

Research Article

Pose Determination for Malfunctioned Satellites Based on Depth Information

Feng Yu , Yi Zhao, and Yanhua Zhang

College of Astronautics, Nanjing University of Aeronautics and Astronautics, 210016, China

Correspondence should be addressed to Feng Yu; yufeng@nuaa.edu.cn

Received 9 November 2018; Revised 22 February 2019; Accepted 1 April 2019; Published 11 June 2019

Academic Editor: Enrico C. Lorenzini

Copyright © 2019 Feng Yu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Autonomous on-orbit servicing is the future space activity which can be utilized to extend the satellite life. Relative pose estimation for a malfunctioned satellite is one of the key technologies to achieve robotic on-orbit servicing. In this paper, a relative pose determination method by using point cloud is presented for the final phase of the rendezvous and docking of malfunctioned satellites. The method consists of three parts: (1) planes are extracted from point cloud by utilizing the random sample consensus algorithm. (2) The eigenvector matrix and the diagonal eigenvalue matrix are calculated by decomposing the point cloud distribution matrix of the extracted plane. The eigenvalues are utilized to recognize rectangular planes, and the eigenvector matrix is the attitude rotation matrix from the sensor to the plane. The solution of multisolution problem is also presented. (3) An extended Kalman filter is designed to estimate the translational states, the rotational states, the location of mass center, and the moment-of-inertia ratios. Because the method only utilizes the local features without observing the whole satellite, it is suitable for the final phase of rendezvous and docking. The algorithm is validated by a series of mathematical simulations.

1. Introduction

The on-orbit servicing technology, which is a very challenging research topic, is commonly regarded to be utilized to extend the life of malfunctioned satellites. Although several on-orbit servicing demonstration cases were performed [1], autonomous on-orbit servicing operation under realistic conditions has not been carried out in space up to now. A series of technologies including autonomous navigation, control, and robotic arm are still immature and should be further developed.

Because the relative pose measurement for a malfunctioned satellite is the precondition of on-orbit servicing [2], many researchers have made many efforts in order to solve this important problem. The relative navigation of TECSAS/DEOS, being studied at the center for robotics and machinery at the German Aerospace Center (DLR), is realized by a stereo vision system. In the automatic mode, a number of images are taken and dumped to ground segment and then are processed to estimate the relative position and the target motion [3]. The Argon system utilizes two different view-size

cameras and a flash LIDAR. Complex image processing techniques are developed to measure the relative position and relative attitude of noncooperative targets [4, 5]. Volpe et al. proposed an on-orbit relative pose and shape estimation with the use of a monocular camera and a distance sensor [6, 7]. In China, Caixiu proposed a method by using a single camera to measure the relative pose by utilizing a solar panel triangle structure and a circular docking ring [8]. Miao et al. utilized solar panel components as the identification objects for non-cooperative spacecraft pose determination [9]. Zhang et al. presented the relative pose measurement algorithm by using a circle and a noncoplanar feature point [10]. Huang et al. proposed a monocular visual servoing control method using only the edge lines of satellite brackets [11]. According to the above research, visual navigation is the main relative pose determination method in the relative state estimation for malfunctioned satellites [12]. However, the visual system sometimes suffers poor imaging quality in the space due to the special optical environment, which leads to the difficulty in subsequent processing. What is worse, the surface of the malfunctioned satellite is often covered by multilayer

insulation (MLI) materials for the thermal protection. Because of this, optical features such as texture change according to the direction of lighting and view; the image features may not be obviously identified [13].

Therefore, other viable image processing method or sensors were utilized by some researchers. Terui et al. used binocular stereo vision to calculate the dense 3D points of target satellite [13], and relative pose was obtained by using Iterative Closest Point (ICP). LIDAR has been employed for full 6DOF pose estimation [14]. Gómez Martínez et al. proposed a pose estimation method for a rocket body in orbit using a time-of-flight (TOF) camera [15]. Hillenbrand and Lampariello realized the motion estimation and model identification of free-floating targets by utilizing range data measurements [16]. The aforementioned literatures intrinsically utilize the target geometrical shape by processing a point cloud of 3D coordinates mapping the sensed target surface. The ICP method is commonly employed to calculate target pose parameters.

However, Lichter and Dubowsky proposed a method to estimate the state, shape, and inertia parameters of noncooperative targets based on a series of distance images [17]. This method includes three parts: firstly, the coarse target pose can be estimated by condensing the range data of the target; secondly, the full dynamic states and inertial parameters are estimated by using Kalman filters; and finally, the shape of the target is estimated by using the estimated pose information and the raw range data. This proposed method is effective only when the whole target is observed. But it is difficult to obtain the complete distance image of the target at the close distance. Therefore, the author advised that a team of range imagers should be distributed uniformly to capture the whole target surface and the relative pose knowledge between sensors mounted on different satellites is required to be known with high accuracy. Literature [15] identified the rocket's geometrical properties including the cylindrical body, nozzle, side tanks, and fairing from the point cloud in the target pose initialization.

Inspired by literatures [15, 17], this paper proposes a new method to estimate the pose of a malfunctioned satellite based on the rectangle feature from point cloud. The complete distance image of the target is not necessary for this method, and only the local rectangular features are extracted to measure the relative position and attitude. Simulation results show the effectiveness of the proposed method.

2. Rectangular Plane Extraction, Recognition, and Pose Determination

2.1. Plane Extraction from Point Cloud. Because satellites are man-made objects, there are some regular components on satellites, such as the main body, solar panels, and docking ring. Although these components are not designed specifically to measure the relative pose of malfunctioned satellites, they have obvious geometric features for recognition and can be utilized to realize relative navigation. The rectangular feature is one of the typical geometric shapes on malfunctioned satellites. For example, the main body of most satellites and solar panels is composed of a variety

of rectangles. Therefore, this paper proposes to select the rectangular planes as the identification object.

The random sample consensus (RANSAC) algorithm is utilized to extract planes from the point cloud dataset. The algorithm can be described as follows: firstly, select a mathematical model; secondly, find the so-called minimum set from the original point cloud sample, and then estimate the parameters of the model by using the minimum set of points; thirdly, check the consistency of the remaining points with the estimated model, and add these qualified points into the point set; and finally, the parameters of the model can be reestimated using all the points that conform to the model.

According to the aforementioned RANSAC algorithm, the specific steps for extracting the plane from the point cloud are proposed as follows:

- (1) Select three points $P_1(x_1, y_1, z_1)$, $P_2(x_2, y_2, z_2)$, and $P_3(x_3, y_3, z_3)$ randomly in the point cloud dataset
- (2) Calculate the parameters of the plane S according to P_1 , P_2 , and P_3
- (3) Suppose the plane thickness as ϵ , and calculate the distance from any point to the plane in the point cloud:

$$d_i = \frac{|Ax_i + By_i + Cz_i + D|}{\sqrt{A^2 + B^2 + C^2}} \quad (1)$$

- (4) Find out how many points are satisfying $d_i < \epsilon$
- (5) Repeat step 1~step 4 K times, and select the plane that includes the most points, where K can be calculated by the following formula:

$$K = \frac{\log(1 - \Phi)}{\log(1 - (1 - \tau)^3)}, \quad (2)$$

where τ is the proportion of the outer point in the point cloud and Φ is the probability that the optimal plane can be found after K calculation

2.2. Rectangular Plane Recognition. Some components of a malfunctioned satellite are composed of rectangular planes. The dimensions of these rectangular planes are known for a specific malfunctioned satellite. After the plane is extracted from the point cloud, the point cloud distribution matrix is utilized to identify if it is a rectangle plane and determine its dimensions.

Before calculating the point cloud distribution matrix, the location of the geometric centroid of the extracted plane can be determined as

$$\mathbf{r}_m = \frac{1}{n} \sum_{i=1}^n \mathbf{r}_i, \quad (3)$$

where n is the point number in the point cloud, \mathbf{r}_i is the position of the n^{th} point in the sensor coordinate system, and \mathbf{r}_m is the centroid location of the extracted plane.

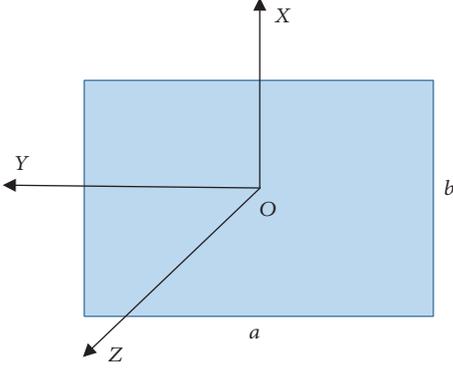


FIGURE 1: The body coordinate system.

In literature [17], the attitude is determined based on the second moment information of point cloud, which is called as geometric moment matrix. However, the point cloud distribution matrix is proposed in this paper, and then, the point cloud distribution matrix is decomposed to calculate the attitude and identify the shape. In essence, both the geometric moment matrix and point cloud distribution matrix reflect the shape information of the point cloud, but the latter requires less processing power. The point cloud distribution matrix is computed as

$$\mathbf{J} = \frac{1}{n} \sum_{i=1}^n \{(\mathbf{r}_i - \mathbf{r}_m) \cdot (\mathbf{r}_i - \mathbf{r}_m)^T\}. \quad (4)$$

The eigenvector matrix and the diagonal eigenvalue matrix can be obtained by utilizing a matrix decomposition operation as

$$\mathbf{J} = \mathbf{R}\mathbf{\Lambda}\mathbf{R}^T, \quad (5)$$

where $\mathbf{\Lambda}$ is the diagonal eigenvalue matrix of \mathbf{J} and \mathbf{R} is the eigenvector matrix.

The eigenvalue matrix reflects the point cloud distribution in the extracted plane body coordinates, which is described in Figure 1. The origin of the coordinate system is the center of the extracted plane; the X-axis, Y-axis and Z-axis are parallel to the principal geometric axes. If the extracted plane is a rectangle, the X-axis and Y-axis are parallel to the short and long sides of the rectangle plane, respectively, and the corresponding side length can be denoted as b and a , respectively. The eigenvalue matrix reflects the dimension parameters of the extracted plane. Therefore, $\mathbf{\Lambda}$ is invariable for a specific plane. That is to say, it does not change with the relative pose.

Similarly, in the body coordinate system, it is obvious that the eigenvalues matrix can be defined as

$$\mathbf{\Lambda} = \frac{1}{n} \sum_{i=1}^n \{(\mathbf{d}_i - \mathbf{d}_m) \cdot (\mathbf{d}_i - \mathbf{d}_m)^T\}, \quad (6)$$

where n is the point number in the point cloud, \mathbf{d}_i is the position of the n^{th} point in the body coordinate system, and \mathbf{d}_m is the origin of the body coordinate system, which also can be defined as

$$\mathbf{d}_m = \frac{1}{n} \sum_{i=1}^n \mathbf{d}_i. \quad (7)$$

It is supposed that the feature points obey a uniform distribution in the feature plane coordinate system and the measurement noise of point cloud is the white Gaussian noise. Therefore, \mathbf{d}_i can be rewritten as

$$\mathbf{d}_i = \bar{\mathbf{d}}_i + \boldsymbol{\varepsilon}_i, \quad (8)$$

where $\bar{\mathbf{d}}_i$ is the true position of the n^{th} point in the body coordinate system and $\boldsymbol{\varepsilon}_i$ is the noise.

Therefore, equation (6) can also be rewritten as

$$\mathbf{\Lambda} = \frac{1}{n} \sum_{i=1}^n \begin{bmatrix} (d_i^x - d_m^x)^2 & (d_i^x - d_m^x)(d_i^y - d_m^y) & (d_i^x - d_m^x)(d_i^z - d_m^z) \\ (d_i^x - d_m^x)(d_i^y - d_m^y) & (d_i^y - d_m^y)^2 & (d_i^y - d_m^y)(d_i^z - d_m^z) \\ (d_i^x - d_m^x)(d_i^z - d_m^z) & (d_i^y - d_m^y)(d_i^z - d_m^z) & (d_i^z - d_m^z)^2 \end{bmatrix}, \quad (9)$$

where (d_i^x, d_i^y, d_i^z) are the three components of \mathbf{d}_i and (d_m^x, d_m^y, d_m^z) are the three components of \mathbf{d}_m .

Suppose $I_y = (1/n) \sum_{i=1}^n (d_i^y - d_m^y)^2$ and then

$$E(I_y) = E \left\{ \frac{1}{n} \sum_{i=1}^n \left((d_i^y)^2 + (d_m^y)^2 - 2 \cdot d_i^y d_m^y \right) \right\}, \quad (10)$$

where $E(I_y)$ is the mathematical expectation of I_y .

Substitute equation (8) into (10), and then, equation (10) can be expressed as

$$E(I_y) = E \left\{ \frac{1}{n} \sum_{i=1}^n (\bar{d}_i^y)^2 \right\} + E \left\{ \frac{1}{n} \sum_{i=1}^n (\varepsilon_i^y)^2 \right\} + 2E \left\{ \frac{1}{n} \sum_{i=1}^n (\bar{d}_i^y \varepsilon_i^y) \right\} - 2E \left\{ \frac{1}{n} \sum_{i=1}^n d_m^y (\bar{d}_i^y + \varepsilon_i^y) \right\} + E \left\{ \frac{1}{n} \sum_{i=1}^n (d_m^y)^2 \right\}. \quad (11)$$

If it is supposed that the extracted plane is a rectangular plane and the number of point n is large enough, the following equations can hold:

$$E\left\{\frac{1}{n}\sum_{i=1}^n (d_m^y)^2\right\} \approx 0, \quad (12)$$

$$E\left\{\frac{1}{n}\sum_{i=1}^n d_m^y (\bar{d}_i^y + \varepsilon_i^y)\right\} \approx 0, \quad (13)$$

$$E\left\{\frac{1}{n}\sum_{i=1}^n (\bar{d}_i^y \varepsilon_i^y)\right\} \approx 0, \quad (14)$$

$$E\left\{\frac{1}{n}\sum_{i=1}^n (\varepsilon_i^y)^2\right\} = (\sigma^y)^2, \quad (15)$$

$$E\left\{\frac{1}{n}\sum_{i=1}^n (\bar{d}_i^y)^2\right\} = \frac{b^2}{12}, \quad (16)$$

where σ^y is the mean square deviation of noise ε_i^y and b is the side length of the edge which is parallel to the Y -axis.

Equations (12), (13), (14), (15), and (16) are substituted into equation (11) and then

$$E(I_y) = \frac{b^2}{12} + (\sigma^y)^2. \quad (17)$$

Similarly, $E(I_x) = a^2/12 + (\sigma^x)^2$ and $E(I_z) = (\sigma^z)^2$.

Suppose $I_{xy} = (1/n)\sum_{i=1}^n (d_i^x - d_m^x)(d_i^y - d_m^y)$ and then

$$\begin{aligned} E(I_{xy}) &= E\left(\frac{1}{n}\sum_{i=1}^n d_i^x d_i^y\right) - E\left(\frac{1}{n}\sum_{i=1}^n d_i^x d_m^y\right) \\ &\quad - E\left(\frac{1}{n}\sum_{i=1}^n d_m^x d_i^y\right) + E\left(\frac{1}{n}\sum_{i=1}^n d_m^x d_m^y\right). \end{aligned} \quad (18)$$

It is regarded that d_i^x and d_i^y are independent of each other. Therefore, the following equation holds:

$$E(I_{xy}) = 0. \quad (19)$$

Similarly, $E(I_{xz}) = 0$ and $E(I_{yz}) = 0$.

It can be seen from the above derivation that the eigenvalue matrix Λ of the rectangular point cloud plane is only related to the size of the rectangular plane and the noise of the sensor. If the mean square deviation of the sensor noise is known, the parameters a and b can be estimated. Therefore, the extracted rectangular surface of the satellite can be identified by calculating the eigenvalue matrix Λ .

2.3. Coarse Pose Determination. The matrix decomposition operation presented in equation (5) can provide the attitude rotation matrix. The eigenvector matrix \mathbf{R} is the attitude rotation matrix from the sensor coordinate system to the body coordinate system of the extracted rectangular plane. It is assumed that the sensor coordinate system is the same as the chaser body coordinate system in this paper.

Thus, the quaternion of the body coordinate system with respect to the chaser body coordinate system, which is denoted as \mathbf{q}_c^r , can be calculated as

$$\begin{aligned} q_{c0}^r &= \frac{1}{2} (1 + R_{xx} + R_{yy} + R_{zz})^{1/2}, \\ q_{c1}^r &= \frac{1}{4q_{c0}^r} (R_{yz} - R_{zy}), \\ q_{c2}^r &= \frac{1}{4q_{c0}^r} (R_{zx} - R_{xz}), \\ q_{c3}^r &= \frac{1}{4q_{c0}^r} (R_{xy} - R_{yx}), \end{aligned} \quad (20)$$

where R_{xx} , R_{yy} , R_{zz} , R_{yz} , R_{zy} , R_{zx} , R_{xz} , R_{xy} , and R_{yx} are the elements of the attitude rotation matrix \mathbf{R} .

If the eigenvalues of \mathbf{J} are arranged as the middle, the largest, and the smallest in the diagonal line of Λ , there are 4 cases for establishing the body coordinate system depending on the axial direction due to the symmetrical structure of the extracted rectangle plane. The 4 cases are shown in Figure 2.

It is supposed that case (1) is the coordinate system of the present time (see Figure 2). Any one of 4 cases of the body coordinate system may appear during the next matrix decomposition operation. The relationship between the target attitude \mathbf{q}_i^t and the attitude quaternion \mathbf{q}_c^r can be described as

$$\mathbf{q}_i^t = \mathbf{q}_c^r \mathbf{q}_c^r \mathbf{q}_i^t, \quad (21)$$

where \mathbf{q}_i^t is the attitude quaternion of the chaser satellite and \mathbf{q}_c^r is the quaternion from the body coordinate system of the extracted rectangular plane to the target body coordinate system, which is known.

Because the target attitude quaternion $\widehat{\mathbf{q}}_i^t$ can be predicted in the filter which is designed in Section 3, the predicted $\widehat{\mathbf{q}}_c^r$ can be derived as

$$\widehat{\mathbf{q}}_c^r = \mathbf{q}_c^r \widehat{\mathbf{q}}_i^t \mathbf{q}_c^r. \quad (22)$$

The predicted $\widehat{\mathbf{q}}_c^r$ can be utilized to solve the multisolution problem presented in Figure 2. Define the difference quaternion between the predicted $\widehat{\mathbf{q}}_c^r$ and the measured \mathbf{q}_c^r as

$$\delta \mathbf{q}_c^r = (\widehat{\mathbf{q}}_c^r)^{-1} \mathbf{q}_c^r. \quad (23)$$

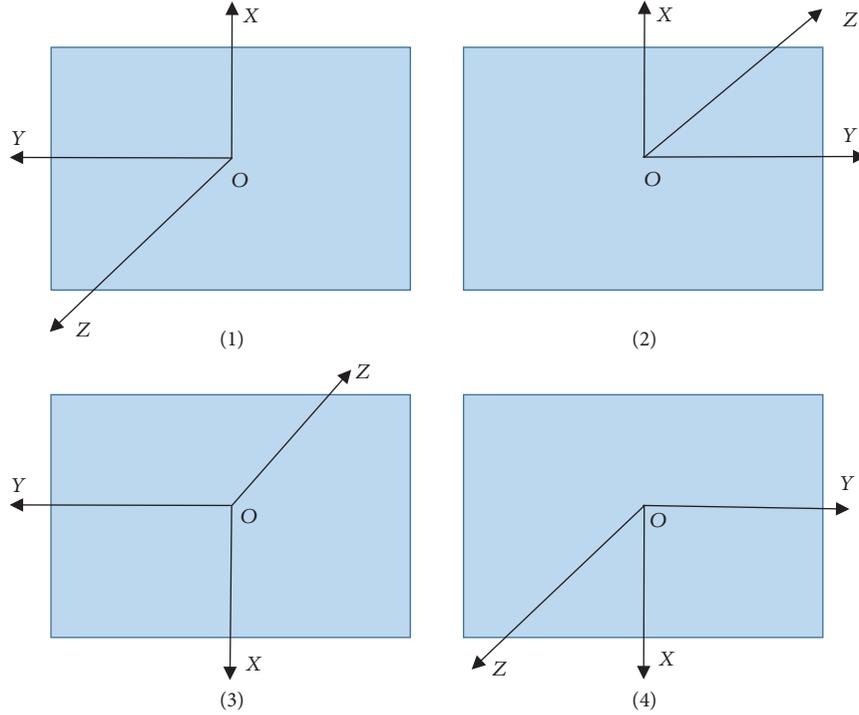


FIGURE 2: Description of the multisolution problem.

If the scalar part of $\delta\mathbf{q}_c^r$ is close to 1, case (1) occurs. If the scalar part of $\delta\mathbf{q}_c^r$ is close to 0, any one of cases (2)~(4) occurs, and the body coordinate system of the extracted plane needs to be changed to case (1) as

$$\tilde{\mathbf{q}}_c^r = \mathbf{q}_c^r \mathbf{q}' \quad (24)$$

where $\tilde{\mathbf{q}}_c^r$ is the adjusted quaternion and \mathbf{q}' is the quaternion which is utilized to adjust the attitude quaternion \mathbf{q}_c^r , satisfying

$$\mathbf{q}' = \mathbf{u} \cdot \sin \frac{\theta}{2} + \cos \frac{\theta}{2}, \quad (25)$$

where \mathbf{u} is the direction of the Euler axis and θ is equal to 180 degrees.

Cases (2), (3), and (4) need to be changed to case (1); the coordinate system should be rotated 180 degrees around the X-, Y-, and Z-axes, respectively. The corresponding quaternions are $\mathbf{q}' = [0 \ 0 \ 1 \ 0]^T$, $\mathbf{q}' = [1 \ 0 \ 0 \ 0]^T$, and $\mathbf{q}' = [0 \ 1 \ 0 \ 0]^T$, respectively.

If there are 2 or more extracted rectangular planes in a single measurement, the similar problem exists. It is supposed that there are 2 extracted rectangular planes. The 2 attitude quaternions are denoted as \mathbf{q}_{c1}^r and \mathbf{q}_{c2}^r , respectively. The corresponding target attitudes \mathbf{q}_{i1}^t and \mathbf{q}_{i2}^t can be derived according equation (21). The difference quaternion between \mathbf{q}_{i1}^t and \mathbf{q}_{i2}^t can be obtained by using equation (23). If the scalar part of the difference quaternion

is close to 0, the similar operation presented in equations (24) and (25) should be performed.

3. Estimation of the Rotational and Translational Parameters

3.1. *State Equations.* When two satellites are nearly in the same circular orbit with a close distance, the Hill equations can perfectly describe the translational motion dynamics as

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\mathbf{v}} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} \\ \mathbf{F}_1 & \mathbf{F}_2 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{v} \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{3 \times 3} \\ \mathbf{I}_{3 \times 3} \end{bmatrix} \mathbf{w}_1, \quad (26)$$

where

$$\mathbf{F}_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -n^2 & 0 \\ 0 & 0 & 3n^2 \end{bmatrix}, \quad \mathbf{F}_2 = \begin{bmatrix} 0 & 0 & 2n \\ 0 & 0 & 0 \\ -2n & 0 & 0 \end{bmatrix}, \quad (27)$$

n is the average orbital angular rate of the malfunctioned satellite and \mathbf{x} and \mathbf{v} are the relative position and velocity between the target and the chaser, which

is described in the target orbital frame. \mathbf{w}_1 is the disturbance force on an unit mass which is modeled as Gaussian white noise.

As the target satellite is a malfunctioned satellite, the angular velocity cannot be obtained directly. The attitude kinematics and dynamics can be written as

$$\begin{cases} \dot{\mathbf{q}}_t = \frac{1}{2} \mathbf{q}_t \otimes \bar{\boldsymbol{\omega}}, \\ \dot{\boldsymbol{\omega}}_t = \mathbf{I}_t^{-1} [-\boldsymbol{\omega}_t \times (\mathbf{I}_t \boldsymbol{\omega}_t)] + \mathbf{I}_t^{-1} \mathbf{w}_2, \end{cases} \quad (28)$$

where \mathbf{q}_t is the target attitude quaternion, $\bar{\boldsymbol{\omega}} = [0 \ \boldsymbol{\omega}_t^T]^T$, $\boldsymbol{\omega}$ is the target angular velocity, \mathbf{I}_t is the moment-of-inertia ratio matrix which can be described as

$$\mathbf{I}_t = \begin{bmatrix} 1 & \frac{I_{xy}}{I_{xx}} & \frac{I_{xz}}{I_{xx}} \\ \frac{I_{xy}}{I_{xx}} & \frac{I_{yy}}{I_{xx}} & \frac{I_{yz}}{I_{xx}} \\ \frac{I_{xz}}{I_{xx}} & \frac{I_{yz}}{I_{xx}} & \frac{I_{zz}}{I_{xx}} \end{bmatrix} = \begin{bmatrix} 1 & \tilde{I}_{xy} & \tilde{I}_{xz} \\ \tilde{I}_{xy} & \tilde{I}_{yy} & \tilde{I}_{yz} \\ \tilde{I}_{xz} & \tilde{I}_{yz} & \tilde{I}_{zz} \end{bmatrix}, \quad (29)$$

and \mathbf{w}_2 is the ratio of the disturbing torque and I_{xx} , which is modeled as Gaussian white noise.

3.2. Measurement Equations. The geometrical relationship between the two satellites is described in Figure 3. $O_c x_c y_c z_c$ is the chaser body coordinate system. \mathbf{r}_m^k is the location of the geometric centroid of an extracted rectangular plane, and it is expressed in the chaser body coordinate system, where k denotes the sequence number of the extracted rectangular plane. $O_r x_r y_r z_r$ is the target nominal body coordinate system. \mathbf{r}_b^k is the location of the geometric centroid of the extracted rectangular plane in the frame $O_r x_r y_r z_r$. Because the target is a malfunctioned satellite, the satellite structure model is known. Therefore, the relative pose between the extracted rectangular plane and frame $O_r x_r y_r z_r$ is known. So \mathbf{r}_b^k is a known value. Because the mass center changes due to the fuel consumption, the location error of the mass center is denoted as \mathbf{b}_k . $O_t x_t y_t z_t$ is the target body coordinate system, whose origin O_t is at the true mass center, and its axes x_t , y_t , and z_t are parallel to the axes x_r , y_r , and z_r , respectively. \mathbf{x} is the displacement vector from the target mass center to the chaser mass center and it is expressed in the target orbital frame. The origin of the orbital frame lies in the mass center of service satellite, the x_o axis is in the flight direction, the z_o axis is in the direction of the Earth center, and the y_o axis which completes the right-handed orthogonal coordinate system is in the opposite direction of the angular momentum of the orbit.

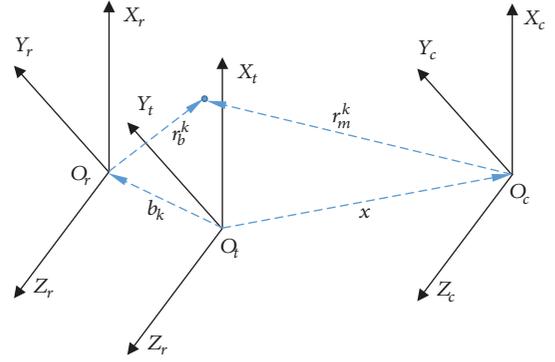


FIGURE 3: Geometrical relationship between two satellites.

According to the geometrical relationship presented in Figure 3, the displacement \mathbf{x} can be described as

$$\mathbf{x} = \mathbf{R}_t^{O_t} \left(\mathbf{r}_b^k + \mathbf{b}_k \right) - \mathbf{R}_c^{O_c} \mathbf{R}_c^{O_c} \left(\mathbf{r}_m^k + \mathbf{v}_k \right), \quad (30)$$

where $\mathbf{R}_t^{O_t}$ is the attitude rotation matrix from the frame $O_t x_t y_t z_t$ to the target orbital frame, $\mathbf{R}_c^{O_c}$ is the rotation matrix from the chaser orbital frame to the target orbital frame, $\mathbf{R}_c^{O_c}$ is the rotation matrix from the frame $O_c x_c y_c z_c$ to the chaser orbital frame and it can be determined by the chaser attitude determination system, and \mathbf{v}_k is the observation noise of the geometric centroid of the extracted rectangular plane.

Because the two satellites are nearly in same circular orbit with a close distance, the rotation matrix $\mathbf{R}_c^{O_c}$ can be regarded as the unit matrix [18].

$$\mathbf{R}_c^{O_c} \approx \mathbf{I}. \quad (31)$$

Therefore, equation (30) can be rewritten as

$$\begin{aligned} \mathbf{R}_t^{O_t} \mathbf{R}_c^{O_c} \mathbf{r}_m^k &\approx \mathbf{R}_c^{O_c} \mathbf{r}_m^k \\ &\approx -\mathbf{x} + \mathbf{R}_t^{O_t} \mathbf{r}_b^k + \mathbf{R}_t^{O_t} \mathbf{b}_k - \mathbf{R}_c^{O_c} \mathbf{v}_k \\ &= -\mathbf{x} + \mathbf{R}_t^{O_t} \mathbf{R}_t^i \mathbf{r}_b^k + \mathbf{R}_t^{O_t} \mathbf{R}_t^i \mathbf{b}_k + \tilde{\mathbf{v}}_k, \end{aligned} \quad (32)$$

where $\mathbf{R}_t^{O_c}$ is the rotation matrix from the inertial frame to the chaser orbital frame and it can be determined according to the orbit information and \mathbf{R}_t^i is target attitude matrix.

It is obvious that \mathbf{b}_k is a constant vector and can be modeled as

$$\dot{\mathbf{b}}_k = 0. \quad (33)$$

As the target is a malfunctioned satellite which is noncooperative, no information including the attitude parameters

The detailed information of the EKF can be concluded as

(1) Initialization:

$$\begin{aligned}\hat{\mathbf{X}}_0 &= E[\mathbf{X}_0], \\ \mathbf{P}_0 &= E[(\mathbf{X}_0 - E[\mathbf{X}_0])(\mathbf{X}_0 - E[\mathbf{X}_0])^T]\end{aligned}\quad (42)$$

(2) Time update:

$$\begin{aligned}\dot{\mathbf{X}}_{k-1} &= f(\mathbf{X}_{k-1}, \mathbf{w}_{k-1}), \\ \mathbf{P}_{k/k-1} &= \Phi_{k/k-1} \cdot \mathbf{P}_{k-1} \cdot \Phi_{k/k-1}^T + \mathbf{Q}_{k-1}\end{aligned}\quad (43)$$

(3) Measurement update:

$$\begin{aligned}\mathbf{K}_k &= \mathbf{P}_{k/k-1} \cdot \mathbf{H}_k^T \cdot [\mathbf{H}_k \cdot \mathbf{P}_{k/k-1} \cdot \mathbf{H}_k^T + \mathbf{R}_k]^{-1}, \\ \delta \hat{\mathbf{X}}_k &= \mathbf{K}_k \mathbf{Z}, \\ \mathbf{P}_k &= (\mathbf{I} - \mathbf{K}_k \cdot \mathbf{H}_k) \cdot \mathbf{P}_{k/k-1}\end{aligned}\quad (44)$$

The estimated states $\hat{\mathbf{x}}$, $\hat{\mathbf{v}}$, $\hat{\mathbf{b}}$, $\hat{\mathbf{q}}_k$, $\hat{\mathbf{w}}_k$, and $\hat{\mathbf{I}}_k$ can be innovated right after measurement update.

4. Simulation and Evaluation

In order to evaluate the effectiveness of extracting a point cloud plane, a point cloud rectangular plane is simulated by the simulation software. The point cloud is randomly simulated obeying a uniform distribution in a rectangular plane, which ranges from -5cm to 5cm in the X-axis, ranges from -9cm to 9cm in the Y-axis, and fixes at 6.5cm in the Z-axis. It is also simulated that there are 2500 points including 100, 500, and 1250 outliers, respectively, in the rectangular plane. The results are shown in Figures 4–7.

In this simulation, the repeat number is set to be 300 times, and the distance threshold is 0.005. The extraction results are shown in Table 1.

It can be seen in Table 1 that the plane can be extracted exactly from the point cloud with outliers. However, the processing speed of the plane extraction is slowed down with the increase of outliers.

The point cloud declines in number as the distance between the target and the sensor increases. In order to validate the effectiveness of a rectangular feature recognition formula as the number of points varies, the number of points in the rectangular plane is set to be 10000, 5000, 2500, 1250, and 625, respectively. The geometric centroid and the point cloud distribution matrix of the extracted plane are calculated, and the three eigenvalues

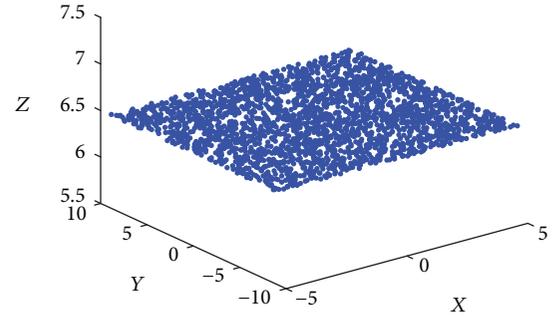


FIGURE 4: The original point cloud.

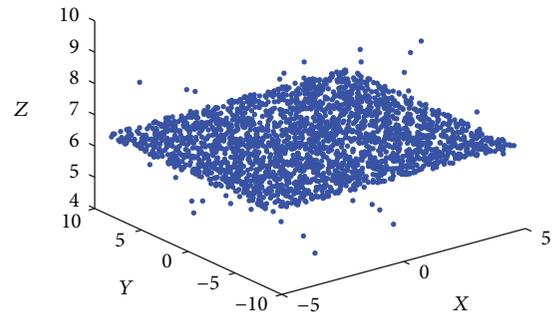


FIGURE 5: Point cloud with 100 outliers.

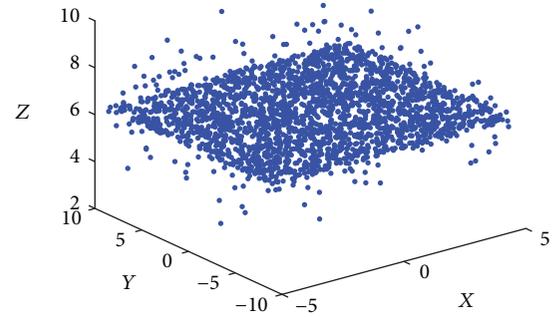


FIGURE 6: Point cloud with 500 outliers.

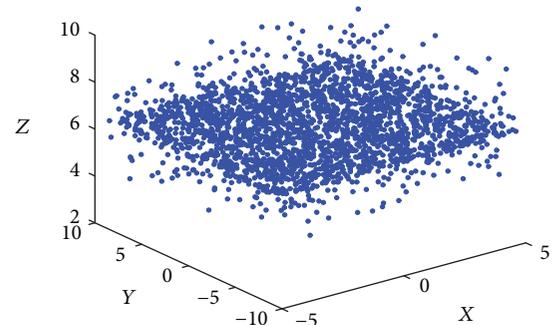


FIGURE 7: Point cloud with 1250 outliers.

TABLE 1: Plane extraction from the point cloud.

Outliers	The number of extracted points	Time (ms)
0	2500	20
100	2400	20
500	2000	30
1250	1252	70

TABLE 2: The geometric centroid and eigenvalues.

Number (points)	The center of gravity	Eigenvalue distribution
10000	(0.0154, -0.0390, 6.5000)	(8.4102, 26.7072, 0)
5000	(-0.0079, -0.0090, 6.5000)	(8.3106, 27.3320, 0)
2500	(-0.0801, -0.0033, 6.5000)	(8.5595, 26.7923, 0)
1250	(0.1542, -0.0438, 6.5000)	(8.4415, 26.4924, 0)
625	(0.0906, -0.3373, 6.5000)	(8.2372, 27.7920, 0)

TABLE 3: The location error of the geometric centroid.

Error (cm)	The geometric centroid (cm)	The location error (cm)
0	(-0.0801, -0.0033, 6.5000)	(-0.0801, -0.0033, 0)
0.01	(0.0817, -0.0833, 6.4999)	(0.0817, -0.0833, 0.0001)
0.1	(0.0541, 0.0661, 6.4984)	(0.0541, 0.0661, 0.0016)
1	(-0.0261, 0.0210, 6.5254)	(-0.0261, 0.0210, 0.0254)

TABLE 4: The eigenvalues.

Error (cm)	Eigenvalues (simulation)	Eigenvalues (theory)
0	(8.5595, 26.7923, 0)	(8.3333, 27, 0)
0.01	(8.5898, 26.9769, 0.0001)	(8.3334, 27.0001, 0.0001)
0.1	(8.3124, 26.9526, 0.0103)	(8.3433, 27.01, 0.01)
1	(9.4664, 27.7494, 0.9765)	(9.3333, 28, 1)

are obtained by using the matrix decomposition operation. The theoretical value of the geometric centroid is (0, 0, 6.5), and the eigenvalues are 8.3333, 27, and 0, respectively. The geometric centroid and the 3 eigenvalues of the rectangular point cloud are calculated and presented in Table 2. Therefore, it can be seen that the geometric centroid and the eigenvalues are almost not obviously affected by the number of points.

And then, the random noise of 0.01 cm, 0.1 cm, and 1 cm is added to the rectangular point cloud plane which contains 2500 points. It can be seen in Table 3 that the location error does not obviously increase as the added random noise increases, because the number of points is large enough.

The theoretical eigenvalues and the eigenvalues computed by using the matrix decomposition operation are

presented in Table 4. The computed eigenvalues are compared with the theoretical eigenvalues, and both the values are almost the same. Therefore, the effectiveness of the proposed recognition algorithm is validated according to Tables 3 and 4.

In order to validate the effectiveness of the relative navigation method, digital simulation software has been written in VC++. In this simulation, it is assumed that the target and the chaser are in the same orbit with different orbital phases. The chaser locates at the position which is about 25 m behind the target satellite. The inertia matrix of the target satellite is set as $\mathbf{I}_t = \text{diag}(30, 15, 20) \text{ kg} \cdot \text{m}^2$. The initial angular velocity of the target is $(0 \ 0 \ 1)^\circ/\text{s}$. The point cloud data is randomly simulated by using uniform distribution on the target surface with Gaussian noise. The number of the points is 1800, and the mean square deviation of the Gaussian error is $[0.1 \ 0.1 \ 0.1] \text{ m}$. The true location error \mathbf{b} is set as $[0.1 \ 0.1 \ 0.1] \text{ m}$.

The PCL (Point Cloud Library) is employed to extract planes. And then, the proposed method is utilized to recognize the feature rectangular plane and determine the relative position and attitude between the feature rectangular plane and the chaser. The relative position and attitude errors are shown in Figures 8–10.

It can be seen from Figure 8 that the accuracy of the relative position is better than 0.01 m, and the triaxial maximum errors are about 0.01 m, 0.01 m, and 0.01 m, respectively. The attitude error with a multisolution problem is shown in Figure 9, and the attitude error after removing the multisolution problem is shown in Figure 10. The multisolution problem occurs, and then the error is about 180° . There is no such phenomenon in Figure 10. Therefore, the method proposed in this paper has solved the attitude multisolution problem caused by the symmetrical rectangular plane. The attitude error presented in Figure 10 is better than 0.2° , and the triaxial maximum error is about 0.13° , 0.12° , and 0.16° .

In the relative navigation filter, $\mathbf{x}(0)$ is set as the first observation of the relative position, and $\dot{\mathbf{x}}(0)$ is set as 0. $\mathbf{q}_t(0)$ is set as the first observation of the target attitude, and $\boldsymbol{\omega}_t(0)$ is set to be the target angular velocity corrupted by initial errors. The covariance matrix $\mathbf{P}_0 = \text{diag}(\boldsymbol{\sigma}_x^2, \boldsymbol{\sigma}_v^2, \boldsymbol{\sigma}_b^2, \boldsymbol{\sigma}_{q_{t,13}}^2, \boldsymbol{\sigma}_{\omega_t}^2, \boldsymbol{\sigma}_I^2)$, where $\boldsymbol{\sigma}_x = [1^2 \ 1^2 \ 1^2] (\text{m}^2)$, $\boldsymbol{\sigma}_v^2 = [0.1^2 \ 0.1^2 \ 0.1^2] (\text{m/s})^2$, $\boldsymbol{\sigma}_b^2 = [0.1^2 \ 0.1^2 \ 0.1^2] (\text{m}^2)$, $\boldsymbol{\sigma}_{q_{t,13}}^2 = [0.1^2 \ 0.1^2 \ 0.1^2] (\text{rad})$, $\boldsymbol{\sigma}_{\omega_t}^2 = 10^{-4} \times [1^2 \ 1^2 \ 1^2] (\text{rad/s})^2$, and $\boldsymbol{\sigma}_I^2 = [0.1^2 \ 0.1^2 \ 0.1^2 \ 0.1^2 \ 0.1^2]$. The system noise is Gaussian noise with zero mean, and the system noise variance matrix is $\mathbf{Q} = 10^{-12} \mathbf{I}_{6 \times 6}$.

It can be seen from Figures 11–13 that not only the relative position and velocity can be estimated but also the mass center location error of the malfunctioned satellite can be estimated. According to the simulation curves, the relative position and velocity errors converge in a very short time, and the fluctuation is very small after convergence. The accuracy of the three axis positions is better than 1.4 mm, 3.2 mm, and 1.4 mm. The triaxial velocity

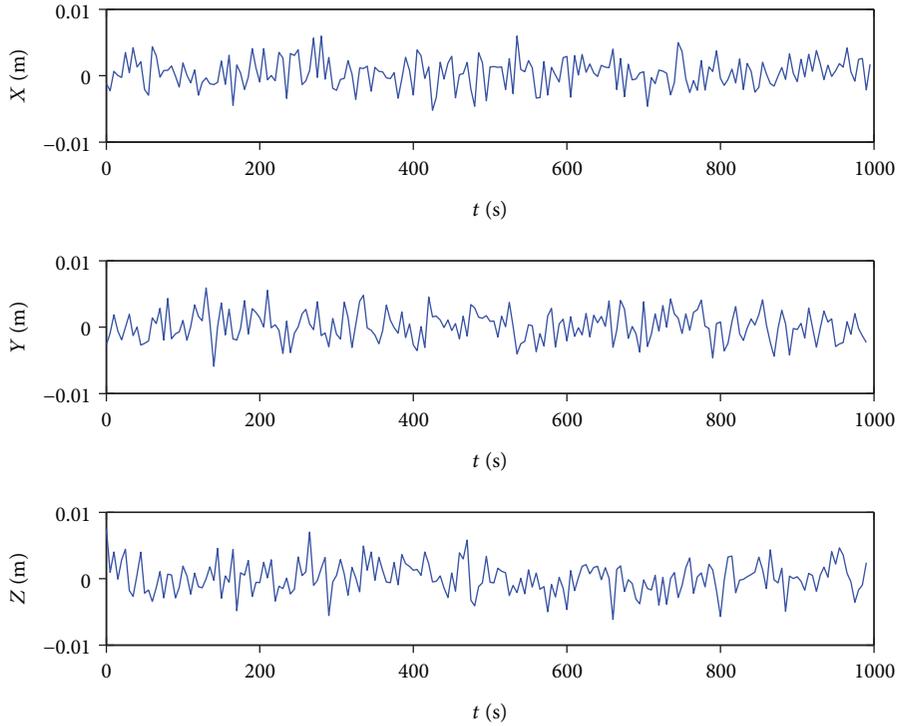


FIGURE 8: Position errors.

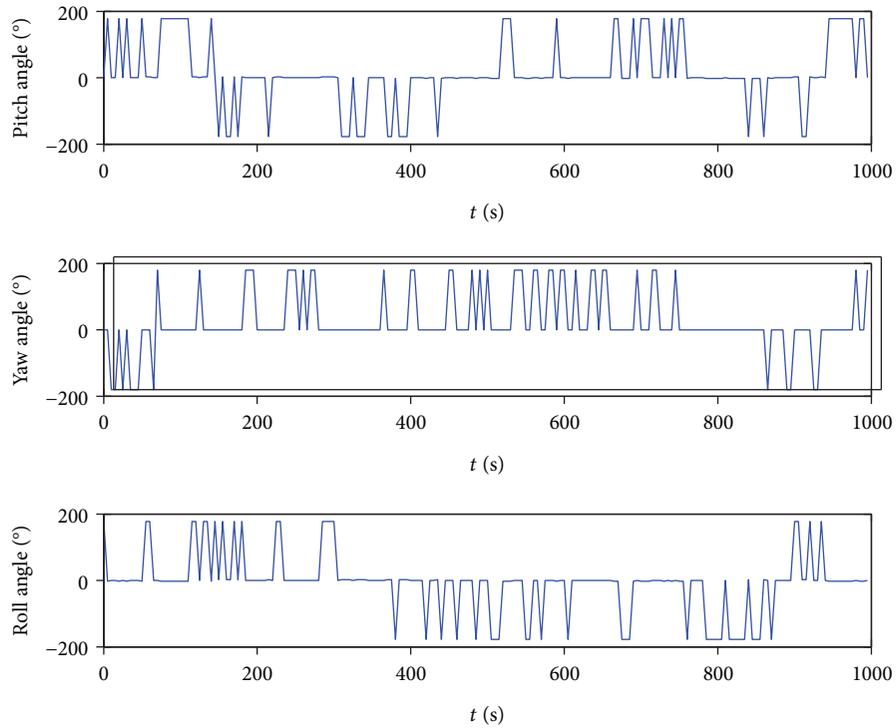


FIGURE 9: Attitude errors with multisolution.

error is better than 0.01 mm/s, 0.01 mm/s, and 0.01 mm/s, respectively. The estimation error of the mass center location is less than 1 mm, 1 mm, and 1 mm, respectively.

The estimation errors of the attitude, the angular velocity, and the moment-of-inertia ratios of the target satellite are presented in Figures 14–16. There are some fluctuations of

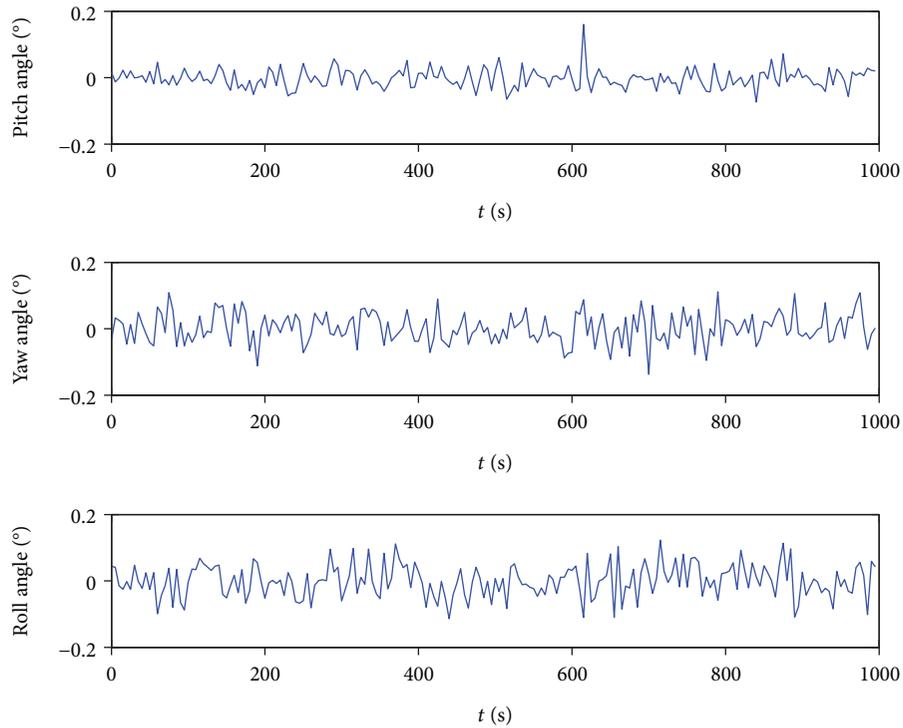


FIGURE 10: Attitude errors removing multisolution.

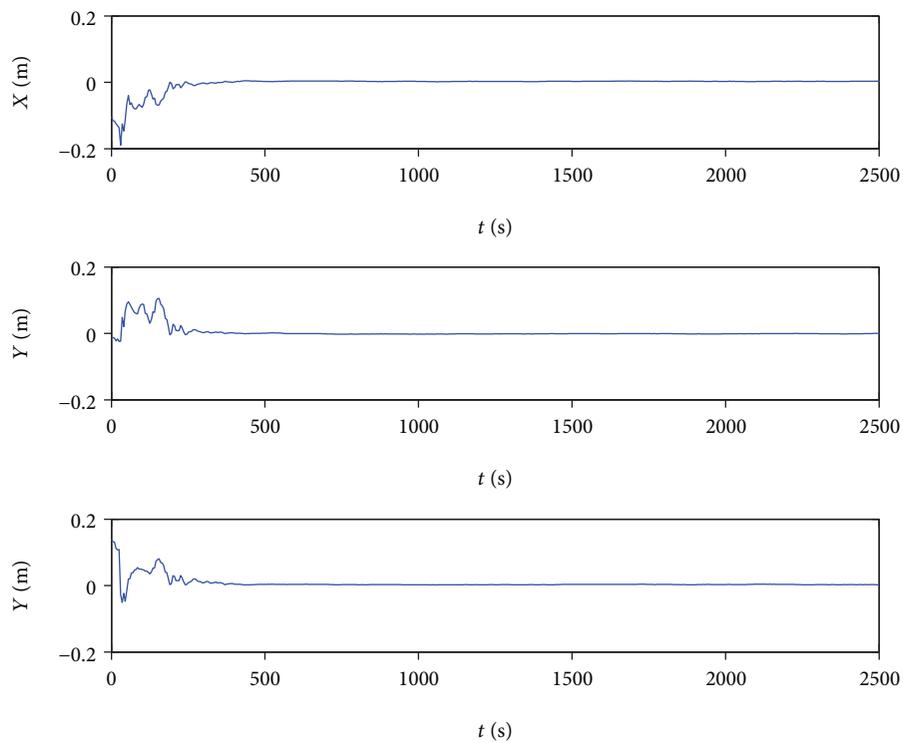


FIGURE 11: Relative position errors.

the malfunctioned satellite attitude, but the accuracy is better than 0.1° , and the angular velocity error is better than $0.01^\circ/s$. The moment-of-inertia ratios also converge to the true values as time goes by.

5. Conclusion

A new method is proposed in this paper to estimate target pose by utilizing local feature rectangular planes when the

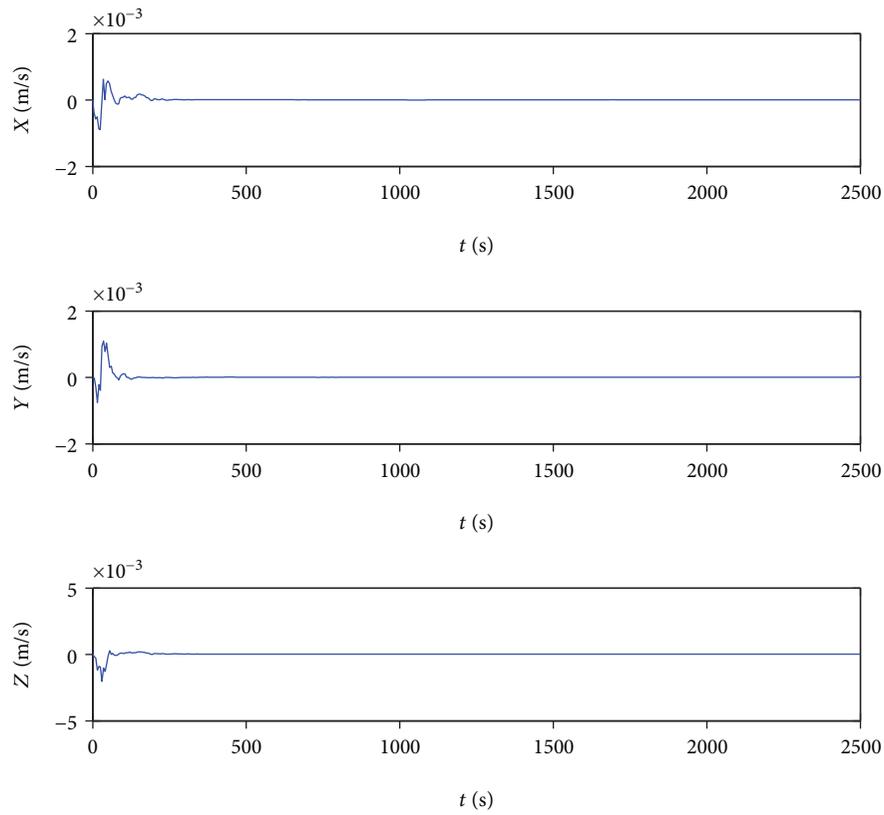


FIGURE 12: Relative velocity errors.

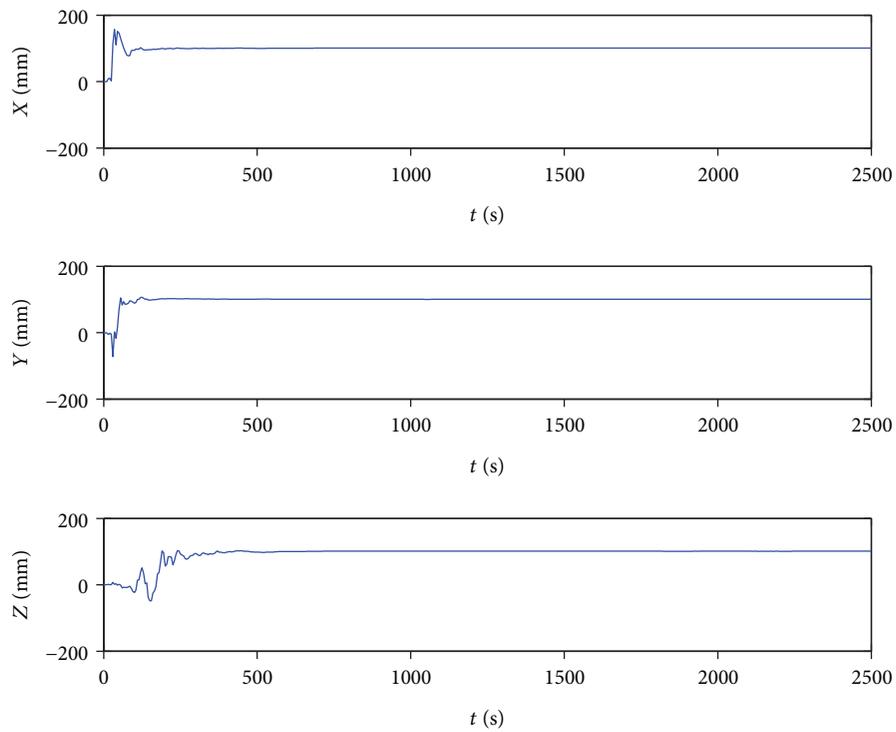


FIGURE 13: Location errors of the mass center.

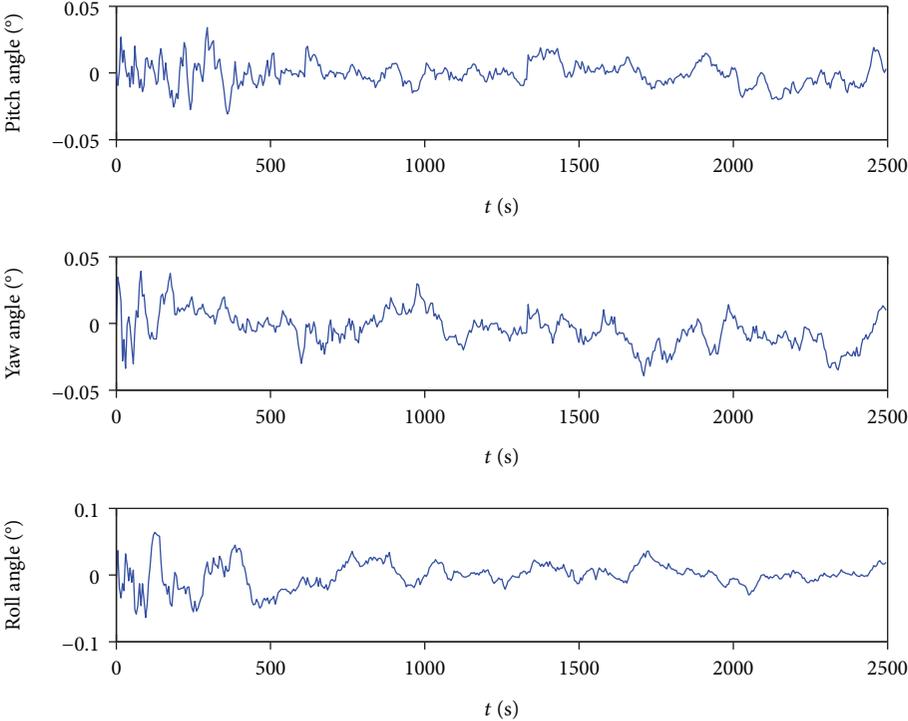


FIGURE 14: Attitude errors of the target.

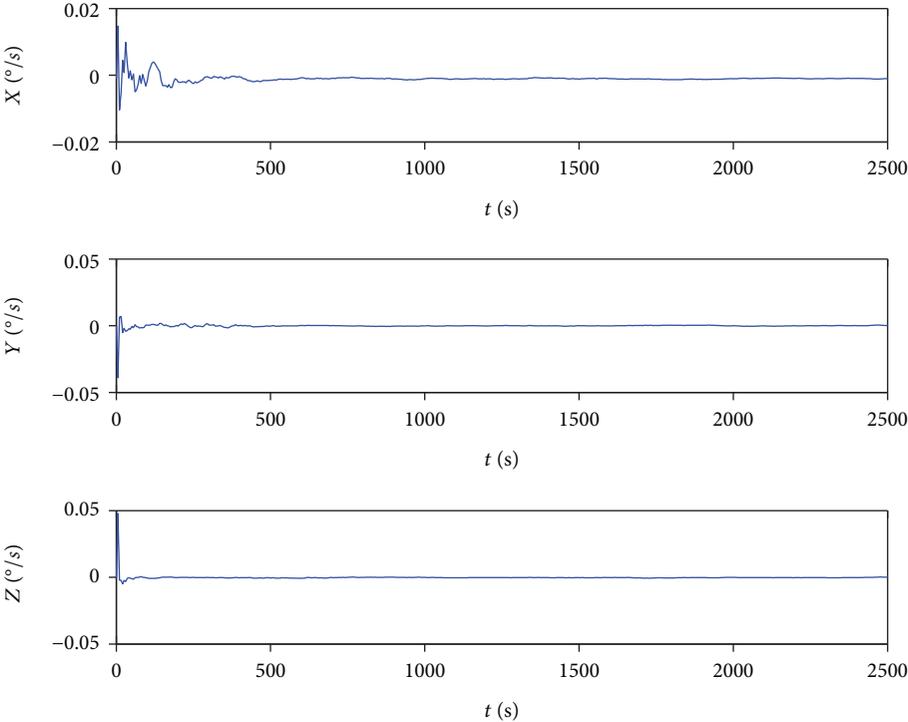


FIGURE 15: Angular velocity errors of the target.

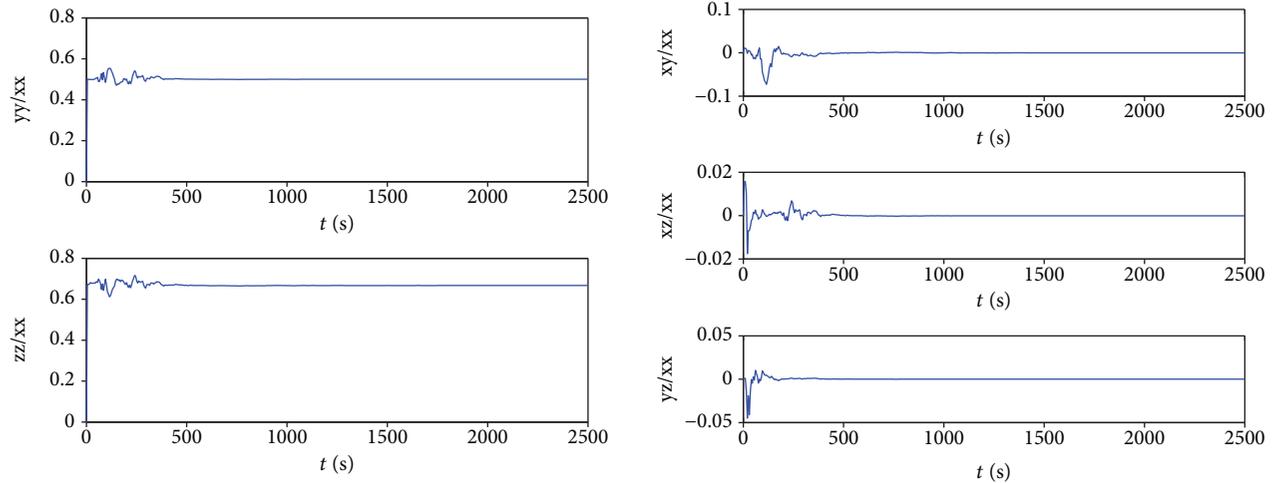


FIGURE 16: Moment-of-inertia ratio errors.

whole satellite cannot be observed. The plane extraction method, the rectangular feature recognition method, and the solution of multisolution problem are presented. Finally, the Kalman filter is designed to estimate the relative position and the attitude of the target satellite. Therefore, this paper provides a feasible algorithm for relative navigation during rendezvous and docking to malfunctioned satellites.

Data Availability

The data used to support the findings of this study are included within the article. The data is generated by using the simulation software which is described in Simulation and Evaluation. The data used to support the findings of this manuscript: “Pose Determination for Malfunctioned Satellites Based on Depth Information”, written by Feng Yu, Yi Zhao, and Yanhua Zhang, is generated by simulation software. The detailed simulation conditions are presented in detail in this manuscript. The simulation software is available from the corresponding author upon request.

Conflicts of Interest

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

Acknowledgments

This study was supported by the “Fundamental Research Funds for the Central Universities,” No. NS2016084. The authors fully appreciate the financial support.

References

- [1] G. Roesler, P. Jaffe, and G. Henshaw, “Inside DARPA’s mission to send a repair robot to geosynchronous orbit,” *IEEE Spectrum*, vol. 8, 2017.
- [2] B. Liang, X. Du, C. Li, and W. Xu, “Advances in space robot on-orbit servicing for non-cooperative spacecraft,” *Robot*, vol. 34, no. 2, pp. 242–256, 2012.
- [3] K. Landzettel, “Ongoing space robotics missions: TECSAS/DEOS[EB/OL],” http://www.dlr.de/rm-neu/en/desktopdefault.aspx/tabid-3825/5963_read-8759.
- [4] J. Galante, J. Van Eepoel, M. Strube et al., “Pose measurement performance of the argon relative navigation sensor suite in simulated-flight conditions,” in *AIAA Guidance, Navigation, and Control Conference*, pp. 27–42, Minneapolis, Minnesota, 2012.
- [5] S. D’Amico, J. S. Ardaens, G. Gaias, H. Benninghoff, B. Schlepp, and J. L. Jørgensen, “Noncooperative rendezvous using angles-only optical navigation: system design and flight results,” *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 6, pp. 1576–1595, 2013.
- [6] R. Volpe, M. Sabatini, and G. B. Palmerini, “Pose and shape reconstruction of a noncooperative spacecraft using camera and range measurements,” *International Journal of Aerospace Engineering*, vol. 2017, Article ID 4535316, 13 pages, 2017.
- [7] R. Volpe, G. B. Palmerini, and M. Sabatini, “A passive camera based determination of a non-cooperative and unknown satellite’s pose and shape,” *Acta Astronautica*, vol. 151, pp. 805–817, 2018.
- [8] C. Caixiu, *Pose On-Orbit Measurement of Non-Cooperative Spacecraft*, Beijing University of Posts and Telecommunications, Beijing, China, 2015.
- [9] X. Miao, F. Zhu, Y. Hao, Q. Wu, and R. Xia, “Vision pose measurement for non-cooperative space vehicles based on solar panel component,” *Chinese High Technology Letters*, vol. 23, no. 4, pp. 400–406, 2013.
- [10] L.-m. Zhang, F. Zhu, Y.-m. Hao, and M.-y. Wang, “Pose measurement based on a circle and a non-coplanar feature point,” *Acta Photonica Sinica*, vol. 44, no. 11, article 1112002, 2015.
- [11] 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.
- [12] S. Segal, A. Carmi, and P. Gurfil, “Stereovision-based estimation of relative dynamics between noncooperative satellites: theory and experiments,” *IEEE Transactions on Control Systems Technology*, vol. 22, no. 2, pp. 568–584, 2014.
- [13] F. Terui, H. Kamimura, and S. i. Nishida, “Motion estimation to a failed satellite on orbit using stereo vision and 3D model

- matching,” in *2006 9th International Conference on Control, Automation, Robotics and Vision*, pp. 1–8, Singapore, 2006.
- [14] J. A. Christian and S. Cryan, “A survey of LIDAR technology and its use in spacecraft relative navigation,” in *AIAA Guidance, Navigation, and Control (GNC) Conference*, pp. 1–7, Boston, MA, USA, 2013.
- [15] H. Gómez Martínez, G. Giorgi, and B. Eissfeller, “Pose estimation and tracking of non-cooperative rocket bodies using time-of-flight cameras,” *Acta Astronautica*, vol. 139, pp. 165–175, 2017.
- [16] U. Hillenbrand and R. Lampariello, “Motion and parameter estimation of a free-floating space object from range data for motion prediction,” in *The 8th international symposium on artificial intelligence, Robotics and Automation in Space*, pp. 1–10, Munich, Germany, 2005.
- [17] M. D. Lichter and S. Dubowsky, “Estimation of state, shape, and inertial parameters of space objects from sequences of range images,” in *Intelligent Robots and Computer Vision XXI: Algorithms, Techniques, and Active Vision*, pp. 194–205, Providence, RI, USA, 2003.
- [18] F. Yu, Z. He, B. Qiao, and X. Yu, “Stereo-vision-based relative pose estimation for the rendezvous and docking of noncooperative satellites,” *Mathematical Problems in Engineering*, vol. 2014, Article ID 461283, 12 pages, 2014.
- [19] T. Bailey, *Mobile robot localization and mapping in extensive outdoor environments*, [Ph.D. thesis], University of Sydney, 2002.

