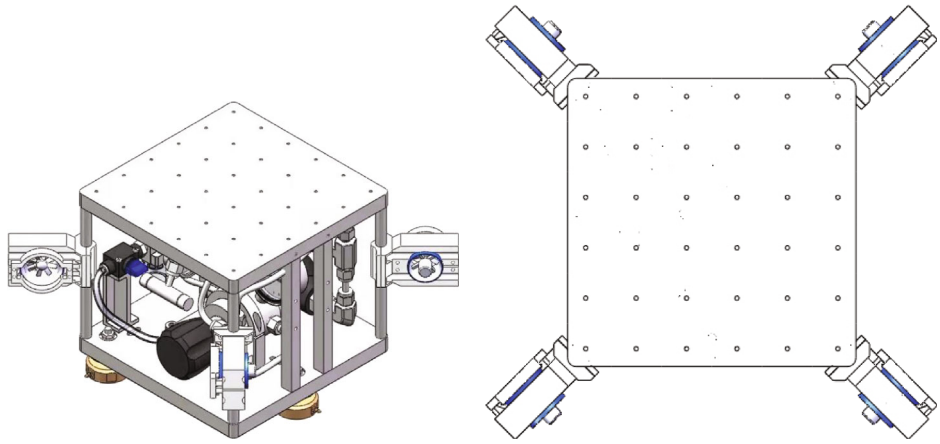
The attitude and translational motion of spacecraft are coupled and highly nonlinear, so the model parameters of spacecraft cannot be accurately obtained, and the spacecraft is always affected by environmental disturbances. All these problems make it difficult for the spacecraft control system to achieve the desired control performance. Due to the high coupling and nonlinearity of the dynamic model, this paper uses the double closed-loop structure to solve the influence of strong coupling on the system and uses the improved PID neural network to solve the problem of nonlinear system. Because the uncertain environmental disturbance always exists in the spacecraft movement, the neural network in machine learning can better cater to the uncertainty to a certain extent and match the changing environment through continuous learning.

The main innovations and contributions of this paper are as follows: (a) we use a double closed-loop system to solve the problem of the strong coupling and environmental disturbance in the model. This control structure can better overcome the impact of uncertainty on the system by matching with the neural network. (b) We optimized the PID neural network algorithm to solve the variable coupling problem and nonlinear problem when the spacecraft position and attitude change, so that the microgravity air flotation system can converge faster and reach the equilibrium position, better simulate the spacecraft motion and meet the experimental needs. (c) A complete control system and algorithm are built on the microgravity air flotation platform, which makes the flight simulation experiment of ground simulation spacecraft more efficient and convenient, and promote the development of ground microgravity environment simulation equipment.

The rest of this paper is organized as follows: in Section 2, the control system of the actual microgravity air flotation platform is designed and the motion principle is analyzed. In Section 3, the dynamic model of the microgravity air flotation platform is built and its motion characteristics are analyzed. In addition, the model is optimized to a certain extent according to the ideal environment of the actual space station. In Section 4, the control algorithm of the microgravity air flotation platform is improved on the basis of PIDnn. Meanwhile, the stability of the system is proved and the feasibility of the algorithm is simulated. In Section 5, the whole improved microgravity air flotation system is simulated. The feasibility of the scheme is verified by comparing other advanced algorithms and combining the theoretical simulation with the actual platform.

2. Design of the Control System Based on Air Flotation Platform

2.1. Control System Hardware Design. The microgravity simulation air flotation platform involved in this paper is shown in Figure 1(a). It can realize two-dimensional microgravity simulation motion on the marble plane through three air flotation pads, gas cylinders, and air pressure transmission system. Four ducted fans are mounted on it to control the movement of the air flotation platform. The installation position is shown in Figure 1(b). Four ducted fans at 45° angles to the side of the air flotation platform are installed at the four corners of the platform. Microspacecraft, such



(a) Air flotation platform model (b) Top view of model

Figure 1: Structural model of air flotation platform.

as astronaut assistance robots AAR-1, AAR-2 [19, 20], etc., can be installed on the air flotation platform, as shown in Figure 2(a). By controlling the changes of the thrust of the four ducted fans, the two-dimensional plane motion of the spacecraft can be simulated in a microgravity environment.

In this paper, the actual platform is shown in Figure 2(b). The STM series chip STM32f1 is selected as the main chip, equipped with two ultrasonic ranging modules HC-SR04, a six-axis gyroscope sensor MPU6050 (built-in DMP attitude calculation), and four ducted fans (brush less motor HL3508).

Figure 3 shows the top view of the air flotation platform, $O-X_nY_nZ_n$ is the inertial coordinate system, $O-X_bY_bZ_b$ is the geometric coordinate system, the arrow direction is the wind direction output by the culvert fan, and the four culvert fans are combined in pairs to control the forward and rotational degrees of freedom of the air flotation platform. The system forms a closed-loop structure through sensors, so as to achieve more efficient control effect.

2.2. Principles of Movement. To correctly describe the motion of the microgravity simulated air flotation platform, the two coordinate systems involved in the system are analyzed and processed. The motion pose of the air flotation platform is determined by the angle between the base point o and the axes X_b and X_n (the initial state is that the x-axis and the y-axis are parallel to each other), the coordinate of the base point is (x_0, y_0), the rigid body gestures are determined by x_0, y_0, and θ. The basic motion unit is composed of translation and rotation in the directions of X_n and Y_n (3-DOF), and the rotation is the rotation around the center of mass. F_1, F_2, F_3, and F_4 are the thrust (wind force) of the four ducted fans.

$$F_{X_b} = m\ddot{X}_b = \frac{\sqrt{2}}{2}(F_1 + F_4 - F_2 - F_3), \tag{1}$$

$$F_{Y_b} = m\ddot{Y}_b = \frac{\sqrt{2}}{2}(F_1 + F_2 - F_3 - F_4), \tag{2}$$

$$m_z^b = J_c\ddot{\theta} = \frac{\sqrt{2}}{2}(F_1 + F_4 - F_2 - F_3)(r + r_s). \tag{3}$$

It is known that the aerial view of the actual air flotation platform is a square, r is the side length, and r_s is the radius of the ducted fan. Let clockwise to be the positive direction, and m_z^b is the torque rotating around Z_b with the center of mass of the air flotation platform.

3. System Dynamics Model

3.1. Coordinate and Attitude Transformation. The pose of the air flotation platform is expressed as

$$\eta_{b,n}^n = \begin{bmatrix} p_{b,n}^n \\ \theta_{nb} \end{bmatrix}. \tag{4}$$

The instantaneous speed is expressed as

$$V_{b,n}^n = \begin{bmatrix} v_{b,n}^n \\ w_{b,n}^n \end{bmatrix}. \tag{5}$$

Then, we can get

$$\dot{\eta}_{b,n}^n = \begin{bmatrix} \dot{p}_{b,n}^n \\ \dot{\theta}_{nb}^n \end{bmatrix} = \begin{bmatrix} R_b^n(\theta_{nb}) & 0_{3*3} \\ 0_{3*3} & T_\theta(\theta_{nb}) \end{bmatrix} \begin{bmatrix} v_{b,n}^n \\ w_{b,n}^n \end{bmatrix}, \tag{6}$$

where $\eta_{b,n}^n$ represents the attitude of b relative to the inertial system n under the inertial system n, $R_b^n(\theta_{nb})$ represents the rotation matrix, and $\theta_{nb} = [\phi \quad \theta \quad \psi]^T$ is the geometric position b Euler angles to inertial position n.

$R_b^n(\theta_{nb})$ and $T_\theta(\theta_{nb})$ are the results of the transformation from the inertial coordinate system to the geometric coordinate system, and there are many ways to obtain them, which are explained in the order of Z-Y-X [21].

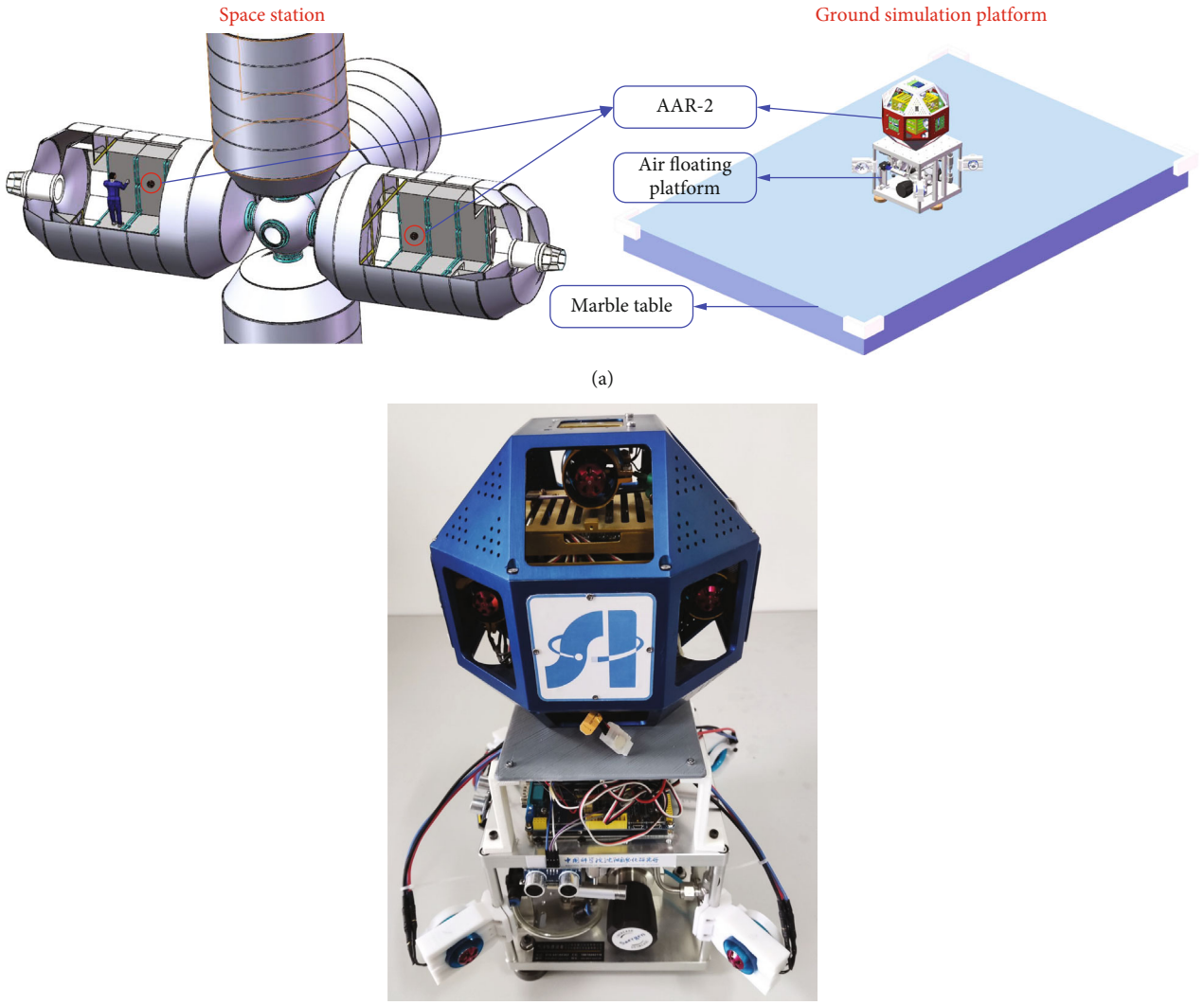


Figure 2: (a) Schematic diagram of space station and ground simulation platform. The AAR-2 is a free flying robot used in the space station cabin to assist astronauts in some space missions. This robot can use the microgravity simulation air flotation platform to simulate the 3-DOF plane motion on the marble table. (b) Actual experimental platform.

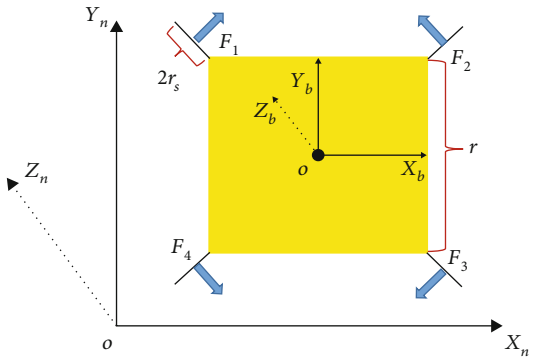


Figure 3: Top view of air flotation platform.

$$R_b^n(\theta_{nb}) = R_\phi R_\theta R_\phi = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\phi & s\phi \\ 0 & -s\phi & c\phi \end{bmatrix} \begin{bmatrix} c\theta & s\theta & 0 \\ -s\theta & c\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\psi & 0 & -s\psi \\ 0 & 1 & 0 \\ s\psi & 0 & c\psi \end{bmatrix}$$
$$= \begin{bmatrix} c\theta c\psi & s\theta & -c\theta s\psi \\ s\phi s\psi - c\phi s\theta c\psi & c\phi c\theta & s\phi c\psi - c\phi s\theta s\psi \\ c\phi s\psi + c\phi s\theta s\psi & -s\phi c\theta & c\phi c\psi - s\phi s\theta s\psi \end{bmatrix}, \tag{7}$$

$$T_\theta(\theta_{nb}) = \begin{bmatrix} 1 & 0 & -s\theta \\ 0 & c\phi & s\phi c\theta \\ 0 & -s\phi & c\phi c\theta \end{bmatrix}^{-1} = \begin{bmatrix} 1 & s\phi t\theta & c\phi t\theta \\ 0 & c\phi & -s\phi \\ 0 & \dfrac{s\phi}{c\theta} & \dfrac{c\phi}{c\theta} \end{bmatrix}. \tag{8}$$

The above sine, cosine, and tangent functions are replaced by s, c, and t. When the motion mode of the air flotation platform is fixed as translation and rotation, then

$$v_{b,n}^{n} = \left[\dot{X}_b, \dot{Y}_b, 0\right]^T, \tag{9}$$

$$\theta_{nb} = [0 \quad 0 \quad \psi]^T, \tag{10}$$

$$R_b^n(\theta_{nb}) = \begin{bmatrix} c\psi & 0 & -s\psi \\ 0 & 1 & 0 \\ s\psi & 0 & c\psi \end{bmatrix}, \tag{11}$$

$$T_\theta(\theta_{nb}) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{12}$$

3.2. Dynamics Model. To further accurately analyze the motion state of the microgravity simulated air flotation platform, the dynamic model of the air flotation platform is deduced and simulated in this paper. The dynamic model of the air flotation platform can be deduced as follows [22–24]:

$$m\left(\dot{v}_{g,n}^{b} + w_{g,n}^{b} \times v_{g,n}^{b}\right) = F^g + f^g, \tag{13}$$

$$I\dot{w}_{g,n}^{b} + w_{g,n}^{b} \times \left(Iv_{g,n}^{b}\right) = M^g + m^g. \tag{14}$$

Among them, g represents the center of gravity and $F^g = [F_x^g, F_y^g, F_z^g]^T$ is used to represent gravity, buoyancy, and other external forces acting on an air flotation platform. $f^g = [f_x^g, f_y^g, f_z^g]^T$ is used to represent the thrust generated by the rotation of the ducted fan. $M^g = [M_x^g, M_y^g, M_z^g]^T$ is used to represent all external moments acting on the air flotation platform except the moment generated by the ducted fan. $m^g = [m_x^g, m_y^g, m_z^g]^T$ indicates the moment acting on the air flotation platform generated by the rotation of the ducted fan.

Since the air flotation platform is a rigid body and the additional mass is the same in all directions, the total inertia matrix about the center of gravity is

$$I = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix}. \tag{15}$$

Since the air flotation platform needs to move freely on the marble tabletop, its motion mode is 3-DOF motion, and the yaw motion can be freely performed while translating on the two-dimensional marble tabletop:

$$v_{g,n}^{b} = \left[v_{x_{g,n}}^{b}, v_{y_{g,n}}^{b}, 0\right]^T, \tag{16}$$

$$w_{g,n}^{b} = \left[0, 0, w_{z_{g,n}}^{b}\right]^T. \tag{17}$$

When the air flotation platform rotates and flies forward at the same time, a lateral force is generated, which is represented by $w_{g,n}^{b} \times v_{g,n}^{b}$. This term is offset by the air resistance created when the microgravity simulated air flotation platform moves. This force can also be ignored under the assumption that the spin speed is almost zero in forward flight and almost zero in spinning motion.

In the inertia matrix, we can get $I_x = I_y$, if the roll and pitch velocities are zero, no rotational moment will be generated. When the rolling and pitching moments are not zero, $w_{g,n}^{b} \times (Iv_{g,n}^{b})$ has little effect on the roll and pitch moments, so they are easily offset by the resistance of gravity, so they are also ignored.

The model can thus be simplified to

$$m\dot{v}_{x_{g,n}}^{b} = f_x^g + F_x^g = f_x^g - k_x\dot{x}, \tag{18}$$

$$m\dot{v}_{y_{g,n}}^{b} = f_y^g + F_y^g = f_y^g - k_y\dot{y}, \tag{19}$$

$$I_z\dot{w}_{z_{g,n}}^{b} = m_z^g + M_z^g = m_z^g + M_{z_{\text{cor}}}^g + M_{z_d}^g - k_z w_z(r + r_s). \tag{20}$$

Among them, $M_{z_{\text{cor}}}^g$ refers to the Coriolis torque in the direction of rotation around the z-axis, $M_{z_d}^g$ refers to the average disturbance torque on the z-axis, and k_z represents the average torque coefficient in the air resistance on the z-axis.

It is easy to design the controller with negligible Coriolis torque, disturbance torque, and air resistance. In the motion state, we are only interested in the dynamic equations of Euler angles in the geometric coordinate system in motion. Therefore, the dynamic model of the air flotation platform is further simplified as follows:

$$m\ddot{X}_b = c\psi f_x^b, \tag{21}$$

$$m\ddot{Y}_b = f_y^b, \tag{22}$$

$$I_z\ddot{\psi} = m_z^b. \tag{23}$$

4. Algorithmic Controller Design

4.1. PIDnn Controller Design. PIDnn is a kind of PID-type controller that relies on the self-adaptation and learning ability of the neural network algorithm [25]. There are various neural network structures that can be designed. This paper adopts a single-layer forward propagation network, in which the input layer, hidden layer, and output layer contain two, three, and one neurons (perceptrons), respectively [26].

The input of the PID neural network system has two parts, which are the expected input and the real-time output of the system. After each iteration, the neural network outputs new values of P, I, and D as three parameters of the

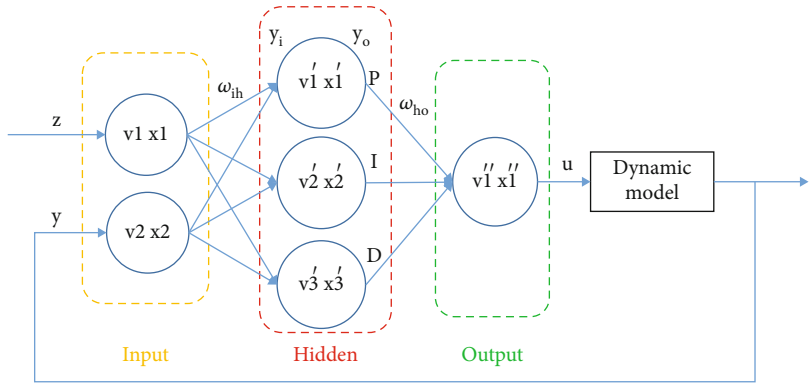


Figure 4: PIDnn control algorithm structure.

PID neural network controller [27, 28]. After processing by the dynamic model, as the new input of the system continues to iterate, the system tends to stabilize after the number of sampling N approaches a larger value [29]. The structure of the PIDnn control algorithm is shown in Figure 4.

4.1.1. Forward Algorithm. When iterating to the nth time, the input and output of neurons in each layer are expressed as follows:

Input layer:

$$v_1(n) = z(n), \tag{24}$$

$$v_2(n) = y(n), \tag{25}$$

$$x_i(n) = \mathrm{NET}[v_i(n)], i = 1, 2, \tag{26}$$

where $\mathrm{NET}(n)$ is an activation function. To reduce the actual hardware system code calculation amount, it can be set as an identity activation function here, namely,

$$\mathrm{NET}[v_i(n)] = v_i(n). \tag{27}$$

Hidden layer:

$$v_h'(n) = \sum_{i=1}^{2} w_{ih} x_i(n), h = 1, 2, 3. \tag{28}$$

The activation function of the hidden layer is special. Considering that the neural network serves the PID system, the three neurons here perform their proportional, integral, and differential functions, respectively. After discretizing the continuous system, we get the following:

Proportion:

$$x_1'(n) = v_1'(n). \tag{29}$$

Integral:

$$x_2'(n) = x_2'(n-1) + v_2'(n). \tag{30}$$

Derivative:

$$x_3'(n) = v_3'(n) - v_3'(n-1). \tag{31}$$

Output layer:

$$v_1''(n) = \sum_{h=1}^{3} W_{ho} x_h'(n), o = 1, \tag{32}$$

$$x_o''(n) = v_o''(n). \tag{33}$$

4.1.2. Back-Propagation Algorithm. The back-propagation algorithm can achieve the effect of learning and memory by modifying the weight value (ω_{ih} and W_{ho}) of the neural network. First, calculate the systematic error (the ideal value minus the actual value) and then update the weight of each layer of the neural network according to the conventional gradient descent method. When taking the $(n+1)$th sampling, the error $E = z(n) - y(n)$. Let learning step size be η, after the nth learning, the weight value between the hidden layer and the output layer is changed to

$$W_{ho}(n+1) = W_{ho}(n) - \eta \frac{dE}{dW_{ho}}, \tag{34}$$

$$\frac{dE}{dW_{ho}} = \frac{dE}{dy_o} \frac{dy_o}{dx_1''} \frac{dx_1''}{dv_1''} \frac{dv_1''}{dW_{ho}}. \tag{35}$$

The weight value between the input layer and the hidden layer is changed to

$$\omega_{ih}(n+1) = W_{ih}(n) - \eta \frac{dE}{d\omega_{ih}}, \tag{36}$$

$$\frac{dE}{d\omega_{ih}} = \frac{dE}{dy_o} \frac{dy_o}{dx_1''} \frac{dx_1''}{dv_1''} \frac{dv_1''}{dx_h'} \frac{dx_h'}{dv_h'} \frac{dv_h'}{d\omega_{ih}}. \tag{37}$$

The weights between hidden layer neurons and output layer neurons are real-time P, I, and D coefficients.

4.2. Momentum and RMSprop Jointly Improve the Algorithm. The iterative principle of the standard gradient descent method is

$$W(n) = W(n-1) - \eta \Delta W(n), \tag{38}$$

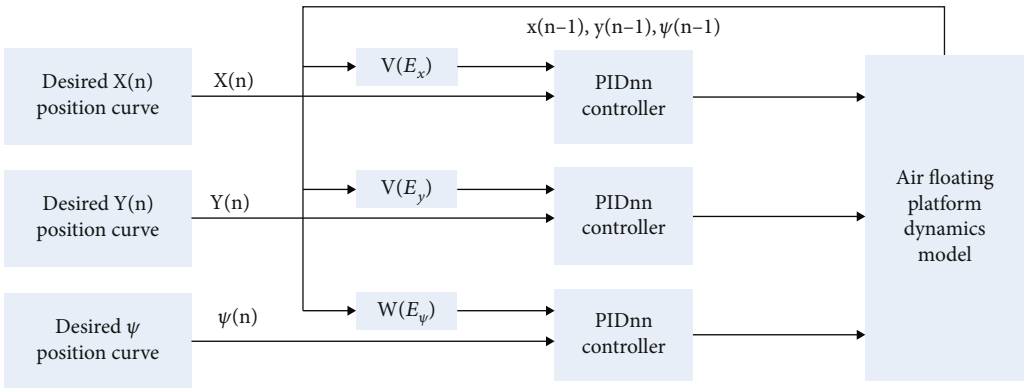


Figure 5: Air flotation platform control system.

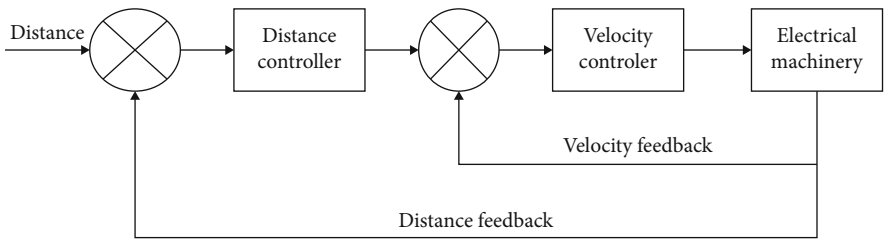


Figure 6: Cascade control structure.

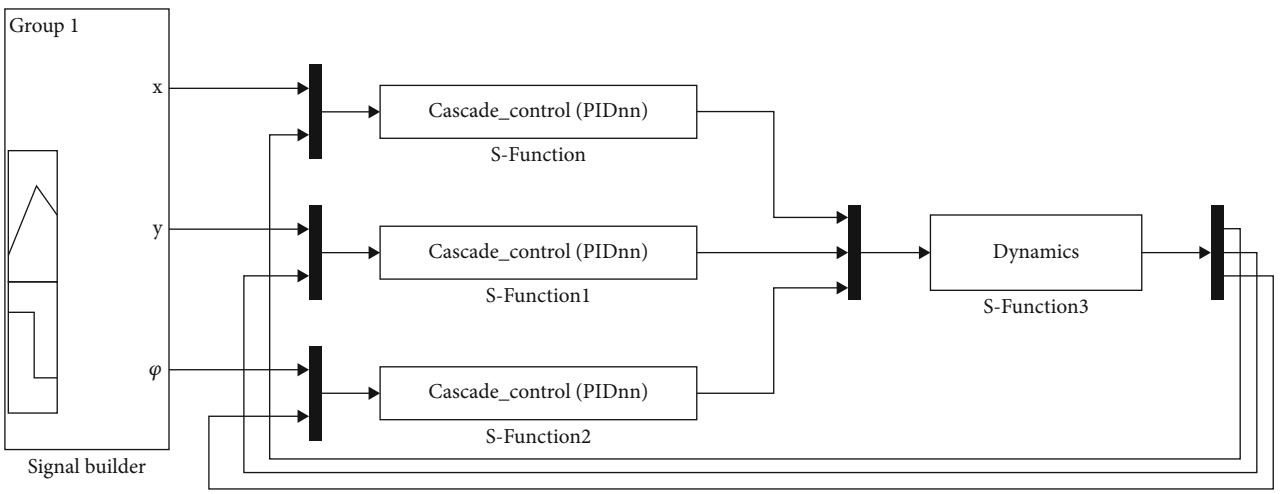


Figure 7: Simulink simulation structural diagram.

$$\Delta W(n) = \frac{\partial E[W(n-1)]}{\partial W}, \mathrm{n} = 1, 2, 3 \cdots . \tag{39}$$

The momentum method changes the weight update law in the standard gradient descent method and introduces an intermediate quantity:

$$V(n) = \beta V(n-1) + (1-\beta)\Delta W(n). \tag{40}$$

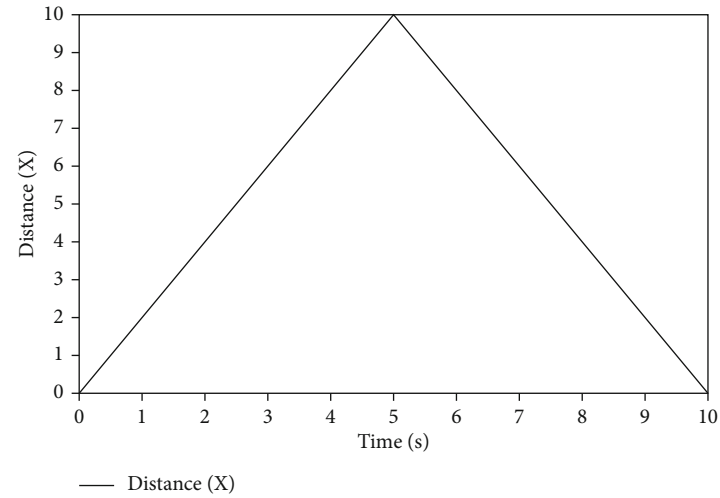
And replacing the original $\Delta W(n)$ with $V(n)$ can get

$$\begin{aligned} W(n) &= W(n-1) - \eta V(n) \\ &= W(n-1) - \eta[\beta V(n-1) + (1-\beta)\Delta W(n)], \end{aligned} \tag{41}$$

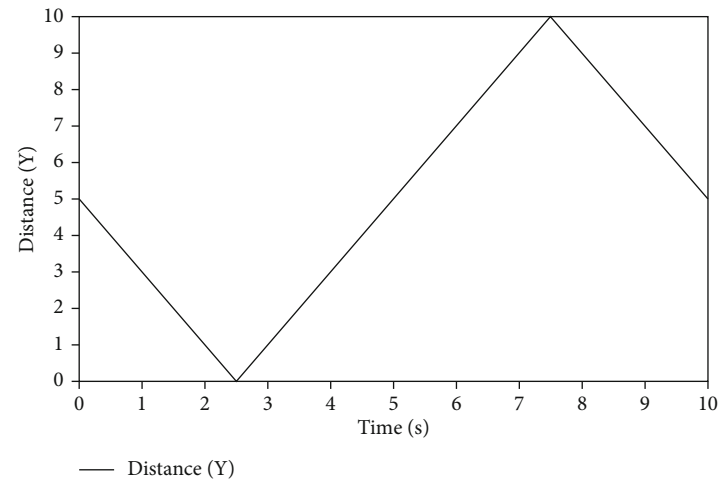
$$\begin{aligned} W(n) = W(n-1) - \eta(1-\beta)[\Delta W(n) + \beta \Delta W(n-1) \\ + \beta^2 \Delta W(n-2) + \cdots + \beta^{n-1} \Delta W(1)]. \end{aligned} \tag{42}$$

The value of β around 0.9 can make the algorithm achieve better results. Doing so is equivalent to a weighted sum of historically processed data. From the nth iteration, the farther historical data has a relatively small impact on the weight update, but the weight updated in the nearby iterations has a greater impact.

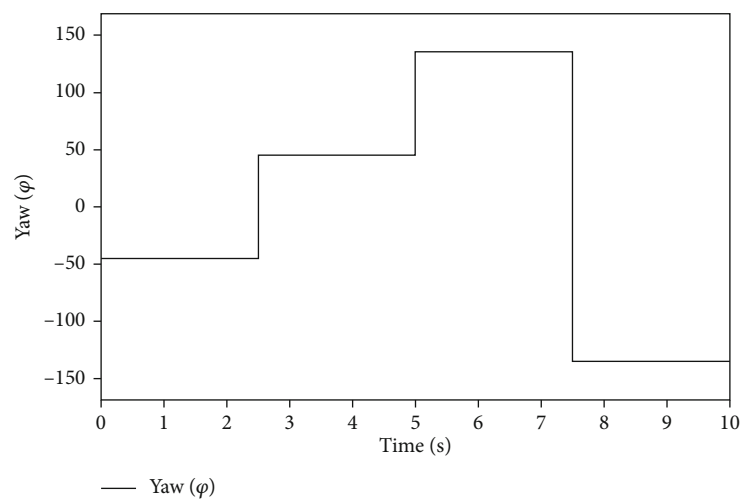
The method of adaptive adjustment of Adagrad learning rate is to convert η (learning rate) in Equation (43) into $\eta / \sqrt{S(n)} + \varepsilon$.



(a) x-axis input curve

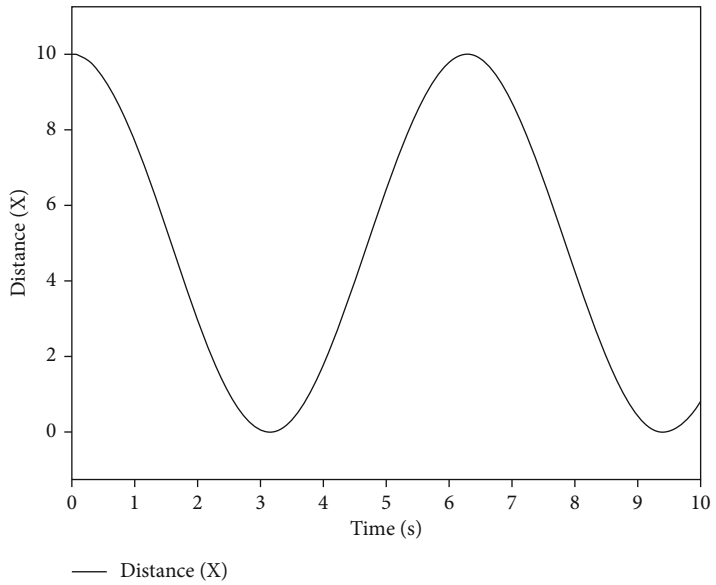


(b) y-axis input curve

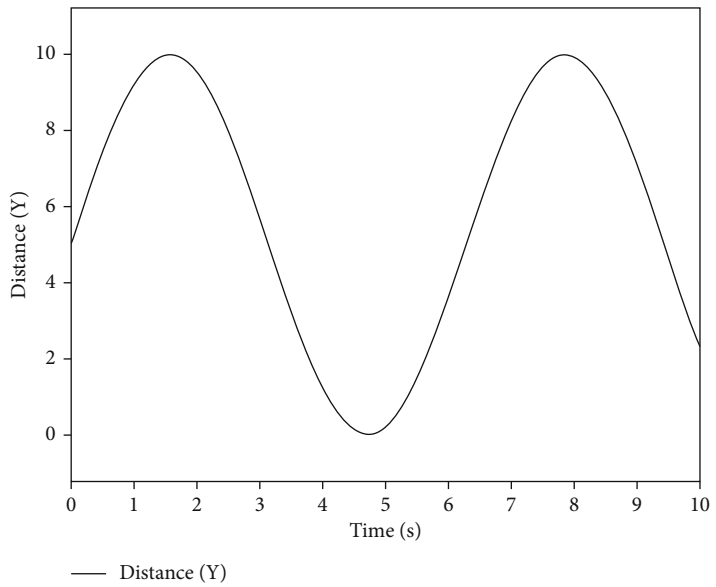


(c) Yaw-axis input curve

Figure 8: Square motion curve design.

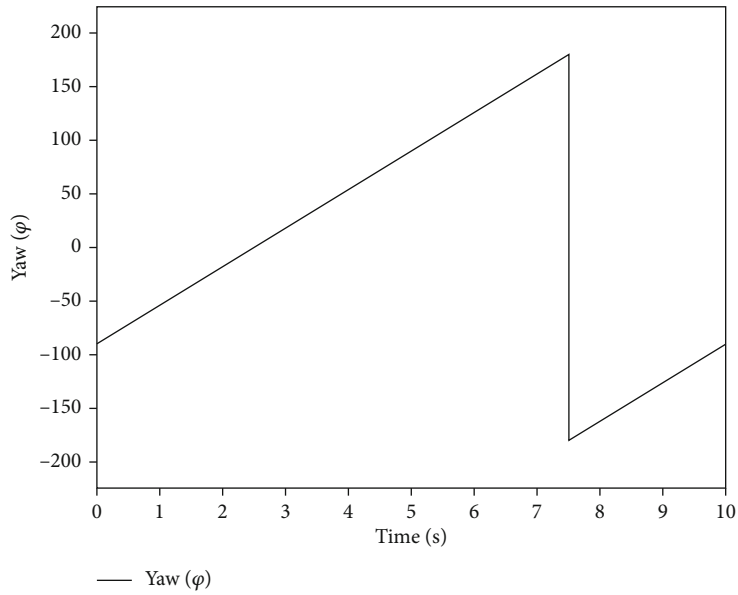


(a) x-axis input curve



(b) y-axis input curve

Figure 9: Continued.



(c) Yaw-axis input curve

Figure 9: Circular curve design.

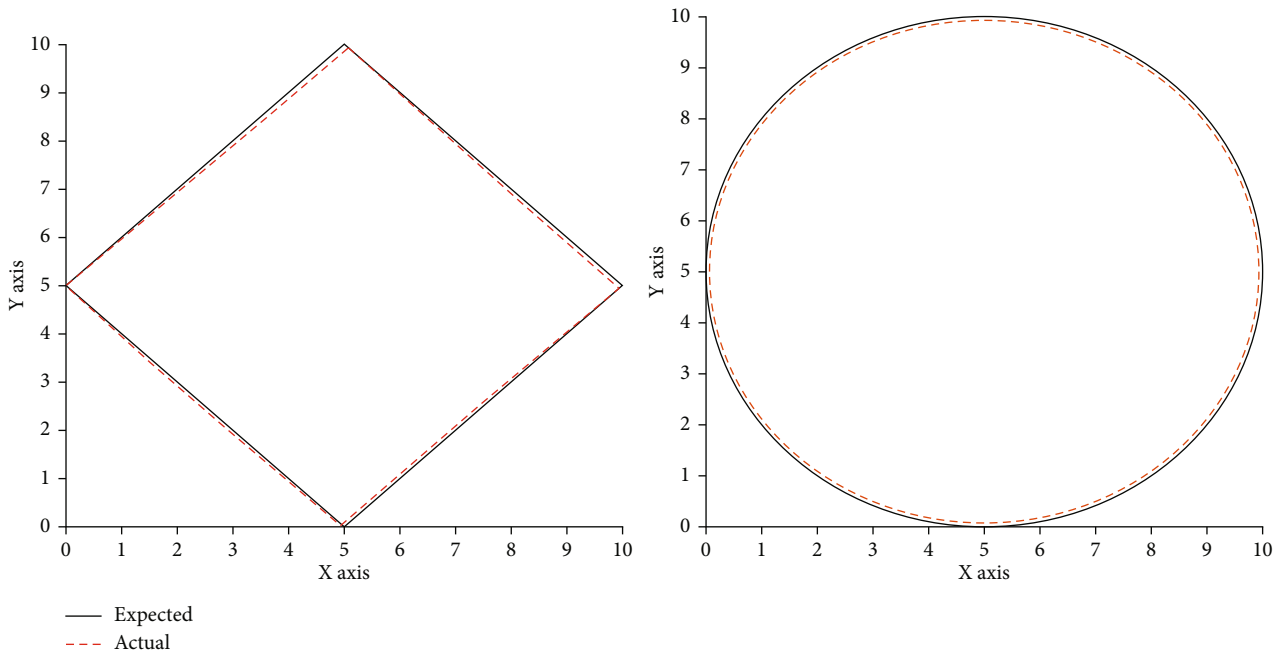


Figure 10: Comparison between control effect and expected effect.

$$W(n) = W(n-1) - \eta \Delta W(n). \tag{43}$$

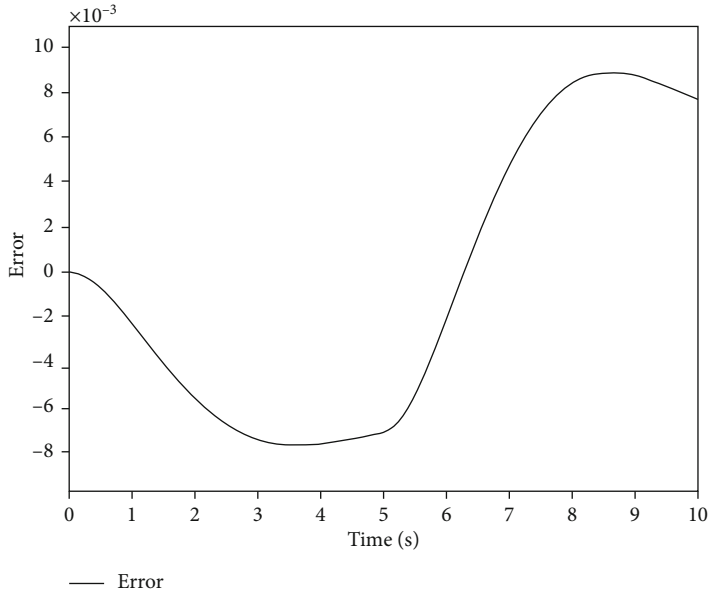
To prevent the denominator from being zero, set ε to a particularly small number.

$$S(n) = S(n-1) + \Delta W(n) \cdot \Delta W(n). \tag{44}$$

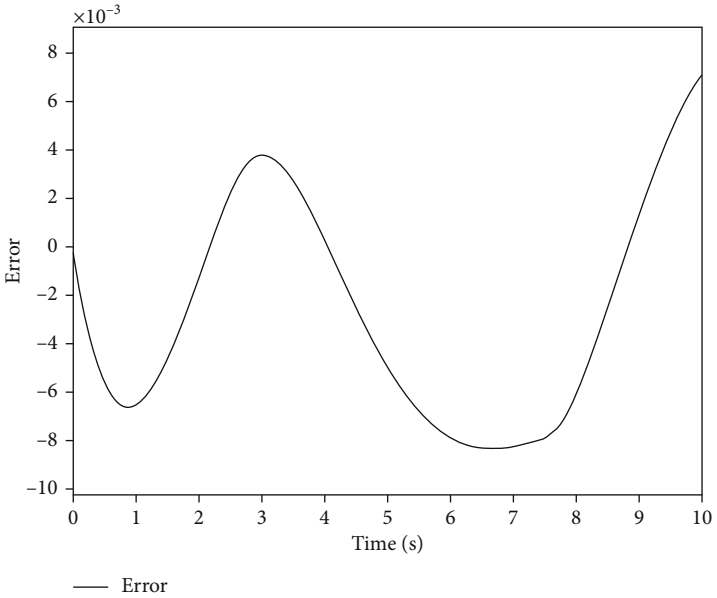
Then, we can get

$$W(n) = W(n-1) - \frac{\eta}{\sqrt{[\Delta W(1)]^2 + \cdots + [\Delta W(n)]^2 + \varepsilon}} \Delta W(n). \tag{45}$$

If the weights are modified too much, the learning rate

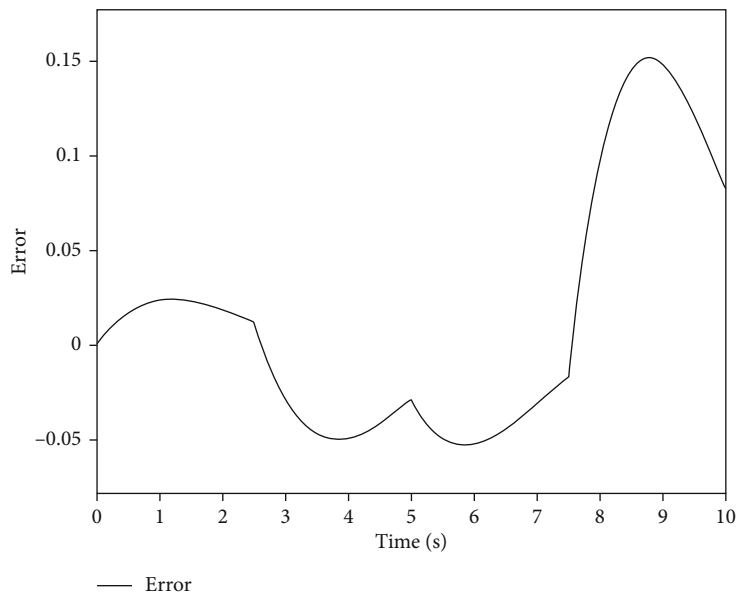


(a) Error curve of x-axis



(b) Error curve of y-axis

Figure 11: Continued.



(c) Error curve in yaw-axis

Figure 11: Error tracking of the first motion curve.

will decrease more. Then, in the process of gradient descent training, after a rapid change period, and then a platform area, the change of the learning rate in the platform area will be very slow. When it enters the fast period again after the plateau period, the decline is still very slow and cannot be modified, and all the historical data of the weights will be taken into account, and then, there is a problem.

So, the RMSprop method introduces Momentum on the learning rate, so that the formula becomes

$$S(n) = \alpha S(n-1) + (1-\alpha)\Delta W(n) \cdot \Delta W(n). \tag{46}$$

It is derived from Equation (46) that the gradient descent method in the PID neural network is finally optimized as

$$W(n) = W(n-1) - \frac{\eta(1-\beta)\left[\beta^{n-1}\Delta W(1) + \cdots + \beta^{0}\Delta W(n)\right]}{\sqrt{(1+\alpha)\left\{\alpha^{n-1}[\Delta W(1)]^2 + \cdots + \alpha^{0}[\Delta W(n)]^2\right\}} + \varepsilon}, \tag{47}$$

where α and β have the same effect, so we let $\alpha = \beta = 0.9$.

4.3. Proof of System Stability. The system control input of the PID neural network controller is

$$u(n) = u(n-1) + \eta(n)\sum_{h=1}^{3} x_h'(n)\, W_{ho}(n), \tag{48}$$

$$E(n) = \frac{1}{2}(r(n) - Y(n))^2, \tag{49}$$

$$\begin{aligned}\frac{dE(n)}{dx_h'(n)} &= \frac{dE(n)}{de(n)}\frac{de(n)}{dY(n)}\frac{dY(n)}{dz(n)}\frac{dz(n)}{dx_h'(n)} \\ &= -e(n)\frac{dy(n)}{dz(n)}W_{ho}(n)\eta(n).\end{aligned} \tag{50}$$

To analyze the stability of the system, a Lyapurov function $V(n) = (1/2)e(n)^2$ is constructed, where $e(n) = z(n) - y(n)$.

$$\Delta \mathrm{V(n)} = \mathrm{V(n+1)} - \mathrm{V(n)} = \frac{1}{2}\left(e(n+1)^2 - e(n)^2\right). \tag{51}$$

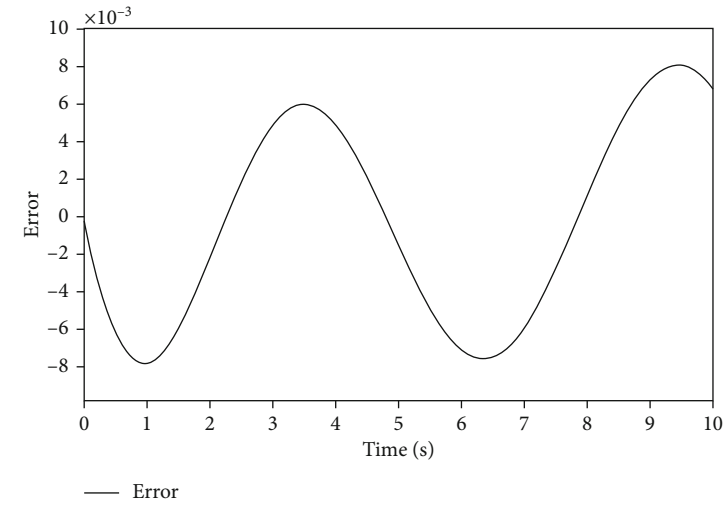
According to the differential theorem,

$$\Delta e(n) = \sum_{j=1}^{3}\frac{de(n)}{dx_h'(n)}\Delta x_h'(n). \tag{52}$$

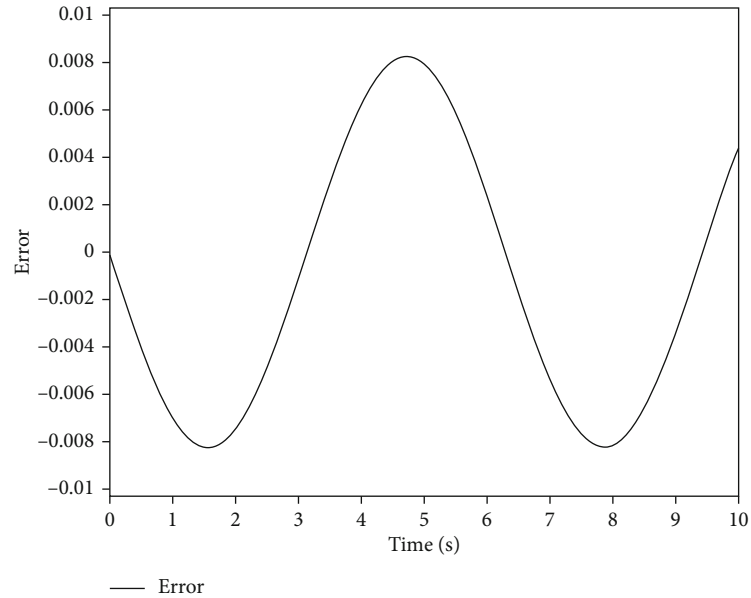
According to Lyapurov function,

$$\begin{aligned}\frac{de(n)}{dx_h'(n)} &= \frac{de(n)}{dY(n)}\frac{dY(n)}{du(n)}\frac{du(n)}{dx_h'(n)} \\ &= -\operatorname{sgn}\left(\frac{dy(n)}{du(n)}\right)\eta(n)W_{ho}(n),\end{aligned} \tag{53}$$

$$\begin{aligned}\Delta e(n) = -\sum_{j=1}^{3}\operatorname{sgn}\left(\frac{dY(n)}{du(n)}\right)\eta(n)W_{ho}(n) \times \eta(k)e(k)\ \operatorname{sgn} \\ \cdot \left(\frac{dy(n)}{du(n)}\right)W_{ho}(n).\end{aligned} \tag{54}$$

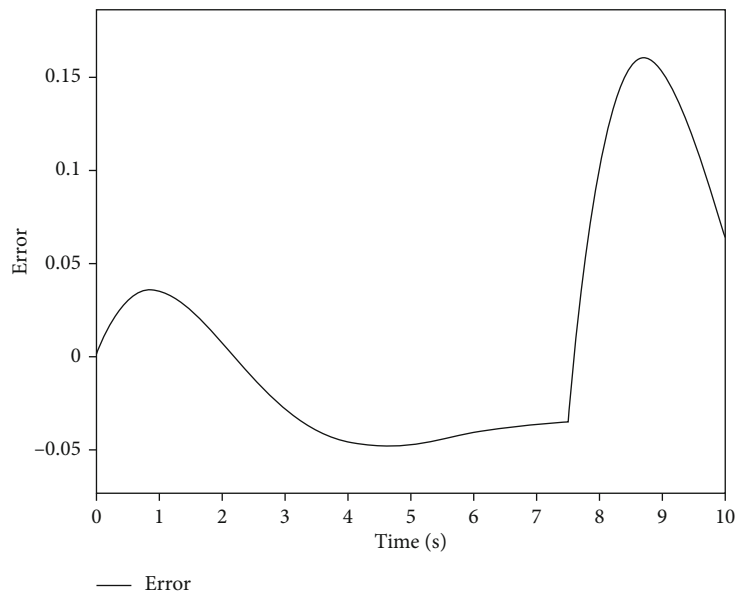


(a) Error curve of x-axis



(b) Error curve of y-axis

Figure 12: Continued.



(c) Error curve in yaw-axis

Figure 12: Error tracking of the second motion curve.

Because of $(\text{sgn}\ (dy(n)/du(n)))^2 = 1$,

$$\Delta e(n) = -\sum_{j=1}^{3} (\eta(n))^2 (W_{ho}(n))^2 e(n), \tag{55}$$

$$e(n+1) = e(n) + \Delta e(n), \tag{56}$$

$$\begin{aligned}\Delta V(n) &= \frac{1}{2}\left(e(n+1)^2 - e(n)^2\right) = \frac{1}{2}\Delta e(n)(\Delta e(n) + 2e(n)) \\ &= e(n)\Delta e(n) + \frac{1}{2}\Delta e(n)^2.\end{aligned} \tag{57}$$

So we can come to a conclusion. The closed loop system is stable at $\Delta V(n) < 0$.

$$e(n)\Delta e(n) < -\frac{1}{2}\Delta e(n)^2, \tag{58}$$

$$\begin{aligned}&-e(n)\sum_{h=1}^{3} (\eta(n))^2 (W_{ho}(n))^2 e(n) \\ &< -\frac{1}{2}\left(-\sum_{h=1}^{3} (\eta(n))^2 (W_{ho}(n))^2 e(n)\right)^2,\end{aligned} \tag{59}$$

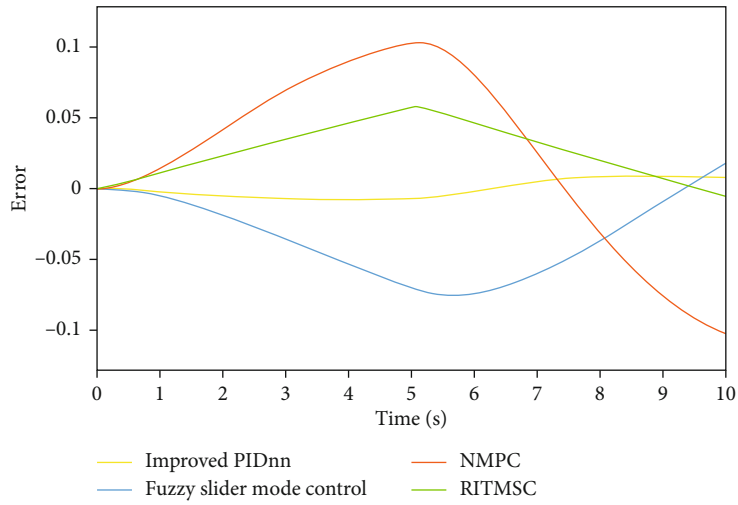
$$2\left(\sum_{h=1}^{3} (W_{ho}(n))^2\right)^{-1} > (\eta(n))^2. \tag{60}$$

Because the Lyapunov function has a lower bound and $\Delta V(n) < 0$, the function monotonically decreases, then when $n \longrightarrow \infty$, $V(n+1) = V(n) = 0, \Delta V(n) = 0$. Therefore, the neural network learning rate is set to a small value to make the system stable. Since the learning rate will affect the optimization speed of the gradient descent method, we have improved the neural network by momentum and RMSprop. By constantly changing the learning rate, the system can be stable and quickly reduced to the best.

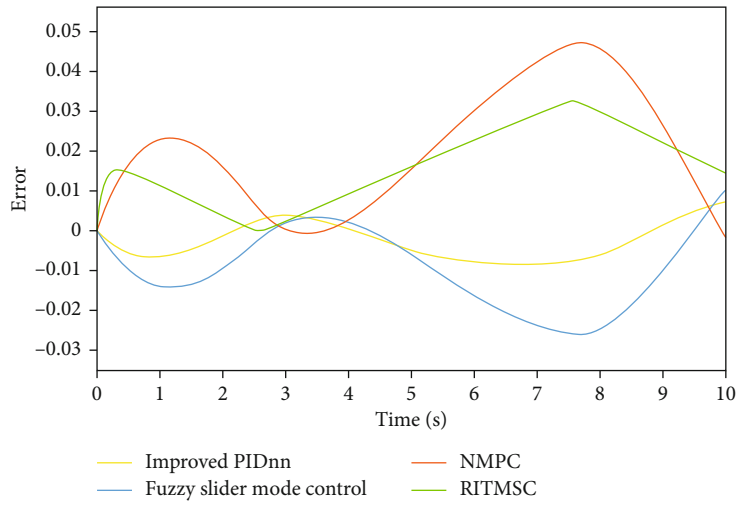
5. Control System and Simulation Experiment

5.1. Control System. According to the actual experimental requirements of fixed-point and fixed-speed, the desired air flotation platform position (x, y), angle (ψ), and speed-error curve $[V(E_x), V(E_y), W(E_\psi)]$ are designed. For three control variables $[x(n), y(n), \psi(n)]$ are designed three cascade double-loop control models. The curve is sent to the algorithm controller for processing and then sent to the algorithm controller through the dynamic model for repeated iterations. The experiment combined the sensor to form a closed loop and combined with the cascade double-loop control to realize the stable control of the air flotation platform to move freely. The air flotation platform control system and cascade control structure are illustrated in Figures 5 and 6.

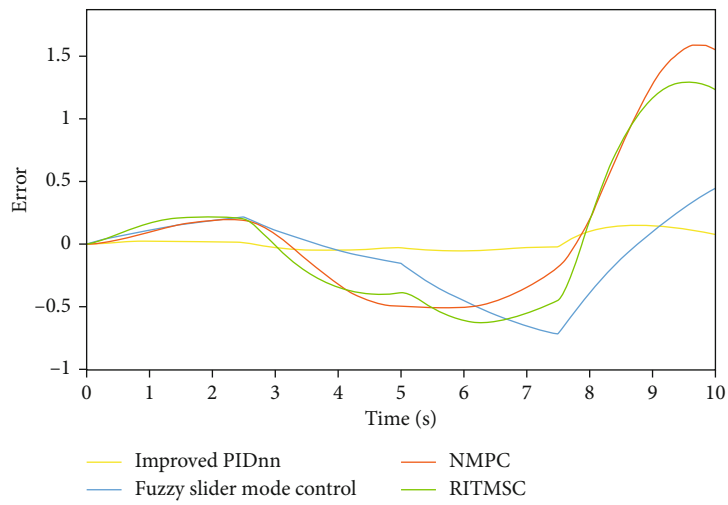
Considering that the actual trajectory may have sharp turns or circular motion, the following two motion models are designed. The first is a right-angled square and the second is a circular motion. In the first one, the motion curve of the air flotation platform starts from point $(0, 5)$ and returns to the origin $(0, 5)$ after moving counterclockwise for a week, and the overall consumption time is 10 seconds. And the initial direction of the air flotation platform is -90°, and the angle calculation is based on the angle between the direction of the air flotation platform and the x-axis. In the second one, the starting point, ending point, and elapsed time conditions of the movement curve of the air flotation platform remains unchanged. The motion curve presents a circular structure with a center of $(5, 5)$ and a radius of 5.



(a) Error curve of x-axis

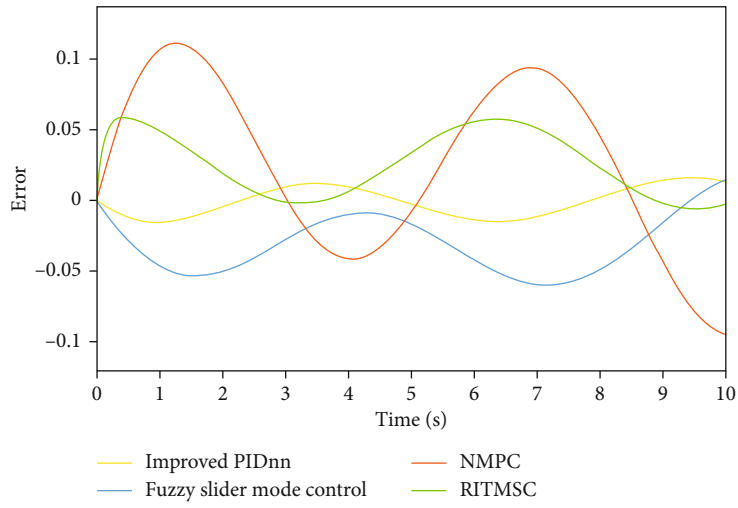


(b) Error curve of y-axis

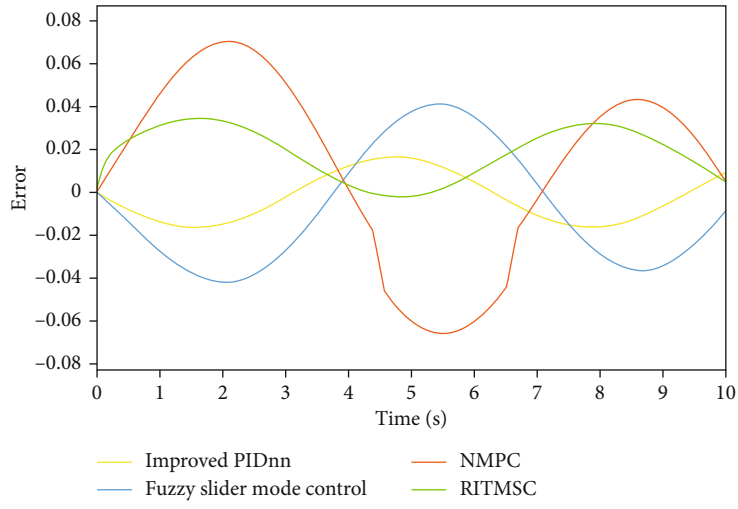


(c) Error curve in yaw-axis

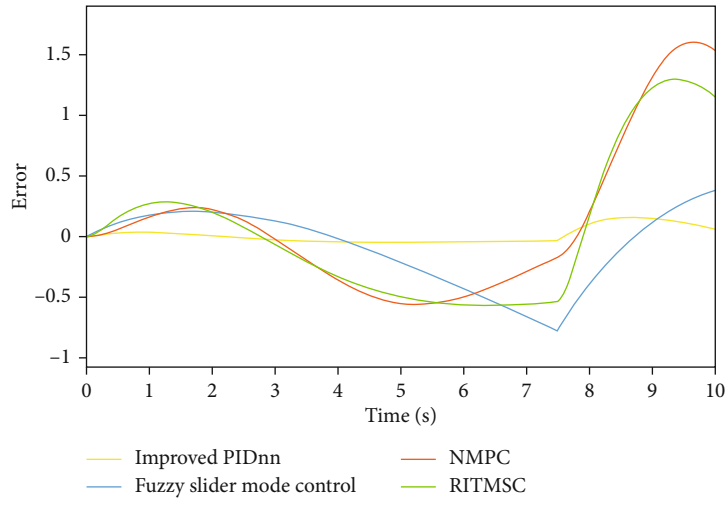
Figure 13: Motion trajectory comparison of advanced algorithms (square).



(a) Error curve of x-axis



(b) Error curve of y-axis



(c) Error curve in yaw-axis

Figure 14: Motion trajectory comparison of advanced algorithms (circle).

5.2. Simulation. To verify the effectiveness of the control algorithm designed for the 3-DOF air flotation platform, MATLAB/Simulink is used for simulation in this paper, and the overall effect is shown in Figure 7. The predicted position curve of the trajectory was input into the system, and then, the algorithm was controlled by the cascade double closed-loop PIDnn system. The obtained control results are introduced into the dynamics model and then returned to the controller for further calculation.

Considering the various movement and work of space robot, there will be a variety of movement curves such as straight line, curve, or sharp angle. So, we designed two kinds of curves, one is the motion curve of the square, and the other is the motion curve of the circle.

As shown in Figures 8 and 9, for each curve, we give the system's 3-DOF input.

After effective control by the controller, the system simulation curve is obtained as shown in Figure 10.

In the simulation experiment, as shown in Figures 11 and 12, we get the error tracking of the three degrees of freedom curve.

We can see from the error curve that the error effect of moving distance can reach the level of 10^{-3} on average in the simulation experiment. In yaw, the maximum error in motion is only 0.17°. It can be observed that using the improved PIDnn based on momentum and RMSprop, the position and attitude control of the air flotation platform has a good control effect. It can quickly and accurately track the input signal.

5.3. Comparison of Advanced Algorithms. In order to analyze the optimality of space robot control system, we compared other advanced control algorithms according to the above two trajectory models. We can see the error comparison curve in Figures 13 and 14. We compare it with three other advanced control algorithms in recent years. The blue line represents the fuzzy slider mode control algorithm [16], the red line represents the nonlinear model predictive control algorithm (NMPC) [17], and the green line represents the robot integral terminal sliding mode control law (RITSMC) [30].

The improvement and variation of the control effect can be obtained from the error curve. The average control error of position control can reach 7.59×10^{-3} unit distance, and the error control accuracy of yaw can reach 8.36×10^{-2}°. The improved PID neural network using momentum and RMSprop can be proved by simulation results. It can effectively control the high-precision movement of the space robot, so that the space robot can have smaller error with the preset curve and achieve accurate control.

6. Conclusion and Future Work

In this paper, according to the needs of the ground preexperiment, the control system is designed and constructed in combination with the 3-DOF air flotation platform. The design allows it to move freely on the marble table. The design of the overall dynamic model and control algorithm is introduced, and the model is simplified and deduced according to the actual experimental environment. The control algorithm adopts the PID neural network to realize the 3-DOF motion of the air flotation platform. And the double-loop cascade control method is adopted to achieve better control effect. Finally, the designed control system is simulated by MATLAB/Simulink. The overall control effect is simulated, and it is known from the simulation effect that the system can control the air flotation platform quickly and effectively. In the era of rapid development of science and technology, the development of space operation technology and robot-assisted technology is gradually improving the hard power of human space exploration. As a ground-assisted experimental configuration, the air flotation platform plays an important role, and the improvement and research of experimental equipment will be further improved and further advance the development and progress of automation.

In the future work, we mainly improve the following contents: (a) optimize the overall control structure and specific hardware equipment of the microgravity air flotation platform and (b) optimize the computing power and sensor accuracy of the single-chip microcomputer, so that it can perform more complex control algorithms and obtain data with higher precision.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no competing interests regarding the publication of this paper.

Authors' Contributions

Siyue Wang and Qing Gao contributed equally to this work. Qing Gao is the co-first author.

Acknowledgments

This work was supported in part by the National Key R & D Program of China (Grant No. 2018YFB1304600), the Natural Science Foundation of China (Grant Nos. 51775541 and 62006204), and CAS Interdisciplinary Innovation Team (Grant No. JCTD-2018-11).

References

[1] M. Shan, J. Guo, and E. Gill, "Review and comparison of active space debris capturing and removal methods," *Progress in Aerospace Sciences*, vol. 80, pp. 18–32, 2016.

[2] X. Ge, Q. Zhou, and Z. Liu, "Assessment of space station on-orbit maintenance task complexity," *Reliability Engineering & System Safety*, vol. 193, article 106661, 2020.

[3] K. Yoshida, "Engineering test satellite VII flight experiments for space robot dynamics and control: theories on laboratory test beds ten years ago, now in orbit," *The International Journal of Robotics Research*, vol. 22, no. 5, pp. 321–335, 2003.

[4] R. Herranz, R. Anken, J. Boonstra et al., "Ground-based facilities for simulation of microgravity: organism-specific recommendations for their use, and recommended terminology," *Astrobiology*, vol. 13, no. 1, pp. 1–17, 2013.

[5] T. Könemann, U. Kaczmarczik, A. Gierse et al., "Concept for a next-generation drop tower system," *Advances in Space Research*, vol. 55, no. 6, pp. 1728–1733, 2015.

[6] J. Liu, Y. Li, W. Li, H. Zhai, and L. Chen, "Testing liquid distribution in a vane-type propellant tank under conditions of microgravity using a drop tower test," *International Journal of Aerospace Engineering*, vol. 2020, Article ID 6402083, 13 pages, 2020.

[7] C. Nieto-Peroy, G. Palmerini, É. J. de Oliveira, P. Gasbarri, M. Sabatini, and M. Milz, "Simulation of spacecraft formation maneuvers by means of floating platforms," in *2021 IEEE Aerospace Conference (50100)*, pp. 1–10, Big Sky, MT, USA, 2021.

[8] L. W. Deng, Z. M. Liu, C. C. Wang, B. W. Chen, and S. M. Song, "The design for position and attitude control system of 5-DOF air-floated platform based on ADRC," in *2011 2nd International Conference on Intelligent Control and Information Processing*, pp. 1061–1065, Harbin, China, 2011.

[9] T. Rybus and K. Seweryn, "Planar air-bearing microgravity simulators: review of applications, existing solutions and design parameters," *Acta Astronautica*, vol. 120, pp. 239–259, 2016.

[10] J. Prado, G. Bisiacchi, L. Reyes et al., "Three-axis air-bearing based platform for small satellite attitude determination and control simulation," *Journal of Applied Research and Technology*, vol. 3, no. 3, pp. 222–237, 2005.

[11] F. Terui, "Position and attitude control of a spacecraft by sliding mode control," in *Proceedings of the American Control Conference*, pp. 217–221, Phildelphia, Pa, USA, 1998.

[12] D. T. Stansbery and J. R. Cloutier, "Position and attitude control of a spacecraft using the state-dependent Riccati equation technique," in *Proceedings of the American Control Conference*, pp. 1867–1871, Chicago, Ill, USA, 2000.

[13] Y. G. H. A. D. Z. Cheng, *Nonlinear System Analysis and Control*, Science Press, Beijing, China, 1st edition, 2005.

[14] G. Q. Wu, S. N. Wu, and Z. G. Wu, "Robust finite-time control for spacecraft with coupled translation and attitude dynamics," *Mathematical Problems in Engineering*, vol. 2013, Article ID 707485, 7 pages, 2013.

[15] J. Liu, Q. Gao, Z. Liu, and Y. Li, "Attitude control for astronaut assisted robot in the space station," *International Journal of Control, Automation and Systems*, vol. 14, no. 4, pp. 1082–1095, 2016.

[16] Q. Gao, J. Liu, T. Tian, and Y. Li, "Free-flying dynamics and control of an astronaut assistant robot based on fuzzy sliding mode algorithm," *Acta Astronautica*, vol. 138, pp. 462–474, 2017.

[17] B. P. Malladi, S. Di Cairano, and A. Weiss, "Nonlinear model predictive control of coupled rotational-translational spacecraft relative motion," in *2019 American Control Conference (ACC)*, pp. 3581–3586, Philadelphia, PA, USA, 2019.

[18] Y. K. Nakka, R. C. Foust, E. S. Lupu et al., "A six degree-of-freedom spacecraft dynamics simulator for formation control research," in *2018 AAS/AIAA Astrodynamics Specialist Conference*, Snowbird, UT, USA, 2018.

[19] J. Zhao, S. Yan, and J. Wu, "Analysis of parameter sensitivity of space manipulator with harmonic drive based on the revised response surface method," *Acta Astronautica*, vol. 98, pp. 86–96, 2014.

[20] J. L. Zhao, S. Z. Yan, J. N. Wu, W. Ma, and Z. Y. Han, "Thermodynamic analysis of a space station remote manipulator with a harmonic drive that considers an integrated thermal protection layer," *SCIENCE CHINA Technological Sciences*, vol. 58, no. 11, pp. 1884–1893, 2015.

[21] H. Liu, L. Guo, and Y. Zhang, "An anti-disturbance PD control scheme for attitude control and stabilization of flexible spacecrafts," *Nonlinear Dynamics*, vol. 67, no. 3, pp. 2081–2088, 2012.

[22] C. Nie, Z. Zheng, and M. Zhu, "Three-dimensional path-following control of a robotic airship with reinforcement learning," *International Journal of Aerospace Engineering*, vol. 2019, Article ID 7854173, 12 pages, 2019.

[23] P. C. Hughes, *Spacecraft attitude dynamics*, Courier Corporation, 2012.

[24] T. Yang, S. Yan, W. Ma, and Z. Han, "Joint dynamic analysis of space manipulator with planetary gear train transmission," *Robotica*, vol. 34, no. 5, pp. 1042–1058, 2016.

[25] Z. B. Zhang, X. H. Li, J. P. An, W. X. Man, and G. H. Zhang, "Model-free attitude control of spacecraft based on PID-guide TD3 algorithm," *International Journal of Aerospace Engineering*, vol. 2020, Article ID 8874619, 13 pages, 2020.

[26] H. Shu, X. Guo, and H. Shu, "PID neural networks in multivariable systems," in *Proceedings of the IEEE Internatinal Symposium on Intelligent Control*, pp. 440–444, Vancouver, BC, Canada, 2002.

[27] H. Yin, W. Yi, J. Wu, K. Wang, and J. Guan, "Adaptive fuzzy neural network PID algorithm for BLDCM speed control system," *Mathematics*, vol. 10, no. 1, p. 118, 2022.

[28] R. Ma, S. Li, B. Zhang, and Z. Li, "Towards fast and robust real image denoising with attentive neural network and PID controller," *IEEE Transactions on Multimedia*, vol. 24, pp. 2366–2377, 2022.

[29] J. Kang, W. Meng, A. Abraham, and H. Liu, "An adaptive PID neural network for complex nonlinear system control," *Neurocomputing*, vol. 135, pp. 79–85, 2014.

[30] M. Labbadi and M. Cherkaoui, "Robust integral terminal sliding mode control for quadrotor UAV with external disturbances," *International Journal of Aerospace Engineering*, vol. 2019, Article ID 2016416, 10 pages, 2019.

Hindawi
International Journal of Aerospace Engineering
Volume 2022, Article ID 9626569, 16 pages
https://doi.org/10.1155/2022/9626569

Research Article

Kinodynamic Trajectory Optimization of Dual-Arm Space Robot for Stabilizing a Tumbling Target

Lei Yan,[1] Han Yuan,[1] Wenfu Xu,[1] Zhonghua Hu,[1] and Bin Liang[2]

[1]*School of Mechanical Engineering and Automation, Harbin Institute of Technology, Shenzhen, China*
[2]*Department of Automation, Tsinghua University, Beijing, China*

Correspondence should be addressed to Wenfu Xu; wfxu@hit.edu.cn

Received 29 April 2022; Accepted 5 July 2022; Published 18 August 2022

Academic Editor: Xin Zhang

Capturing and stabilizing tumbling targets using dual-arm space robots are very crucial to on-orbit servicing task. However, it is still very challenging due to the complex dynamics coupling and closed-chain constraints between the manipulators, the base, and the target. In this paper, a kinodynamic trajectory optimization method is proposed to generate the motion of a dual-arm space robot for stabilizing the captured tumbling target, which is formulated and solved as a nonlinear programming problem using direct collocation. Instead of optimizing the trajectory of each joint with the dynamics model of space robot, this method optimizes the trajectory of the tumbling target while considering the kinematics and dynamics constraints between the two arms and the target simultaneously. The objective function of the optimization is defined as weighted detumbling time, base disturbance, and manipulability, in order to avoid singularity and save the energy of space robot for further manipulation. Several physical simulations are carried out to validate the proposed method.

1. Introduction

With growing scientific research and commercial applications in space, more and more malfunctioning satellites and space debris are occupying precious orbital resources which will bring a great threat to the safety of on-orbit spacecrafts [1, 2]. In order to utilize or remove them, space robots have been studied and developed for many years [3]. For capturing a space target, the space robotic system mainly adopts the following two capturing methods, i.e., stiff-connection capturing and flexible-connection capturing [4]. The flexible-connection capturing method using tethered flying net [5] or flying gripper [6] can deal with a variety of targets even without any requirement on rendezvous and docking. This method also allows a long capture distance between the target and the servicing spacecraft and a broad range of size and shape of the target object. However, it has limitations on dexterous manipulation of the captured target, such as on-orbit maintenance and space assembly. Therefore, the stiff-connection capturing method using robotic arms [7] will still be a promising method for on-orbit servicing of noncooperative space target.

Moosavian and Papadopoulos summarize the modeling, planning, and control methods for free-floating space robots [8]. In order to solve the dynamics coupling between the base and the manipulator, the generalized Jacobian matrix is proposed for single-arm space robots [9]. Zhang et al. design an efficient decoupling controller based on the time-delay estimation (TDE) and the supertwisting control (STC), which can linearize the nonlinear dynamics of space robot and drive the state variables to converge to the equilibrium point robustly [10]. Compared with single-arm space robot, multiarm robotic system can perform more dexterous, flexible, and complex tasks [11, 12]. Yoshida et al. designs a resolved motion-rate coordinated control method for dual-arm space robot in which one of the manipulators tracks the desired trajectory while the other maintains the orientation of the satellite base. However, these methods can not be used for multiarm coordinated planning of space robots. Similar to relative Jacobian matrix [13] for terrestrial

dual-arm robots, a generalized relative Jacobian matrix is proposed for multiarm space robots which can easily plan and control the relative motion between the arms while considering the dynamics coupling between the arms and the base [14]. For multiarm space robots, the base can be kept to be inertially stable during multiarm coordinated manipulation by reactionless motion planning [15]. However, most of these methods with Jacobian projection can be only used for nonconstrained quasistatic planning and control problem.

One of the most challenging on-orbit servicing tasks for space robots is to capture a space target with nonzero momentum. During the capture phase, several components need to be considered, including path planning to capture the tumbling target, hybrid control of the motion and the contact force for the end-effectors, coordinated control of the base attitude, and parameter identification of the tumbling target [16]. Considering the grasping force limitation, parameter uncertainty of the target and arbitrary detumbling motion, a time-optimal control method [17] is proposed for free-floating space robots stabilizing a tumbling target. However, the limitation of this method is the parameterized end-effector velocity. For capturing and stabilizing a tumbling space target with uncertain dynamics, Aghili [18] proposes an optimal motion planning scheme which will generate the end-effector trajectories for both pre- and postcapture phases. Further, a two-layer optimization is proposed to yield both end-effector forces and contact locations for cooperative manipulation of an on-orbit passive objects [19]. For the postcontact phase of capturing a tumbling target in space, Zhang et al. [20] present a control scheme and parameter identification technique for postcapture stabilization of unknown tumbling target, in which the manipulator's motion is used to compensate torque limitation. A detumbling strategy is also proposed to minimize the detumbling time and control torques, in which the target's trajectory is represented by quartic Bézier curves and the optimal solution is found by adaptive particle swarm optimization algorithm [21]. Joint-velocity limits are further considered in the detumbling and stabilization manipulation [22]. In order to limit the target attitude motion as well as interaction torque at the grasping point, a time-optimal control problem (OCP) is formulated and solved using the calculus of variations method with a highly accurate solution [23]. Taking advantage of the coupling between dynamics of translational and rotational systems, Aghili proposes an optimal controller which can damp out both translational and rotational motions collaboratively and simultaneously [24].

During postcapture manipulation, the optimal detumbling motion of space robot should be generated to reduce the momentum of the tumbling target with minimal base disturbance, while satisfying equality and inequality constraints simultaneously. A purely kinematic trajectory optimization method is proposed to manipulate the in-grasp object with relaxed-rigidity constraints [25]. However, it can not be used for heavy object manipulation with nonnegligible dynamics. Recently, a nonlinear trajectory optimization method is proposed to generate the trajectory for approaching the tumbling target during precontact phase [26] and solved by direct collocation method [27]. Similarly, neither the object dynamics nor interaction between the object and the space robot is considered for this precontact trajectory optimization. However, the closed-chain constraints and coupling dynamics between the object and space robot make the postcapture manipulation much more challenging. In this paper, we will formulate the postcapture manipulation as a trajectory optimization problem in which the base disturbance will be minimized. Betts et al. [28] reviewed the numerical methods for trajectory optimization and discussed the direct and indirect methods. In this paper, we will adopt the direct collocation method to solve the trajectory optimization problem of postcapture manipulation.

Kinodynamic motion planning [29] is first proposed to solve motion planning problem subject to simultaneous kinematics and dynamics constraints. In this paper, a kinodynamic trajectory optimization framework is proposed for generating dual-arm detumbling motion while satisfying the closed-chain kinodynamic constraints between the object and dual-arm space robot. The main contributions of this paper are as follows: (1) a kinodynamic trajectory optimization framework is proposed to minimize the base disturbance of dual-arm space robot for postcapture manipulation of tumbling target; (2) the base disturbance of dual-arm space robot during detumbling manipulation is derived as a function of the position vector of the tumbling target and the total detumbling force exerted on the tumbling target, without calculating dual-arm operational forces, respectively; (3) instead of optimizing the trajectory for each single joint of space robot, the optimal detumbling motion of dual-arm space robot is generated from the optimal trajectory of the tumbling target according to closed-chain kinodynamic constraints.

The remainder of this paper is organized as follows. In Section 2, the dynamics modelling of dual-arm space robot and the tumbling target is presented. In Section 3, the kinodynamic trajectory optimization framework for postcapture manipulation is introduced, and the detailed formulation is presented. In Section 4, the proposed method is verified and compared through physical simulations with different objective functions and initial conditions. Finally, the conclusion and future work are presented in Section 5.

2. Modelling of Dual-Arm Space Robots and Tumbling Target

2.1. Modelling of Dual-Arm Space Robots. As shown in Figure 1(a), the dual-arm space robotic system consists of a satellite base and two central-symmetrically mounted manipulators. The initial and final states of dual-arm space robot and tumbling target during detumbling manipulation are shown in Figure 1(b). The degrees of freedom (DOF) of manipulator Arm-k is denoted by n_k. In this paper, each arm is an S-R-S (spherical-revolute-spherical) 7 DOF redundant manipulator. Moreover, any two adjacent joints are perpendicular without offset. All the variables in Figure 1 are defined in Table 1. The reference coordinate system of

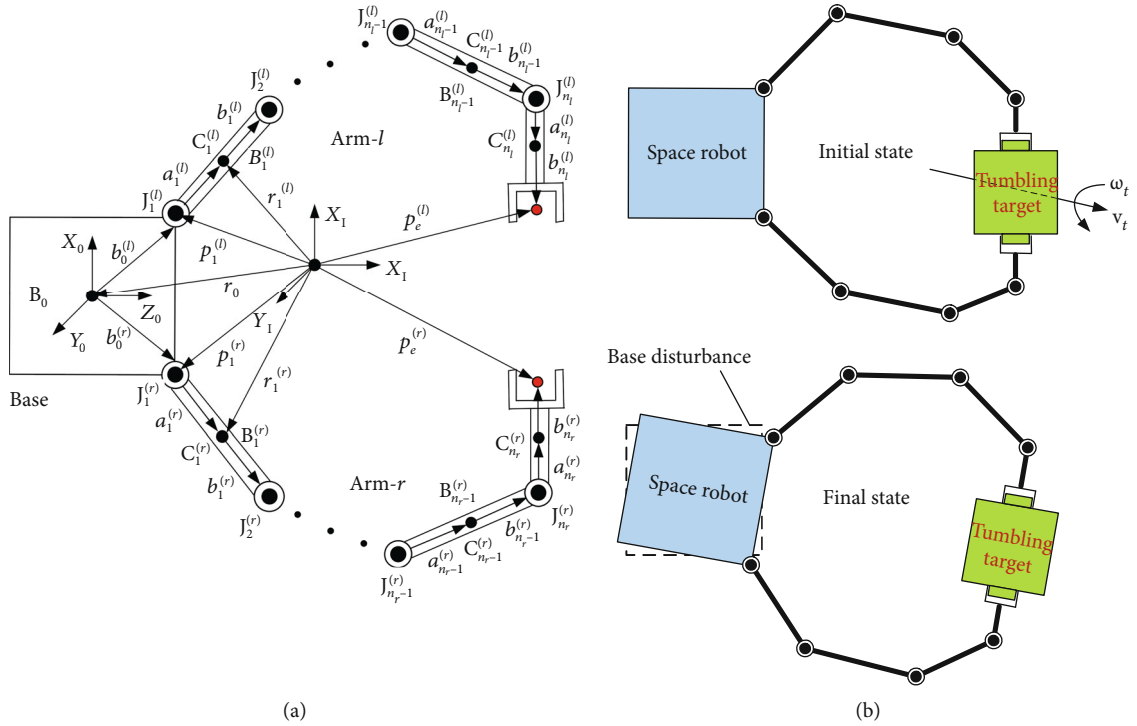


Figure 1: Schematic diagram of dual-arm space robot and the detumbling manipulation.

Table 1: Nomenclature.

Variable	Description
$J_i^{(k)}$	The i^{th} joints of Arm-k ($i = 1, \cdots, n_k$; $k = l, r$)
$\theta_i^{(k)}$	The joint angle of the i^{th} joint of Arm-k
$\mathbf{k}_i^{(k)}$	The unit vector representing the rotation direction of $J_i^{(k)}$
$C_i^{(k)}$	The CoM of rigid body $B_i^{(k)}$
$\mathbf{a}_i^{(k)}, \mathbf{b}_i^{(k)}$	The position vectors from $J_i^{(k)}$ to $C_i^{(k)}$ and $C_i^{(k)}$ to $J_{i+1}^{(k)}$, respectively
$\mathbf{b}_0^{(k)}$	The position vector from the CoM of B_0 to joint $J_1^{(k)}$
$\mathbf{r}_i^{(k)}, \mathbf{r}_0, \mathbf{r}_g$	The position vector of $C_i^{(k)}$, base's CoM, and space robot's CoM, respectively
$\mathbf{p}_i^{(k)}, \mathbf{p}_e^{(k)}$	The position vector of $J_i^{(k)}$ and the end-effector of Arm-k, respectively
$\boldsymbol{\Theta}^{(k)}$	The joint angle vector of Arm-k; $\boldsymbol{\Theta}^{(k)} = \left[\theta_1^{(k)}\theta_2^{(k)} \cdots \theta_7^{(k)}\right]^T$
$\mathbf{v}_0, \boldsymbol{\omega}_0$	The linear and angular velocities of B_0
$\mathbf{v}_i^{(k)}, \boldsymbol{\omega}_i^{(k)}$	The linear and angular velocities of $B_i^{(k)}$
$\mathbf{v}_e^{(k)}, \boldsymbol{\omega}_e^{(k)}$	The linear and angular velocities of the end-effector of Arm-k
$\mathbf{v}_t, \boldsymbol{\omega}_t$	The linear and angular velocities of the target
$m_0, m_i^{(k)}, M$	The masses of B_0, $B_i^{(k)}$, and the whole system, respectively
$\mathbf{I}_0, \mathbf{I}_i^{(k)}$	The inertia matrices of B_0 and $B_i^{(k)}$ in terms of the body CoM frame
$\mathbf{E}_n$, $\mathbf{O}_n$	The $n \times n$ identity and zero matrices

Arm-k is the same as the coordinate system of the base of Arm-k. The center of mass (CoM) coordinate system of each body $B_i^{(k)}$ has the same orientation with the coordinate system of each joint $J_i^{(k)}$. The reference coordinate system of each variable and its corresponding derivative is denoted by the left superscript in the rest of this paper. Unless specified, all the variables are expressed in the inertial coordinate system "$\sum I$" which is omitted for simplification.

As shown in Figure 1, the end-effector velocity of each manipulator can be derived as follows:

$$\dot{\mathbf{x}}_e^{(k)} = \begin{bmatrix} \mathbf{v}_e^{(k)} \\ \boldsymbol{\omega}_e^{(k)} \end{bmatrix} = \mathbf{J}_b^{(k)} \begin{bmatrix} \mathbf{v}_0 \\ \boldsymbol{\omega}_0 \end{bmatrix} + \mathbf{J}_m^{(k)} \dot{\boldsymbol{\Theta}}^{(k)}, \tag{1}$$

where $\mathbf{J}_b^{(k)}$ and $\mathbf{J}_m^{(k)}$ are the Jacobian matrices related to the base and the manipulator, respectively, which can be calculated by the following equations:

$$\mathbf{J}_b^{(k)} = \begin{pmatrix} \mathbf{E}_3 & -\mathbf{p}_{0e}^{\times\,(k)} \\ \mathbf{O}_3 & \mathbf{E}_3 \end{pmatrix},$$
$$\mathbf{J}_m^{(k)} = \begin{bmatrix} \mathbf{k}_1^{(k)} \times \left(\mathbf{p}_e^{(k)} - \mathbf{p}_1^{(k)}\right) & \cdots & \mathbf{k}_{n_k}^{(k)} \times \left(\mathbf{p}_e^{(k)} - \mathbf{p}_{n_k}^{(k)}\right) \\ \mathbf{k}_1^{(k)} & \cdots & \mathbf{k}_{n_k}^{(k)} \end{bmatrix}, \tag{2}$$

where $\mathbf{p}_{0e}^{(k)}$ is the vector from the CoM of the base to the end-effector of Arm-k, $\mathbf{p}_{0e}^{(k)} = \mathbf{p}_e^{(k)} - \mathbf{r}_0$; $\mathbf{r}^\times$ is the skew symmetric matrix of $\mathbf{r}$, i.e.,

$$\text{if } r = \begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix}, \text{ then } r^\times = \begin{bmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{bmatrix}. \tag{3}$$

The linear and angular momentums of dual-arm space robots can be expressed as follows:

$$\mathbf{P} = \left[M\mathbf{E}_3 \; M\mathbf{r}_{0g}^{\times\,\mathrm{T}}\right] \begin{bmatrix} \mathbf{v}_0 \\ \boldsymbol{\omega}_0 \end{bmatrix} + \mathbf{J}_{Tw}^l \dot{\boldsymbol{\Theta}}^l + \mathbf{J}_{Tw}^r \dot{\boldsymbol{\Theta}}^r,$$
$$\mathbf{L} = \mathbf{I}_w \boldsymbol{\omega}_0 + M\mathbf{r}_g^\times \mathbf{v}_0 + \mathbf{I}_\phi^l \dot{\boldsymbol{\Theta}}^l + \mathbf{I}_\phi^r \dot{\boldsymbol{\Theta}}^r = \left[M\mathbf{r}_g^\times \; \mathbf{I}_w\right] \begin{bmatrix} \mathbf{v}_0 \\ \boldsymbol{\omega}_0 \end{bmatrix} + \mathbf{I}_\phi^l \dot{\boldsymbol{\Theta}}^l + \mathbf{I}_\phi^r \dot{\boldsymbol{\Theta}}^r, \tag{4}$$

where

$$\mathbf{J}_{Tw}^{(k)} = \sum_{i=1}^{n_k} \left(m_i^{(k)} \mathbf{J}_{Ti}^{(k)}\right),$$
$$\mathbf{J}_{Ti}^{(k)} = \left[\mathbf{k}_1^{(k)} \times \left(\mathbf{r}_i^{(k)} - \mathbf{p}_1^{(k)}\right), \cdots, \mathbf{k}_i^{(k)} \times \left(\mathbf{r}_i^{(k)} - \mathbf{p}_i^{(k)}\right), 0, \cdots, 0\right],$$
$$\mathbf{J}_{Ri}^{(k)} = \left[\mathbf{k}_1^{(k)}, \cdots, \mathbf{k}_i^{(k)}, 0, \cdots, 0\right],$$
$$\mathbf{I}_w = \mathbf{I}_0 + \sum_{k=l,r} \left\{ \sum_{i=1}^{n_k} \left[\mathbf{I}_i^{(k)} + m_i^{(k)} r_i^{\times(k)} \left(\mathbf{r}_{0i}^{\times\,(k)}\right)^T\right] \right\},$$
$$\mathbf{I}_\phi^{(k)} = \sum_{i=1}^{n_k} \left(\mathbf{I}_i^{(k)} \mathbf{J}_{Ri}^{(k)} + m_i^{(k)} \mathbf{r}_i^{\times(k)} \mathbf{J}_{Ti}^{(k)}\right),$$
$$\mathbf{r}_{0g} = \mathbf{r}_g - \mathbf{r}_0. \tag{5}$$

For the free-floating space robots, the linear and angular momentums are conserved as the environmental force caused by solar pressure, air drag, and microgravity is negligible. The whole system satisfies the following holonomic and nonholonomic constraints:

$$\begin{bmatrix} \mathbf{P} \\ \mathbf{L} \end{bmatrix} = \begin{pmatrix} M\mathbf{E}_3 & M\mathbf{r}_{0g}^{\times\,T} \\ M\mathbf{r}_g^\times & \mathbf{I}_w \end{pmatrix} \begin{bmatrix} \mathbf{v}_0 \\ \boldsymbol{\omega}_0 \end{bmatrix} + \begin{bmatrix} \mathbf{J}_{Tw}^l \\ \mathbf{I}_\phi^l \end{bmatrix} \dot{\boldsymbol{\Theta}}^l + \begin{bmatrix} \mathbf{J}_{Tw}^r \\ \mathbf{I}_\phi^r \end{bmatrix} \dot{\boldsymbol{\Theta}}^r$$
$$= \mathbf{H}_B \begin{bmatrix} \mathbf{v}_0 \\ \boldsymbol{\omega}_0 \end{bmatrix} + \mathbf{H}_{bm}^l \dot{\boldsymbol{\Theta}}^l + \mathbf{H}_{bm}^r \dot{\boldsymbol{\Theta}}^r. \tag{6}$$

From (6), the following relationship between the base's motion and arms' motion can be obtained:

$$\begin{bmatrix} \mathbf{v}_0 \\ \boldsymbol{\omega}_0 \end{bmatrix} = \mathbf{J}_{bm}^l \dot{\boldsymbol{\Theta}}^l + \mathbf{J}_{bm}^r \dot{\boldsymbol{\Theta}}^r = \left[\mathbf{J}_{bm}^l \; \mathbf{J}_{bm}^r\right] \begin{bmatrix} \dot{\boldsymbol{\Theta}}^l \\ \dot{\boldsymbol{\Theta}}^r \end{bmatrix} = \mathbf{J}_{bm} \begin{bmatrix} \dot{\boldsymbol{\Theta}}^l \\ \dot{\boldsymbol{\Theta}}^r \end{bmatrix}, \tag{7}$$

where

$$\mathbf{J}_{bm}^{(k)} = \begin{bmatrix} -\mathbf{r}_{0g}^\times \mathbf{I}_s^{-1} \mathbf{I}_\Theta^{(k)} - \dfrac{\mathbf{J}_{Tw}^{(k)}}{M} \\ -\mathbf{I}_s^{-1} \mathbf{I}_\Theta^{(k)} \end{bmatrix},$$
$$\mathbf{I}_s = \left(M\mathbf{r}_g^\times \mathbf{r}_{0g}^\times + \mathbf{I}_w\right),$$
$$\mathbf{I}_\Theta^{(k)} = \left(\mathbf{I}_\phi^{(k)} - \mathbf{r}_g^\times \mathbf{J}_{Tw}^{(k)}\right). \tag{8}$$

Substituting (7) into (1) yields the end-effector velocity of each manipulator:

$$\dot{\mathbf{x}}_e^l = \mathbf{J}_b^l \begin{bmatrix} \mathbf{v}_0 \\ \boldsymbol{\omega}_0 \end{bmatrix} + \mathbf{J}_m^l \dot{\boldsymbol{\Theta}}^l = \left(\mathbf{J}_b^l \mathbf{J}_{bm}^l + \mathbf{J}_m^l\right) \dot{\boldsymbol{\Theta}}^l + \mathbf{J}_b^l \mathbf{J}_{bm}^r \dot{\boldsymbol{\Theta}}^r = \mathbf{J}_{ll} \dot{\boldsymbol{\Theta}}^l + \mathbf{J}_{lr} \dot{\boldsymbol{\Theta}}^r,$$
$$\dot{\mathbf{x}}_e^r = \mathbf{J}_b^r \begin{bmatrix} \mathbf{v}_0 \\ \boldsymbol{\omega}_0 \end{bmatrix} + \mathbf{J}_m^r \dot{\boldsymbol{\Theta}}^r = \mathbf{J}_b^r \mathbf{J}_{bm}^l \dot{\boldsymbol{\Theta}}^l + (\mathbf{J}_b^r \mathbf{J}_{bm}^r + \mathbf{J}_m^r) \dot{\boldsymbol{\Theta}}^r = \mathbf{J}_{rl} \dot{\boldsymbol{\Theta}}^l + \mathbf{J}_{rr} \dot{\boldsymbol{\Theta}}^r. \tag{9}$$

Therefore, the generalized Jacobian matrix of dual-arm space robots can be derived as

$$\begin{bmatrix} \dot{\mathbf{x}}_e^l \\ \dot{\mathbf{x}}_e^r \end{bmatrix} = \begin{bmatrix} \mathbf{J}_{ll} & \mathbf{J}_{lr} \\ \mathbf{J}_{rl} & \mathbf{J}_{rr} \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{\Theta}}^l \\ \dot{\boldsymbol{\Theta}}^r \end{bmatrix} = \mathbf{J}_g \dot{\boldsymbol{\Theta}}, \tag{10}$$

where $\mathbf{J}_{ll} = \mathbf{J}_b^l \mathbf{J}_{bm}^l + \mathbf{J}_m^l$, $\mathbf{J}_{lr} = \mathbf{J}_b^l \mathbf{J}_{bm}^r$, $\mathbf{J}_{rl} = \mathbf{J}_b^r \mathbf{J}_{bm}^l$, $\mathbf{J}_{rr} = \mathbf{J}_b^r \mathbf{J}_{bm}^r + \mathbf{J}_m^r$, and $\mathbf{J}_{bm}^{(k)}$ is the coupling Jacobian matrix between the base and the manipulator. The derivation details can be found in [14].

2.2. Motion Equation of Tumbling Target. Assuming that the target is tumbling with an initial velocity, its inertia parameters m_t and ${}^0\mathbf{I}_t$ are known or can be estimated during the

precontact phase and contact phase [30, 31]. Therefore, the motion equation of the target which is captured by the two arms of space robot can be expressed in the base frame as

$$ {}^0\mathbf{M}_t\,{}^0\ddot{\mathbf{x}}_t + {}^0\mathbf{C}_t = {}^0\mathbf{F}_t, \tag{11} $$

$$ {}^0\boldsymbol{F}_t = {}^0\boldsymbol{G}_l\,{}^0\boldsymbol{F}_l + {}^0\boldsymbol{G}_r\,{}^0\boldsymbol{F}_r + {}^0\boldsymbol{G}_{\text{ext}}\,{}^0\boldsymbol{F}_{\text{ext}}, \tag{12} $$

where ${}^0\mathbf{M}_t$ is the inertia matrix, ${}^0\mathbf{C}_t$ is the Coriolis and centrifugal force, and ${}^0\mathbf{F}_t$ is the total force exerted on the target; it also refers to the detumbling force of the target in this paper; ${}^0\mathbf{F}_k = \begin{bmatrix} {}^0\mathbf{f}_k^T & {}^0\boldsymbol{\tau}_k^T \end{bmatrix}^T$ is the operational force of Arm-k; ${}^0F_{\text{ext}}$, ${}^0G_{\text{ext}}$ are the external force exerted on the target and the corresponding grasp matrix, respectively. The external force ${}^0\mathbf{F}_{\text{ext}}$ caused by solar pressure, air drag, and microgravity is order of magnitude less than the operational forces exerted by the manipulator's end-effector, and hence, is negligible. The inertia matrix and the Coriolis and centrifugal force can be obtained as follows:

$$ {}^0\mathbf{M}_t = \begin{bmatrix} m_t\mathbf{E}_3 & \mathbf{O}_3 \\ \mathbf{O}_3 & {}^0\mathbf{I}_t \end{bmatrix}, {}^0\mathbf{C}_t = \begin{bmatrix} \mathbf{O}_3 \\ {}^0\boldsymbol{\omega}_t \times \left({}^0\mathbf{I}_t\,{}^0\boldsymbol{\omega}_t\right) \end{bmatrix}. \tag{13} $$

On the other hand, when the end-effectors of dual-arm space robot and the grasping points of the target are connected, the dual-arm space robot and the target form a closed-chain constraint. Considering the postcapture manipulation, the end-effectors of the two arms will be fixed with the grasping points. Therefore, we can have the following position-level constraint:

$$ {}^0\mathbf{T}_t = {}^0\mathbf{T}_1^{(l)}\,{}^1\mathbf{T}_e^{(l)}\,{}^e\mathbf{T}_t^{(l)} = {}^0\mathbf{T}_1^{(r)}\,{}^1\mathbf{T}_e^{(r)}\,{}^e\mathbf{T}_t^{(r)}, \tag{14} $$

where ${}^0\mathbf{T}_t$ represents the homogeneous transformation matrix (HTM) of the target with respect to the base, ${}^0\mathbf{T}_1^{(k)}$ represents the HTM of the reference coordinate system of Arm-k with respect to the base, ${}^1\mathbf{T}_e^{(k)}$ represents the HTM of the end-effector with respect to the reference coordinate system of Arm-k, and ${}^e\mathbf{T}_t^{(k)}$ represents the HTM of the target with respect to the grasp coordinate system of Arm-k.

Therefore, given the desired velocity of the target ${}^0\dot{\mathbf{x}}_t$ with respect to the base frame of dual-arm space robot, the corresponding velocity of the end-effector of Arm-k ${}^0\dot{\mathbf{x}}_e^{(k)}$ can be obtained as

$$ {}^0\dot{\mathbf{x}}_e^{(k)} = \begin{bmatrix} \mathbf{E}_3 & -\left({}^0\mathbf{r}_{ek}^t\right)^\times \\ \mathbf{O}_3 & \mathbf{E}_3 \end{bmatrix} {}^0\dot{\mathbf{x}}_t = {}^0\mathbf{G}_k^T\,{}^0\dot{\mathbf{x}}_t, \tag{15} $$

where ${}^0\mathbf{r}_{ek}^t$ is the position vector from the CoM of target to the grasping point; ${}^0\mathbf{G}_k (k = l, r)$ is the grasp matrix [32] of Arm-k.

Furthermore, by differentiating (15), the acceleration constraint can be obtained as

$$ {}^0\ddot{\mathbf{x}}_e^{(k)} = {}^0\mathbf{G}_k^T\,{}^0\ddot{\mathbf{x}}_t + {}^0\dot{\mathbf{G}}_k^T\,{}^0\dot{\mathbf{x}}_t. \tag{16} $$

The kinematics of each manipulator of dual-arm space robot can be written as

$$ \begin{aligned} {}^1\dot{\mathbf{x}}_e^{(k)} &= {}^1\mathbf{R}_0^{(k)}\,{}^0\dot{\mathbf{x}}_e^{(k)} = \mathbf{J}_m^{(k)}\dot{\boldsymbol{\Theta}}_d^{(k)}, \\ {}^1\ddot{\mathbf{x}}_e^{(k)} &= {}^1\mathbf{R}_0^{(k)}\,{}^0\ddot{\mathbf{x}}_e^{(k)} = \mathbf{J}_m^{(k)}\ddot{\boldsymbol{\Theta}}_d^{(k)} + \dot{\mathbf{J}}_m^{(k)}\dot{\boldsymbol{\Theta}}_d^{(k)}, \end{aligned} \tag{17} $$

where ${}^1\mathbf{R}_0^{(k)}$ is the rotation matrix from the base frame to the reference (base) frame of Arm-k, which is a constant matrix as the base of Arm-k is fixed with the satellite base.

Given the desired velocity and acceleration of the end-effector, the desired joint velocity and acceleration can be obtained directly by the inverse kinematics of the manipulator:

$$ \begin{aligned} \dot{\boldsymbol{\Theta}}_d^{(k)} &= \left(\mathbf{J}_m^{(k)}\right)^\dagger {}^1\dot{\mathbf{x}}_e^{(k)}, \\ \ddot{\boldsymbol{\Theta}}_d^{(k)} &= \left(\mathbf{J}_m^{(k)}\right)^\dagger \left({}^1\ddot{\mathbf{x}}_e^{(k)} - \dot{\mathbf{J}}_m^{(k)}\dot{\boldsymbol{\Theta}}_d^{(k)}\right), \end{aligned} \tag{18} $$

where $\left(\mathbf{J}_m^{(k)}\right)^\dagger$ is the Moore-Penrose pseudoinverse of $\mathbf{J}_m^{(k)}$; for each 7 DOF redundant manipulator of dual-arm space robot, the Moore-Penrose pseudoinverse is used to obtain the least-square solution of differential kinematics with minimum norm.

2.3. Base Disturbance Caused by Detumbling Manipulation. During the detumbling manipulation of dual-arm space robot, the base disturbance resulted from the operational forces of two arms to detumble the tumbling target is analyzed in this section. The dynamic constraints between the two arms and the target are shown in (12), where ${}^0\mathbf{F}_{\text{ext}}$ is equal to zero. Therefore, given the desired motion of the target, the operational forces of the two arms can be obtained. However, there is no unique solution for (12). Many existed algorithms can be used to solve this problem, for example, the master-slave or shared force control proposed in [33] and the optimal distribution method which minimized the squared operational forces proposed in [32].

As the main purpose in this paper is to minimize the detumbling time and base disturbance caused by the detumbling maneuver during the postcapture phase, we only consider the total disturbance force exerted on the base. Therefore, the total disturbance force exerted on the base which is caused by the operational forces of two arms is calculated as follows:

$$
\begin{aligned}
{}^0\boldsymbol{F}_{\mathrm{dis}}^{f} &= -\begin{bmatrix} \boldsymbol{E}_3 & \boldsymbol{O}_3 \\ {}^0\boldsymbol{r}_{el}^{\times} & \boldsymbol{E}_3 \end{bmatrix} {}^0\mathbf{F}_l - \begin{bmatrix} \boldsymbol{E}_3 & \boldsymbol{O}_3 \\ {}^0\boldsymbol{r}_{er}^{\times} & \boldsymbol{E}_3 \end{bmatrix} {}^0\boldsymbol{F}_r \\
&= -\begin{bmatrix} \boldsymbol{E}_3 & \boldsymbol{O}_3 \\ \left({}^0\boldsymbol{r}_t + {}^0\boldsymbol{r}_{el}^{t}\right)^{\times} & \boldsymbol{E}_3 \end{bmatrix} {}^0\boldsymbol{F}_l - \begin{bmatrix} \boldsymbol{E}_3 & \boldsymbol{O}_3 \\ \left({}^0\boldsymbol{r}_t + {}^0\boldsymbol{r}_{er}^{t}\right)^{\times} & \boldsymbol{E}_3 \end{bmatrix} {}^0\boldsymbol{F}_r \\
&= -\begin{bmatrix} \boldsymbol{E}_3 & \boldsymbol{O}_3 \\ {}^0\boldsymbol{r}_t^{\times} & \boldsymbol{E}_3 \end{bmatrix} \begin{bmatrix} \boldsymbol{E}_3 & \boldsymbol{O}_3 \\ {}^0\mathbf{r}_{el}^{t\times} & \boldsymbol{E}_3 \end{bmatrix} {}^0\boldsymbol{F}_l - \begin{bmatrix} \boldsymbol{E}_3 & \boldsymbol{O}_3 \\ {}^0\boldsymbol{r}_t^{\times} & \boldsymbol{E}_3 \end{bmatrix} \begin{bmatrix} \boldsymbol{E}_3 & \boldsymbol{O}_3 \\ {}^0\boldsymbol{r}_{er}^{t\times} & \boldsymbol{E}_3 \end{bmatrix} {}^0\boldsymbol{F}_r \\
&= -\begin{bmatrix} \boldsymbol{E}_3 & \boldsymbol{O}_3 \\ {}^0\boldsymbol{r}_t^{\times} & \boldsymbol{E}_3 \end{bmatrix} \left\{ \begin{bmatrix} \boldsymbol{E}_3 & \boldsymbol{O}_3 \\ {}^0\mathbf{r}_{el}^{t\times} & \boldsymbol{E}_3 \end{bmatrix} {}^0\boldsymbol{F}_l + \begin{bmatrix} \boldsymbol{E}_3 & \boldsymbol{O}_3 \\ {}^0\boldsymbol{r}_{er}^{t\times} & \boldsymbol{E}_3 \end{bmatrix} {}^0\boldsymbol{F}_r \right\},
\end{aligned}
\tag{19}
$$

where ${}^0\mathbf{r}_{ek}$ is the position vector from the CoM of base to the end-effector of Arm-k and ${}^0\mathbf{r}_t$ is the position vector from the CoM of base to the CoM of target.

Combining equations (12) and (19) and the external force ${}^0\mathbf{F}_{\mathrm{ext}}$ on the target is equal to zero, then we can have:

$$
{}^0\mathbf{F}_{\mathrm{dis}}^{f} = -\begin{bmatrix} \mathbf{E}_3 & \mathbf{O}_3 \\ {}^0\mathbf{r}_t^{\times} & \mathbf{E}_3 \end{bmatrix} \left({}^0\mathbf{G}_l\, {}^0\mathbf{F}_l + {}^0\mathbf{G}_r\, {}^0\mathbf{F}_r \right) = -\begin{bmatrix} \mathbf{E}_3 & \mathbf{O}_3 \\ {}^0\mathbf{r}_t^{\times} & \mathbf{E}_3 \end{bmatrix} {}^0\mathbf{F}_t. \tag{20}
$$

It can be seen from (20) that the disturbance force ${}^0\mathbf{F}_{\mathrm{dis}}^{f}$ exerted on the base is related to both the position vector ${}^0\mathbf{r}_t$ and the detumbling force ${}^0\mathbf{F}_t$ of the target. The disturbance force should be minimized in order to decrease the base disturbance. Therefore, in the kinodynamic trajectory optimization method presented in the Section 3, the base disturbance force ${}^0\mathbf{F}_{\mathrm{dis}}^{f}$ will be minimized as an objective function.

3. Kinodynamic Trajectory Optimization for Detumbling Manipulation

Generally speaking, the process of capturing a tumbling target in space can be decomposed into three phases: the precontact, contact, and postcontact phases. However, the precontact and contact phases are not in the scope of this paper. In order to study the kinodynamic trajectory optimization problem for the postcontact phase, the following assumptions are presented:

(1) In the precontact phase, the two arms can reach the grasping point by generalized relative Jacobian [14] or reactionless motion planning method [11]

(2) In the contact phase, the two arms and the target can form a stable connection for further manipulation [30, 34]

(3) For a tumbling target, the initial velocity and inertia parameters can be estimated during the precontact phase [30] and the postcontact phase [31]

In this section, the kinodynamic trajectory optimization problem for stabilizing a tumbling target in the postcapture phase will be formulated, in which only the trajectory of the target is optimized while the detumbling motion of two arms is generated from the optimal trajectory of the target by considering the kinematic and dynamic constraints between the target and the two arms. This trajectory optimization problem is transformed into a nonlinear programming problem (NLP) by the direct collocation method [27]. Then, the solution of NLP can be found by the NLP solver *fmincon* in the Optimization Toolbox of MATLAB. The kinodynamic trajectory optimization algorithm is developed based on open-source trajectory optimization library *OptimTraj* [35]. The detailed formulation of the kinodynamic trajectory optimization method is shown in the following sections.

3.1. Kinodynamic Trajectory Optimization Framework. For postcontact/capture phase, the space robot servicer and target will form a closed-chain constraint. In order to stabilize the tumbling target in the postcapture phase, the kinodynamic trajectory optimization framework is proposed for generating detumbling motion of dual-arm space robot. Given the initial conditions of the target and the dual-arm space robot, the deceleration trajectory of the tumbling target should be optimized to minimize the detumbling time and base disturbance and avoid singularity of dual-arm space robot, while the detumbling motions of dual-arm space robot can be generated according to the closed-chain kinodynamic constraints between the two arms and the target. The framework of kinodynamic trajectory optimization is shown in Figure 2.

3.2. System Dynamics and Decision Variables. To solve the trajectory optimization problem of the tumbling target, we use direct collocation to discrete the continuous trajectory. For each collocation point, we define the following state variable $\mathbf{s}$ and control variable $\mathbf{F}$ as follows:

$$
\begin{aligned}
\mathbf{s} &= \begin{bmatrix} {}^0\mathbf{x}_t^T & {}^0\dot{\mathbf{x}}_t^T \end{bmatrix}, \\
\dot{\mathbf{s}} &= \begin{bmatrix} {}^0\dot{\mathbf{x}}_t^T & {}^0\ddot{\mathbf{x}}_t^T \end{bmatrix}, \\
\mathbf{F} &= \begin{bmatrix} {}^0\mathbf{F}_l^T & {}^0\mathbf{F}_r^T \end{bmatrix},
\end{aligned}
\tag{21}
$$

where ${}^0\mathbf{x}_t$, ${}^0\dot{\mathbf{x}}_t$, and ${}^0\ddot{\mathbf{x}}_t$ are the pose, velocity, and acceleration of the target object; the states of two adjacent collocation points are constrained by the following dynamics equation of the object:

$$
\begin{bmatrix} m_t\mathbf{E}_3 & \mathbf{O}_3 \\ \mathbf{O}_3 & {}^0\mathbf{I}_t \end{bmatrix} \begin{bmatrix} {}^0\dot{\mathbf{v}}_t \\ {}^0\dot{\boldsymbol{\omega}}_t \end{bmatrix} + \begin{bmatrix} \mathbf{O}_3 \\ {}^0\boldsymbol{\omega}_t \times \left({}^0\mathbf{I}_t\, {}^0\boldsymbol{\omega}_t \right) \end{bmatrix} = {}^0\mathbf{G}_l\, {}^0\mathbf{F}_l + {}^0\mathbf{G}_r\, {}^0\mathbf{F}_r, \tag{22}
$$

where ${}^0\mathbf{x}_t = \begin{bmatrix} {}^0\mathbf{p}_t^T & {}^0\boldsymbol{\psi}_t^T \end{bmatrix}^T = [p_x, p_y, p_z, \alpha, \beta, \gamma]^T$ and ${}^0\dot{\mathbf{x}}_t = \begin{bmatrix} {}^0\mathbf{v}_t^T & {}^0\dot{\boldsymbol{\psi}}_t^T \end{bmatrix}^T$. The attitude ${}^0\boldsymbol{\psi}_t$ in ${}^0\mathbf{x}_t$ is represented by the

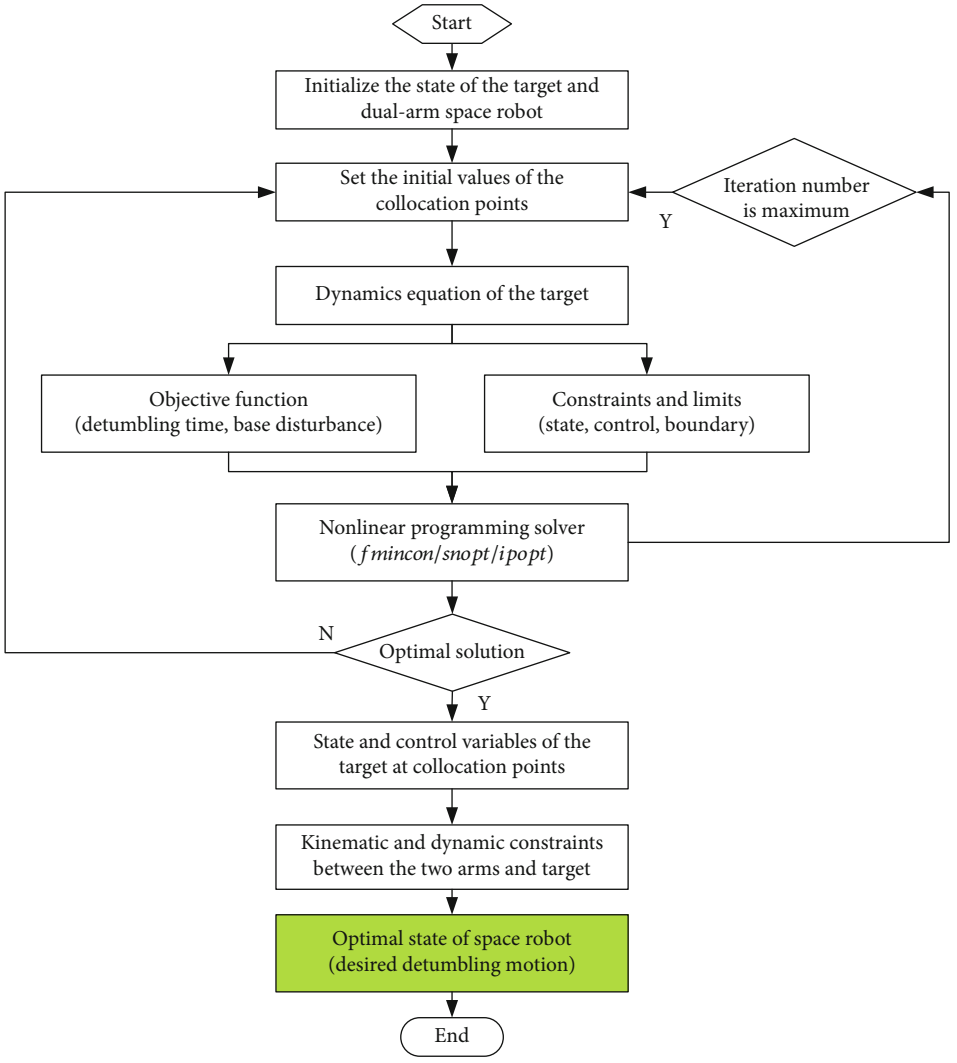


Figure 2: Kinodynamic trajectory optimization framework for detumbling a tumbling target with dual-arm space robot.

z-y-x Euler angles; the angular velocity in the dynamic equation is represented by ${}^0\boldsymbol{\omega}_t = \begin{bmatrix} {}^0\omega_{tx} & {}^0\omega_{ty} & {}^0\omega_{tz} \end{bmatrix}^T$.

In order to deal with the nonholonomic property of the angular velocity, the transformation between the Euler angle rate and the angular velocity is derived as follows:

$$ {}^0\boldsymbol{\omega}_t = \begin{bmatrix} 0 & -s_\alpha & c_\alpha c_\beta \\ 0 & c_\alpha & s_\alpha c_\beta \\ 1 & 0 & -s_\beta \end{bmatrix} \begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix} = \mathbf{N}^{\omega}_{zyx}(\alpha, \beta, \gamma)\, {}^0\dot{\boldsymbol{\psi}}_t, \tag{23} $$

where $\mathbf{N}^{\omega}_{zyx}$ is the matrix which projects the Euler angle rate ${}^0\dot{\boldsymbol{\psi}}_t$ to the angular velocity ${}^0\boldsymbol{\omega}_t$; $s_\alpha, c_\alpha, s_\beta, c_\beta$ are the abbreviations of $\sin(\alpha)$, $\cos(\alpha)$, $\sin(\beta)$, and $\cos(\beta)$.

Therefore, we can obtain the following equation:

$$ {}^0\dot{\boldsymbol{\psi}}_t = {\mathbf{N}^{\omega}_{zyx}}^{-1}(\alpha, \beta, \gamma)\, {}^0\boldsymbol{\omega}_t. \tag{24} $$

Furthermore, we can have the following equation by differentiating (23):

$$ \begin{aligned} {}^0\dot{\boldsymbol{\omega}}_t &= \mathbf{N}^{\omega}_{zyx}(\alpha, \beta, \gamma)\, {}^0\ddot{\boldsymbol{\psi}}_t + \dot{\mathbf{N}}^{\omega}_{zyx}(\alpha, \beta, \gamma)\, {}^0\dot{\boldsymbol{\psi}}_t, \\ {}^0\ddot{\boldsymbol{\psi}}_t &= {\mathbf{N}^{\omega}_{zyx}}^{-1}(\alpha, \beta, \gamma)\left\{ {}^0\dot{\boldsymbol{\omega}}_t - \dot{\mathbf{N}}^{\omega}_{zyx}(\alpha, \beta, \gamma)\, {}^0\dot{\boldsymbol{\psi}}_t \right\}. \end{aligned} \tag{25} $$

3.3. Constraints of the Trajectory Optimization. For deaccelerate the tumbling target, we can specify the initial and final states of the tumbling target as follows:

$$ \begin{aligned} {}^0\mathbf{x}_t(t_0) &= {}^0\mathbf{x}_{\text{ini}}, \\ {}^0\mathbf{x}_t(t_f) &= {}^0\mathbf{x}_{\text{fin}}, \\ {}^0\dot{\mathbf{x}}_t(t_0) &= {}^0\dot{\mathbf{x}}_{\text{ini}}, \\ {}^0\dot{\mathbf{x}}_t(t_f) &= {}^0\dot{\mathbf{x}}_{\text{fin}}, \end{aligned} \tag{26} $$

where ${}^0\mathbf{x}_{\text{ini}}, {}^0\dot{\mathbf{x}}_{\text{ini}}$ and ${}^0\mathbf{x}_{\text{fin}}, {}^0\dot{\mathbf{x}}_{\text{fin}}$ are the initial and final (desired) states of the target, respectively.

Additionally, in order to ensure that the target is within the workspace of the dual-arm space robot, we use box bounds to approximate them in this paper. The state limits of the target object in the trajectory optimization are introduced as follows:

$$\begin{aligned} {}^0\mathbf{x}_{\min} \le {}^0\mathbf{x}_t(t) \le {}^0\mathbf{x}_{\max}, \\ {}^0\dot{\mathbf{x}}_{\min} \le {}^0\dot{\mathbf{x}}_t(t) \le {}^0\dot{\mathbf{x}}_{\max}, \end{aligned} \tag{27}$$

where ${}^0\mathbf{x}_{\min}$, ${}^0\mathbf{x}_{\max}$ are the minimum and maximum poses of the target and ${}^0\dot{\mathbf{x}}_{\min}$, ${}^0\dot{\mathbf{x}}_{\max}$ are the minimum and maximum velocities of the target.

During the detumbling manipulation, the force magnitude of each manipulator applied to the target is constrained as follows:

$$-{}^0\mathbf{F}_{k,\max} \le {}^0\mathbf{F}_k(t) \le {}^0\mathbf{F}_{k,\max}, \tag{28}$$

where ${}^0\mathbf{F}_{k,\max}$ is the maximum force of Arm-k. Through equation (28), the optimal trajectory generation of the target is decoupled from the dynamics of space robot [21]. Additionally, as the proposed kinodynamic trajectory optimization method can not handle the time-variant constraints, the corresponding joint torque can be guaranteed to be below its limit by setting strict end-effector force/torque limit. Therefore, the prescribed maximum end-effector force (28) of each manipulator is designed to guarantee joint torque limits of space robot during manipulation.

In addition to the above explicit constraints, the implicit constraints are also included in the kinodynamic trajectory optimization. As shown in Section 2.3, the base disturbance is calculated according to the kinematic and dynamic closed-chain constraints between two arms and the target.

3.4. Objective Function of the Trajectory Optimization. For space robot, the attitude of the base is generally required to keep fixed with respect to the sun and the earth for communication and observation purposes. However, the fuel of thrusters for attitude control is very limited and mainly reserved for orbital maneuvers. Therefore, the trajectory optimization problem of postcapture phase is formulated to minimize the detumbling time and base disturbance force, i.e., minimize the energy consumption during the whole detumbling manipulation. For the dual-arm space robot, the base disturbance mainly comes from the operational forces of two arms for detumbling the tumbling target as shown in Section 2.3.

Furthermore, the inverse kinematic equation of dual-arm space robot can be obtained from (10):

$$\dot{\boldsymbol{\Theta}} = \mathbf{J}_g^{\dagger}\dot{\mathbf{x}}_e. \tag{29}$$

The singularity of dual-arm space robot occurs if the generalized Jacobian matrix $\mathbf{J}_g$ is not full ranked. In order to avoid the singularity, we added another weighted function into the objective function of trajectory optimization. This function is the negative manipulability of dual-arm space robot based on the generalized Jacobian matrix, which is defined as follows:

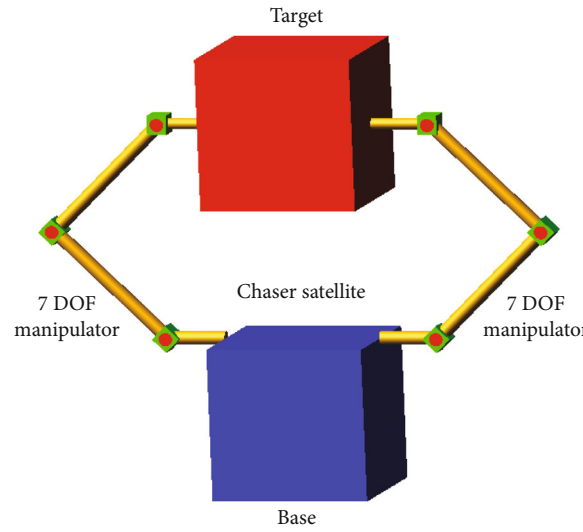


Figure 3: The initial configuration of dual arm space robot and the target.

$$M_{J_g} = -\sqrt{\det\left(\mathbf{J}_g\mathbf{J}_g^T\right)}. \tag{30}$$

By minimizing M_{J_g} (maximizing the manipulability), dual-arm space robot can keep away from the singular configuration. Therefore, the objective function of this trajectory optimization for stabilizing the tumbling target can be written as the following (equality constraints (22) and (26) and inequality constraints (27) and (28)):

$$\begin{aligned} \min_{\mathbf{s}(t),\mathbf{F}(t)} \quad & w_1 T + w_2 \int_{t_0}^{t_f} {}^0\mathbf{F}_{\text{dis}}^{fT}\,{}^0\mathbf{F}_{\text{dis}}^{f} dt + w_3 \int_{t_0}^{t_f} M_{J_g} dt \\ \text{s.t.} \quad & \textbf{\textit{equality constriants}} \\ & \textbf{\textit{inequality constriants}}, \end{aligned} \tag{31}$$

where $T = t_f - t_0$ is the detumbling time and ${}^0\mathbf{F}_{\text{dis}}^{f}$ is the disturbance force of the base during the detumbling manipulation. Among the three items in the objective function, the base disturbance force and manipulability are much more important than the detumbling time; w_1, w_2, and w_3 are the weights to trade off the detumbling time, base disturbance, and manipulability. Unless specified, w_1, w_2, and w_3 are set to 1 in this paper.

4. Simulation Study

In order to verify the kinodynamic trajectory optimization method proposed in this paper, simulation studies with different objective functions and initial conditions are carried out. In Section 4.1, the proposed kinodynamic trajectory optimization framework is used to minimize the detumbling time and the base disturbance of dual-arm space robot during detumbling manipulation. In Section 4.2, the proposed method is

Table 2: Mass properties of space robotic system and target.

		Base	Redundant manipulator (as shown in Figure 3)							Target
			B_1	B_2	B_3	B_4	B_5	B_6	B_7	
Mass (kg)		500	4	2	6	2	6	2	2	100
$^i\mathbf{I}_i$ (kg.m^2)	$^iI_{xx}$	50	0.012	0.003	0.052	0.003	0.052	0.003	0.003	20
	$^iI_{yy}$	50	0.012	0.003	0.052	0.003	0.052	0.003	0.003	20
	$^iI_{zz}$	50	0.002	0.0008	0.006	0.0008	0.006	0.0008	0.0008	20
	$^iI_{xy}$	0	0	0	0	0	0	0	0	0
	$^iI_{xz}$	0	0	0	0	0	0	0	0	0
	$^iI_{yz}$	0	0	0	0	0	0	0	0	0

Table 3: The constraints for trajectory optimization.

Description	Variables	Values
Bound constraint	$\mathbf{s}_{\text{ini}}(m, \text{rad})$	{0, 0, 0.4, 0, 0, 0; 0.1, 0.05, 0, 0, 0, 0}
	$\mathbf{s}_{\text{fin}}(m, \text{rad})$	{0, 0, 0.3, 0, 0, 0; 0, 0, 0, 0, 0, 0}
State limit	$\mathbf{s}_{\min}(m, \text{rad})$	{-0.6, -0.6, 0.2, -π, -π, -π; -∞, -∞, -∞, -∞, -∞, -∞}
	$\mathbf{s}_{\max}(m, \text{rad})$	{-0.6, 0.6, 0.6, π, π, π; ∞, ∞, ∞, ∞, ∞, ∞}
Control limit	$\mathbf{F}_{\min}(N, N^*m)$	{-30, -30, -30, -10, -10, -10, -30, -30, -30, -10, -10, -10}
	$\mathbf{F}_{\max}(N, N^*m)$	{30, 30, 30, 10, 10, 10, 30, 30, 30, 10, 10, 10}

used to deal with a general case of stabilizing the tumbling target while considering the singularity avoidance.

For detumbling manipulation in postcapture phase, the two arms have formed a stable connection with the target. Without loss of generality, the initial configuration of the dual-arm space robot and the target in the simulation study are shown in Figure 3. The initial state of Arm-l and Arm-r are set to $[0, 45, 0, 90, 0, 45, 0]^T * \pi/180$ and $[0,-45, 0,-90, 0,-45, 0]^T * \pi/180$, respectively. The mass and inertia parameters of the base, each manipulator, and the tumbling target are shown in Table 2. Unless specified, the length and angle units are m and rad, respectively. In order to evaluate the results generated from trajectory optimization, the base disturbance metric is defined as follows:

$$M_d = w_p\sqrt{\delta p_x^2 + \delta p_y^2 + \delta p_z^2} + w_o\sqrt{\delta o_x^2 + \delta o_y^2 + \delta o_z^2}, \quad (32)$$

where δp and δo are the position and orientation disturbance of the base and w_p and w_o are the corresponding weights for position disturbance and orientation disturbance.

4.1. Minimal Detumbling Time and Base Disturbance. In this section, the objective function of the trajectory optimization is set to minimize the detumbling time and the base disturbance caused by the detumbling force of two arms. The kinodynamic trajectory optimization is carried out with the dynamic equation of the tumbling target while the corresponding motions and forces of the two arms are generated by the kinematic and dynamic constraints between the two arms and the target. The constraints for the trajectory optimization are listed in Table 3. The initial guesses for control variable are set to zero.

For space robots, the attitude stabilization is much more important than position as the specific attitude is required to guarantee communication and solar energy utilization. Without loss of generality, the weight coefficients for base disturbance force and base disturbance torque in the objective function are set to 0 and 1 correspondingly. The generated trajectory for stabilizing a moving target is performed in the gravity-free simulation environment as shown in Figure 4.

The position and linear velocity trajectories of the target are shown in Figure 5. The corresponding position and attitude disturbances of the base of dual-arm space robot during the manipulation are shown in Figure 6. This result is compared with the optimization result of minimizing the detumbling time and the detumbling force at the end-effector of each manipulator [21]. As shown in Figure 6, the attitude disturbances of the base resulted from these two different objective functions are [-0.09, -0.34, -0.79] and [-0.12, -0.38, -0.85], respectively. It should be noted that minimizing detumbling force is not equivalent to minimizing the base disturbance. The trajectories of the tumbling target and corresponding disturbance metrics in the above-mentioned two different cases are shown in Figure 7. It can be seen that the trajectory which minimizes the base disturbance is different from the one minimizing detumbling force and consequently has smaller base disturbance.

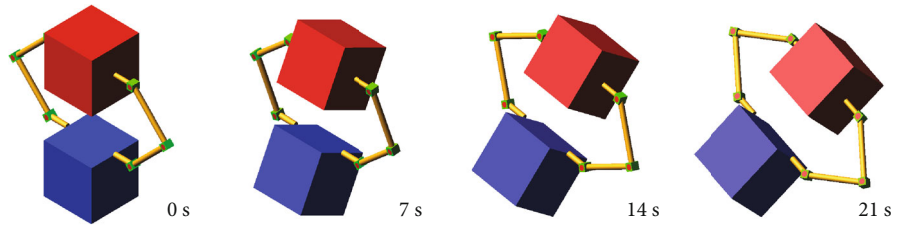


Figure 4: Simulation result with trajectory optimization. Four sequentially selected snapshots.

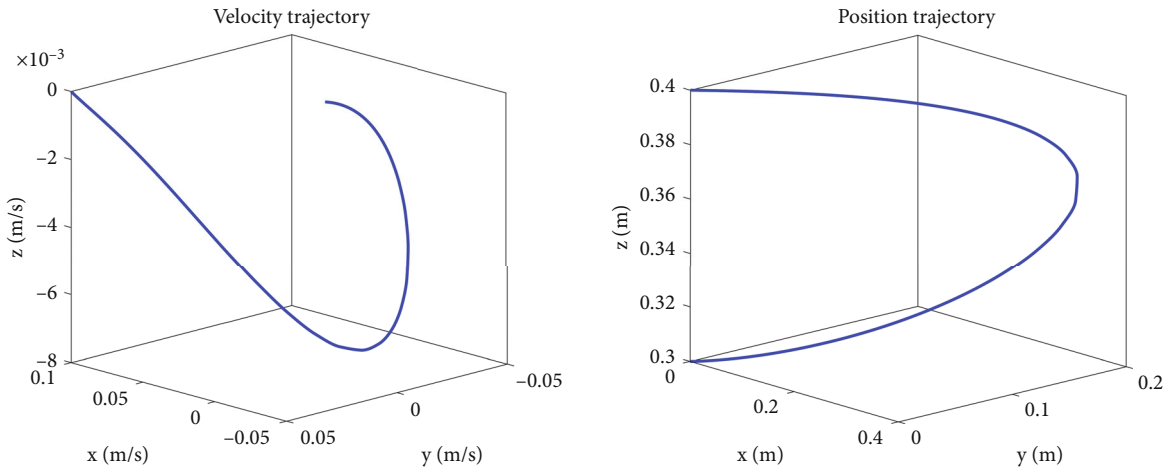


Figure 5: The velocity and position trajectories of the target during the detumbling manipulation.

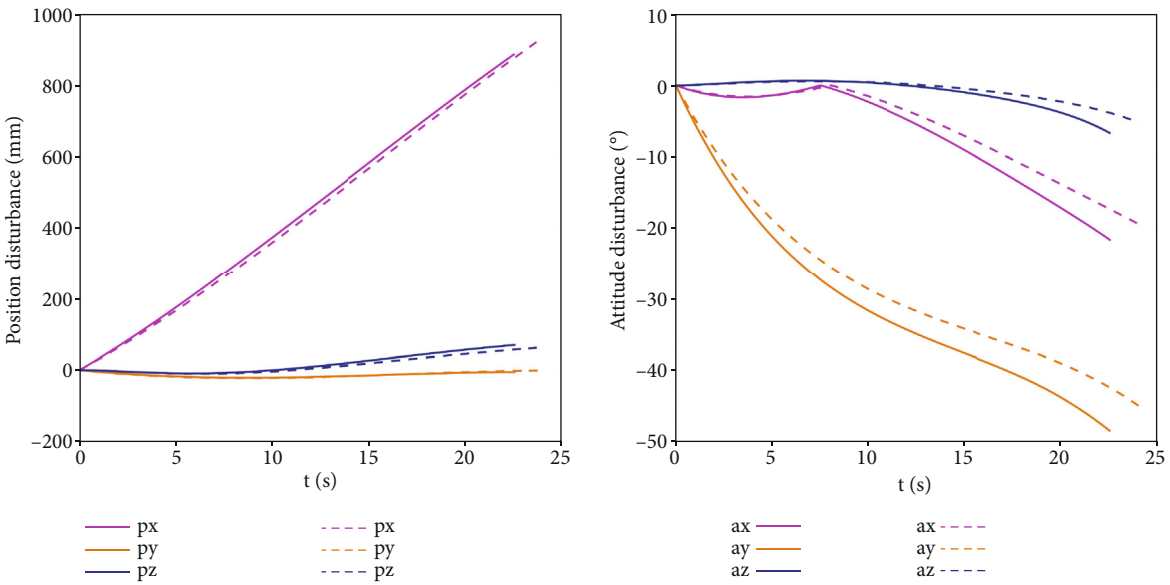


Figure 6: The position disturbance and attitude disturbance of the base of dual-arm space robot. The solid lines represent the result minimizing detumbling force; the dashed lines represent the result minimizing base disturbance.

In order to show the capability and robustness of the proposed kinodynamic trajectory optimization method, we carry out several simulations in which the mass and inertia ratio between the target and the base of dual-arm space robot is set to 1 while the uncertainties in mass and inertia parameters of the target are also considered as shown in Table 4. The optimal trajectories corresponding to different mass and inertia of the target are shown in Figure 8, which can be generated from the proposed kinodynamic trajectory optimization. However, in order to show the robustness of this method, dual-arm space robot only adopted the optimal trajectory (solid line in Figure 8) where the mass and inertia of the target and base are both set to 100 kg and [20, 20, 20] kg.m^2. Considering 0 (solid line), 10% (dashed line),

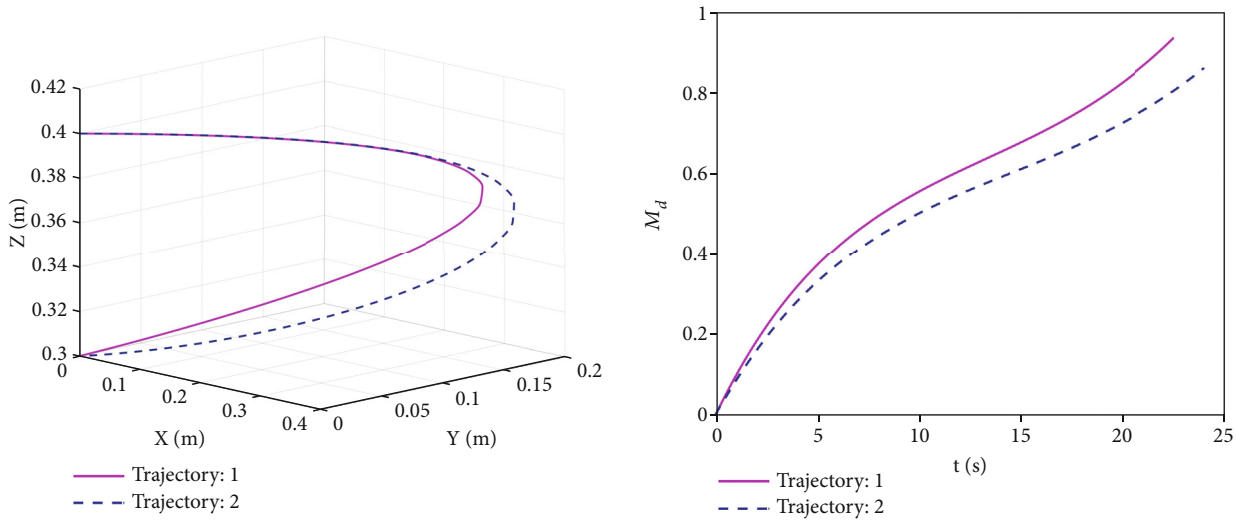


Figure 7: The trajectory and base disturbance metric corresponding to minimizing the detumbling time and detumbling force (trajectory 1) and minimizing the detumbling time and the base disturbance (trajectory 2).

Table 4: Parameter uncertainties of target's model.

Mass and inertia of base (kg, kg.m^2)	Mass and inertia of target (kg, kg.m^2)	Parameter uncertainty
Mass: 100; inertia: [20, 20, 20]	Mass: 100; inertia: [20, 20, 20]	0
Mass: 100; inertia: [20, 20, 20]	Mass: 110; inertia: [22, 22, 22]	10% (mass and inertia)
Mass: 100; inertia: [20, 20, 20]	Mass: 120; inertia: [24, 24, 24]	20% (mass and inertia)

and 20% (dash-dotted line) mass and inertia uncertainties of the target, the corresponding base disturbance force is shown in Figure 9. It can be seen that for the same optimal trajectory of the tumbling target, the smaller mass and inertia parameters will result in a smaller base disturbance. Therefore, considering the detumbling manipulation of space target with initial mass and inertia uncertainty, we can choose the maximum value for mass and inertia in the optimization to get a conservative detumbling solution and increase the robustness of the optimal solution.

Furthermore, we consider different initial velocities in the trajectory optimization where the mass and inertia of the target and base are both set to 100 kg and [20, 20, 20] kg.m^2. For different initial linear velocities [0.20, 0.10. 0.0], [0.22, 0.11, 0.0], and [0.24, 0.12, 0.0], the optimal trajectories generated from the trajectory optimization are shown in Figure 10(a). The corresponding base disturbance metric is shown in Figure 10(b). It can be seen that for the same tumbling target, the larger initial velocity will result in a larger base disturbance.

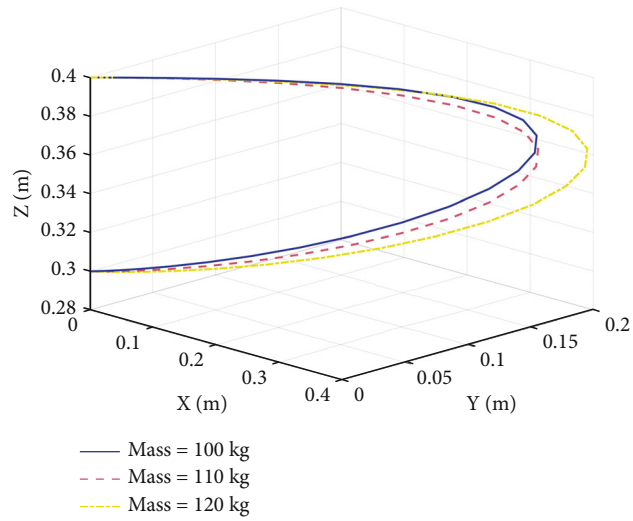


Figure 8: Optimal trajectories corresponding to different masses and inertias of the target.

4.2. A General Case for Singularity Avoidance. For stabilizing the tumbling target, a general case is considered in which the objective function is the same as Section 4.1 and the constraints condition is shown in Table 3. However, in order to verify the singularity avoidance capacity of the trajectory optimization framework, the initial velocity of the target is set to [0.15 0.10 0 0.05 0.04 0.03], which may cause the singularity of dual-arm space robot because of the initial linear velocity and angular velocity.

The simulation results of trajectory optimization without singularity avoidance (i.e., $w_3 = 0$) are shown in Figure 11. It can be seen that the singularity happens around 8 s. The joint angular velocities under singularity and manipulability of dual-arm space robot are shown in Figure 12, from which it can be seen that the singularity happens twice at 5.6 s and 8.4 s, respectively. On the other hand, the simulation results of trajectory optimization with singularity avoidance are shown in Figure 13. The optimal trajectories for the velocity

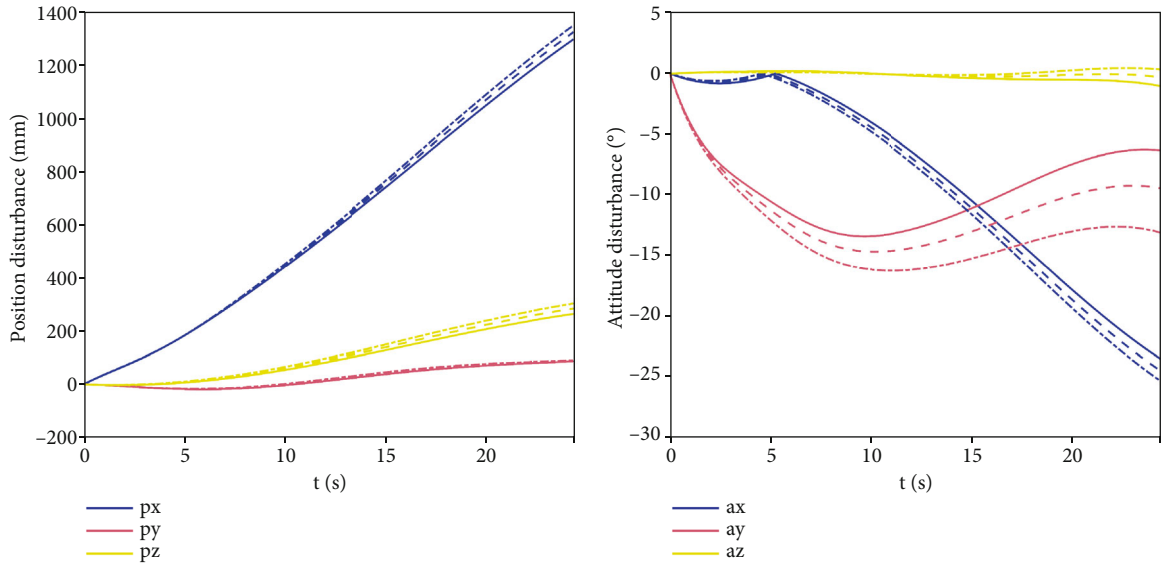


Figure 9: Base disturbances under different mass and inertia uncertainties (solid line: no uncertainty; dashed line: 10% uncertainty; dash-dotted line: 20% uncertainty).

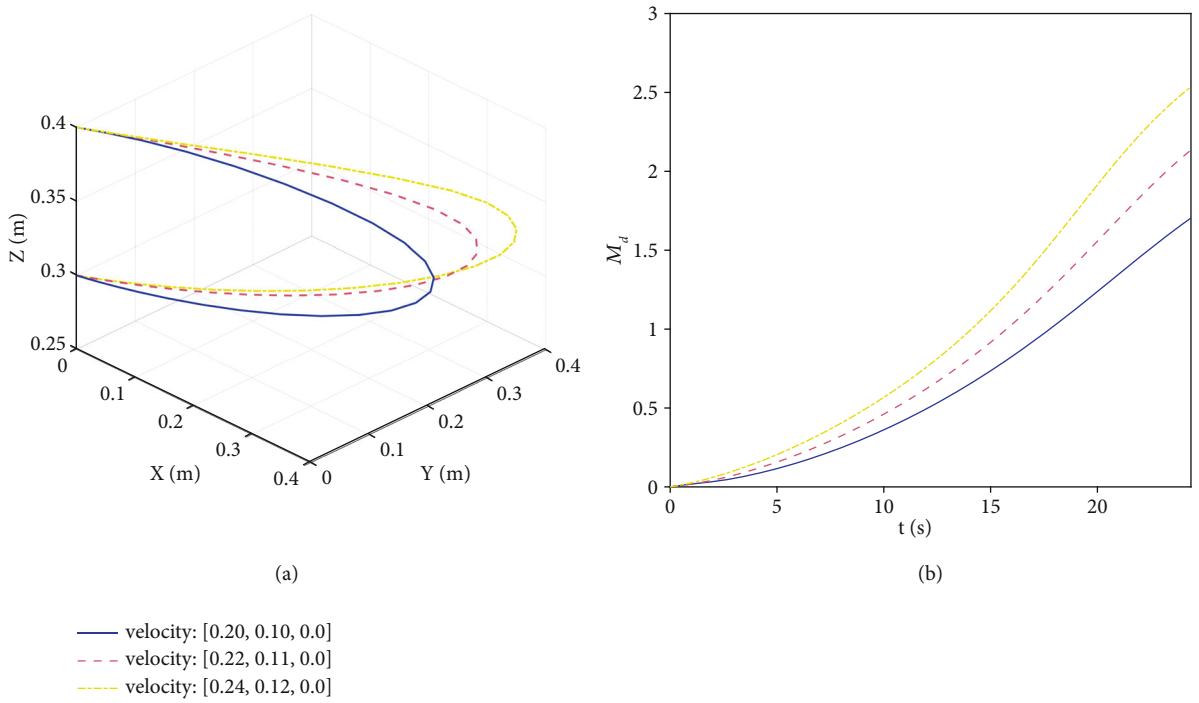


Figure 10: Optimal trajectories and disturbance metrics corresponding to different initial velocities of the target.

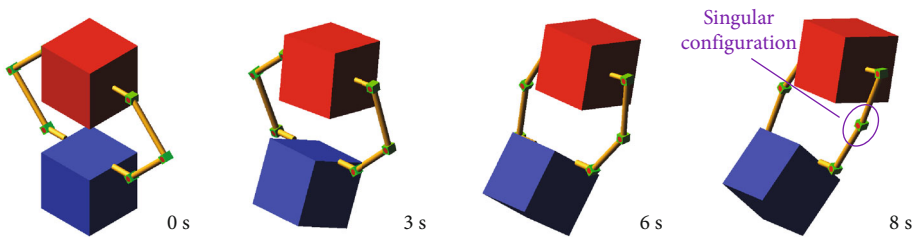
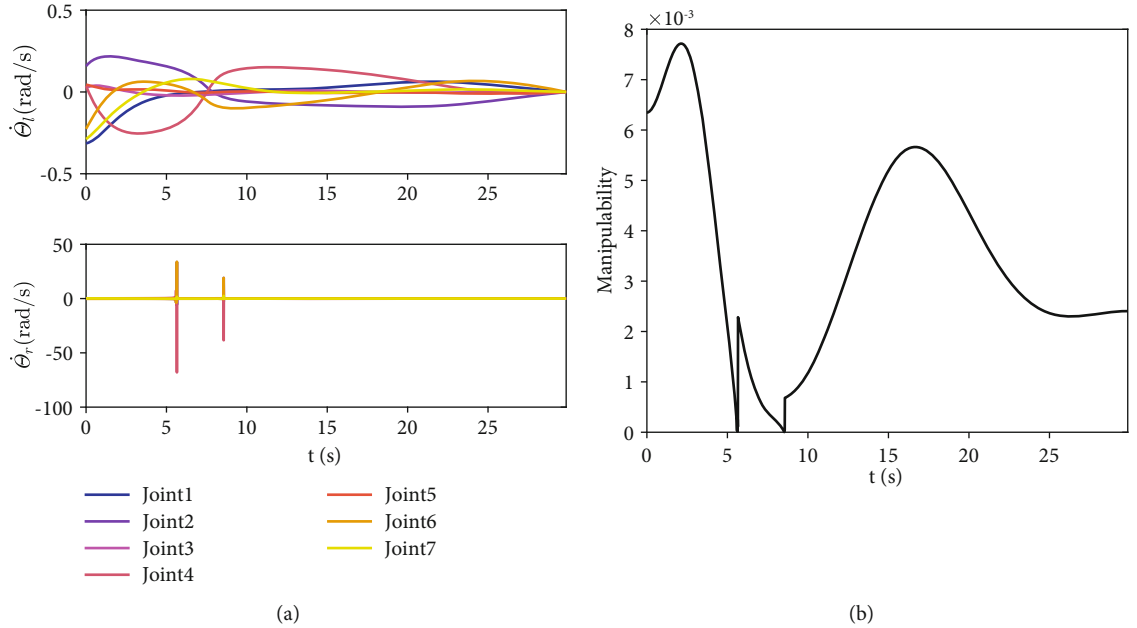


Figure 11: Simulation results of trajectory optimization without singularity avoidance.



(a) (b)

Figure 12: Joint angular velocities of each arm (a) and manipulability of dual-arm space robot (b).

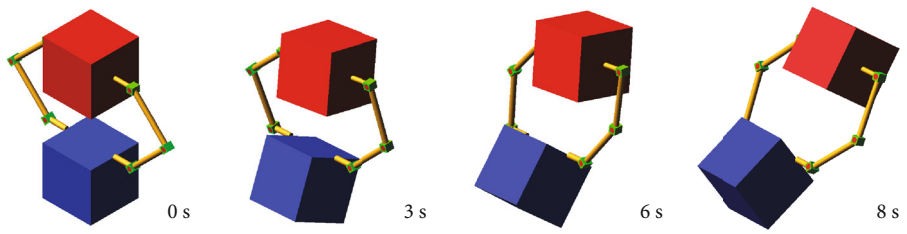
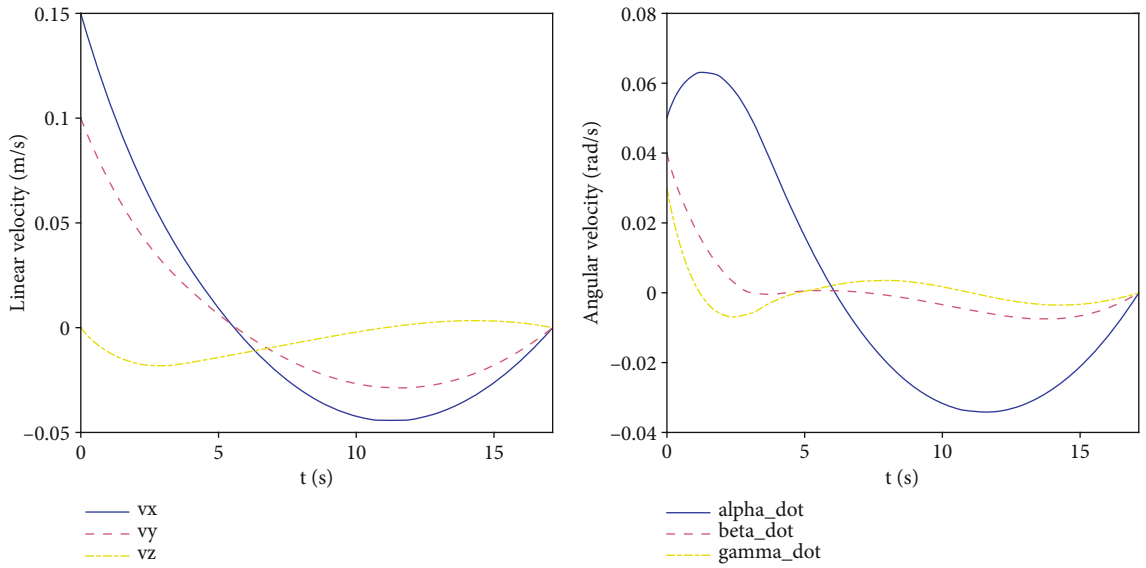


Figure 13: Simulation results of trajectory optimization with singularity avoidance.



(a) (b)

Figure 14: The linear velocity (a) and angular velocity (b) trajectories of the target.

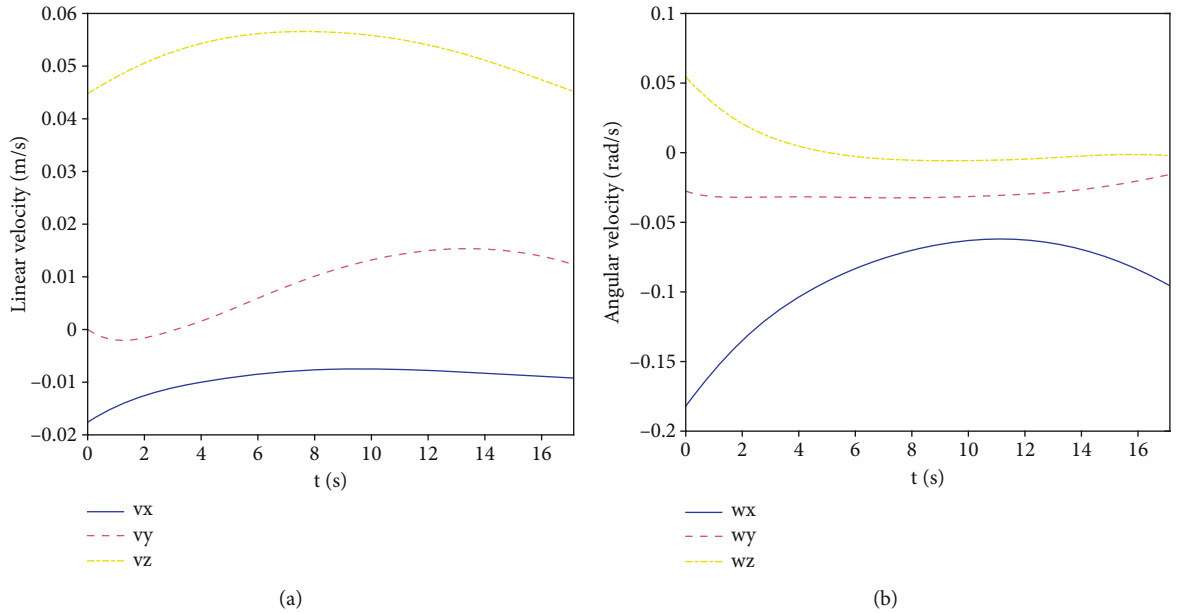


Figure 15: The linear velocity (a) and angular velocity (b) of the base of dual-arm space robot.

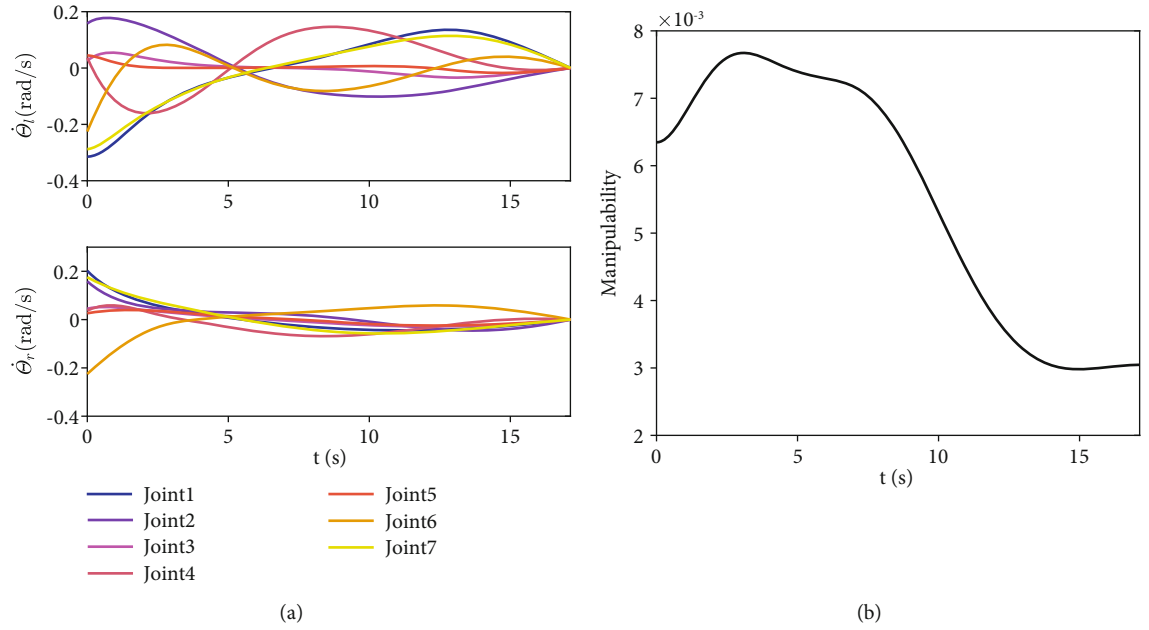


Figure 16: Joint angular velocities of each arm (a) and manipulability of dual-arm space robot (b).

of the target are shown in Figure 14. The corresponding base linear and angular velocity are shown in Figure 15. It can be seen that the dual-arm space will move with the tumbling target during the detumbling manipulation as the base is not actively controlled. The corresponding joint angular velocities and manipulability of dual-arm space robot are shown in Figure 16. The simulation results show that dual-arm space robot can keep away from the singular configuration by maximizing the manipulability. Therefore, the singularity avoidance problem of dual-arm space robot can also be solved in the kinodynamic trajectory optimization framework.

5. Conclusion

In order to stabilize the tumbling target to a desired pose for further maintenance and manipulation, the kinodynamic trajectory optimization method is proposed for postcapture phase in this paper. Instead of minimizing the detumbling time and detumbling force, the objective function is formulated to minimize the detumbling time and base disturbance caused by the dual-arm detumbling force of dual-arm space robot. To verify the proposed method, several physical simulations with different initial conditions and objective functions are carried out. The results show that the trajectory

generated from the proposed kinodynamic trajectory optimization method which minimizes the base disturbance force can result in smaller base disturbance than other objective functions and avoid singularities of dual-arm space robot. Therefore, the energy of dual-arm space robot can be saved for further manipulation. The proposed kinodynamic trajectory optimization method can be used to plan the trajectory of space robots for on-orbit manipulation. For implementing the proposed method on real space robot system, warm start is needed to decrease the computation time. The multiple capturing phases, including precontact, contact, and postcontact phases, will be also considered into the whole trajectory optimization in future work.

Data Availability

Data is available on request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Key R&D Program of China (Grant No. 2018YFB1304600), the Shenzhen Excellent Scientific and Technological Innovation Talent Training Project (Grant No. RCJC20200714114436040), and the Basic Research Program of Shenzhen (Grant No. JSGG20200103103401723). This work was also partially supported by the Shenzhen Municipal Basic Research Project for Natural Science Foundation (Grant No. JCYJ20190806143408992) and the Guangdong Basic and Applied Basic Research Foundation (Grant No. 2019A1515110680).

References

[1] X. Bai, F. Zhou, and Y. Hui, "Obtaining JTF-signature of space-debris from incomplete and phase-corrupted data," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 53, no. 3, pp. 1169–1180, 2017.

[2] M. Grassi, E. Cetin, and A. G. Dempster, "Enabling orbit determination of space debris using narrowband radar," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 2, pp. 1231–1240, 2015.

[3] A. Flores-Abad, O. Ma, K. Pham, and S. Ulrich, "A review of space robotics technologies for on-orbit servicing," *Progress in Aerospace Sciences*, vol. 68, pp. 1–26, 2014.

[4] M. Shan, J. Guo, and E. Gill, "Review and comparison of active space debris capturing and removal methods," *Progress in Aerospace Sciences*, vol. 80, pp. 18–32, 2016.

[5] P. Huang, D. Wang, Z. Meng, F. Zhang, and J. Guo, "Adaptive postcapture backstepping control for tumbling tethered space robot–target combination," *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 1, pp. 150–156, 2016.

[6] P. Huang, D. Wang, Z. Meng, F. Zhang, and Z. Liu, "Impact dynamic modeling and adaptive target capturing control for tethered space robots with uncertainties," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 5, pp. 2260–2271, 2016.

[7] W. Xu, B. Liang, B. Li, and Y. Xu, "A universal on-orbit servicing system used in the geostationary orbit," *Advances in Space Research*, vol. 48, no. 1, pp. 95–119, 2011.

[8] S. A. A. Moosavian and E. Papadopoulos, "Free-flying robots in space: an overview of dynamics modeling, planning and control," *Robotica*, vol. 25, no. 5, pp. 537–547, 2007.

[9] K. Yoshida, R. Kurazume, and Y. Umetani, "Dual arm coordination in space free-flying robot," in *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2516–2521, Sacramento, CA, USA, 1991.

[10] X. Zhang, J. Liu, Y. Tong, Y. Liu, and Z. Ju, "Attitude decoupling control of semifloating space robots using time-delay estimation and supertwisting control," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 57, no. 6, pp. 4280–4295, 2021.

[11] S. V. Shah, I. Sharf, and A. Misra, "Reactionless path planning strategies for capture of tumbling objects in space using a dual-arm robotic system," in *AIAA Guidance, Navigation, and Control (GNC) Conference*, vol. 4521, Boston, MA, 2013.

[12] L. Shi, S. Kayastha, and J. Katupitiya, "Robust coordinated control of a dual-arm space robot," *Acta Astronautica*, vol. 138, pp. 475–489, 2017.

[13] R. S. Jamisola Jr. and R. G. Roberts, "A more compact expression of relative Jacobian based on individual manipulator Jacobians," *Robotics and Autonomous Systems*, vol. 63, pp. 158–164, 2015.

[14] L. Yan, H. Yuan, W. Xu, Z. Hu, and B. Liang, "Generalized relative Jacobian matrix of space robot for dual-arm coordinated capture," *Journal of Guidance, Control, and Dynamics*, vol. 41, no. 5, pp. 1202–1208, 2018.

[15] L. Yan, W. Xu, Z. Hu, and B. Liang, "Virtual-base modeling and coordinated control of a dual-arm space robot for target capturing and manipulation," *Multibody System Dynamics*, vol. 45, no. 4, pp. 431–455, 2019.

[16] X. Wang, L. Shi, and J. Katupitiya, "A strategy to decelerate and capture a spinning object by a dual-arm space robot," *Aerospace Science and Technology*, vol. 113, article 106682, 2021.

[17] T. Oki, S. Abiko, H. Nakanishi, and K. Yoshida, "Time-optimal detumbling maneuver along an arbitrary arm motion during the capture of a target satellite," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 625–630, San Francisco, CA, USA, 2011.

[18] F. Aghili, "Pre-and post-grasping robot motion planning to capture and stabilize a tumbling/drifting free-floater with uncertain dynamics," in *IEEE International Conference on Robotics and Automation*, pp. 5461–5468, Karlsruhe, Germany, 2013.

[19] G. Rekleitis and E. Papadopoulos, "On-orbit cooperating space robotic servicers handling a passive object," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 2, pp. 802–814, 2015.

[20] T. Zhang, X. Yue, X. Ning, and J. Yuan, "Stabilization and parameter identification of tumbling space debris with bounded torque in postcapture," *Acta Astronautica*, vol. 123, pp. 301–309, 2016.

[21] M. Wang, J. Luo, J. Yuan, and U. Walter, "Detumbling strategy and coordination control of kinematically redundant space robot after capturing a tumbling target," *Nonlinear Dynamics*, vol. 92, no. 3, pp. 1023–1043, 2018.

[22] J. Luo, R. Xu, and M. Wang, "Detumbling and stabilization of a tumbling target using a space manipulator with joint-velocity

limits," *Advances in Space Research*, vol. 66, no. 7, pp. 1689–1699, 2020.

[23] L. Zong, J. Luo, and M. Wang, "Optimal detumbling trajectory generation and coordinated control after space manipulator capturing tumbling targets," *Aerospace Science and Technology*, vol. 112, article 106626, 2021.

[24] F. Aghili, "Optimal trajectories and robot control for detumbling a non-cooperative satellite," *Journal of Guidance, Control, and Dynamics*, vol. 43, no. 5, pp. 981–988, 2020.

[25] B. Sundaralingam and T. Hermans, "Relaxed-rigidity constraints: in-grasp manipulation using purely kinematic trajectory optimization," in *Robotics: Science and Systems XIII*, pp. 1–10, Cambridge MA, USA, 2017.

[26] M. Jankovic and F. Kirchner, "Trajectory generation method for robotic free-floating capture of a non-cooperative, tumbling target," in *Stardust Final Conference*, pp. 111–127, Springer, 2018.

[27] M. Kelly, "An introduction to trajectory optimization: how to do your own direct collocation," *SIAM Review*, vol. 59, no. 4, pp. 849–904, 2017.

[28] J. T. Betts, "Survey of numerical methods for trajectory optimization," *Journal of Guidance, Control, and Dynamics*, vol. 21, no. 2, pp. 193–207, 1998.

[29] B. Donald, P. Xavier, J. Canny, J. Canny, J. Reif, and J. Reif, "Kinodynamic motion planning," *Journal of the ACM*, vol. 40, no. 5, pp. 1048–1066, 1993.

[30] F. Aghili, "A prediction and motion-planning scheme for visually guided robotic capturing of free-floating tumbling objects with uncertain dynamics," *IEEE Transactions on Robotics*, vol. 28, no. 3, pp. 634–649, 2012.

[31] L. Zong, J. Luo, M. Wang, and J. Yuan, "Parameters concurrent learning and reactionless control in post-capture of unknown targets by space manipulators," *Nonlinear Dynamics*, vol. 96, no. 1, pp. 443–457, 2019.

[32] S. Erhart and S. Hirche, "Internal force analysis and load distribution for cooperative multi-robot manipulation," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1238–1243, 2015.

[33] L. Yan, Z. Mu, W. Xu, and B. Yang, "Coordinated compliance control of dual-arm robot for payload manipulation: master-slave and shared force control," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2697–2702, Daejeon, Korea (South), 2016.

[34] W. Xu, J. Peng, B. Liang, and Z. Mu, "Hybrid modeling and analysis method for dynamic coupling of space robots," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, no. 1, pp. 85–98, 2016.

[35] M. Kelly, "OptimTraj – trajectory optimization library GitHub," 2020, https://github.com/MatthewPeterKelly/OptimTraj.

Hindawi
International Journal of Aerospace Engineering
Volume 2022, Article ID 3396811, 11 pages
https://doi.org/10.1155/2022/3396811

Research Article

Fast and Accurate Hand Visual Detection by Using a Spatial-Channel Attention SSD for Hand-Based Space Robot Teleoperation

Qing Gao[1,2] **Xin Zhang**[3,4] **and Wenrao Pang**[1,2]

[1]*Institute of Robotics and Intelligent Manufacturing & School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen, Shenzhen 518172, China*
[2]*Shenzhen Institute of Artificial Intelligence and Robotics for Society, Shenzhen 518129, China*
[3]*State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Science, Shenyang 110016, China*
[4]*Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110169, China*

Correspondence should be addressed to Qing Gao; gaoqing@cuhk.edu.cn

Received 16 March 2022; Revised 7 April 2022; Accepted 15 April 2022; Published 4 May 2022

Academic Editor: Angelo Cervone

Space robot teleoperation is an important technology in the space human-robot interaction and collaboration. Hand-based visual teleoperation can make the operation more natural and convenient. The fast and accuracy hand detection is one of the most difficult and important problem in the hand-based space robot teleoperation. In this work, we propose a fast and accurate hand detection method by using a spatial-channel attention single shot multibox detector (SCA-SSD). The SSD framework is used and improved in our method by introducing spatial-channel attentions with feature fusion. To increase the restricted receptive field in shallow layers, two shallow layers are fused with deep layers by using feature fusion modules. And spatial attention and channel-wise attention are also used to extract more efficient features. This method can not only ease the computational burden but also bring more contextual information. To evaluate the effectiveness of the proposed method, experiments on some public datasets and a custom astronaut hand detection dataset (AHD) are conducted. The results show that our method can improve the hand detection accuracy by 2.7% compared with the original SSD with only 15 fps speed drops. In addition, the space robot teleoperation experiment proves that our hand detection method can be well utilized in the space robot teleoperation system.

1. Introduction

Due to the limited intelligence of space robots, space human-robot interaction plays an important role in the application of space tasks [1]. Teleoperation is one of a widely used space human-robot interaction method [2]. Teleoperation does not depend on the high intelligence capabilities of space robots. It can effectively combine the human decision-making ability with the space robot precise operation ability to improve the operation ability of space robots. There are some devices for space teleoperation. Some traditional devices, such as haptic feedback controllers [3–5], have stable and robust performance but lack of convenience. Some hand-based teleoperation devices [6], such as data gloves [7–9] and surface electromyography (SEMG) wristbands [10–12], have good convenience performance. However, because they are wearable devices, the performance on different people is very different. So, complex calibration work is required before using them. Hand-based visual teleoperation [6] is an emerging teleoperation method. It has the advantages of noncontact, natural, and convenience.

Hand detection is an important and difficult issue in the hand-based visual teleoperation. Because (1) space robot teleoperation needs real-time and robust operation. So, the hand detection should balance fast and accurate performances. (2) Complex backgrounds and changing illumination inside and outside the space station cabin make the astronaut hands

difficult to detect and locate. (3) Hand is a small object. Detection for small objects has always been a difficult problem in computer vision.

To deal with the above problems, a fast and accurate hand detection method is proposed in this paper. SSD framework [13] is used to design the hand detector since its good balance of speed and precision and ease of structural improvement. However, the SSD is not good at detecting small objects. Because it uses shallow layers to detect small objects, and shallow layers have enough contextual information but lack of semantic information. To address the lack of semantic information in the shallow layers, a multiattention module with feature fusion (MA-FF) is proposed to combine shallow layers with deep layers. The multiattention module extract channel attention features from deep and low-resolution feature maps and extract spatial attention features from high resolution layers, respectively. Then, the feature fusion module fuses these features to obtain new shallow layer feature maps with enough contextual and semantic information.

The main contributions and innovations are shown as follows. (1) A spatial-channel attention SSD (SCA-SSD) is proposed to deal with fast and accurate hand detection. The layers for object detection in the SSD structure are visualized to find out which layers play the most important role for small object detection. And these layers are improved and fused with deep layers. A multiattention module with feature fusion (MA-FF) is proposed. It includes a channel attention branch, a spatial attention branch, and a feature fusion branch. (2) A custom astronaut hand detection dataset (AHD) is designed. This dataset collects a large number of astronaut hand images and is used for hand detection verification for space robot teleoperation. (3) The experiments on hand detection datasets proves that the proposed SCA-SSD has fast and accurate hand detection performance, which is superior to some state-of-the-art method. And the experiments on the space robot teleoperation platform prove that the designed hand detector can be well used in the hand-based space robot teleoperation.

The rest of this paper is structured as follows. Section 2 reviews the prior work of hand-based robot teleoperation and hand detection methods. In Section 3, we first describe and visualize the original SSD and then elaborate the structure details of the proposed hand detection method. In Section 4, we provide the results of ablation experiments and comparative experiments on public datasets and a custom AHD dataset. And we also provide the application experiment on hand-based space robot teleoperation platform. Finally, we draw the conclusions and future work in Section 5.

2. Related Work

2.1. Hand-Based Robot Teleoperation. The hand-based robot teleoperation methods include contact and noncontact methods. The mainly contact methods include haptic feedback-based, sEMG-based, and data glove-based methods. Haptic feedback-based teleoperation [3–5] is a traditional teleoperation method. It transmits the 6-Dof position and orientation of human hand to the robot through the haptic feedback controller. For example, the da Vinci surgical telemanipulator [3] can transmit the dual-hand motion information of the chief surgeon through two main joysticks to control the instruments and a 3D high-definition endoscope. The principle of the sEMG-based teleoperation [10–12] is that when hand moves, the arm will generate corresponding motor neuron information, which can be obtained by decoding the sEMG signal. For example, Raspopovic et al. [10] used sEMG equipment to collect sEMG signal of hand gestures and used these gestures to control a dexterous hand. Data glove-based teleoperation [7–9] uses curvature sensors to collect the bending degrees of the fingers and the posture change of the entire human hand, to decode the movement of the hand. Fang et al. [7] designed a novel data glove to control a robotic hand-arm teleoperation system. The above contact teleoperation methods are lack of robustness for different people. The visual teleoperation is robust to different people due to its noncontact advantage [14–16]. For example, Li et al. [14] designed a mobile robot hand-arm teleoperation system by using vision and IMU. Handa et al. [15] designed a vision-based teleoperation method for a dexterous robotic hand-arm system. Table 1 shows the comparison and summary of the above hand-based robot teleoperation methods.

2.2. Hand Detection Methods. Traditional visual hand detection methods [17] mainly include skin color-based hand detection, motion flow information-based hand detection, and shape model-based hand detection. These methods only extract the shallow information of hands, which are subject to many conditions. Nowadays, deep learning-based hand detection methods can achieve better detection performance in complex environment [18–20]. Hand detection can be regarded as a kind of object detection. There are some typical deep learning-based object detectors, such as RCNN series [21, 22], YOLO series [23–25], and SSD series [13, 26, 27]. Among them, the SSD is a light weight one-stage network, which considers speed and accuracy trade-off and is easy to modify. For example, Gao et al. [18] designed a feature-map-fused SSD for robust real-time hand detection and localization. He also used SSD and body pose estimation for dual-hand detection [19]. Yu et al. designed a deep temporal model-based identity-aware hand detector by using the SSD framework for space human-robot interaction [20]. However, the SSD is stuck with the speed and accuracy dilemma for small object detection. Some useful methods and tricks are proposed to resolve this dilemma. DSSD [26] attempts to recover higher resolution features and adds with the primary features through shortcut connection. FSSD [28], DF-SSD [29], RSSD [30], and ESSD [31] provided many feature fusion methods to add more contextual information into shallow feature maps. Table 2 shows the comparison and summary of the above hand detection methods.

3. Spatial-Channel Attention SSD

In this section, first, the original SSD is introduced and visualized. Then, the proposed SCA-SSD is introduced, which includes the multi-attention module and feature fusion module.

3.1. SSD Introduction and Visualization. In this subsection, the SSD architecture is introduced first. And then, the

Table 1: Comparison and summary of the hand-based robot teleoperation methods.

Method	Brief methodology	Highlights	Limitations
Haptic feedback-based method [3–5]	This method transmits the 6-Dof position and orientation of human hand to the robot through the haptic feedback controller.	High accuracy and mature technology.	Contact method and only be used for 6-DOF control.
sEMG-based method [10–12]	When hand moves, the arm will generate corresponding motor neuron information, which can be obtained by decoding the sEMG signal.	High accuracy.	Contact method and lack of robustness.
Data glove-based method [7–9]	This method uses curvature sensors to collect the bending degrees of the fingers and the posture change of the entire human hand, to decode the movement of the hand.	High accuracy and mature technology.	Contact method and lack of robustness.
Vision-based method [14–16]	This method uses camera to capture the movements of human hands and maps the human hand movement information to the robot.	Strong noise and immature technology.	Noncontact method, strong robustness, and naturalness.

Table 2: Comparison and summary of the hand detection methods.

Method	Brief methodology	Highlights	Limitations
Traditional visual hand detection methods [17]	They mainly include skin color-based hand detection, motion flow information-based hand detection, and shape model-based hand detection.	Small amount of calculation, mature technology.	These methods only extract the shallow information of hands, which are subject to many conditions.
Deep learning-based hand detection methods [18–20]	They autonomously extract deep features of hand images through deep neural networks.	High detection performance.	They are stuck with the speed and accuracy dilemma for small object detection.

detection visualization in SSD is shown to find out which layers are suitable for improving.

3.1.1. SSD Architecture. The SSD [13] is one of the outstanding one-stage detectors with high speed and accuracy. The architecture is shown in Figure 1. The VGG-16 is used as its backbone, and several extra convolution layers on the top of the network are used for prediction and classification by filters directly. Unlike other detectors, SSD uses pyramidal multiresolution feature maps as convolutional detector input, which means it handles different scales in different resolution feature maps. The SSD brings significant improvement on speed because of its one-stage architecture. However, it cannot get a high detection accuracy on small object. Because the shallow layers for detection have much contextual information but less semantic information. While the deep layers for detection are reverse. Small object detection needs enough semantic and contextual information for its low resolution. So, feature maps with enough semantic and contextual information should be designed for hand detection.

3.1.2. Detection Visualization in SSD. To find out which layers are suitable for improving for small object detection, the results of feature maps for object detection in SSD are visualized. We select one convolution layer as the input of the detector and block other convolution layers which means we only use one specific convolution layer to detect objects. The results are shown in Figure 2, which shows that the small objects are easier detected in shallow layers (conv4_3 and conv7 layers), and large objects are easier detected in deep layers (conv8_2, conv9_2, and conv10_2 layers). Because the contextual information is vital to small object detection and shallow layers have enough contextual information. However, due to the lack of semantic information, there are some missing detection results of small objects in conv4_3 and conv7 layers. Once it misses the object in shallow layers, it has no chance to be detected in the subsequent deep layers. To increase the accuracy of small object detection, we propose the SCA-SSD. A multiattention module is employed on conv4_3 and conv7 layers and then fuses them with conv8_2 and conv11_2, separately. The details are presented below.

3.2. SCA-SSD Architecture. In this subsection, the overview of the SCA-SSD is introduced first. Then, the multiattention module and feature fusion module are introduced, respectively.

3.2.1. Overview of SCA-SSD. The architecture of our proposed SCA-SSD is introduced and shown in Figure 3. From the figure, we can see that the SCA-SSD reuses the multiscale and one-stage architecture of the original SSD. Two multiattention branch with feature fusion (MA-FF) modules are employed on the shallow layers conv4_3 and conv7, respectively. They use the multiattention modules to extract channel and spatial features and use feature fusion modules to fuse the two shallow layers (conv4_3 and conv7) with the deep layers (conv8_2 and conv11_2). Finally, the two new feature maps output from the MA-FF modules are mainly used for small object detection.

3.2.2. Multiattention Module. To address the lack of information in shallow layers, we propose a multiattention module with feature fusion, and the improved structure of conv4_3

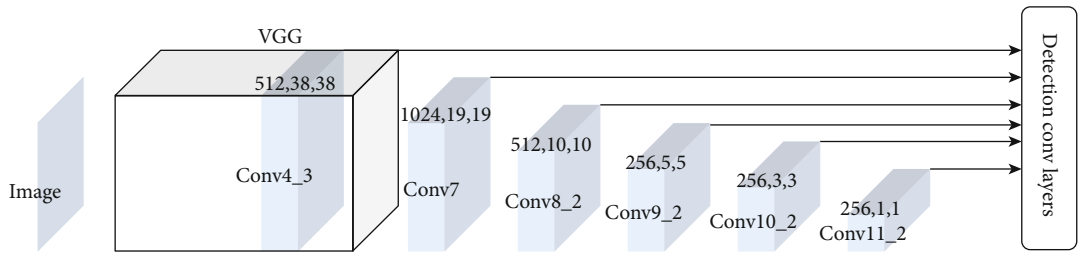


Figure 1: Original SSD architecture.

Figure 2: Detection results for each layer of SSD. In SSD, shallow layers are usually used to detect small objects and deep layers respond to objects in large scales.

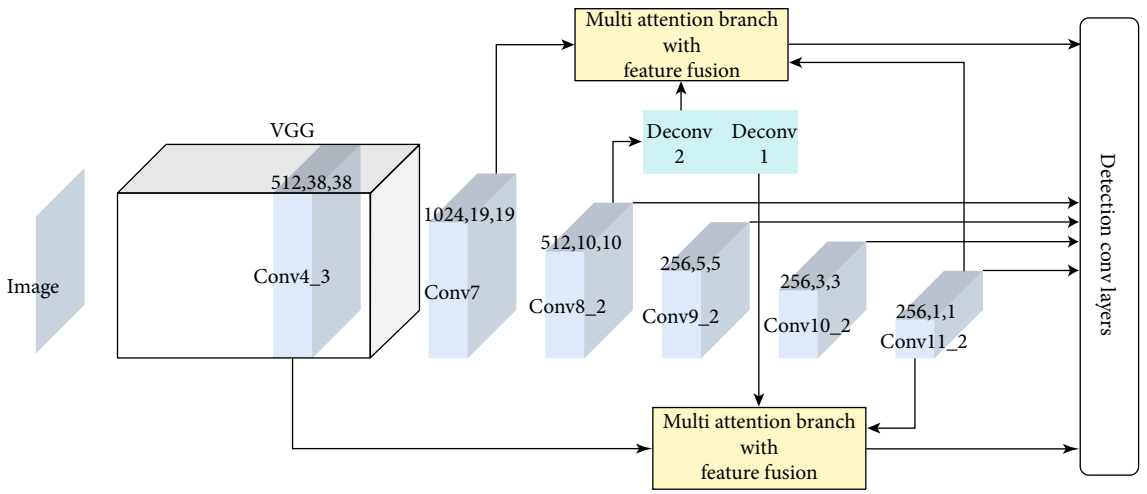


Figure 3: The architecture of the SCA-SSD.

is shown in Figure 4 as an example. The design of the attention module is inspired by bottleneck attention module (BAM) [32]. To be specific, first, the spatial attention branch Atts is employed after conv4_3 and conv7, respectively. After that, channel attention branch Att_c is employed after the conv11_2 whose resolution is 1×1 so that we can skip the global pooling operation in the squeeze stage coincidentally. Then, the Atts and Att_c are combined by element wise add operation to generate the cross resolution spatial-channel attention which terms Att_{sc}. Finally, the sigmoid is applied for Att_{sc} to obtain the weighted Att_{sc} and then multiply with feature maps from the corresponding feature map. For instance, as shown in Figure 4, weighted Att_{sc} is obtained from conv4_3 so that it multiplies and adds with conv4_3.

The spatial branch structure is shown in Figure 5(1). This branch follows encoder-decoder structure, but we do not down the resolution of the feature map to preserve more information. Each branch consists of a 1×1 convolution layer to reduce the dimensions of channels, and two 3×3 dilated convolution layers are employed for obtaining long-range information with a widely receptive field. Then, it will restore the number of channels as input by another 1×1 convolution layer. In practice, each convolution layer and dilated convolution layer are followed by a batch normalization and a ReLU activation function except for the last 1×1 convolution layer. Set the input feature map is F, the output Att_s can be expressed as

$$Att_s(F) = \sigma\left(f^{7\times7}([\text{AvgPool}(F); \text{MaxPool}(F)])\right), \quad (1)$$

where σ denotes the sigmoid function, $f^{7\times7}$ denotes a conclusion operation with the filter size of 7×7.

The structure of the channel branch is shown in Figure 5(2). In Att_c, in order not to affect the value of conv11_2 feature map, one 1×1 convolution layer following

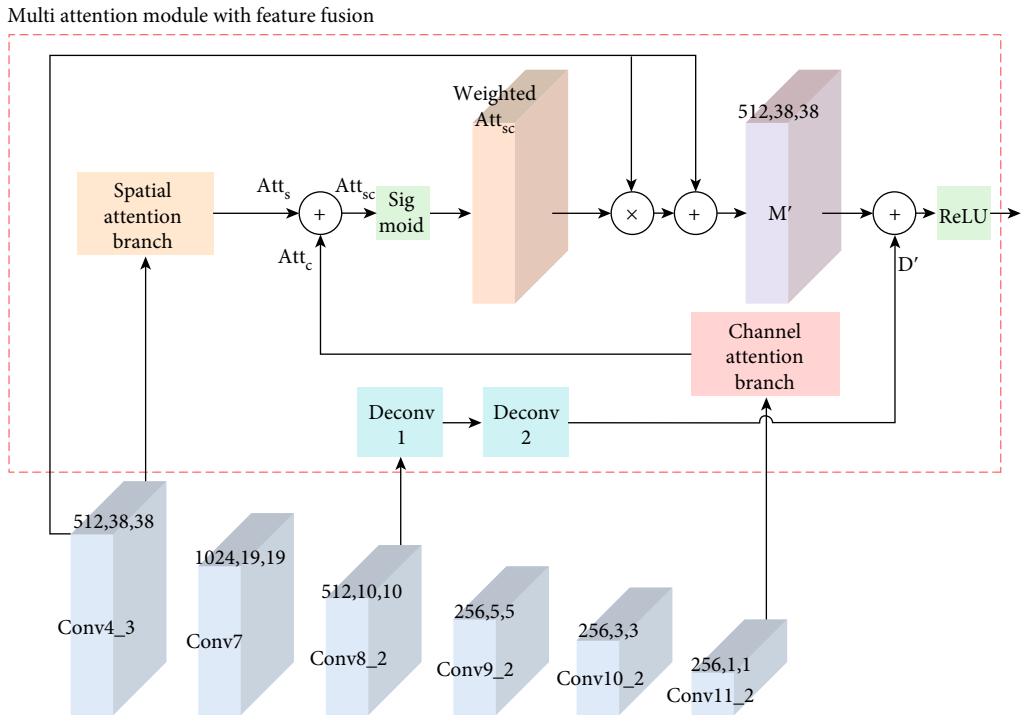


FIGURE 4: Multiattention module with feature fusion. The structure is the MA-FF on conv4_3 layer. In practice, the output of Deconv-1 adds with conv7, and the output of Deconv-2 adds with conv4_3. + means the symbol of element wise add operation, and × means the corresponding element multiply operation.

ReLU after the conv11_2 is employed. In excitation, two fully connected layers are used to blend different values in different channels. Then, they are expanded to match the sizes of conv4_3 and conv7. Set the input feature map is F, the output Att_c can be expressed as

$$Att_c(F) = \sigma(MLP(\text{AvgPool}(F)) + MLP(\text{MaxPool}(F))), \tag{2}$$

where σ denotes the sigmoid function.

3.2.3. Feature Fusion Module. Even though the multiattention module brings extra contextual information to shallow layers, the spatial attention branch still has a drawback. The context is encoded as an attention mask so that the value is limited between zero and one. By multiplying with input feature map, it can enhance the useful information for detection. However, context and long-range information are encoded as attention mask Att_{sc} which only provides weighted value. So, to capture more context, a feature fusion module which can be embedded within the multiattention module is proposed, and it is shown in Figure 4.

In the feature fusion module, two deconvolution layers are employed to restore the size of feature map from 10×10 to 19×19 and 38×38, so that it can match the size of conv7 and conv4_3. Our feature fusion module is inspired by DSSD [26], and two deconvolution layers are only used to avoid increasing much computational burden. In each Deconv-n block, it includes a deconvolution layer and a batch normalization (BN). After deconvolution, fusion operation is employed to merge a reweighted feature map $M^{'}$ with the output of deconvolution $D^{'}$. Follow the feature-fusion SSD [27], element-wise add is used as the fuse operation. It can be proved that element-wise add outperforms the concatenate operation in the feature-fusion SSD [27]. At the end of this module, the ReLU activation function is employed.

4. Experiments and Analysis

In this section, to compare performance with the state-of-the-art object detection methods, experiments are conducted on Pascal VOC dataset [33] first. Then, experiments are conducted on the Oxford hands dataset [34] to demonstrate the effectiveness of our proposed method on public hand detection datasets. After that, the AHD dataset will be introduced, and experiments will be conducted on this dataset to prove the performance of astronaut hand detection. The mean average precision (mAP) is adopted as evaluation metric to evaluate our model prediction performance.

We implement the MA-SSD based on PyTorch [35]. The data augmentation method is followed with SSD [13], and the VGG-16 is used as the pretrained backbone. All experiments are performed on 4 NVIDIA RTX 2080 Ti GPU.

4.1. Experiments on the Pascal VOC Dataset

4.1.1. Training. In training stage, the batch size is set to 32, and the learning rate is set to $1 \times 10 - 3$ with a warm-up phase at the first 500 times iteration. However, the experiment resulted

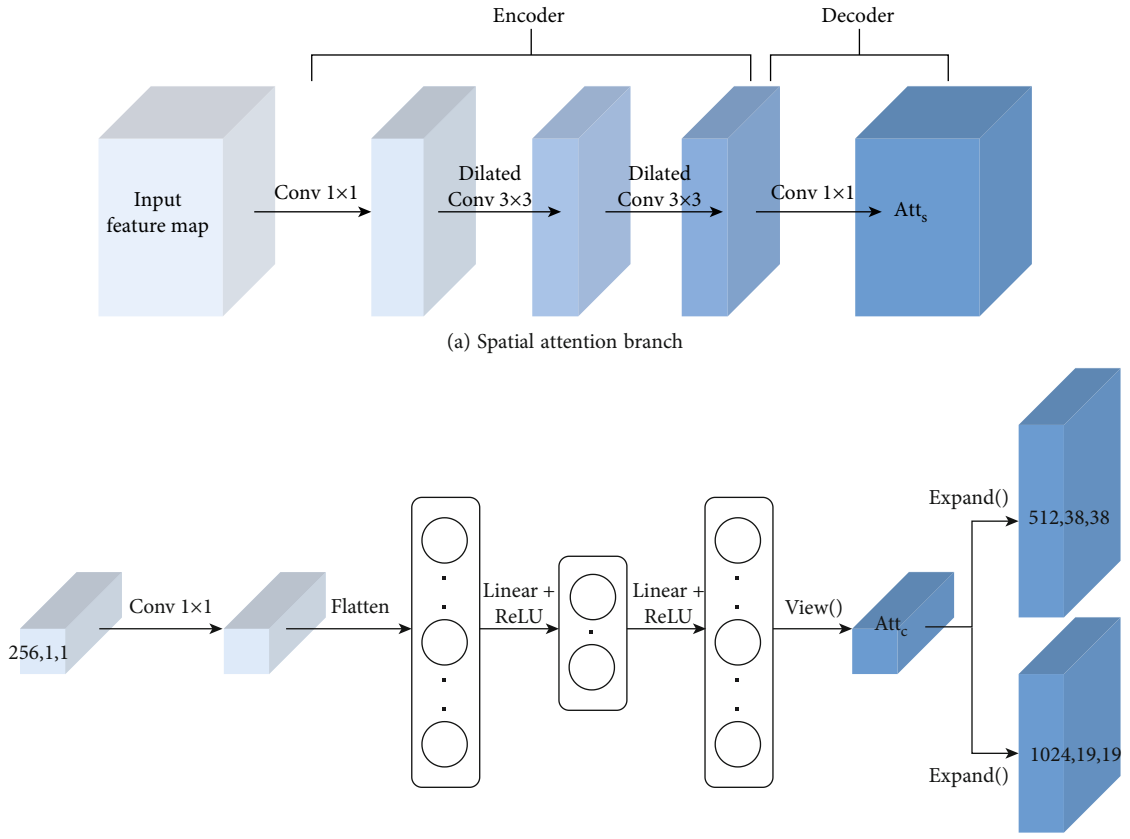


Figure 5: Spatial branch and channel branch in the MA-FF module.

the default learning rate is too small. Instead, the learning rate is set to $4 \times 10 - 3$ with a warm-up phase at first 2800 iterations. Learning rate should be increased from $1 \times 10 - 6$ with warm-up factor as 0.03333 gradually. The training step is set to 140 k iterations totally, and the learning rate is divided by 10 at 84 k and 112 k iterations which is different from original SSD [13] but similar to RFB-Net [36]. Following the trick in RFB-Net, the number of prior boxes in conv4_3 is increased to 6.

4.1.2. Introduction of the Pascal VOC Dataset. The objects in the Pascal VOC 2007 dataset include 4 categories and 20 subcategories, which are vehicle (car, bus, bicycle, motorbike, airplane, boat, and train), household (chair, sofa, dining table, TV, bottle, and potted plant), animal (cat, dog, cow, horse, sheep, and bird), and person. These images are collected from flickr and Microsoft Research Cambridge (MSRC) dataset. The dataset includes 9,963 images containing 24,640 annotated objects.

4.1.3. Comparative Experiments. To demonstrate the performance of the proposed SCA-SSD, some other state-of-the-art methods are compared. The results are shown in Table 3. For a fair comparison, the updated SSD [37] is used as our baseline, which can get a 77.7% mAP on the VOC test dataset. It is slightly higher than that of the original SSD [13], which mAP is 77.2%. By employing multiattention and fusion modules on the SSD, it achieves a 79.9% mAP, which is 2.7% higher than that of the original SSD [13] and 2.2% higher than that of the baseline [37]. The SCA-SSD brings significant improvement into SSD with the least impact on speed. It is only 15 FPS slower than the original SSD. And the mAP of the SCA-SSD is even higher than that of the SSD512, which has a higher input resolution (512×512) than that of the SCA-SSD (300×300). We also compare the results of the proposed SCA-SSD with some state-of-the-art object detection methods like faster-RCNN [21], YOLO v4 [25], R-FCN [38], and StairNet [39]. From Table 1, we can see that the performance of the SCA-SSD is higher than most of the state-of-the-art methods both on accuracy and seed. In addition, we also show the results of some SSD-series methods like DSSD [26] and FSSD [28]. To the best of our knowledge, our SCA-SSD achieves the best performance within SSD-series methods. It proves that the proposed SCA-SSD can achieve a great performance for object detection both on speed and accuracy.

4.2. Experiments on Oxford Hands Dataset

4.2.1. Introduction of the Oxford Hands Dataset. The hand detection is different with normal object detection. It has small size and changeable shape. To better prove the performance of the SCA-SSD for hand detection, the experiments on hand detection dataset are also conducted. The Oxford hands dataset [34] which is a public hand detection dataset is used for training and testing. In the dataset, a total of 13050 hand

Table 3: Comparison of object detection methods on VOC 2007 and VOC 2012 test dataset. In this table, the SCA-SSD is compared with some state-of-the-art methods and other SSD-based methods to illustrate the promising performance on object detection.

Method	Backbone	Input size	mAP	Device	FPS
Fast RCNN [21]	ResNet-101	600 × 1000	76.4	K40	2.4
YOLO v4 [25]	DarkNet-19	352 × 352	78.2	Titan X	81
R-FCN [38]	ResNet-101	600 × 1000	79.5	K40	5.8
StairNet [39]	VGG-16	300 × 300	78.8	Titan X	30
SSD300 [13]	VGG-16	300 × 300	77.2	2080Ti	119
Baseline [37]	VGG-16	300 × 300	77.7	2080Ti	119
DSSD321 [26]	ResNet-101	321 × 321	78.6	Titan X	9.5
FSSD300 [28]	VGG-16	300 × 300	78.8	1080Ti	65.8
FA-SSD300 [40]	ResNet-101	300 × 300	78.3	—	34.7
FF-SSD300 [27]	VGG-16	300 × 300	78.9	—	43
Shift SSD300 [41]	VGG-16	300 × 300	78.3	Titan X	77
ESSD++300 [24]	VGG-16	300 × 300	79.2	—	52
DF-SSD300 [29]	Dense-32-S-1	300 × 300	78.9	Titan X	11.6
RSSD300 [30]	VGG-16	300 × 300	78.5	—	35
SCA-SSD300	VGG-16	300 × 300	79.9	2080Ti	104

instances are annotated. Hand instances larger than a fixed area of bounding box (1500 sq. pixels) are considered "big" enough for detections and are used for evaluation. This gives around 4170 high-quality hand instances. In each image, all the hands that can be perceived clearly by humans are annotated.

4.2.2. Ablation Experiment. To understanding SCA-SSD structure deeper and better, several ablation experiments are conducted to show the effectiveness of each module of the network on hand detection. The results are summarized in Table 4. In this experiment, first, we add channel attention and spatial attention models on the baseline structure, respectively. The mAP can increase 1.6% and 1.3% compared with the baseline method. And the speeds only drop by 3FPS. It proves that the proposed channel attention and spatial attention models are effective in hand detection. Second, we take the feature fusion module away from the SCA-SSD, which terms as SCA-SSD w/o fusion. The result decreases from 44.6% to 43.8% compared with the SCA-SSD w/fusion, which indicates the feature fusion module is effective in hand detection. The feature fusion module can improve 0.8% of mAP but it has little impact on the speed of inference, the speed still keeps on over 100 FPS (104FPS). So, the ablation experiment results show that the proposed channel attention, spatial attention, and feature fusion modules are effective to improve the performance of hand detection.

4.3. Experiments on AHD Dataset

4.3.1. AHD Dataset. To further verify the effectiveness of the designed SCA-SSD hand detector in hand-based space robot teleoperation, the experiment on the space environment images should be conducted. Since there is no such hand detection dataset, we customize a set of astronaut hand images in various intra/extravehicular activities from some sci-fi movies and YouTube resource. We named it AHD dataset. The dataset includes a total of 2000 images and more than 4000 instances. All hands in the images are labelled as "hand."

Table 4: Ablation experiment results on the Oxford hand dataset.

Model	mAP	FPS
Baseline [37]	40.2	119
w/ channel	41.8	116
w/ spatial	41.5	116
SCA-SSD w/o fusion	43.8	114
SCA-SSD w/fusion	44.6	104

Table 5: The verification results of the SCA-SSD hand detector on the AHD dataset.

IoU	Area	mAP
0.50 : 0.95	All	0.69
0.50	All	0.88
0.75	All	0.82
0.50 : 0.95	Small	0.56
0.50 : 0.95	Medium	0.62
0.50 : 0.95	Large	0.81

4.3.2. Verification Experiment and Visualization. The AHD dataset is just used for verification. The hand detector trained on the Oxford hands dataset is verification on the AHD dataset, and the results are shown in Table 3. From Table 5, we can see that when the IoU is 0.50 : 0.95, the hand detect accuracy is 0.69. And when the IoU is 0.50, the hand detection accuracy is 0.88. It is proved that the SCA-SSD

Figure 6: Some hand detection results on the AHD dataset.

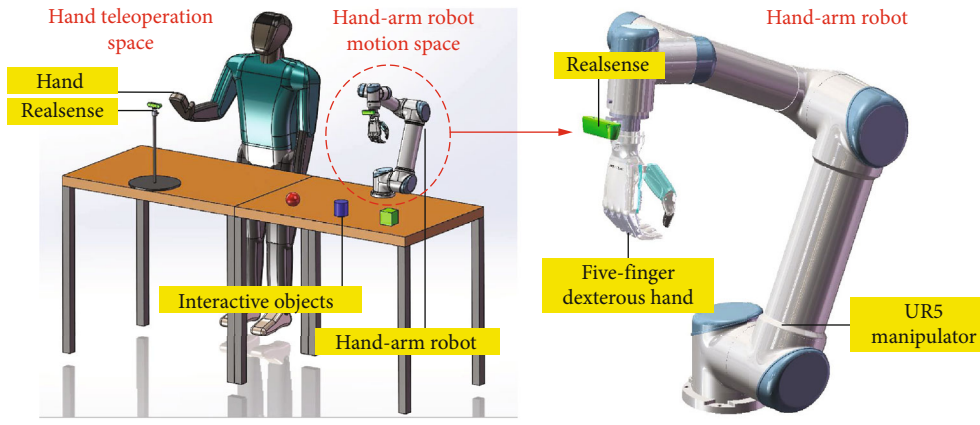


Figure 7: Space robot teleoperation platform. (a) The platform includes a hand teleoperation space and a hand-arm robot motion space. (b) The hand-arm robot includes a UR5 manipulator, a five-finger dexterous hand, and a RealSense.

hand detector can achieve good performance on the AHD dataset. And when the hand areas are small, medium, and large, the hand detection accuracies are 0.56, 0.62, and 0.81, respectively. It is proved that the SCA-SSD hand detector can achieve good performance on hands with various areas.

To better show the results of the hand detection for astronaut's hand, some of the result images are visualized as follows. From Figure 6, we can see that the proposed SCA-SSD hand detector can detect astronaut's hands in various scenes.

4.4. Experiments on Space Robot Teleoperation Platform. The SCA-SSD hand detector is utilized in a designed space robot teleoperation platform, which is shown in Figure 7. The teleoperation platform includes a hand teleoperation space and a hand-arm robot motion space. A RealSense camera can

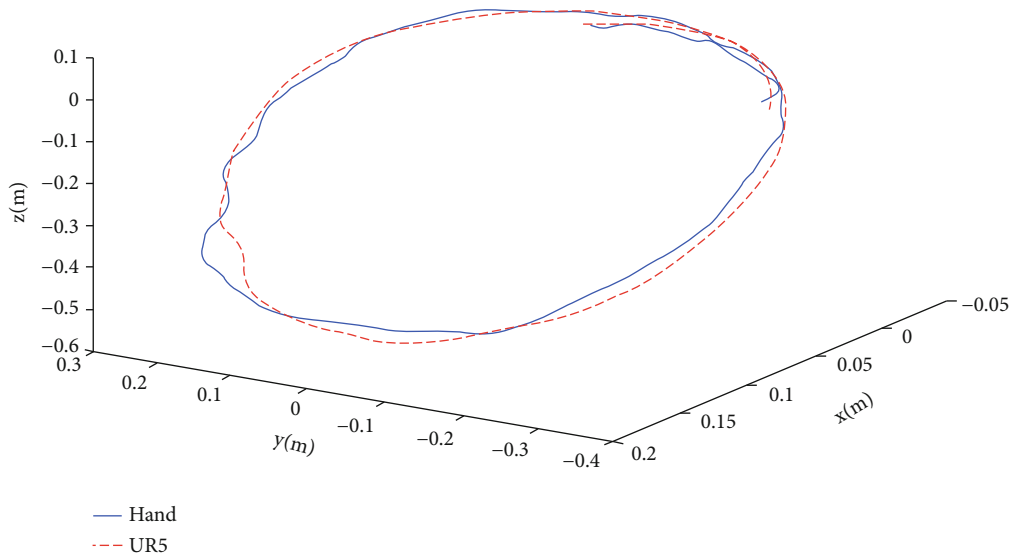


Figure 8: Motion trajectories of hand and the end effector of UR5.

capture the astronaut's hands in real time. After that, the SCA-SSD hand detector can detect hands on the RGB images, and then, the 2D hand positions can be mapped to the corresponding depth images to obtain the 3D hand positions. Then, the real-time hand positions in the hand teleoperation space can be transferred to the hand-arm robot motion space by using the following mapping relationship equation.

$$\begin{cases} x_i^R = x_{i-1}^R + \lambda\left[x_i^H - x_{i-1}^H\right], \\ y_i^R = y_{i-1}^R + \lambda\left[y_i^H - y_{i-1}^H\right], \\ z_i^R = z_{i-1}^R + \lambda\left[z_i^H - z_{i-1}^H\right], \end{cases} \tag{3}$$

where the (x^R, y^R, z^R) is the position of the end effector of the robot, and the (x^H, y^H, z^H) is the hand position in the camera coordinate system. (x_i^R, y_i^R, z_i^R) is the hand position in i-th frame, and $(x_{i-1}^R, y_{i-1}^R, z_{i-1}^R)$ is the hand position in $(i-1)$-th frame. λ is a scale factor, and we set $\lambda = 1$ in the teleoperation experiment.

By collecting the motion trajectories of hand in the camera coordinate system and robot end effector in the robot coordinate system, the trajectories are shown in Figure 8.

From Figure 8, we can see that the end effector of the robot can track the movement trajectory of the hand very well. And the maximum error is only 9.3 mm.

5. Conclusion and Future Work

In this work, a fast and accurate hand detection method was proposed by using a spatial-channel attention single shot multibox detector (SCA-SSD). And the proposed hand detector was utilized in a hand-based space robot teleoperation system. Specifically, two shallow layers were fused with deep layers by using feature fusion modules to increase the restricted receptive field in shallow layers. And spatial attention and channel-wise attention were also used to extract more efficient features. This method can not only ease the computational burden but also bring more contextual information. The comparative experiment, ablation experiment, and verification experiment have proved the good performance of the proposed SCA-SSD hand detector. Finally, the experiment on space robot teleoperation platform has demonstrated that the proposed SCA-SSD hand detector can be applied well in the space robot teleoperation. There are some limitations of the proposed hand detection and teleoperation method. First, the proposed method is only trained on public datasets, and due to the small sizes of the public datasets, the generalization ability of hand detection is not strong. Second, only the detection and localization of hands cannot control the space robots well, and the subsequent recognition of hand gestures and poses is also required.

In the future, hand gesture recognition methods need further research to realize space robot teleoperation for complex tasks. In addition, skeleton-based hand detection and pose estimation also require further research to achieve more precise teleoperation.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no competing interests regarding the publication of this paper.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (62006204 and 62103407) and partly supported by the Shenzhen Outstanding Scientific and Technological Innovation Talents Training Project (RCBS20210609104516043).

References

[1] K. Hambuchen, J. Marquez, and T. Fong, "A review of NASA human-robot interaction in space," *Current Robotics Reports*, vol. 2, no. 3, pp. 265–272, 2021.

[2] M. Shahbazi, S. F. Atashzar, and R. V. Patel, "A systematic review of multilateral teleoperation systems," *IEEE Transactions on Haptics*, vol. 11, no. 3, pp. 338–356, 2018.

[3] C. Freschi, V. Ferrari, F. Melfi, M. Ferrari, F. Mosca, and A. Cuschieri, "Technical review of the da Vinci surgical telemanipulator," *The International Journal of Medical Robotics and Computer Assisted Surgery*, vol. 9, no. 4, pp. 396–406, 2013.

[4] A. Bolopion and S. Régnier, "A review of haptic feedback teleoperation systems for micromanipulation and microassembly," *IEEE Transactions on Automation Science and Engineering*, vol. 10, no. 3, pp. 496–502, 2013.

[5] Y. Liang, G. Du, C. Li, C. Chen, X. Wang, and P. X. Liu, "A gesture-based natural human-robot interaction interface with unrestricted force feedback," *IEEE Transactions on Instrumentation and Measurement*, 2022.

[6] R. Li, H. Wang, and Z. Liu, "Survey on mapping human hand motion to robotic hands for teleoperation," *IEEE Transactions on Circuits and Systems for Video Technology*, p. 1, 2021.

[7] B. Fang, D. Guo, F. Sun, H. Liu, and Y. Wu, "A robotic hand-arm teleoperation system using human arm/hand with a novel data glove," in *In 2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 2483–2488, Zhuhai, China, 2015.

[8] L. Dipietro, A. M. Sabatini, and P. Dario, "A survey of glove-based systems and their applications," *Ieee transactions on systems, man, and cybernetics, part c (applications and reviews)*, vol. 38, no. 4, pp. 461–482, 2008.

[9] C. Mizera, T. Delrieu, V. Weistroffer, C. Andriot, A. Decatoire, and J.-P. Gazeau, "Evaluation of hand-tracking systems in teleoperation and virtual dexterous manipulation," *IEEE Sensors Journal*, vol. 20, no. 3, pp. 1642–1655, 2019.

[10] S. Raspopovic, M. Capogrosso, F. M. Petrini et al., "Restoring natural sensory feedback in real-time bidirectional hand prostheses," *Science Translational Medicine*, vol. 6, no. 222, p. 222ra19, 2014.

[11] X. Lv, C. Dai, H. Liu et al., "Gesture recognition based on sEMG using multi-attention mechanism for remote control," *Neural Computing and Applications*, pp. 1–11, 2022.

[12] S. E. Ovur, X. Zhou, W. Qi et al., "A novel autonomous learning framework to enhance sEMG-based hand gesture recognition using depth information," *Biomedical Signal Processing and Control*, vol. 66, p. 102444, 2021.

[13] W. Liu, D. Anguelov, D. Erhan et al., "Ssd: single shot multibox detector," in *In European conference on computer vision, Computer Vision – ECCV 2016*, pp. 21–37, Springer, Cham, 2016.

[14] S. Li, J. Jiang, P. Ruppel et al., "A mobile robot hand-arm teleoperation system by vision and imu," in *In 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 10900–10906, Las Vegas, NV, USA, 2021.

[15] A. Handa, K. V. Wyk, and W. Yang, "Dexpilot: vision-based teleoperation of dexterous robotic hand-arm system," in *In 2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 9164–9170, Paris, France, 2020.

[16] A. Sivakumar, K. Shaw, and D. Pathak, "Robotic telekinesis: learning a robotic hand imitator by watching humans on Youtube," 2022, http://arxiv.org/abs/2202.10448.

[17] M. Kölsch and M. A. Turk, "Robust hand detection," *Current Robotics Reports*, vol. 4, pp. 614–619, 2004.

[18] Q. Gao, J. Liu, and Z. Ju, "Robust real-time hand detection and localization for space human–robot interaction based on deep learning," *Neurocomputing*, vol. 390, pp. 198–206, 2020.

[19] Q. Gao, J. Liu, Z. Ju, and X. Zhang, "Dual-hand detection for human–robot interaction by a parallel network based on hand detection and body pose estimation," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 12, pp. 9663–9672, 2019.

[20] J. Yu, H. Gao, D. Zhou, J. Liu, Q. Gao, and Z. Ju, "Deep temporal model-based identity-aware hand detection for space human-robot interaction," *IEEE Transactions on Cybernetics*, pp. 1–14, 2021.

[21] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: towards real-time object detection with region proposal networks," *Advances In Neural Information Processing Systems*, vol. 28, 2015.

[22] Z. Cai and N. Vasconcelos, "Cascade r-cnn: delving into high quality object detection," in *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6154–6162, Salt Lake City, 2018.

[23] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: unified, real-time object detection," in *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, Silicon Valley, 2016.

[24] J. Redmon and A. Farhadi, "Yolov3: an incremental improvement," 2018, http://arxiv.org/abs/1804.02767.

[25] A. Bochkovskiy, C. Wang, and H. M. Liao, "Yolov4: optimal speed and accuracy of object detection," 2020, http://arxiv.org/abs/2004.10934.

[26] C. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg, "Dssd: deconvolutional single shot detector," 2017, http://arxiv.org/abs/1701.06659.

[27] G. Cao, X. Xie, W. Yang, Q. Liao, G. Shi, and J. Wu, "Feature-fused SSD: fast detection for small objects," *In Ninth International Conference on Graphic and Image Processing (ICGIP 2017)*, vol. 10615, p. 106151E, 2018.

[28] Z. Li and F. Zhou, "FSSD: feature fusion single shot multibox detector," 2017, http://arxiv.org/abs/1712.00960.

[29] S. Zhai, D. Shang, S. Wang, and S. Dong, "DF-SSD: an improved SSD object detection algorithm based on DenseNet and feature fusion," *IEEE access*, vol. 8, pp. 24344–24357, 2020.

[30] J. Jeong, H. Park, and N. Kwak, "Enhancement of SSD by concatenating feature maps for object detection," 2017, http://arxiv.org/abs/1705.09587.

[31] J. Leng and Y. Liu, "An enhanced SSD with feature fusion and visual reasoning for object detection," *Neural Computing and Applications*, vol. 31, no. 10, pp. 6549–6558, 2019.

[32] J. Park, S. Woo, J. Lee, and I. S. Kweon, "Bam: bottleneck attention module," 2018, http://arxiv.org/abs/1807.06514.

[33] M. Everingham, L. V. Gool, C. K. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes (voc) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.

[34] A. Mittal, A. Zisserman, and P. H. Torr, "Hand detection using multiple proposals," *In Bmvc*, vol. 2, no. 3, p. 5, 2011.

[35] A. Paszke, S. Gross, F. Massa et al., "Pytorch: an imperative style, high-performance deep learning library," *Advances In Neural Information Processing Systems*, vol. 32, 2019.

[36] S. Liu and D. Huang, "Receptive field block net for accurate and fast object detection," in *In Proceedings of the European conference on computer vision (ECCV)*, pp. 385–400, 2018.

[37] C. Li, *High Quality, Fast, Modular Reference Implementation of SSD in PyTorch*, 2018, https://github.com/lufficc/SSD.

[38] J. Dai, Y. Li, K. He, and J. Sun, "R-fcn: object detection via region-based fully convolutional networks," *Advances In Neural Information Processing Systems*, vol. 29, 2016.

[39] S. Woo, S. Hwang, and I. S. Kweon, "Stairnet: top-down semantic aggregation for accurate one shot detection," in *In 2018 IEEE winter conference on applications of computer vision (WACV)*, pp. 1093–1102, Lake Tahoe, NV, USA, 2018.

[40] J. Lim, M. Astrid, H. Yoon, and S. Lee, "Small object detection using context and attention," in *In 2021 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, pp. 181–186, Jeju Island, Korea (South), 2021.

[41] L. Fang, X. Zhao, and S. Zhang, "Small-objectness sensitive detection based on shifted single shot detector," *Multimedia Tools and Applications*, vol. 78, no. 10, pp. 13227–13245, 2019.