sensors

MDPI

Article

Target Capturing Control for Space Robots with Unknown Mass Properties: A Self-Tuning Method Based on Gyros and Cameras

Zhenyu Li *, Bin Wang * and Hong Liu

State Key Laboratory of Robotics and System, Harbin Institute of Technology, Harbin 150000, China; dlrhitlab@aliyun.com

* Correspondence: lizhenyuhit@sohu.com (Z.L.); wbhit@hit.edu.cn (B.W.); Tel.: +86-177-1021-4133 (Z.L.); +86-186-4653-0985 (B.W.)

Academic Editor: Dan Zhang

Received: 24 June 2016; Accepted: 25 August 2016; Published: 30 August 2016

Abstract: Satellite capturing with free-floating space robots is still a challenging task due to the non-fixed base and unknown mass property issues. In this paper gyro and eye-in-hand camera data are adopted as an alternative choice for solving this problem. For this improved system, a new modeling approach that reduces the complexity of system control and identification is proposed. With the newly developed model, the space robot is equivalent to a ground-fixed manipulator system. Accordingly, a self-tuning control scheme is applied to handle such a control problem including unknown parameters. To determine the controller parameters, an estimator is designed based on the least-squares technique for identifying the unknown mass properties in real time. The proposed method is tested with a credible 3-dimensional ground verification experimental system, and the experimental results confirm the effectiveness of the proposed control scheme.

Keywords: free-floating space robot; autonomous target capturing; space robot modeling; self-tuning control; mass property estimation

1. Introduction

With development of space technology, modern space missions are becoming much more complex than ever before, including tasks such as on-orbit refueling, damaged module replacement, target satellite capture, etc. To complete such challenging missions, many attempts have been made to design special robots for space applications. The free-floating space robot was invented as a particular kind of robotic system for on-orbit satellite recovery and maintenance. It employs a free-flying satellite and robotic manipulators mounted on it, so that it can perform on-orbit approaching and target operating tasks. For extending the service life, the position and attitude of the carrier satellite are totally uncontrolled during operations. There have been several successful free-flying space robots, such as Engineering Test Satellite VII (ETS–VII) which is the first space robot project that successfully captured an autonomous on-orbit target [1–3] and the orbital express plan of NASA [4–6], etc. Other free-floating space robot projects have also been carried out as described in [7,8].

The kinematics, dynamics and control of a free-floating space robot are much more complex compared to their ground counterparts, because of the free-floating base satellite. Since the base satellite is free-floating, any robot arm movements may result in changes to the base satellite position and attitude due to the dynamical interactions between the manipulator and the satellite carrier [9,10]. To solve this problem, a modeling technique was proposed by Umetani and Yoshida [11]. They introduced the momentum conservation law into the space robot kinematics formulation, and put forward the concept of the generalized Jacobian matrix. Another successful modeling technique known as the Virtual Manipulator (VM) technique was proposed by Vafa and Dubowsky [12]. They simplified the kinematics equation by decoupling the system centroid translational degrees of freedom. The VM technique sets up a virtual ground-fixed manipulator to describe the motions of the free-floating space robot. Dubowsky and Papadopoulos compared the structure of the motion equations for space robots and ground-fixed robots in [13]. They concluded that if the carrier satellite attitude can be measured or calculated, almost all the ground robot control strategies can be applied to a free-floating space robot. Based on these models and concepts, a number of control methods have been proposed for free-floating space robots. A variable structure control strategy was presented by Fang [14], where a neural network controller is used as the dynamic compensator. Huang et al. studied the tethered space robot (TSR), and presented several trajectory planning and control methods [15–17]. To stabilize the TSR during capture impact with target, Huang derived the impact dynamic model for target capturing and an adaptive robust controller was designed accordingly [18]. Because the collision during the capture and the original target rotation leads to a tumbling of the tethered space robot–target combination system, a robust adaptive backstepping controller was designed to realize stabilization after the target is captured [19]. Huang also investigated the spacecraft attitude takeover control for extending the life time of fuel-exhausted spacecraft [20]. He designed a reconfigurable control system handling the attitude control problem including the mass property changes. Pathak proposed a robust overwhelming control method for space robot and verified it by numerical simulations [21]. Tsuchiya studied the satellite attitude dynamics of space robot and proposed an attitude control scheme based on the reaction wheels [22]. Rastegari and Moosavian suggested a multiple impedance control approach for free-flying space robots to track the path and tune the inner forces at the same time [23]. Zarafshan and Moosavian investigated the dynamics of space robot with flexible elements and proposed a hybrid suppression control method [24]. Considering that pure motion control is not applicable for the satellite-capturing tasks, sensor-based control methods are also applied in space robot engineering. One of the most effective sensor-based robot control strategies is the visual servoing. Instead of the "looking" then "moving" mode which combines the visual sensor and the robotic system in an open-loop fashion, visual servoing introduces a visual feedback control loop to increase the accuracy of the overall system. The basic conceptual framework of visual servoing for robotic manipulators is introduced in Hutchinson's article [25]. In this paper, two major classes of visual servoing systems were discussed in detail. Chaumette and Hutchinson described the basic approaches and advanced techniques of visual servoing in [26,27] and the performance and stability issues of the two visual servoing schemes were discussed. The visual servoing techniques have also been approved in space robot engineering by several scholars as reported in [28–31].

Although a lot of achievements have been made in space robot modelling and control, there are still several issues to be considered in practical engineering. One practical problem is that the mass properties of the carrier satellite keep changing due to the fuel consumption, solar panel adjustments, carrying a captured payload with undetermined mass, etc. These unknown properties are critical and recognized as a challenging problem in space robot target capture control. Since few efforts have been made for controller designing of space robots with unknown satellite mass properties, the undetermined properties of robotic systems are need to be known in advance. Yoshida and Abiko presented an approach for determining space robot inertia parameters [32]. Their identification algorithm is designed based on the conservation of momentum and the gravity gradient torque. In this method, only the reaction wheel motion rates need to be measured. Ma and Dang proposed another method to identify the carrier satellite inertia properties [33]. By using the linear and angular satellite velocities, this issue is treated as a linear identification problem. According to the existing literatures, the unknown satellite mass properties are generally determined based on the conservation-of-momentum principle. The largest advantages of such identification methods are: (1) they don't consume any fuel because the properties are estimated by manipulator motions; (2) since the accelerations and forces aren't directly involved in the calculation, in theory, only velocities need to be measured, which are generally less noisy. However, several difficulties still must be considered in identifying the satellite mass, which plays a very important role in space robot dynamics. One difficulty is that the carrier-satellite mass has quite low sensitivity to the angular motions of the satellite, which means it can't be identified only by reaction wheel motion rates or gyro signals. The other problem is that although the satellite mass can be identified by the linear velocities of the carrier satellite and no acceleration or force is directly involved in the computation, in fact, the linear velocities can't be measured as easily as the angular velocities. In engineering, linear velocities are usually integrated from accelerometer data, which brings drifting errors. Based on such considerations, new identification methods should be proposed trying to handle these issues.

In this paper, a new identification strategy is proposed to estimate the unknown mass properties without computing the linear velocity of the carrier satellite. The eye-in-hand camera signals, which are commonly used in space robot target capture, are adopted as well as the gyro data for identifying both the satellite mass and the centroid position. Because the unknown mass properties are estimated in real time, the self-tuning control scheme is applied to handle this capturing control problem including unknown parameters. The whole approach is established based on a new space robot model for reducing the computation complexity. The rest of this paper is organized as follows: Section 2 describes the new space robot modeling method proposed in this paper; Section 3 designs the self-tuning scheme and presents the new identification method; Section 4 introduces the ground verification experimental system and gives the simulation results; and the results are discussed in Section 5.

2. Free-Floating Space Robot Kinematics Modeling

The space robot system, consisting of a carrier satellite and a space manipulator mounted on it, is shown in Figure 1.

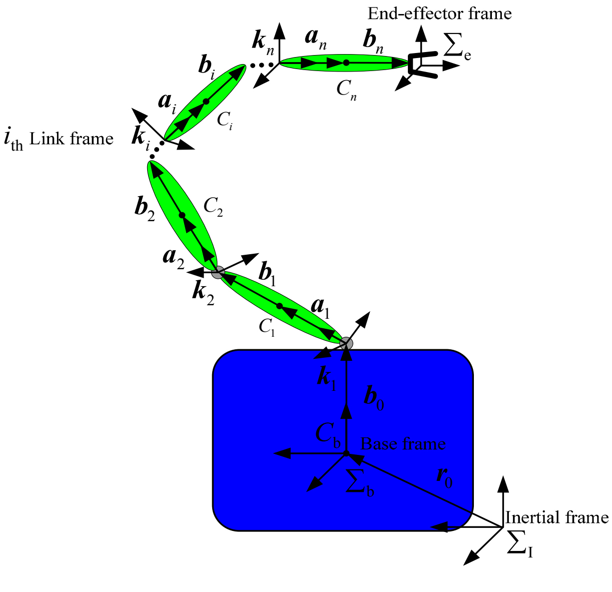

Figure 1. Space robot system.

$\boldsymbol{a}_i$ is the position vector from the i_{th} joint of the robotic arm to the centroid of the i_{th} link. $\boldsymbol{b}_i$ is the position vector from the centroid of the i_{th} link to the $i+1_{\text{th}}$ joint. $\boldsymbol{k}_i$ is the rotation vector of the i_{th} joint. $\boldsymbol{r}_0$ represents the position of the carrier satellite. $\boldsymbol{b}_0$ is the position vector from the carrier satellite centroid to the first joint.

According to Figure 1, the position of space robot end-effector can be determined as:

$$\boldsymbol{P}_e = \boldsymbol{r}_0 + \sum_{i=1}^{n} \boldsymbol{a}_i + \sum_{i=0}^{n} \boldsymbol{b}_i \tag{1}$$

where $\boldsymbol{r}_0$ is not a constant vector, because the carrier satellite is free-floating in space. The centroid position of the space robot system can be expressed as:

$$\boldsymbol{r}_{\mathrm{g}} = \frac{\sum\limits_{i=1}^{n} m_i \boldsymbol{r}_i + m_0 \boldsymbol{r}_0}{M} \tag{2}$$

where m_i is the mass of the i_{th} link; m_0 is the mass of the carrier satellite; $\boldsymbol{r}_i$ is the centroid position of the i_{th} link; M is the total mass of the space robot system. They can be obtained by:

$$\boldsymbol{r}_i = \boldsymbol{r}_0 + \sum_{k=0}^{i-1} \boldsymbol{b}_k + \sum_{k=1}^{i} \boldsymbol{a}_k \tag{3}$$

$$M = \sum_{i=0}^{n} m_i \tag{4}$$

Assuming that there is no external force acting on the space robot, the centroid position of the space robot system will not change. By Equations (2)–(4), the position of the carrier satellite can be determined as:

$$\boldsymbol{r}_0 = \boldsymbol{r}_{\mathrm{g}} - \frac{1}{M}\left(\sum_{i=1}^{n}\left(\sum_{k=i}^{n} m_k\right)\boldsymbol{a}_i\right) - \frac{1}{M}\left(\sum_{i=0}^{n-1}\left(\sum_{k=i+1}^{n} m_k\right)\boldsymbol{b}_i\right) \tag{5}$$

According to Equation (5), a hypothetical manipulator can be set up to describe the linear motion of the carrier satellite. The length vector from the i_{th} joint of the hypothetical manipulator to the joint $i+1$, is defined as:

$$\boldsymbol{l}_{i'} = \begin{cases} \gamma \overset{\frown}{\boldsymbol{b}}_i & i = 0 \\ \gamma \left(\overset{\frown}{\boldsymbol{a}}_i + \overset{\frown}{\boldsymbol{b}}_i\right) & 0 < i < n \\ \gamma \overset{\frown}{\boldsymbol{a}}_i & i = n \end{cases} \tag{6}$$

where:

$$\begin{cases} \overset{\frown}{\boldsymbol{a}}_i = \left(\sum\limits_{k=i}^{n} m_k\right)\boldsymbol{a}_i \\ \overset{\frown}{\boldsymbol{b}}_i = \left(\sum\limits_{k=i+1}^{n} m_k\right)\boldsymbol{b}_i \\ \gamma = -\frac{1}{M} \end{cases} \tag{7}$$

Attaching the hypothetical manipulator to the centroid of the space robot system, the end position of the hypothetical manipulator can be calculated as:

$$\begin{aligned} \boldsymbol{P}_{\mathrm{e}'} &= \boldsymbol{r}_{\mathrm{g}} + \gamma \sum_{i=1}^{n} \overset{\frown}{\boldsymbol{a}}_i + \gamma \sum_{i=0}^{n-1} \overset{\frown}{\boldsymbol{b}}_i \\ &= \boldsymbol{r}_0 \end{aligned} \tag{8}$$

According to Equation (8), the carrier satellite position can be obtained by the hypothetical manipulator motion. In this case, the hypothetical manipulator is referred as the carrier manipulator. The position of the space robot end-effector can be expressed as:

$$\begin{aligned} \boldsymbol{P}_e &= \boldsymbol{r}_0 + \sum_{i=1}^{n} \boldsymbol{a}_i + \sum_{i=0}^{n} \boldsymbol{b}_i \\ &= \boldsymbol{P}_{\mathrm{e}'} + \boldsymbol{L}_{0\mathrm{e}} \end{aligned} \tag{9}$$

where $\boldsymbol{L}_{0e}$ is the position vector from the carrier satellite centroid to the space robot end. If the carrier satellite is considered as the 0_{th} link, the space robot can be considered as another hypothetical manipulator mounted on the carrier manipulator end. Since it performs the same operation motions as the space manipulator, it is referred as the service manipulator. Consequently, the free-floating space robot system is equivalent to a hypothetical ground-fixed robotic system including two manipulators. The hypothetical robotic system can be described as Figure 2.

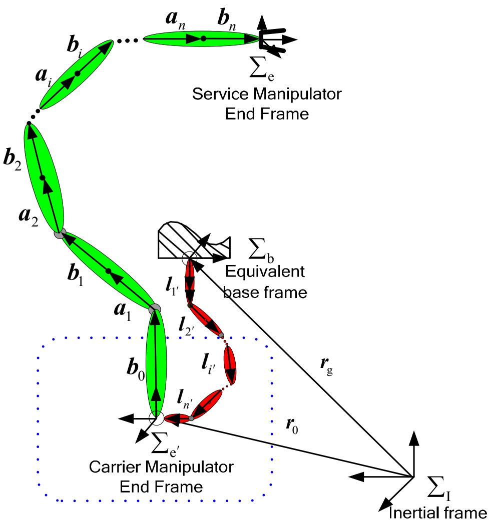

Figure 2. Equivalent robotic system.

According to Equation (9), the velocity vector of the service manipulator end, representing the space robot end-effector, can be expressed as:

$$v_e = \dot{\boldsymbol{P}}_e = \dot{\boldsymbol{P}}_{e'} + \dot{\boldsymbol{L}}_{0e} \tag{10}$$

where:

$$\dot{\boldsymbol{L}}_{0e} = \sum_{i=1}^{n} \dot{\boldsymbol{a}}_i + \sum_{i=0}^{n} \dot{\boldsymbol{b}}_i \tag{11}$$

$$\dot{\boldsymbol{P}}_{e'} = \gamma \sum_{k=1}^{n} \dot{\widehat{\boldsymbol{a}}}_k + \gamma \sum_{k=0}^{n-1} \dot{\widehat{\boldsymbol{b}}}_k \tag{12}$$

and $\dot{\boldsymbol{a}}_i$, as well as $\dot{\boldsymbol{b}}_i$ can be determined by:

$$\begin{cases} \dot{\boldsymbol{a}}_i = \omega_i \times \boldsymbol{a}_i \\ \dot{\boldsymbol{b}}_i = \omega_i \times \boldsymbol{b}_i \end{cases} \tag{13}$$

where ω_i is the angular velocity of the i_{th} link and can be described as:

$$\omega_i = \omega_0 + \sum_{k=1}^{i} \boldsymbol{k}_k \dot{\theta}_k \tag{14}$$

and ω_0 is the satellite angular velocity; $\dot{\theta}_k$ is the angle velocity of the k_{th} link. Substituting Equations (13) and (14) into Equation (11) gives:

$$\begin{aligned} \dot{\boldsymbol{L}}_{0e} &= \sum_{i=1}^{n} \left[\left(\omega_0 + \sum_{k=1}^{i} \boldsymbol{k}_k \dot{\theta}_k \right) \times \boldsymbol{a}_i \right] + \sum_{i=1}^{n} \left[\left(\omega_0 + \sum_{k=1}^{i} \boldsymbol{k}_k \dot{\theta}_k \right) \times \boldsymbol{b}_i \right] + \omega_0 \times \boldsymbol{b}_0 \\ &= \omega_0 \times \left(\sum_{i=0}^{n} \boldsymbol{b}_i + \sum_{i=1}^{n} \boldsymbol{a}_i \right) + \sum_{i=1}^{n} \left[\boldsymbol{k}_i \times \left(\sum_{k=i}^{n} \boldsymbol{b}_k + \sum_{k=i}^{n} \boldsymbol{a}_k \right) \dot{\theta}_i \right] \end{aligned} \tag{15}$$

Defining $\boldsymbol{L}_{i\mathrm{e}}$ as the position vector from the i_{th} joint to the service manipulator end, it can be determined as:

$$\boldsymbol{L}_{ie} = \sum_{k=i}^{n} \boldsymbol{a}_k + \sum_{k=i}^{n} \boldsymbol{b}_k \quad n \geq i > 1 \tag{16}$$

According to Equations (15) and (16), $\dot{\boldsymbol{L}}_{0\mathrm{e}}$ is derived as:

$$\begin{aligned} \dot{\boldsymbol{L}}_{0\mathrm{e}} &= \omega_0 \times \boldsymbol{L}_{0\mathrm{e}} + \sum_{i=1}^{n} \left[\boldsymbol{k}_i \times \boldsymbol{L}_{i\mathrm{e}} \dot{\theta}_i \right] \\ &= \left(\boldsymbol{J}_{\mathrm{r}} - \widetilde{\boldsymbol{b}}_0 \right) \omega_0 + \boldsymbol{J}_{\mathrm{m}} \dot{\theta} \end{aligned} \tag{17}$$

where:

$$\begin{cases} \boldsymbol{J}_{\mathrm{r}} = -\widetilde{\boldsymbol{L}}_{1\mathrm{e}} \\ \boldsymbol{J}_{\mathrm{m}} = \begin{bmatrix} \boldsymbol{k}_1 \times \boldsymbol{L}_{1\mathrm{e}} & \cdots & \boldsymbol{k}_i \times \boldsymbol{L}_{i\mathrm{e}} & \cdots & \boldsymbol{k}_n \times \boldsymbol{L}_{n\mathrm{e}} \end{bmatrix} \end{cases}$$

According to Equations (7) and (13), $\dot{\widehat{\boldsymbol{a}}}_k$ and $\dot{\widehat{\boldsymbol{b}}}_k$ can be defined by:

$$\begin{cases} \dot{\widehat{\boldsymbol{a}}}_k = \omega_k \times \left(\sum_{q=k}^{n} m_q \right) \boldsymbol{a}_k = \omega_k \times \widehat{\boldsymbol{a}}_k \\ \dot{\widehat{\boldsymbol{b}}}_k = \omega_k \times \left(\sum_{q=k+1}^{n} m_q \right) \boldsymbol{b}_k = \omega_k \times \widehat{\boldsymbol{b}}_k \end{cases} \tag{18}$$

Substituting Equations (14) and (18) into Equation (12), we have:

$$\begin{aligned} \dot{\boldsymbol{P}}_{\mathrm{e}'} &= \gamma \sum_{k=1}^{n} \left[\left(\omega_0 + \sum_{i=1}^{k} \boldsymbol{k}_i \dot{\theta}_i \right) \times \widehat{\boldsymbol{a}}_k \right] + \gamma \sum_{k=1}^{n-1} \left[\left(\omega_0 + \sum_{i=1}^{k} \boldsymbol{k}_i \dot{\theta}_i \right) \times \widehat{\boldsymbol{b}}_k \right] + \gamma \omega_0 \times \widehat{\boldsymbol{b}}_0 \\ &= \gamma \omega_0 \times \left(\sum_{k=0}^{n-1} \widehat{\boldsymbol{b}}_k + \sum_{k=1}^{n} \widehat{\boldsymbol{a}}_k \right) + \gamma \sum_{i=1}^{n} \left[\boldsymbol{k}_i \times \left(\sum_{k=i}^{n-1} \widehat{\boldsymbol{b}}_k + \sum_{k=i}^{n} \widehat{\boldsymbol{a}}_k \right) \dot{\theta}_i \right] \end{aligned} \tag{19}$$

Defining $\boldsymbol{L}_{i'e'}$ as the position vector from the i_{th} joint to the end of the carrier manipulator, it can be expressed as:

$$\boldsymbol{L}_{i'e'} = \gamma \widehat{\boldsymbol{L}}_{ie} \tag{20}$$

where:

$$\widehat{\boldsymbol{L}}_{ie} = \begin{cases} \widehat{\boldsymbol{a}}_n & i = n \\ \left(\sum_{k=i}^{n} \widehat{\boldsymbol{a}}_k + \sum_{k=i}^{n-1} \widehat{\boldsymbol{b}}_k \right) & 0 < i < n \\ \left(\sum_{k=1}^{n} \widehat{\boldsymbol{a}}_k + \sum_{k=0}^{n-1} \widehat{\boldsymbol{b}}_k \right) & i = 0 \end{cases} \tag{21}$$

According to Equations (19) and (20), $\dot{\boldsymbol{P}}_{\mathrm{e}'}$ can be further derived as:

$$\begin{aligned} \dot{\boldsymbol{P}}_{\mathrm{e}'} &= \omega_0 \times \boldsymbol{L}_{0'\mathrm{e}'} + \sum_{i=1}^{n} \left[\boldsymbol{k}_i \times \boldsymbol{L}_{i'\mathrm{e}'} \dot{\theta}_i \right] \\ &= \left(\boldsymbol{J}_{\mathrm{r}'} - \gamma M_{\mathrm{m}} \widetilde{\boldsymbol{b}}_0 \right) \omega_0 + \boldsymbol{J}_{\mathrm{m}'} \dot{\theta} \end{aligned} \tag{22}$$

where:

$$\begin{cases} \boldsymbol{J}_{\mathrm{r}'} = -\widetilde{\boldsymbol{L}}_{1'\mathrm{e}'} \\ \boldsymbol{J}_{\mathrm{m}'} = \begin{bmatrix} \boldsymbol{k}_1 \times \boldsymbol{L}_{1'\mathrm{e}'} & \cdots & \boldsymbol{k}_i \times \boldsymbol{L}_{i'\mathrm{e}'} & \cdots & \boldsymbol{k}_n \times \boldsymbol{L}_{n'\mathrm{e}'} \end{bmatrix} \end{cases}$$

and M_{m} is the total mass of the manipulator, known as:

$$M_{\mathrm{m}} = \sum_{i=1}^{n} m_i \tag{23}$$

Defining the Jacobian matrices $\overset{\frown}{\boldsymbol{J}}_{\rm r}$ and $\overset{\frown}{\boldsymbol{J}}_{\rm m}$ as:

$$\begin{cases} \overset{\frown}{\boldsymbol{J}}_{\rm r} = -\widetilde{\overset{\frown}{\boldsymbol{L}}}_{\rm 1e} = \frac{1}{\gamma}\boldsymbol{J}_{\rm r'} \\ \overset{\frown}{\boldsymbol{J}}_{\rm m} = \left[\begin{array}{ccccc} \boldsymbol{k}_1 \times \overset{\frown}{\boldsymbol{L}}_{\rm 1e} & \cdots & \boldsymbol{k}_i \times \overset{\frown}{\boldsymbol{L}}_{i\rm e} & \cdots & \boldsymbol{k}_n \times \overset{\frown}{\boldsymbol{L}}_{n\rm e} \end{array}\right] = \frac{1}{\gamma}\boldsymbol{J}_{\rm m'} \end{cases}$$

as a result, $\dot{\boldsymbol{P}}_{\rm e'}$ also takes the following form:

$$\dot{\boldsymbol{P}}_{\rm e'} = \gamma \left[\left(\overset{\frown}{\boldsymbol{J}}_{\rm r} - M_{\rm m}\widetilde{\boldsymbol{b}}_0 \right) \omega_0 + \overset{\frown}{\boldsymbol{J}}_{\rm m}\dot{\theta} \right] \tag{24}$$

In Equation (24), the unknown satellite mass is only contained in the parameter γ. According to Equations (17) and (24), the differential kinematics equation of free-floating space robot translational motion can be expressed as:

$$\boldsymbol{v}_{\rm e} = \left[\boldsymbol{J}_{\rm r} + \gamma\overset{\frown}{\boldsymbol{J}}_{\rm r} - (1+\gamma M_{\rm m})\,\widetilde{\boldsymbol{b}}_0 \right] \omega_0 + \left(\boldsymbol{J}_{\rm m} + \gamma\overset{\frown}{\boldsymbol{J}}_{\rm m} \right) \dot{\theta} \tag{25}$$

where $\boldsymbol{J}_{\rm m}$ and $\overset{\frown}{\boldsymbol{J}}_{\rm m}$ are exactly the same as the Jacobian matrix of ground-fixed manipulators.

According to Equation (25), two linear velocities are defined as:

$$\begin{cases} \boldsymbol{v}_{\rm eo} = \boldsymbol{J}_{\rm m}\dot{\theta} \\ \boldsymbol{v}_{\rm ec} = \overset{\frown}{\boldsymbol{J}}_{\rm m}\dot{\theta} \end{cases} \tag{26}$$

where $\boldsymbol{v}_{\rm eo}$, as well as $\boldsymbol{v}_{\rm ec}$ can be considered as the linear velocity of a ground-fixed manipulator. Consequently, the end-effector velocity of space robot can be further expressed as:

$$\boldsymbol{v}_{\rm e} = \boldsymbol{v}_{\rm eo} + \omega_0 \times \boldsymbol{L}_{\rm 1e} + \gamma \left(\boldsymbol{v}_{\rm ec} + \omega_0 \times \overset{\frown}{\boldsymbol{L}}_{\rm 1e} \right) + \omega_0 \times \boldsymbol{b}_0 + \gamma M_{\rm m}\omega_0 \times \boldsymbol{b}_0 \tag{27}$$

The main advantage in using Equation (27) is that the unknown satellite mass properties will not be involved in the calculation of $\boldsymbol{v}_{\rm eo}$ and $\boldsymbol{v}_{\rm ec}$, and only two parameters, γ and $\boldsymbol{b}_0$ relating to the undetermined satellite mass and centroid are to be determined. Accordingly, in this paper, γ will be estimated instead of identifying the satellite mass. m_0 can still be determined by Equations (4) and (7) if necessary.

The proposed modeling method is derived by describing the carrier satellite translational motions with a hypothetical manipulator fixed on the centroid of the robotic system and taking the free-floating space robot as an equivalent ground-fixed manipulator system. Comparing with existing modeling methods presented in [11] and [12], several advantages of using this new method are described as follows:

(1) Since the free-floating space robot system is kinematically equivalent to the hypothetical robotic system, this new space robot model is not complex to compute because only ground-fixed robot kinematics are involved in the calculations.

(2) In the new translational-motion equations, the undetermined carrier satellite mass, which is a challenge in parameter estimating as suggested in [32], only impacts a constant factor, namely γ. Accordingly, this new modeling method is more convenient in identifying the satellite mass properties.

3. Self-Tuning Control Designing

Because the mass properties of the carrier satellite are changing throughout the whole service life, precise target capturing of free-floating space robots is considered as a challenging problem.

For improving the control performance, a self-tuning target capturing control scheme is applied by adopting the eye-in-hand camera and gyros. The self-tuning control concept is obtained based on the certainty equivalence principle. By coupling a motion controller with an online parameter estimator, the self-tuning controller can perform simultaneous identification of unknown properties. Accordingly, instead of the unknown true values, the controller parameters are determined by the estimations.

Because the dynamic and kinematic parameters of the space manipulator mounted on the carrier satellite are all constants, the satellite mass properties can be estimated in real time by end-effector translations and satellite rotations. As a consequence, the proposed space robot self-tuning control scheme is as shown in Figure 3.

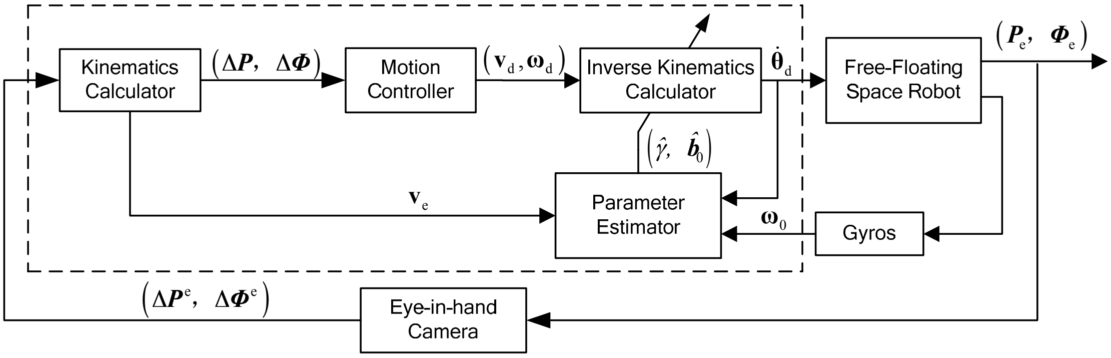
Kinematics Calculator
$(\Delta \boldsymbol{P},\ \Delta \boldsymbol{\Phi})$
Motion Controller
$(\mathbf{v}_{\mathrm{d}}, \boldsymbol{\omega}_{\mathrm{d}})$
Inverse Kinematics Calculator
$\dot{\boldsymbol{\theta}}_{\mathrm{d}}$
Free-Floating Space Robot
$(\boldsymbol{P}_{\mathrm{e}},\ \boldsymbol{\Phi}_{\mathrm{e}})$
$(\hat{\gamma},\ \hat{\boldsymbol{b}}_0)$
$\mathbf{v}_{\mathrm{e}}$
Parameter Estimator
$\boldsymbol{\omega}_0$
Gyros
$(\Delta \boldsymbol{P}^{\mathrm{e}},\ \Delta \boldsymbol{\Phi}^{\mathrm{e}})$
Eye-in-hand Camera

Figure 3. Block diagram of space robot self-tuning control scheme.

The self-tuning control operation is described as follows: the motion controller plans the space robot end-effector motion based on the current relative position and attitude after transforming the camera signals to the inertial frame. At each time instant, a set of property estimations identified by the parameter estimator are sent to the inverse kinematics calculator, which is obtained from the past joint motions and the sensor data from eye-in-hand camera and gyros. Based on these estimated parameters and desired end-effector motions, the joint trajectories are planned by the space robot inverse kinematics calculator. The free-floating space robot joints move following the control input and generate a new output, updating the input data of the parameter estimator and motion controller.

3.1. Kinematics Calculator Designing

Because the outputs of the eye-in-hand camera are the relative position and attitude at the end-effector frame, a kinematics calculator is applied to transform the camera signals to the inertial frame. The relative position and attitude at the inertial frame are computed as:

$$\begin{cases} \Delta \boldsymbol{P} = \boldsymbol{A}_{\mathrm{e}}\left(\theta, \psi\right) \Delta \boldsymbol{P}^{\mathrm{e}} \\ \Delta \Phi = \boldsymbol{A}_{\mathrm{e}}\left(\theta, \psi\right) \Delta \Phi^{\mathrm{e}} \end{cases} \tag{28}$$

where $\boldsymbol{A}_{\mathrm{e}}$ is the rotation matrix; $\Delta \boldsymbol{P}^{\mathrm{e}}$ and $\Delta \Phi^{\mathrm{e}}$ are the relative position and attitude at the end-effector frame; ψ is the satellite attitude. Based on the Euler axis/angle, $\Delta \Phi$ is expressed as:

$$\Delta \Phi = \chi \rho \tag{29}$$

where χ is the Euler rotation angle; ρ is the Euler equivalent axis.

The linear end-effector velocity, which is applied as the parameter estimator input, is also computed by this calculator as:

$$v_{\mathrm{e}} = -\Delta \dot{\boldsymbol{P}} + v_{\mathrm{t}} \tag{30}$$

where v_{t} is the linear velocity of the target satellite, which is assuming to be known or identified by other approaches.

3.2. Motion Controller Designing

Defining the relative pose as the control error of the free-floating space robot, it is written as:

$$\boldsymbol{e} = \begin{bmatrix} \Delta \boldsymbol{P} \\ \Delta \Phi \end{bmatrix} \tag{31}$$

The ideal feed-back response of $\boldsymbol{e}$ is designed as follows:

$$\boldsymbol{K}\boldsymbol{e} + \dot{\boldsymbol{e}} = 0 \tag{32}$$

where $\boldsymbol{K}$ is the matrix of control factors reflecting performance specifications. According to Equations (31) and (32), the motion controller of the free-floating space robot is given by ignoring the target rotation as:

$$\begin{bmatrix} \boldsymbol{v}_{\mathrm{d}} \\ \omega_{\mathrm{d}} \end{bmatrix} = \boldsymbol{K} \begin{bmatrix} \Delta \boldsymbol{P} \\ \Delta \Phi \end{bmatrix} + \begin{bmatrix} \boldsymbol{v}_{\mathrm{t}} \\ 0 \end{bmatrix} \tag{33}$$

3.3. Inverse Kinematics Calculating

According to Equation (25), the carrier satellite angular velocity is to be determined for performing inverse kinematics calculations. There are several ways to obtain the satellite angular velocity. One way is to substitute the angular momentum conservation equation into the kinematics equation. This, however, requires cumbersome computations. Another way is measuring the angular motion directly by gyros, which is especially convenient because almost all satellites are equipped with gyroscopes. Another advantage in adopting gyro information is that it will be unnecessary to identify the satellite inertia tensor matrix for calculating the inverse kinematics. Defining the measured satellite angular velocity as $\overline{\omega}_0$, the angular velocity of space robot end-effector is calculated as:

$$\omega_{\mathrm{e}} = \overline{\omega}_0 + \boldsymbol{J}_{\omega} \dot{\theta} \tag{34}$$

where:

$$\boldsymbol{J}_{\omega} = \begin{bmatrix} \boldsymbol{k}_1 & \cdots & \boldsymbol{k}_i & \cdots & \boldsymbol{k}_1 \end{bmatrix} \tag{35}$$

According to Equations (25) and (34), we have the following differential kinematics equation of the free-floating space robot as:

$$\begin{bmatrix} \boldsymbol{v}_{\mathrm{e}} \\ \omega_{\mathrm{e}} \end{bmatrix} = \begin{bmatrix} J_{\mathrm{r}} + \gamma \overset{\frown}{\boldsymbol{J}}_{\mathrm{r}} - (1 + \gamma M_{\mathrm{m}}) \widetilde{\boldsymbol{b}}_0 \\ \boldsymbol{E} \end{bmatrix} \overline{\omega}_0 + \begin{bmatrix} J_{\mathrm{m}} + \gamma \overset{\frown}{\boldsymbol{J}}_{\mathrm{m}} \\ \boldsymbol{J}_{\omega} \end{bmatrix} \dot{\theta} \tag{36}$$

Consequently, the desired manipulator joint velocities can be obtained as:

$$\dot{\theta}_{\mathrm{d}} = \begin{bmatrix} J_{\mathrm{m}} + \gamma \overset{\frown}{\boldsymbol{J}}_{\mathrm{m}} \\ \boldsymbol{J}_{\omega} \end{bmatrix}^{-1} \left(\begin{bmatrix} \boldsymbol{v}_{\mathrm{d}} \\ \omega_{\mathrm{d}} \end{bmatrix} - \begin{bmatrix} J_{\mathrm{r}} + \gamma \overset{\frown}{\boldsymbol{J}}_{\mathrm{r}} - (1 + \gamma M_{\mathrm{m}}) \widetilde{\boldsymbol{b}}_0 \\ \boldsymbol{E} \end{bmatrix} \overline{\omega}_0 \right) \tag{37}$$

By Equation (37), once the mass properties, namely γ and $\boldsymbol{b}_0$, are determined, space manipulator joint motions can be planned.

3.4. Mass Property Estimating

In this section, a real time estimator is proposed identifying the unknown mass properties relating to the inverse kinematics calculation. According to Equation (37), only γ and $\boldsymbol{b}_0$ are to be estimated which indicate the system total mass and the satellite centroid position, respectively. Defining the linear velocity of the end-effector as a function of γ and $\boldsymbol{b}_0$, it gives:

$$\boldsymbol{v}_{\mathrm{e}} = f\left(\ \gamma, \quad \boldsymbol{b}_0 \ \right) \tag{38}$$

Note that the unknown parameters γ and $\boldsymbol{b}_0$ appear nonlinearly in Equation (38), linear identification techniques cannot be applied directly.

The estimation error is defined as:

$$\boldsymbol{e} = \hat{\boldsymbol{v}}_{\mathrm{e}} - \boldsymbol{v}_{\mathrm{e}} \tag{39}$$

where $\boldsymbol{v}_{\mathrm{e}}$ is the true value of space robot end-effector velocity obtained by the eye-in-hand camera; $\hat{\boldsymbol{v}}_{\mathrm{e}}$ is the calculation employing the estimated mass properties $\hat{\gamma}$ and $\hat{\boldsymbol{b}}_0$. Defining temporarily identified mass properties as $\hat{\gamma}_{\mathrm{tem}}$ and $\hat{\boldsymbol{b}}_{0\mathrm{tem}}$, according to Equation (38), $\boldsymbol{e}$ can be linearized as:

$$\boldsymbol{e} = \left[f\left(\ \hat{\gamma}_{\mathrm{tem}}, \quad \hat{\boldsymbol{b}}_{0\mathrm{tem}} \ \right) + \frac{\partial f\left(\ \hat{\gamma}_{\mathrm{tem}}, \quad \hat{\boldsymbol{b}}_{0\mathrm{tem}} \ \right)}{\partial \gamma} \Delta\hat{\gamma} + \frac{\partial f\left(\ \hat{\gamma}_{\mathrm{tem}}, \quad \hat{\boldsymbol{b}}_{0\mathrm{tem}} \ \right)}{\partial \boldsymbol{b}_0} \Delta\hat{\boldsymbol{b}}_0 \right] - \boldsymbol{v}_{\mathrm{e}} \tag{40}$$

where:

$$\begin{cases} \Delta\hat{\gamma} = \hat{\gamma} - \hat{\gamma}_{\mathrm{tem}} \\ \Delta\hat{\boldsymbol{b}}_0 = \hat{\boldsymbol{b}}_0 - \hat{\boldsymbol{b}}_{0\mathrm{tem}} \end{cases} \tag{41}$$

According to Equation (27), $\frac{\partial f\left(\ \hat{\gamma}_{\mathrm{tem}}, \quad \hat{\boldsymbol{b}}_{0\mathrm{tem}} \ \right)}{\partial \gamma}$ can be simply determined as:

$$\frac{\partial f\left(\ \hat{\gamma}_{\mathrm{tem}}, \quad \hat{\boldsymbol{b}}_{0\mathrm{tem}} \ \right)}{\partial \gamma} = \boldsymbol{v}_{\mathrm{ec}} + \omega_0 \times \overset{\frown}{\boldsymbol{L}}_{1\mathrm{e}} + M_{\mathrm{m}}\omega_0 \times \hat{\boldsymbol{b}}_{0\mathrm{tem}} \tag{42}$$

Defining a new robot linear velocity as:

$$\overset{\frown}{\boldsymbol{v}} = \boldsymbol{v}_{\mathrm{ec}} + \omega_0 \times \overset{\frown}{\boldsymbol{L}}_{1\mathrm{e}} + M_{\mathrm{m}}\omega_0 \times \boldsymbol{b}_0 \tag{43}$$

Equation (42) can be further derived as:

$$\frac{\partial f\left(\ \hat{\gamma}_{\mathrm{tem}}, \quad \hat{\boldsymbol{b}}_{0\mathrm{tem}} \ \right)}{\partial \gamma} = \overset{\frown}{\boldsymbol{v}} \left(\hat{\boldsymbol{b}}_{0\mathrm{tem}} \right) \tag{44}$$

Substituting Equation (27) into Equation (38), $\frac{\partial f\left(\ \hat{\gamma}_{\mathrm{tem}}, \quad \hat{\boldsymbol{b}}_{0\mathrm{tem}} \ \right)}{\partial \boldsymbol{b}_0}$ can be determined as:

$$\frac{\partial f\left(\ \hat{\gamma}_{\mathrm{tem}}, \quad \hat{\boldsymbol{b}}_{0\mathrm{tem}} \ \right)}{\partial \boldsymbol{b}_0} = \left[1 + M_{\mathrm{m}}\hat{\gamma}_{\mathrm{tem}} \right] \tilde{\omega}_0 \tag{45}$$

Defining a robot angular velocity $\overset{\frown}{\omega}$ as:

$$\overset{\frown}{\omega} = \left[1 + M_{\mathrm{m}}\gamma \right] \omega_0 \tag{46}$$

Equation (45) can be further expressed as:

$$\frac{\partial f\left(\ \hat{\gamma}_{\mathrm{tem}}, \quad \hat{\boldsymbol{b}}_{0\mathrm{tem}} \ \right)}{\partial \boldsymbol{b}_0} = \tilde{\overset{\frown}{\omega}} \left(\hat{\gamma}_{\mathrm{tem}} \right) \tag{47}$$

Substituting Equations (44) and (47) into Equation (40), the estimation error can be calculated as:

$$\boldsymbol{e} = f\left(\hat{\gamma}_{\mathrm{tem}}, \quad \hat{\boldsymbol{b}}_{0\mathrm{tem}} \right) + \overset{\frown}{\boldsymbol{v}} \left(\hat{\boldsymbol{b}}_{0\mathrm{tem}}\right) \Delta\hat{\gamma} + \overset{\sim}{\overset{\frown}{\omega}} \left(\hat{\gamma}_{\mathrm{tem}}\right) \Delta\hat{\boldsymbol{b}}_0 - \boldsymbol{v}_{\mathrm{e}} \tag{48}$$

Defining the end-effector velocity calculated by temporarily identified mass properties as $\hat{\boldsymbol{v}}_{\mathrm{tem}}\left(\hat{\gamma}_{\mathrm{tem}}, \quad \hat{\boldsymbol{b}}_{0\mathrm{tem}} \right)$, Equation (48) can be further expressed as:

$$\boldsymbol{e} = \hat{\boldsymbol{v}}_{\mathrm{tem}}\left(\hat{\gamma}_{\mathrm{tem}}, \quad \hat{\boldsymbol{b}}_{0\mathrm{tem}} \right) - \boldsymbol{v}_{\mathrm{e}} + \overset{\frown}{\boldsymbol{v}}_{\mathrm{ec}} \left(\hat{\boldsymbol{b}}_{0\mathrm{tem}}\right) \Delta\hat{\gamma} + \overset{\sim}{\overset{\frown}{\omega}}_0 \left(\hat{\gamma}_{\mathrm{tem}}\right) \Delta\hat{\boldsymbol{b}}_0 \tag{49}$$

According to Equation (41), once parameter errors $\Delta\hat{\gamma}$ and $\Delta\hat{\boldsymbol{b}}_0$ are determined, the estimated mass properties $\hat{\gamma}$ and $\hat{\boldsymbol{b}}_0$ can be simply identified. Since the number of the parameters to be identified is larger than the number of equations described by Equation (49), $\Delta\hat{\gamma}$ and $\Delta\hat{\boldsymbol{b}}_0$ can't be determined by simply assuming that the estimation error vector is $\boldsymbol{0}$. To solve this problem, other constraints need to be introduced.

Employing the least-squares technique, the total estimation error during the time interval $\left(t - \Delta t, \quad t \right)$ is defined as:

$$\boldsymbol{Q} = \int_{t-\Delta t}^{t} \boldsymbol{e}\left(r\right)^{\mathrm{T}} \boldsymbol{e}\left(r\right) dr \tag{50}$$

where Δt is the length of the integral interval. To ensure the total estimation error is minimized, first order partial differential equations of $\boldsymbol{Q}$ can be determined by:

$$\begin{cases} \frac{\partial \boldsymbol{Q}}{\partial \Delta\gamma} = 2\int_{t-\Delta t}^{t} \boldsymbol{e}\left(r\right)^{\mathrm{T}} \frac{\partial \boldsymbol{e}(r)}{\partial \Delta\gamma} dr = 0 \\ \frac{\partial \boldsymbol{Q}}{\partial \Delta\boldsymbol{b}_0} = 2\int_{t-\Delta t}^{t} \frac{\partial \boldsymbol{e}(r)}{\partial \Delta\boldsymbol{b}_0} \boldsymbol{e}\left(r\right) dr = 0_{3\times1} \end{cases} \tag{51}$$

Substituting Equations (44), (47) and (48) into Equation (51), the following equations hold as:

$$\begin{cases} \int_{t-\Delta t}^{t} \overset{\frown}{\boldsymbol{v}}^{\mathrm{T}} \overset{\frown}{\boldsymbol{v}} dr \Delta\hat{\gamma} + \int_{t-\Delta t}^{t} \overset{\frown}{\boldsymbol{v}}^{\mathrm{T}} \overset{\sim}{\overset{\frown}{\omega}} dr \Delta\hat{\boldsymbol{b}}_0 = \int_{t-\Delta t}^{t} \overset{\frown}{\boldsymbol{v}}^{\mathrm{T}} \left(\boldsymbol{v}_{\mathrm{e}} - \hat{\boldsymbol{v}}_{\mathrm{etem}}\right) dr \\ \int_{t-\Delta t}^{t} \overset{\sim}{\overset{\frown}{\omega}} \overset{\frown}{\boldsymbol{v}} dr \Delta\hat{\gamma} + \int_{t-\Delta t}^{t} \overset{\sim}{\overset{\frown}{\omega}} \overset{\sim}{\overset{\frown}{\omega}} dr \Delta\hat{\boldsymbol{b}}_0 = \int_{t-\Delta t}^{t} \overset{\sim}{\overset{\frown}{\omega}} \left(\boldsymbol{v}_{\mathrm{e}} - \hat{\boldsymbol{v}}_{\mathrm{etem}}\right) dr \end{cases} \tag{52}$$

Defining the parameter error vector as:

$$\boldsymbol{X} = \begin{bmatrix} \Delta\hat{\gamma} \\ \Delta\hat{\boldsymbol{b}}_0 \end{bmatrix} \tag{53}$$

according to Equation (52), the identification equation can be formed as:

$$\boldsymbol{A}\boldsymbol{X} = \boldsymbol{B} \tag{54}$$

where:

$$\boldsymbol{A} = \begin{bmatrix} \int_{t-\Delta t}^{t} \overset{\frown}{\boldsymbol{v}}^{\mathrm{T}} \overset{\frown}{\boldsymbol{v}} dr & \int_{t-\Delta t}^{t} \overset{\frown}{\boldsymbol{v}}^{\mathrm{T}} \overset{\sim}{\overset{\frown}{\omega}} dr \\ \int_{t-\Delta t}^{t} \overset{\sim}{\overset{\frown}{\omega}} \overset{\frown}{\boldsymbol{v}} dr & \int_{t-\Delta t}^{t} \overset{\sim}{\overset{\frown}{\omega}} \overset{\sim}{\overset{\frown}{\omega}} dr \end{bmatrix} \tag{55}$$

$$\boldsymbol{B} = \begin{bmatrix} \int_{t-\Delta t}^{t} \overset{\frown}{\boldsymbol{v}}^{\mathrm{T}} \left(\boldsymbol{v}_{\mathrm{e}} - \boldsymbol{v}_{\mathrm{enom}}\right) dr \\ \int_{t-\Delta t}^{t} \overset{\sim}{\overset{\frown}{\omega}} \left(\boldsymbol{v}_{\mathrm{e}} - \boldsymbol{v}_{\mathrm{enom}}\right) dr \end{bmatrix} \tag{56}$$

According to Equation (54), the parameter error vector can be determined by:

$$\boldsymbol{X} = \boldsymbol{A}^{-1}\boldsymbol{B} \tag{57}$$

After all, the estimated mass properties $\hat{\gamma}$ and $\hat{\boldsymbol{b}}_0$ can be computed according to Equations (41), (53) and (57).

Note that, the estimated mass properties are initialized by the nominal mass and centroid position in this work. After that, the temporarily identified mass properties must be updated in real time for more accurate estimations. In addition, the length of the integral interval has to be carefully designed according to the specified system properties and the practical mission requirements because the condition number of matrix $\boldsymbol{A}$ can be very large, even singular when Δt is too small. On the other hand, Δt cannot be too large either. Although the velocities as $\overset{\frown}{v}$ and $\overset{\frown}{\omega}$ are quite simple in calculation, longer integral intervals still require greater computations which are against the on-line application. Because the tuning process is based on the feedback of eye-in-hand camera images, which are usually quite noisy, the identification error is investigated to indicate the impacts of the measurement errors. The upper bound estimation of the identification error is given in the Appendix A of this paper.

4. Ground Testing Based on Hardware-in-the-Loop Simulation System

Generally, space robot target capture methods are tested with ground test systems [34–36]. Thus a hardware-in-the-loop simulation system, illustrated by Figure 4, is employed for simulating the satellite capture process. Two industrial robots are used to simulate the space robot motions and the target satellite respectively: industrial robot A represents the space robot motions and the target satellite is mounted on industrial robot B.

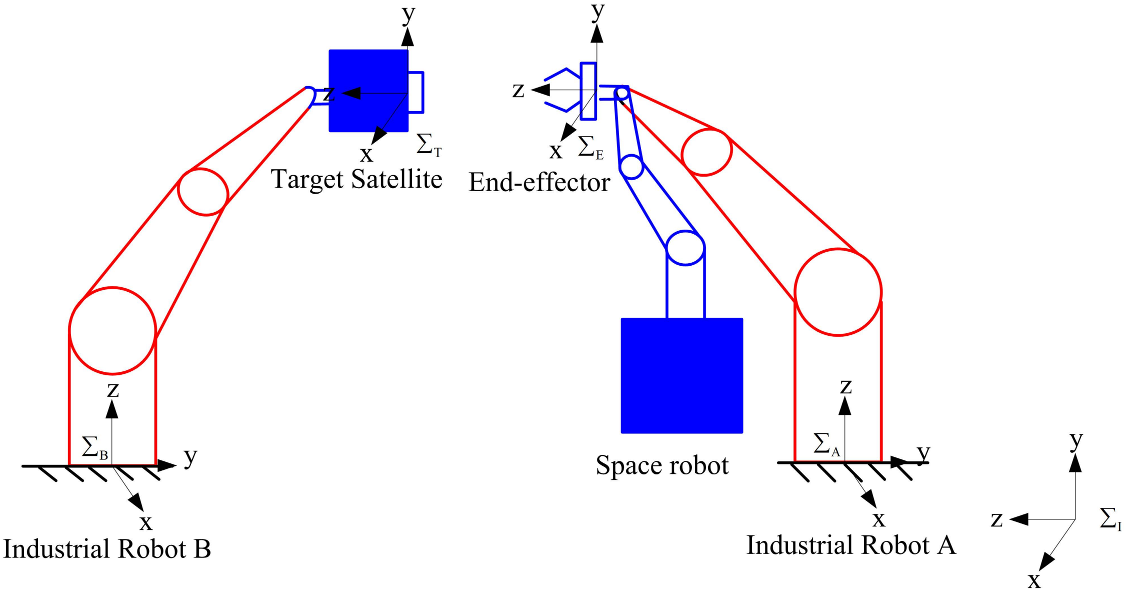

Figure 4. Kinematic equivalence diagram of hard-ware-in-the-loop simulation system.

To mimic the weightless condition in space, which are quite difficult to achieve in a ground environment, a space robot dynamics simulation program is established for calculating the space robot motions under microgravity. Since a carrier satellite was not available, gyro data is also output by this program. Moreover, a joint electronic simulator employing the same electronic interface with real space robot joints is used to simulate the robotic joint dynamics since the actual space robot is missing too. The eye-in-hand camera system and the space robot central controller are both real.

The space robot hardware-in-the-loop simulation system consists of two industrial robots, the space robot motion controller, the inverse kinematics calculator, an electronic simulator, the space robot dynamics simulation system, the kinematic equivalence module, the eye-in-hand camera system and the mass property estimator. The space robot motion controller, inverse kinematics calculations and the proposed mass property estimation algorithm are all realized by the space robot central controller. The block diagram of designed experimental system is shown in Figure 5.

In the satellite capturing experimental system, the needed linear and angular velocities are calculated through the relative position and attitude measured by the vision system. Meanwhile, the mass property estimator determines the unknown parameters from the past joint trajectories and sensor information in real time. Then, the joint motions are planned based on the estimations and the desired end-effector motion. According to the joint motions, the electronic simulator determines the output torques of the joints. Then the motions of the space robot are simulated by the dynamic simulation system. Finally, the relative motions between the space robot and the target satellite are demonstrated by the industrial robots based on kinematic equivalence.

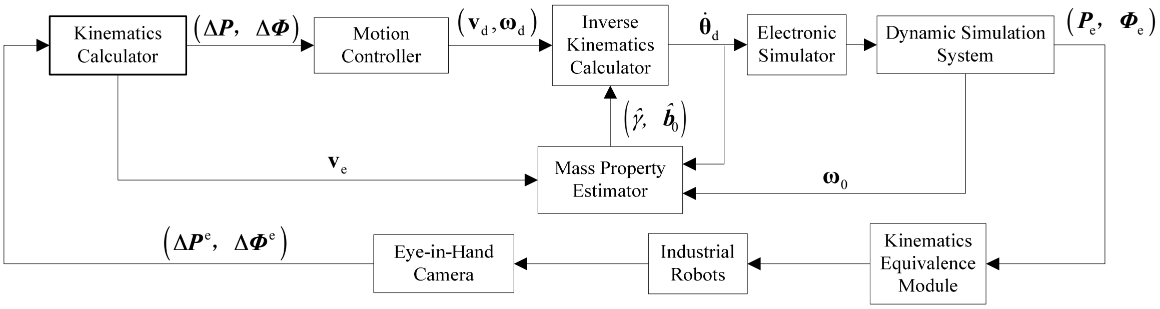
Kinematics Calculator
$(\Delta \boldsymbol{P}, \Delta \boldsymbol{\Phi})$
Motion Controller
$(\mathbf{v}_d, \boldsymbol{\omega}_d)$
Inverse Kinematics Calculator
$\dot{\boldsymbol{\theta}}_d$
Electronic Simulator
Dynamic Simulation System
$(\boldsymbol{P}_e, \boldsymbol{\Phi}_e)$
$(\hat{\gamma}, \hat{\boldsymbol{b}}_0)$
$\mathbf{v}_e$
Mass Property Estimator
$\boldsymbol{\omega}_0$
$(\Delta \boldsymbol{P}^e, \Delta \boldsymbol{\Phi}^e)$
Eye-in-Hand Camera
Industrial Robots
Kinematics Equivalence Module

Figure 5. Block diagram of space robot ground test system.

The instruction cycle of the designed space robot central controller is 250 ms. The joint electronic simulator's control cycle is 25 ms. The required measurement accuracies of the eye-in-hand camera system are 1 mm in distance and 1 deg in orientation. The measurement frequency is 4 Hz. A picture of the laboratory setup is shown in Figure 6. Space robot kinematic and dynamic parameters are expressed as Table 1.

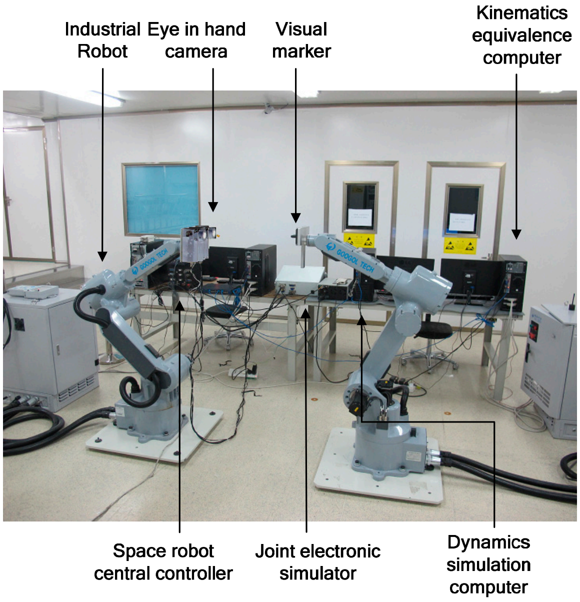
Industrial Robot
Eye in hand camera
Visual marker
Kinematics equivalence computer
Space robot central controller
Joint electronic simulator
Dynamics simulation computer

Figure 6. Laboratory with the space robot ground experimental system.

Table 1. Kinematic and dynamic parameters of space robot.

Parameter (unit)	Base	Pole 1	Pole 2	Pole 3	Pole 4	Pole 5	Pole 6
M (kg)	648	1.5	9.6	1.5	9.0	1.5	10.5
a_x (mm)	0	0	−493.5	0	289	0	−112
a_y (mm)	0	0	56	−123	−123	127	0
a_z (mm)	0	120	0	0	0	0	0
b_x (mm)	539	0	−493.5	123	333	−123	−123
b_y (mm)	5	189	−56	0	12	0	0
b_z (mm)	813	0	0	0	0	0	0
I_{xx} (kg·m^2)	198	3.12×10^{-3}	3.71×10^{-2}	3.47×10^{-3}	0.82	3.42×10^{-3}	0.91
I_{yy} (kg·m^2)	198	3.12×10^{-3}	1.92	3.47×10^{-3}	0.64	3.42×10^{-3}	0.91
I_{zz} (kg·m^2)	198	3.12×10^{-3}	1.92	3.47×10^{-3}	0.67	3.42×10^{-3}	0.11

In this test, the space robot is required to move the end-effector from the initial position to the satellite with suitable posture. The nominal mass properties of carrier satellite are defined as:

$$\begin{cases} m_{0nom} = 680(\text{kg}) \\ \boldsymbol{b}_{0nom} = \begin{bmatrix} 500 & 0 & 800 \end{bmatrix}^{\text{T}} (\text{mm}) \end{cases} \tag{58}$$

Although the satellite mass is not necessary in the self-tuning control scheme, it is computed by $\hat{\gamma}$ for verifying the estimator. According to Equations (4) and (7), the estimated satellite mass can be calculated as:

$$\hat{m}_0 = -\frac{1}{\hat{\gamma}} - M_{\text{m}} \tag{59}$$

Accordingly, the property estimation errors are computed as follows:

$$\begin{cases} \delta m_0 = \hat{m}_0 - m_0 \\ \delta \boldsymbol{b}_0 = \hat{\boldsymbol{b}}_0 - \boldsymbol{b}_0 \end{cases} \tag{60}$$

To validate the proposed control scheme, both the closed-loop responses and the parameter estimation errors are tested by the ground experimental system. The experimental results are shown in Figures 7 and 8.

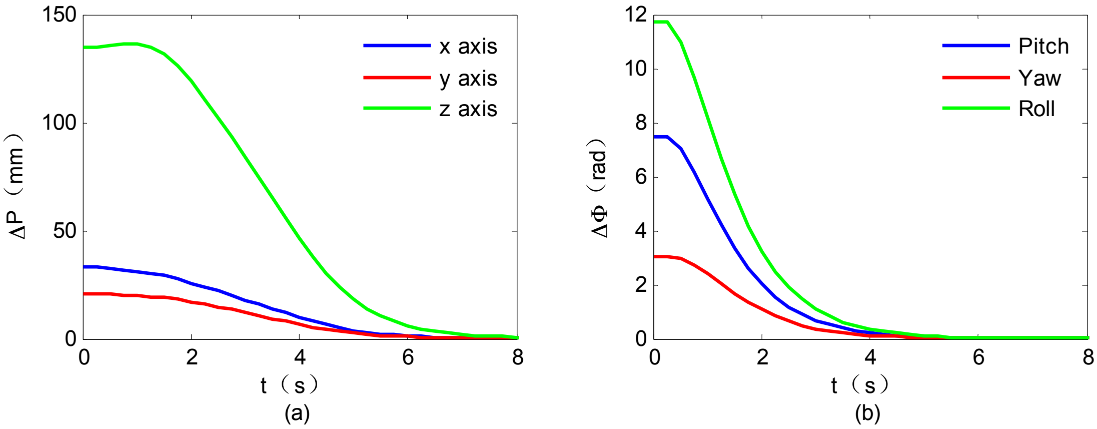

Figure 7. Closed-loop responses of space robot system: (**a**) Position errors; (**b**) Angle errors.

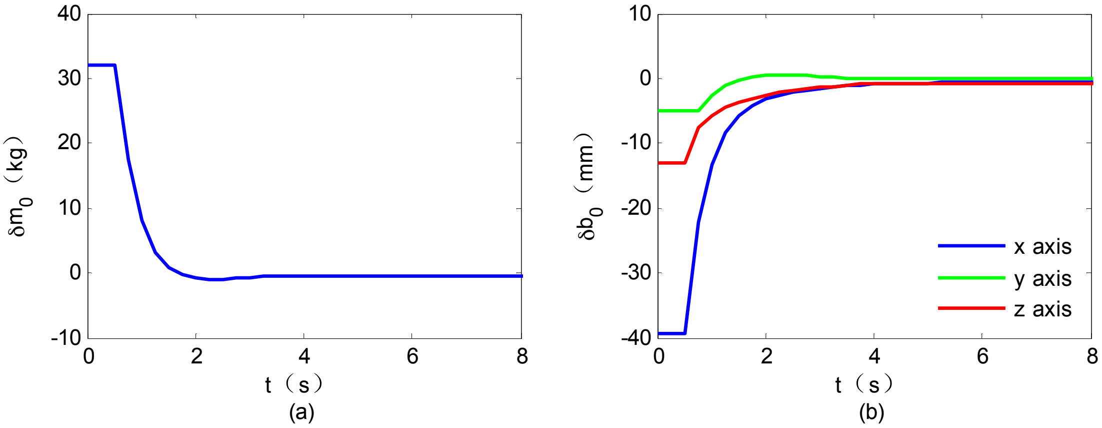

Figure 8. Property estimation errors: (**a**) estimation error of m_0; (**b**) estimation error of $\boldsymbol{b}_0$.

The closed-loop responses are shown in Figure 7. It is seen that, the space robot end-effector can approach the target satellite smoothly with the desired attitude, illustrating the effectiveness of the proposed target capture control method. The mass property estimation errors are shown in Figure 8. They suggest that although the nominal mass properties are initialized with obvious errors, the estimated mass properties gradually approach the real values.

As the first attempt for estimating the satellite mass properties by eye-in-hand camera signals, the following conclusions can be made by comparing the proposed method with other existing identification methods:

(1) Compared with the propulsion-based methods as presented in [37], because the thrusters are not applied in the proposed method, no satellite fuel will be consumed.

(2) Unlike the direct torque-sensing method proposed in [38], this proposed method doesn't demand any torque or acceleration measurements, not only in theory but also in engineering.

(3) Compared with the method based on measuring the reaction wheel motion rates presented in [32], which has difficulty estimating the satellite mass, the proposed method identifies both satellite mass and centroid position by adopting eye-in-hand camera signals and gyro information.

(4) Compared with the method presented in [33], which estimates the satellite mass properties base on sensing the satellite rotation and translation, this proposed method doesn't require the linear velocity of the carrier satellite, which is usually integrated from accelerometer data and brings drifting errors.

5. Conclusions

Target satellite capture is a challenging problem, especially for space robots with unknown mass properties. Since most existing works for space robot motion control require accurate property values, new efforts are being made for handling such control problems including unknown parameters. In this paper, gyro and camera signals are adopted to improve the control performance. For this improved system, a novel space robot modelling technique is proposed. By this newly established model, the free-floating space robot is equivalent to a ground-fixed manipulator system, thus simplifying the issue. Accordingly, a self-tuning target capturing controller is designed taking unknown parameters into count. The control parameters are determined in real time by the estimator established based on the least-squares technique. The experimental results suggest that the designed space robot target capturing controller is effective. Because the proposed method does not demand accurate satellite mass properties, it can be applied when the fuel consumption is unknown or carrying an undetermined payload, etc. As further research, the proposed method has a potential to be applied in identifying other space robot parameters, such as manipulator link lengths, to cope with contingent requirements.

Acknowledgments: This work is supported by a grant from the National Program on Key Research Program (No. 2013CB733105).

Author Contributions: H.L. raised the challenge target capturing control problem of free-floating space robot with unknown mass properties based on project needs; Z.L. designed the space robot modeling technique and the self-tuning control scheme including the mass property estimation algorithm; Z.L. and B.W. conceived and designed the experiments; Z.L. and B.W. analyzed the data; Z.L. contributed the software tools; Z.L. wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

The measured velocity data is expressed as:

$$\begin{cases} \overline{\boldsymbol{v}}_{\mathrm{e}} = \boldsymbol{v}_{\mathrm{e}} + \Delta\boldsymbol{v}_{\mathrm{e}} \\ \overline{\omega}_0 = \omega_0 + \Delta\omega_0 \end{cases} \tag{A1}$$

where $\Delta\boldsymbol{v}_{\mathrm{e}}$, as well as $\Delta\omega_0$, is the measurement error. According to Equations (43) and (46), the defined velocities are computed as:

$$\begin{cases} \overline{\overset{\frown}{\boldsymbol{v}}} = \boldsymbol{v}_{\mathrm{ec}} - \overset{\frown}{\boldsymbol{L}}_{0\mathrm{e}}\left(\hat{\boldsymbol{b}}_{0\mathrm{tem}}\right) \times \overline{\omega}_0 \\ \overline{\overset{\frown}{\omega}} = \left[1 + M_{\mathrm{m}}\hat{\gamma}_{\mathrm{tem}}\right]\overline{\omega}_0 \end{cases} \tag{A2}$$

According to Equation (54), the identification equation is determined as:

$$\overline{\boldsymbol{A}}\,\overline{\boldsymbol{X}} = \overline{\boldsymbol{B}} \tag{A3}$$

where:

$$\overline{\boldsymbol{A}} = \boldsymbol{A} + \Delta\boldsymbol{A} = \begin{bmatrix} \int_{t-\Delta t}^{t} \overline{\overset{\frown}{\boldsymbol{v}}} \cdot \overline{\overset{\frown}{\boldsymbol{v}}}\,dr & -\int_{t-\Delta t}^{t} \left(\overline{\overset{\frown}{\omega}} \times \overline{\overset{\frown}{\boldsymbol{v}}}\right)^{\mathrm{T}} dr \\ \int_{t-\Delta t}^{t} \overline{\overset{\frown}{\omega}} \times \overline{\overset{\frown}{\boldsymbol{v}}}\,dr & \int_{t-\Delta t}^{t} \overline{\overset{\frown}{\omega}} \times \left(\overline{\overset{\frown}{\omega}} \times\right) dr \end{bmatrix} \tag{A4}$$

$$\overline{\boldsymbol{B}} = \boldsymbol{B} + \Delta\boldsymbol{B} = \begin{bmatrix} \int_{t-\Delta t}^{t} \overline{\overset{\frown}{\boldsymbol{v}}} \cdot (\overline{\boldsymbol{v}}_{\mathrm{e}} - \boldsymbol{v}_{\mathrm{enom}})dr \\ \int_{t-\Delta t}^{t} \overline{\overset{\frown}{\omega}} \times (\overline{\boldsymbol{v}}_{\mathrm{e}} - \boldsymbol{v}_{\mathrm{enom}})dr \end{bmatrix} \tag{A5}$$

and $\overline{\boldsymbol{X}}$ is the identification result with the error $\delta\boldsymbol{X}$; $\Delta\boldsymbol{A}$ and $\Delta\boldsymbol{B}$ are the errors of $\boldsymbol{A}$ and $\boldsymbol{B}$ caused by the measurement errors $\Delta\boldsymbol{v}_{\mathrm{e}}$ and $\Delta\omega_0$. The error matrixes $\Delta\boldsymbol{A}$ and $\Delta\boldsymbol{B}$ is expressed as follows:

$$\begin{cases} \Delta\boldsymbol{A} = \varepsilon_{\mathrm{A}}\boldsymbol{E}_{\mathrm{A}} \\ \Delta\boldsymbol{B} = \varepsilon_{\mathrm{B}}\boldsymbol{E}_{\mathrm{B}} \end{cases} \tag{A6}$$

where:

$$\begin{cases} \varepsilon_{\mathrm{A}} = ||\Delta\boldsymbol{A}|| \\ \varepsilon_{\mathrm{B}} = ||\Delta\boldsymbol{B}|| \\ ||\boldsymbol{E}_{\mathrm{A}}|| = ||\boldsymbol{E}_{\mathrm{B}}|| = 1 \end{cases} \tag{A7}$$

In accordance with the theory presented in [33], the upper bound of the error for the identification problem represented by Equation (A3) can be estimated by:

$$||\delta\boldsymbol{X}|| \leq ||\boldsymbol{A}^{-1}||^2 ||\boldsymbol{r}||\varepsilon_{\mathrm{A}} + ||\boldsymbol{A}^{-1}||\left(||\boldsymbol{X}||\varepsilon_{\mathrm{A}} + \varepsilon_{\mathrm{B}}\right) + N_{1\varepsilon} \tag{A8}$$

and the upper bound of the relative error in the identified parameters is:

$$\frac{||\delta\boldsymbol{X}||}{||\boldsymbol{X}||} \leq \left(\kappa\left(\boldsymbol{A}\right)\right)^2 \frac{\varepsilon_{\mathrm{A}}}{||\boldsymbol{A}||}\zeta + \kappa\left(\boldsymbol{A}\right)\left(\frac{\varepsilon_{\mathrm{A}}}{||\boldsymbol{A}||} + \frac{\varepsilon_{\mathrm{B}}}{||\boldsymbol{B}||}\gamma\right) + N_{2\varepsilon} \tag{A9}$$

where $\boldsymbol{r}$ is the residual vector; κ is the condition number; $N_{1\varepsilon}$, as well as $N_{2\varepsilon}$, represents the second- and higher-order terms of ε, which can be practically ignored. They and others are computed as follows:

$$\begin{cases} \boldsymbol{r} = \boldsymbol{B} - \boldsymbol{A}\boldsymbol{X} \\ \kappa\left(\boldsymbol{A}\right) = \|\boldsymbol{A}^{-1}\|\,\|\boldsymbol{A}\| \\ \zeta = \frac{\|\boldsymbol{r}\|}{\|\boldsymbol{A}\|\,\|\boldsymbol{X}\|} \\ \gamma = \frac{\|\boldsymbol{B}\|}{\|\boldsymbol{A}\|\,\|\boldsymbol{X}\|} \end{cases} \tag{A10}$$

To determine the error bounds, $\Delta\boldsymbol{A}$ as well as $\Delta\boldsymbol{B}$ should be computed in terms of the measurement errors $\Delta\boldsymbol{v}_{\mathrm{e}}$ and $\Delta\omega_0$. Substituting Equations (A1) and (A2) into Equations (A4) and (A5), $\Delta\boldsymbol{A}$ and $\Delta\boldsymbol{B}$ are expressed as:

$$\Delta\boldsymbol{A} = \begin{bmatrix} \int_{t-\Delta t}^{t} \Delta\overparen{\boldsymbol{v}} \cdot \left(2\overline{\overparen{\boldsymbol{v}}} - \Delta\overparen{\boldsymbol{v}}\right) dr & -\int_{t-\Delta t}^{t} \left(\Delta\overparen{\omega} \times \overline{\overparen{\boldsymbol{v}}} + \overline{\overparen{\omega}} \times \Delta\overparen{\boldsymbol{v}} - \Delta\overparen{\omega} \times \Delta\overparen{\boldsymbol{v}}\right)^{\mathrm{T}} dr \\ \int_{t-\Delta t}^{t} \left(\Delta\overparen{\omega} \times \overline{\overparen{\boldsymbol{v}}} + \overline{\overparen{\omega}} \times \Delta\overparen{\boldsymbol{v}} - \Delta\overparen{\omega} \times \Delta\overparen{\boldsymbol{v}}\right) dr & \int_{t-\Delta t}^{t} \left(\overline{\overparen{\omega}} \times \Delta\overparen{\omega} \times + \Delta\overparen{\omega} \times \overline{\overparen{\omega}} \times - \Delta\overparen{\omega} \times \Delta\overparen{\omega} \times\right) dr \end{bmatrix} \tag{A11}$$

$$\Delta\boldsymbol{B} = \begin{bmatrix} \int_{t-\Delta t}^{t} \left(\overline{\overparen{\boldsymbol{v}}} \cdot \Delta\boldsymbol{v}_{\mathrm{e}} + \Delta\overparen{\boldsymbol{v}} \cdot \delta\boldsymbol{v}_{\mathrm{e}} - \Delta\overparen{\boldsymbol{v}} \cdot \Delta\boldsymbol{v}_{\mathrm{e}}\right) dr \\ \int_{t-\Delta t}^{t} \left(\overline{\overparen{\omega}} \times \Delta\boldsymbol{v}_{\mathrm{e}} + \Delta\overparen{\omega} \times \delta\boldsymbol{v}_{\mathrm{e}} - \Delta\overparen{\omega} \times \Delta\boldsymbol{v}_{\mathrm{e}}\right) dr \end{bmatrix} \tag{A12}$$

where:

$$\begin{cases} \Delta\overparen{\boldsymbol{v}} = \overline{\overparen{\boldsymbol{v}}} - \overparen{\boldsymbol{v}} = -\overparen{\boldsymbol{L}}_{0\mathrm{e}} \times \Delta\omega_0 \\ \Delta\overparen{\omega} = \overline{\overparen{\omega}} - \overparen{\omega} = \left[1 + M_{\mathrm{m}}\hat{\gamma}_{\mathrm{tem}}\right] \Delta\omega_0 \\ \delta\boldsymbol{v}_{\mathrm{e}} = \overline{\boldsymbol{v}}_{\mathrm{e}} - \boldsymbol{v}_{\mathrm{enom}} \end{cases} \tag{A13}$$

Ignoring the second-order terms, $\Delta\boldsymbol{A}$ and $\Delta\boldsymbol{B}$ are rewritten as functions of $\Delta\omega_0$ and $\Delta\boldsymbol{v}_{\mathrm{e}}$:

$$\Delta\boldsymbol{A} = \begin{bmatrix} -\int_{t-\Delta t}^{t} \left(2\overline{\overparen{\boldsymbol{v}}}^{\mathrm{T}} \widetilde{\overparen{\boldsymbol{L}}}_{0\mathrm{e}}\right) \Delta\omega_0 dr & \int_{t-\Delta t}^{t} \left\{\left[\left(1 + M_{\mathrm{m}}\hat{\gamma}_{\mathrm{tem}}\right) \widetilde{\overline{\overparen{\boldsymbol{v}}}} + \widetilde{\overline{\overparen{\omega}}}\,\widetilde{\overparen{\boldsymbol{L}}}_{0\mathrm{e}}\right] \Delta\omega_0\right\}^{\mathrm{T}} dr \\ -\int_{t-\Delta t}^{t} \left[\left(1 + M_{\mathrm{m}}\hat{\gamma}_{\mathrm{tem}}\right) \widetilde{\overline{\overparen{\boldsymbol{v}}}} + \widetilde{\overline{\overparen{\omega}}}\,\widetilde{\overparen{\boldsymbol{L}}}_{0\mathrm{e}}\right] \Delta\omega_0 dr & \left(1 + M_{\mathrm{m}}\hat{\gamma}_{\mathrm{tem}}\right) \int_{t-\Delta t}^{t} \left[\widetilde{\overline{\overparen{\omega}}} \Delta\tilde{\omega}_0 + \left(\widetilde{\overline{\overparen{\omega}}} \Delta\tilde{\omega}_0\right)^{\mathrm{T}}\right] dr \end{bmatrix} \tag{A14}$$

$$\Delta\boldsymbol{B} = \begin{bmatrix} \int_{t-\Delta t}^{t} \overline{\overparen{\boldsymbol{v}}}^{\mathrm{T}} \Delta\boldsymbol{v}_{\mathrm{e}} - \delta\boldsymbol{v}_{\mathrm{e}}^{\mathrm{T}} \widetilde{\overparen{\boldsymbol{L}}}_{0\mathrm{e}} \Delta\omega_0 dr \\ \int_{t-\Delta t}^{t} \widetilde{\overline{\overparen{\omega}}} \Delta\boldsymbol{v}_{\mathrm{e}} - \left[1 + M_{\mathrm{m}}\hat{\gamma}_{\mathrm{tem}}\right] \delta\tilde{\boldsymbol{v}}_{\mathrm{e}} \Delta\omega_0 dr \end{bmatrix} \tag{A15}$$

References

1. Yoon, W.; Goshozono, T.; Kawabe, H.; Kinami, M.; Tsumaki, Y.; Uchiyama, M.; Oda, M.; Doi, T. Model based space robot teleoperation of ETS-VII manipulator. *IEEE Trans. Robot. Autom.* **2004**, *20*, 602–612. [CrossRef]
2. Yoshida, K.; Hasizume, K.; Nenchev, D.N.; Inaba, N.; Oda, M. Control of a space manipulator for autonomous target capture-ETS-VII flight experiments and analysis. *Guid. Navig. Control* **2000**. [CrossRef]
3. Yoshida, K.; Hashizume, K.; Abiko, S. Zero reaction maneuver: Flight validation with ETS-VII space robot and extension to kinematically redundant arm. In Proceedings of the IEEE International Conference on Robotics & Automation, Seoul, Korea, 21–26 May 2001; IEEE Computer Society: Washington, DC, USA.
4. Motaghedi, P. On-orbit performance of the Orbital Express Capture System. In Proceedings of the SPIE—The International Society for Optical Engineering, Orlando, FL, USA, 15 April 2008; SPIE: Bellingham, WA, USA.
5. Ogilvie, A.; Allport, J.; Hannah, M.; Lymer, J. Autonomous satellite servicing using the Orbital Express demonstration manipulator system. In Proceedings of the 9th International Symposium on Artificial Intelligence, Robotics and Automation in Space, Hollywood, SC, USA, 25–29 February 2008; IEEE Computer Society: Washington, DC, USA.

6. Pinson, R.; Howard, R.; Heaton, A. Orbital Express Advanced Video Guidance Sensor: Ground Testing, Flight Results and Comparisons. In Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit, Honolulu, HI, USA, 18–21 August 2008; AIAA: Reston, VA, USA.
7. Jin, M.; Yang, H.; Xie, Z.; Sun, K.; Liu, H. The Ground-based Verification System of Visual Servoing Control for a Space Robot. In Proceedings of the IEEE International Conference on Mechatronics and Automation, Takamatsu, Japan, 4–7 August 2013; IEEE Computer Society: Washington, DC, USA.
8. Liang, B.; Li, C.; Xue, L.; Qiang, W. A Chinese Small Intelligent Space Robotic System for On-Orbit Servicing. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, China, 9–15 October 2006; IEEE Computer Society: Washington, DC, USA.
9. Dubowsky, S.; Papadopoulos, E. The kinematics, dynamics and control of free-flying and free-floating space robotics system. *IEEE Trans. Robot. Autom.* **1993**, *9*, 531–543. [CrossRef]
10. Oda, M.; Ohkami, Y. Coordinated control of spacecraft attitude and space manipulators. *Control Eng. Pract.* **1997**, *5*, 11–21. [CrossRef]
11. Umetani, Y.; Yoshida, K. Resolved motion rate control of space manipulators with generalized jacobian matrix. *IEEE Trans. Robot. Autom.* **1989**, *5*, 303–314. [CrossRef]
12. Vafa, Z.; Dubowsky, S. The kinematics and dynamics of Space manipulators: The virtual manipulator approach. *Int. J. Robot. Res.* **1990**, *9*, 3–21. [CrossRef]
13. Papadopoulos, E.; Dubowsky, S. On the nature of control algorithms for free-floating space manipulators. *IEEE Trans. Robot. Autom.* **1991**, *7*, 750–758. [CrossRef]
14. Fang, Y.; Zhang, W.; Ye, X. Variable Structure Control for Space Robots Based on Neural Networks. *Int. J. Adv. Robot. Syst.* **2014**, *11*. [CrossRef]
15. Huang, P.; Xu, X.; Meng, Z. Optimal Trajectory Planning and Coordinated Tracking Control Method of Tethered Space Robot Based on Velocity Impulse. *Int. J. Adv. Robot. Syst.* **2014**, *11*. [CrossRef]
16. Meng, Z.; Huang, P. An Effective Approach Control Scheme for the Tethered Space Robot System. *Int. J. Adv. Robot. Syst.* **2014**, *11*. [CrossRef]
17. Wang, D.; Huang, P.; Cai, J.; Meng, Z. Coordinated control of tethered space robot using mobile tether attachment point in approaching phase. *Adv. Space Res.* **2014**, *54*, 1077–1091. [CrossRef]
18. Huang, P.; Wang, D.; Meng, Z.; Zhang, F.; Liu, Z. Impact dynamic modelling and adaptive target capturing control for tethered space robots with uncertainties. *IEEE/ASME Trans. Mechatron.* **2016**, *21*, 2260–2271. [CrossRef]
19. Huang, P.; Wang, D.; Meng, Z.; Zhang, F. Adaptive postcapture backstepping control for tumbling tethered space robot–target combination. *J. Guid. Control Dyn.* **2016**, *39*, 150–155. [CrossRef]
20. Huang, P.; Wang, M.; Meng, Z.; Zhang, F.; Liu, Z.; Chang, H. Reconfigurable spacecraft attitude takeover control in post-capture of target by space manipulators. *J. Frankl. Inst.* **2016**, *353*, 1985–2008. [CrossRef]
21. Pathak, P.M.; Kumar, R.P.; Mukherjee, A.; Dasgupta, A. A scheme for robust trajectory control of space robots. *Simul. Model. Pract. Theory* **2008**, *16*, 1337–1349. [CrossRef]
22. Tsuchiya, K. Breakwell Memorial Lecture: Attitude dynamics of satellite—From spinning satellite to space robot. *Acta Astronaut.* **2008**, *62*, 131–139. [CrossRef]
23. Rastegari, R.; Moosavian, S.A.A. Multiple impedance control of space free-flying robots via virtual linkages. *Acta Astronaut.* **2010**, *66*, 748–759. [CrossRef]
24. Zarafshan, P.; Moosavian, S.A.A. Dynamics modeling and Hybrid Suppression Control of space robots performing cooperative object manipulation. *Commun. Nonlinear Sci. Numer. Simulat.* **2013**, *18*, 2807–2824. [CrossRef]
25. Hutchinson, S.; Hager, G.D.; Corke, P.I. A tutorial on visual servo control. *IEEE Trans. Robot. Autom.* **1996**, *12*, 651–670. [CrossRef]
26. Chaumette, F.; Hutchinson, S. Visual servo control. I. Basic approaches. *IEEE Robot. Autom. Mag.* **2006**, *13*, 82–90. [CrossRef]
27. Chaumette, F.; Hutchinson, S. Visual servo control. II. Advanced approaches. *IEEE Robot. Autom. Mag.* **2007**, *14*, 109–118. [CrossRef]
28. Rouleau, G.; Rekleitis, I.; L'Archeveque, R.; Martin, E. Autonomous capture of a tumbling satellite. *J. Field Robot.* **2007**, *24*, 275–296.
29. Sabatini, M.; Monti, R.; Gasbarri, P.; Palmerini, G. Adaptive and robust algorithms and tests for visual-based navigation of a space robotic manipulator. *Acta Astronaut.* **2013**, *83*, 65–84. [CrossRef]

30. Sabatini, M.; Monti, R.; Gasbarri, P.; Palmerini, G. Deployable space manipulator commanded by means of visual-based guidance and navigation. *Acta Astronaut.* **2013**, *83*, 27–43. [CrossRef]
31. Dong, G.; Zhu, Z.H. Position-based visual servo control of autonomous robotic manipulators. *Acta Astronaut.* **2015**, *115*, 291–302. [CrossRef]
32. Yoshida, K.; Abiko, S. Inertia Paramter Identification for a Free-flying Space Robot. *AIAA Guid. Navig. Control Conf. Exhib.* **2002**, *38*, 1–8.
33. Ma, O.; Dang, H.; Pham, K. On-Orbit Identification of Inertia Properties of Spacecraft Using a Robotic Arm. *J. Guid. Control Dyn.* **2008**, *31*, 1761–1771. [CrossRef]
34. Ma, O.; Wang, J.; Misra, S.; Liu, M. On the validation of SPDM task verification facility. *J. Robot. Syst.* **2004**, *21*, 219–235. [CrossRef]
35. Xu, W.; Liu, Y.; Liang, B.; Xu, Y.; Qiang, W. Autonomous Path Planning and Experiment Study of Free-floating Space Robot for Target Capturing. *J. Intell. Robot. Syst.* **2008**, *51*, 303–331. [CrossRef]
36. Boge, T.; Ma, O. Using Advanced Industrial Robotics for Spacecraft Rendezvous and Docking Simulation. In Proceedings of the IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; IEEE Computer Society: Washington, DC, USA.
37. Rackl, W.; Lampariello, R.; Albu-Schäffer, A. Parameter identification methods for free-floating space robots with direct torque sensing. In Proceedings of the IFAC Symposium on Automatic Control in Aerospace, Würzburg, Germany, 2–6 September 2013; Elsevier: Amsterdam, The Netherlands.
38. Xu, W.; Hu, Z.; Zhang, Y.; Wang, Z.; Wu, X. A practical and effective method for identifying the complete inertia parameters of space robots. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Hamburg, Germany, 28 September–2 October 2015; IEEE Computer Society: Washington, DC, USA.