We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

State Feature Extraction and Relative Navigation Algorithms for Spacecraft

Kezhao Li[1,2], Qin Zhang[1] and Jianping Yuan[3]
[1]Dept.of Geomatics, Chang'an University,
[2]Henan Polytechnic University,
[3]Northwestern Polytechnical University,
China

1. Introduction

Since 1957 when the first manmade satellite launched, humankind has made splendid progress in space exploration. However, we must face some new problems, which have affected or will affect new space activities: (i) space debris problem. There are more than 8700 objects larger than 10~30 cm in Low Earth Orbit (LEO) and larger than 1m in Geostationary Orbit (GEO) registered in the US Space Command Satellite Catalogue (D.Mehrholz, 2002). Among these space objects, approximately 6% are operational spacecrafts, that is to say, about 94% of the catalogued objects no longer serve any useful purpose and are collectively referred to as 'space debris'. If we don't track, detect, model for these space debris, the hazards of on-orbit spacecrafts or future spacecrafts will be enhanced. Fortunately, this problem has been recognized; (ii) maintenance for disable satellites. Sometimes an operational spacecraft is out of use only due to some simple faults. If it is maintained properly, it can still work as usual. So this is an economical way to use space resource. For example, a tyre of an expensive car has been broken, we can take a few of money to maintain it, and it can work as well as before. First of all, the problem of tacking, detecting and relative posing for disable spacecrafts must be solved, and then we can capture them or do some on-orbit service; (iii)on-orbit assembling of large-scale space platform. Along with the space exploring, it is a challenge and profound space project to build a large-scale space platform through launching in batches and assembling in orbit, and this will provide a valid platform for human to explore deep space. Whereas, the key technology of on-orbit assembling of large-scale space platform is space rendezvous and docking, it is also needed tracking, detecting and relative posing space objects. To solve those above problems successfully, the problem about space detection and relative posing must be researched and solved firstly. In recent twenty years, a series of important plans for space operations, including Demonstration of Autonomous Rendezvous Technology (DART) (Ben Iannotta, 2005 ; Richard P. Kornfeld, 2002 ; LiYingju, 2006),Orbital Express (OE) (Kornfeld, 2002 ; Michael A. Dornheim, 2006 ; Joseph W. Evans, 2006 ; Richard T. Howard, 2008), HII Transfer Vehicle (HTV) (Isao Kawano, 1999 ; Yoshihiko Torano, 2010), Automated Transfer Vehicle (ATV) (Gianni Casonato, 2004) etc, are paid greatly attention to by National Aeronautics and Space Administration (NASA) and Defense Advanced Research Projects Agency of America (DARPA) or National Space Development Agency of

Japan (NASDA) or European Space Agency (ESA) etc. And the operations, such as autonomous rendezvous and docking (AR&D), capturing, maintaining, assembling and attacking etc, have been involved in the plans above. As mentioned above, autonomous relative navigation is one of key technologies in all these space activities. And autonomous relative navigation based on machine vision is a direction all over the world currently. But there are some disadvantages of some traditional algorithms, such as complicated description, huge calculation burden, and lack of real-time ability etc (Wang Guangjun, 2004; Li Guokuan, 2000 ; H. P. Xu , 2006).
In order to overcome these disadvantages above, the algorithms of shape & state feature extraction and relative navigation for spacecraft are emphatically researched in this chapter.

2. Shape & state feature extraction algorithm based-on mathematical morphology

Mathematical morphology (MM) is a new discipline for imaging analysis and processing. Based on these characters, such as the character of nonlinear, morphological analysis, fast and parallel processing, simple and apt operation etc., mathematical morphology is very suitable for automation and intelligence object detection, and make it become a hotspot in imaging processing and correlation field. Recently, some successful applications of mathematical morphology have been made at home and abroad (Richard Alan Peters II, 1995; Joonki Paik, 2002; Ulisses Braga-Neto, 2003).

2.1 Basic four operation of MM
MM is a theory for the analysis of spatial structures which is a tool for extracting image components. It is called "Morphology" since it aims at analyzing the shape and form of object. The four basic morphological set transformations are dilation, erosion, opening and closing.

2.1.1 Dilation
Let $\mathbf{A}$ be an original image, and $\mathbf{B}$ be a SE. The dilation of $\mathbf{A}$ by $\mathbf{B}$ is defined as follows,

$$\mathbf{A} \oplus \mathbf{B} = \bigcup_{b_i \in \mathbf{B}} \mathbf{A}_{b_i} \tag{1}$$

Where $\mathbf{A}_{b_i} = \{a + b_i \mid a \in \mathbf{A}\}$.

2.1.2 Erosion
The erosion of $\mathbf{A}$ by $\mathbf{B}$ is defined as follows,

$$\mathbf{A} \ominus \mathbf{B} = \mathbf{A}^{C} \oplus \breve{\mathbf{B}}^{C}$$

$$= \{p \mid \coprod_{b_i \in \mathbf{B}} \coprod_{a_j \in \mathbf{A}^C} (p = a_j - b_i)\}^{C} \tag{2}$$

The superscript C in $\mathbf{A}^{C}$ stands for the complement of $\mathbf{A}$ such that $\mathbf{A}^{C} + \mathbf{A}$ =constant;
$\breve{\mathbf{B}}$ stands for the reflection of $\mathbf{B}$, that is, $\breve{\mathbf{B}} = \{-b_i \mid b_i \in \mathbf{B}\}$; The superscript C in $(\cdots)$ and $\{\cdots\}$ also stand for the complements of them.

2.1.3 Opening

The opening of **A** by **B** is defined as follows,

$$A \circ B = (A \ominus B) \oplus B \tag{3}$$

2.1.4 Closing

The closing of **A** by **B** is defined as follows,

$$A \bullet B = [A \oplus (-B)] \ominus (-B) \tag{4}$$

2.2 The vital function of the structuring element (SE)

Using a probe called as SE to detect the image information is the principle idea of MM. When the probe is moving in the image, we can find and know the correlation the structure feature of the image each part. This method is similar to the human FOA (Focus of Attention) from detecting thought. As a probe, SE can be included some knowledge directly, such as shape, size, further more, the information of gray and colour, and we can use the knowledge and information to detect and study the characters of the image (Cui Yi, 2002). So how to select a convenient SE is very important.

Fig. 1 gives the different feature extraction results of the satellite according to the different SEs. From the Fig. 1, we can see that the feature extraction result from SE (b1) is better than the result from (b2). Therefore it is necessary to select SE according to the different applications. In the feature extraction of distributed spacecraft system, we can select the convenient structure element according to the character and the approximate attitude and orbital information of the spacecraft. Additionally, spacecraft move regularly in orbit, the relative position and attitude is changed every time. Thus dynamically re-structured element based-on the approximate attitude and orbital information of the spacecraft system is one of the research directions.
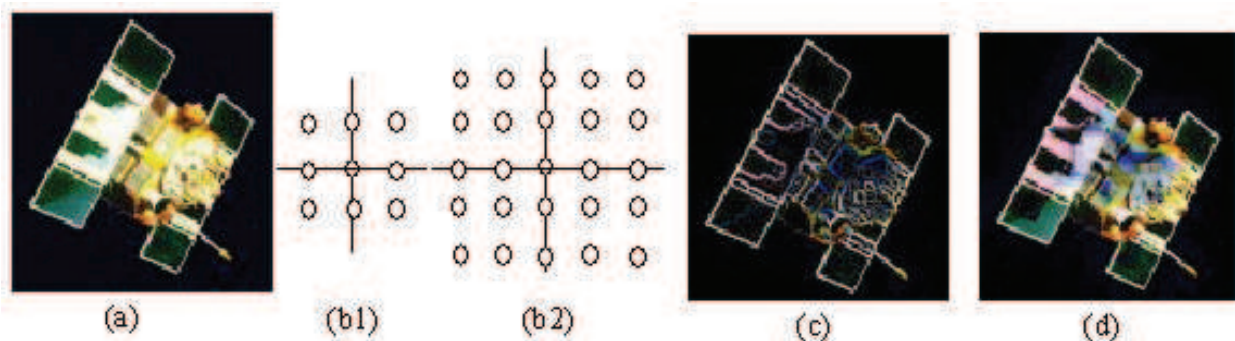

Fig. 1. The different feature extraction results of the satellite according to the different SEs. (a) The original image of satellite; (b1) and (b2) are two kind of SEs; (c) and (d) are the feature extraction results according to (b1) and (b2)

2.3 Dynamically re-structured element based-on the approximate attitude and orbital information of the spacecraft system

In the idea conditions, spacecraft move regularly in orbit according to their six basis orbital elements (semi-major axis: a; excentricity: e; ascending node: Ω; inclination of orbit: i; argument of perigee: ω; time of perigee passage: t_p) and their relative navigation angles

(yaw angle: ψ; roll angle: ϕ; pitch angle: θ;). As mentioned above, it is very important to select a valid SE in feature extraction of distributed spacecraft system, thus we can build the relationship between the movement rule of the spacecraft and dynamically re-structured element by using the SE database built beforehand. Considering that function $\Gamma(a,e,\Omega,i,\omega,f,\psi,\varphi,\theta)$ to stand for the spacecraft transformation form time t_1 to time t_2 (see Figure 2). On the basis theory of the attitude dynamics of spacecraft (Y. L. Xiao, 2003), we will build the function Γ as follow.

Two frame must be defined when the relative attitude described. Commonly, one is the space reference frame $ox_ry_rz_r$, and the other is body frame $ox_by_bz_b$ of the spacecraft. Thus the attitude Euler form is described as

$$\begin{cases} \psi = \arctan[-\dfrac{\mathbf{A}_{yx}}{\mathbf{A}_{yy}}] \\ \phi = \arcsin[\mathbf{A}_{yz}] \\ \theta = \arctan[-\dfrac{\mathbf{A}_{xz}}{\mathbf{A}_{zz}}] \end{cases} \tag{5}$$

$\mathbf{A}_{xz}$, $\mathbf{A}_{yx}$, $\mathbf{A}_{yy}$, $\mathbf{A}_{yz}$, $\mathbf{A}_{zz}$ stand for the cosine between $ox_ry_rz_r$ and $ox_by_bz_b$.

The spacecraft attitude differential equation can be calculated form this equation,

$$\begin{bmatrix} \dot{\psi} \\ \dot{\phi} \\ \dot{\theta} \end{bmatrix} = \frac{1}{\cos\phi}\begin{bmatrix} -\omega_x\sin\theta + \omega_z\cos\theta \\ \omega_x\cos\theta\cos\phi + \omega_z\sin\theta\cos\phi \\ \omega_x\sin\theta\sin\phi + \omega_y\cos\phi - \omega_z\cos\theta\sin\phi \end{bmatrix} \tag{6}$$

ω_x, ω_y, ω_z is the angle velocity.

So the absolute attitude expression of time t_k can be deduced from eq. (5) and (6),

$$\begin{bmatrix} \psi_{t_k} \\ \phi_{t_k} \\ \theta_{t_k} \end{bmatrix} = \begin{bmatrix} \psi_{t_0} \\ \phi_{t_0} \\ \theta_{t_0} \end{bmatrix} + (t_k - t_0)\begin{bmatrix} \dot{\psi} \\ \dot{\phi} \\ \dot{\theta} \end{bmatrix} \tag{7}$$

To calculate the relative attitude of spacecraft, we always build the relationship by geocentric equatorial inertial frame, the transformation formulation can be described as follow,

$$\begin{aligned} \mathbf{R}_{oi} &= \mathbf{R}_z(\omega + f)\mathbf{R}_x(i)\mathbf{R}_z(\Omega) \\ &= \begin{bmatrix} \cos(\omega+f) & \sin(\omega+f) & 0 \\ -\sin(\omega+f) & \cos(\omega+f) & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos i & \sin i \\ 0 & -\sin i & \cos i \end{bmatrix}\begin{bmatrix} \cos\Omega & \sin\Omega & 0 \\ -\sin\Omega & \cos\Omega & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{8}$$

Thus the absolute attitude angle of t_k defined in geocentric equatorial inertial frame can be calculated from eq. (9),

$$\Gamma_{t_k}(a,e,\Omega,i,\omega,f,\psi,\phi,\theta) = \mathbf{R}_{oi} \begin{bmatrix} \psi_{t_k} \\ \phi_{t_k} \\ \theta_{t_k} \end{bmatrix} \tag{9}$$

When the $\Gamma(a,e,\Omega,i,\omega,f,\psi,\varphi,\theta)$ is calculated, how to select the SE dynamically? As Fig. 2 shows, consider the track spacecraft attitude of time t_1 and time t_2 are orderly Λ_1 and Λ_2, the tracked spacecraft attitude of time t_1 and time t_2 are orderly Θ_1 and Θ_2, then we can build the expression as follows,

$$\Delta\Lambda_1\Theta_1 = \Gamma_{\Theta_1} - \Gamma_{\Lambda_1} \tag{10}$$

$$\Delta\Lambda_2\Theta_2 = \Gamma_{\Theta_2} - \Gamma_{\Lambda_2} \tag{11}$$

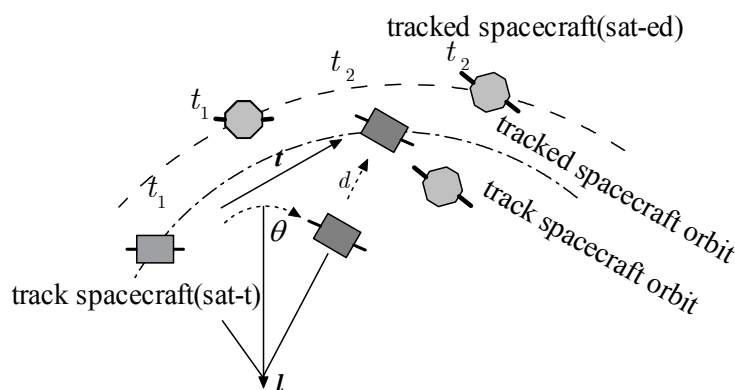$$\Delta\Lambda_{1,2}\Theta_{1,2} = \Delta\Lambda_2\Theta_2 - \Delta\Lambda_1\Theta_1 \tag{12}$$

Fig. 2. Track and tracked spacecraft sketch map

$\nabla\,\Delta\Lambda_{1,2}\Theta_{1,2}$ stands for the relative attitude between track and tracked spacecraft from time t_1 to time t_2. So dynamically re-structured element can be implemented from eq. (12).

2.4 Simulations and analyses
To prove the algorithm above, a simulation about a track and tracked satellites formation is studied in this section.

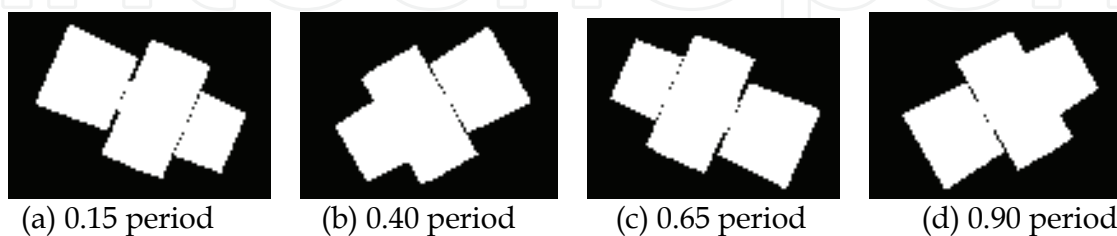| (a) 0.15 period | (b) 0.40 period | (c) 0.65 period | (d) 0.90 period |

Fig. 3. The original image of tracked satellite corresponding periods

According to Fig. 3, the corresponding SEs are designed from the solar panels character of the tracked spacecraft corresponding period (see Fig. 4). On the basis of these SEs, the feature extraction results are described as Figure 5 and Figure 6.
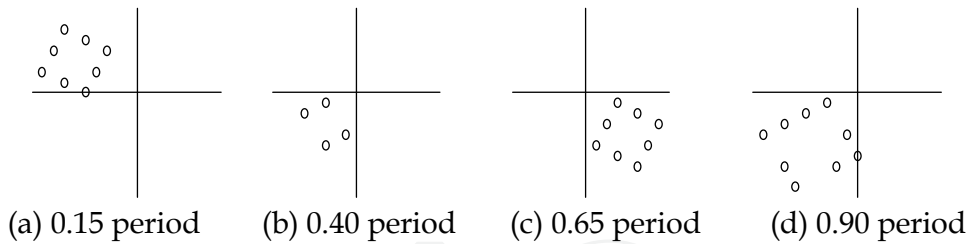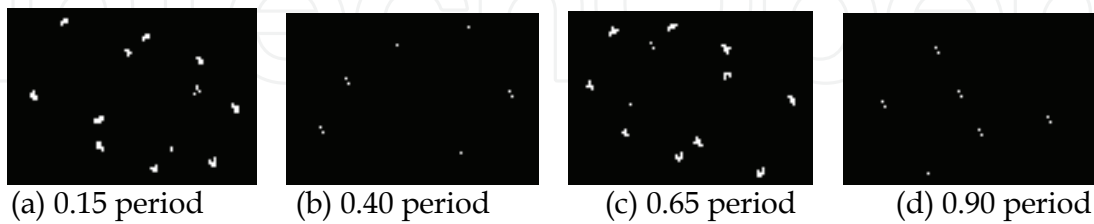
(a) 0.15 period   (b) 0.40 period   (c) 0.65 period   (d) 0.90 period

Fig. 4. SEs of corresponding periods

(a) 0.15 period   (b) 0.40 period   (c) 0.65 period   (d) 0.90 period

Fig. 5. The feature extraction results corresponding SE of Fig. 4 (a)

(a) 0.15 period   (b) 0.40 period   (c) 0.65 period   (d) 0.90 period
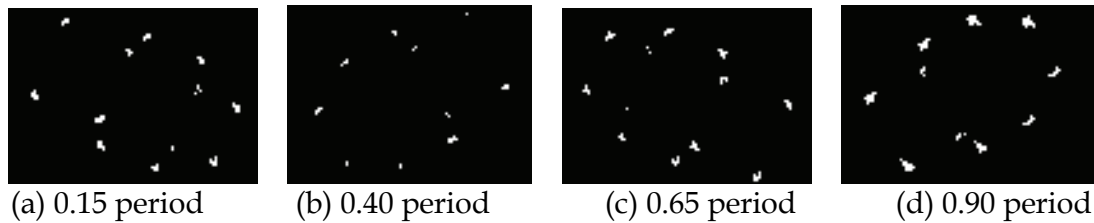
Fig. 6. The feature extraction results corresponding period SEs of Fig. 4

From Fig. 5 and Fig. 6, we can see: ( ) the feature extraction results of Fig. 5 are worse, especially the results (b) and (d) are distorted by using the SE of Figure 4 (a), because the shape of (b) and (d) are different from the SE of Fig. 4 (a); ( ) the feature extraction results of Fig. 6 are better because the corresponding period SEs are used in data processing.

3. Static forecast algorithms based-on quaternion and Rodrigues

3.1 Static forecast algorithm based-on quaternion

There already exists Hall algorithm for positioning and posing (Schwab A. L,2002). We now propose a new algorithm that we believe in better than Hall's. In this section, we explain in some detail our algorithm. We just add some pertinent remarks to listing the two topics of explanation. The first topic is: quaternion based method for determining position and attitude. Its two subtopics are: the quaternion based description of the rotational transformation for three dimensional bodies (subtopic 3.1.1), the camera model and the basic equation for machine vision for determining position and attitude (subtopic 3.1.2) and the quaternion based model for determining position and attitude by machine vision (subtopic 3.1.3). In subtopic 3.1.3, the initial position values are calculated by eq.(25) in this section; eq.(25) is based on Taylor expansion and least squares method. The second topic is: the algorithm for positioning and posing based on quaternion and spacecraft orbit and attitude information. Finally we give an example of numerical simulation, whose results are given in Figs. 8 through 10 in this section. These results show preliminarily that our proposed algorithm is much faster than Hall's.

3.1.1 Representation of 3D vector transformation by quaternion

Considering that $\mathbf{x}$ stands for 3D vector, and $\mathbf{x}'$ is a 3D vector from $\mathbf{x}$ by transformation matrix $\mathbf{R}$, this transformation can be represented as

$$\mathbf{Q_x} = (0, \mathbf{x}) \tag{13}$$

$$\mathbf{Q_{x'}} = (0, \mathbf{x}') \tag{14}$$

$$\mathbf{Q} = (q_0, \mathbf{q}) = q_0 + q_1 \mathbf{i} + q_2 \mathbf{j} + q_3 \mathbf{k} \tag{15}$$

$$\mathbf{Q_{x'}} = \mathbf{Q} \circ \mathbf{Q_x} \circ \mathbf{Q}^{-1} = \mathbf{Q} \circ \mathbf{Q_x} \circ \bar{\mathbf{Q}} \tag{16}$$

Where $\mathbf{Q_x}, \mathbf{Q_{x'}}, \mathbf{Q}$ are all quaternions, $\mathbf{Q}^{-1}, \bar{\mathbf{Q}}$ are inverse and onjugate of $\mathbf{Q}$, and $\mathbf{Q}$ is the corresponding quaternion of matrix $\mathbf{R}$.

The relation of matrix $\mathbf{R}$ and $\mathbf{Q}$ can be described as

$$\mathbf{x}' = (q_0^2 - \mathbf{q} \cdot \mathbf{q})\mathbf{x} + 2q_0(\mathbf{q} \times \mathbf{x}) + 2(\mathbf{q} \cdot \mathbf{x})\mathbf{q} = \mathbf{R}\mathbf{x} \tag{17}$$

3.1.2 The camera model and the basic equation of computer vision

The process of relative position and pose based on computer vision is: first to extract and match the feature of the image; secondly to calculate the position and pose between the camera and the object. Therefore, camera model is the basis model of relative position and pose based on computer vision. And camera model is a simple style of optics imaging. This model represents the transformation from 3D to 2D object. Usually, two kinds of camera model, viz. linear and nonlinear camera model, are classified by the imaging process, whether object point, centre point and image point are co-lined or not. Nonlinear camera model is from linear camera model added by the aberration correction. In this paper we will apply linear camera model. The detail of nonlinear camera model can be see literature (Z.G. Zhu, 1995 ; S. D. Ma, 1998 ; G. J. Zhang, 2005 ; Marc Pollefeys, 2002).

Fig. 7 shows the projection relation of object point, centre point and image point. Where $O_I - uv$ stands for image frame, $O_i - x_i y_i$ stands for physical image frame, $O_C - x_C y_C z_C$ stands for camera frame, $O_W - x_W y_W z_W$ stands for object frame, this is consistent with body frame of objective spacecraft later.
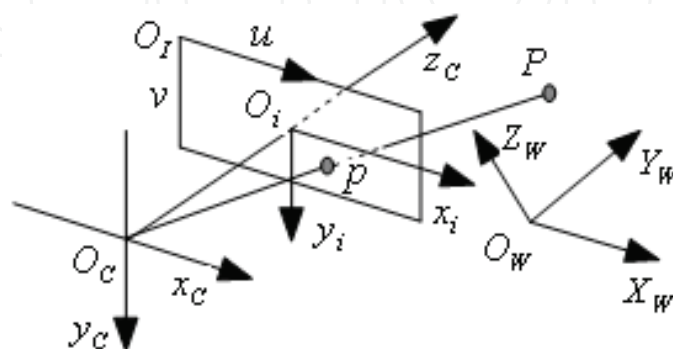

Fig. 7. Sketch of image frame, camera frame and object frame

The projection relation of object point, centre point and image point can be represented as,

$$\begin{cases} x_i = \dfrac{f x_C}{z_C} \\ y_i = \dfrac{f y_C}{z_C} \end{cases} \tag{18}$$

Where f is the focus of camera.

According to eq. (17) and (18), the relation of object frame and camera frame can be describe as

$$z_C \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} x_W \\ y_W \\ z_W \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} l_{11} & l_{12} & l_{13} & t_1 \\ l_{21} & l_{22} & l_{23} & t_2 \\ l_{31} & l_{32} & l_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_W \\ y_W \\ z_W \\ 1 \end{bmatrix} \tag{19}$$

From formula (19), co-line equation is get as follows

$$\left. \begin{aligned} x_i &= f \frac{l_{11} x_W + l_{12} y_W + l_{13} z_W + t_1}{l_{31} x_W + l_{32} y_W + l_{33} z_W + t_3} \\ y_i &= f \frac{l_{21} x_W + l_{22} y_W + l_{23} z_W + t_2}{l_{31} x_W + l_{32} y_W + l_{33} z_W + t_3} \end{aligned} \right\} \tag{20}$$

Eq. (20) is the basic equation of computer vision, $l_{ij}(i = 1, 2, 3; j = 1, 2, 3)$ is the relative attitude, $t_i(i = 1, 2, 3)$ is the objective spacecraft coordinate defined in camera frame.

3.1.3 The relative position and pose model based-on quaternion

In eq. (20), there are six absolute parameters: three attitude parameters and three translation parameters. In order to reduce the calculation parameters, quaternion is applied here. Let

$$\left. \begin{aligned} \bar{X} &= l_{11} x_W + l_{12} y_W + l_{13} z_W + t_1 = (q_0^2 + q_1^2 - q_2^2 - q_3^2) x_W + 2(q_1 q_2 + q_0 q_3) y_W + 2(q_3 q_1 - q_0 q_2) z_W + t_1 \\ \bar{Y} &= l_{21} x_W + l_{22} y_W + l_{23} z_W + t_2 = 2(q_1 q_2 - q_0 q_3) x_W + (q_0^2 - q_1^2 + q_2^2 - q_3^2) y_W + 2(q_2 q_3 + q_0 q_1) z_W + t_2 \\ \bar{Z} &= l_{31} x_W + l_{32} y_W + l_{33} z_W + t_3 = 2(q_3 q_1 + q_0 q_2) x_W + 2(q_2 q_3 - q_0 q_1) y_W + (q_0^2 - q_1^2 - q_2^2 + q_3^2) z_W + t_3 \end{aligned} \right\} \tag{21}$$

Then eq. (20) can be rewritten as

$$\left. \begin{aligned} x_i &= f \frac{\bar{X}}{\bar{Z}} \\ y_i &= f \frac{\bar{Y}}{\bar{Z}} \end{aligned} \right\} \tag{22}$$

Obviously, eq. (22) are nonlinear equations, then the linearisations are accomplisationed by expanding the function in a Taylor series to the first order (linear term) as,

$$\left. \begin{aligned} x_i &= F x_0 + \Delta F x \\ y_i &= F y_0 + \Delta F y \end{aligned} \right\} \tag{23}$$

Where Fx_0, Fy_0 are the results of entered the initial value of q_0, q_1, q_2, q_3, t_1, t_2, t_3 into eq. (22). $\Delta Fx, \Delta Fy$ are calculated as follows,

$$\left.\begin{aligned} \Delta Fx &= \frac{\partial x}{\partial q_0} \Delta q_0 + \frac{\partial x}{\partial \mathbf{q}} \Delta \mathbf{q} + \frac{\partial x}{\partial \mathbf{t}} \Delta \mathbf{t} \\ \Delta Fy &= \frac{\partial y}{\partial q_0} \Delta q_0 + \frac{\partial y}{\partial \mathbf{q}} \Delta \mathbf{q} + \frac{\partial y}{\partial \mathbf{t}} \Delta \mathbf{t} \end{aligned}\right\}$$

(24)

In eq. (24), $\Delta \mathbf{q} = (\Delta q_1, \Delta q_2, \Delta q_3)^{\mathrm{T}}$, $\Delta \mathbf{t} = (\Delta t_1, \Delta t_2, \Delta t_3)^{\mathrm{T}}$.

The iterative calculation above will be continued until the corrections less than the threshold values.

When observation point number $n > 4$, the results will be calculated by using least squares method. According to literature (Z.G. Zhu, 1995), the results of least squares method can be get from eq. (25) as follow,

$$\Delta \mathbf{X} = -(\mathbf{A}^{\mathrm{T}} \mathbf{P} \mathbf{A})^{-} \mathbf{A}^{\mathrm{T}} \mathbf{P} \mathbf{l}$$

(25)

Where $\Delta \mathbf{X} = \left[\Delta q_0, \Delta q_1, \Delta q_2, \Delta q_3, \Delta t_1, \Delta t_2, \Delta t_3 \right]^{\mathrm{T}}$; $\mathbf{A}$ is the coefficient matrix of $\mathbf{X}$ coefficients of eq. (24). $\mathbf{P}$ stands for weight matrix; $\mathbf{l} = [((Fx_0)_1 \quad (Fy_0)_1 \quad \cdots \quad (Fx_0)_n \quad (Fy_0)_n)^{\mathrm{T}} -(x_1 \quad y_1 \quad \cdots \quad x_n \quad y_n)^{\mathrm{T}}]^{T}$; n is the number of observation.

3.1.4 Relative navigation based on quaternion and spacecraft orbit & attitude information

From above, we can see that the calculation speed of the relative position and pose algorithm based-on quaternion of least squares method depends on the initial value selection. In this section, we look spacecraft orbit & attitude information as initial values. And next section will introduce how to calculate the relative position and pose of spacecraft according to spacecraft orbit & attitude information.

a.   Relative position calculated by using differential method

Considering there are active spacecraft A and objective spacecraft P, and their orbital elements are known, according to literature (Y. L. Xiao, 2003), the coordinates (x_A, y_A, z_A), (x_P, y_P, z_P) of inertial frame of active spacecraft A and objective spacecraft P can be calculated. So the relative position can be described as

$$\begin{bmatrix} \Delta x_{AP} \\ \Delta y_{AP} \\ \Delta z_{AP} \end{bmatrix} = \begin{bmatrix} x_P \\ y_P \\ z_P \end{bmatrix} - \begin{bmatrix} x_A \\ y_A \\ z_A \end{bmatrix}$$

(26)

Finally we can transform $\begin{bmatrix} \Delta x_{AP} & \Delta y_{AP} & \Delta z_{AP} \end{bmatrix}^{T}$ from inertial frame to body frame defined in active spacecraft A, and the relative position between spacecraft A and P is calculated.

b.   Relative pose calculated by using quaternion

Considering S_A is the body frame of spacecraft A, S_P is the body frame of spacecraft P, the relation of S_A, S_P and inertial frame S_i can be represented by using Rodrigues as

$$\left.\begin{array}{ccc} S_i & \mathbf{Q}_{Pi} & S_P \\ S_i & \mathbf{Q}_{Ai} & S_A \\ S_P & \mathbf{Q}_{AP} & S_A \end{array}\right\} \tag{27}$$

Thus relative attitude of spacecraft A and P can be described as

$$\mathbf{Q}_{AP} = \mathbf{Q}_{iP} \circ \mathbf{Q}_{Ai} \tag{28}$$

Where $\circ$ is quaternion multiplication sign.

Usually, camera is fixed on the active spacecraft A, we can transform $[x_b \quad y_b \quad z_b]^T$ from the body frame of the active spacecraft A to the camera frame $O_C - x_C y_C z_C$ as follows

$$\begin{bmatrix} x_C \\ y_C \\ z_C \end{bmatrix} = \mathbf{M} \begin{bmatrix} x_b \\ y_b \\ z_b \end{bmatrix} + \mathbf{T} \tag{29}$$

Where $\mathbf{M}$ is the attitude transition matrix, and $\mathbf{T}$ is the transition matrix from body frame defined in active spacecraft A to camera frame. They can be designed or measured.

Hereto, the relative attitude and position parameters between objective spacecraft P and camera frame are calculated. Then let these parameters as the initial value of eq. (25). And then the relative attitude and position between active spacecraft A and objective spacecraft P can be determined quickly.

3.1.5 Simulations and analyses

On the basis of the algorithm above, let the camera focus $f = 350$mm, the objective spacecraft P is a $2m \times 2m \times 2m$ cube, and its body frame coordinates of feature points are respectively $\{-1,-1,1\}$, $\{-1,-1,-1\}$, $\{-1,1,1\}$, $\{-1,1,-1\}$, $\{-1,0,-1\}$, $\{0,0,1\}$, $\{1,-1,1\}$, $\{1,1,1\}$. Table 1 lists the initial parameters of the simulations. According to the parameters of table 1, calculate the relative position and pose parameters between active spacecraft A and objective spacecraft P by eq. (26) and (28). And let these relative parameters as true value $\tilde{\mathbf{X}}$. Then calculate image coordinates by eq. (22), and add one pixel white noise to the image coordinates and let them as the simulation observations. Finally, calculate the relative position and pose parameters $\hat{\mathbf{X}}$ between active spacecraft A and objective spacecraft P by eq. (23) and (25). The simulation time is 1200 seconds. Fig. 8 – Fig. 10 are the simulation results. It is not intuitionistic to represent the attitude results by quaternion, yet the attitude results are described as their Euler form. In Fig. 8 – Fig. 10, (a) stands for the results based on spacecraft orbit & attitude information, (b) stands for the results based on optional value. The results of simulation are calculated by using the computer of HP Pavilion Intel (R), Pentium (R) 4, CPU 3.06GHz, 512 MB, the consumable times of method (a) and (b) are 4662 ms and 7874 ms respectively.

	Active spacecraft A	Objective spacecraft B
ascending node/deg	0.0	0.0
inclination of orbit/deg	96.498	96.498005
argument of perigee/deg	0.0	0.0
excentricity	0.0	0.00000001
semi-major axis/km	7146.768	7146.768
time of perigee passage/s	0	0.008
yaw/deg	0.5	0.5
pitch/deg	0.2	0.2
roll/deg	0.4	0.5
yaw angle velocity/(deg/s)	5×10^{-7}	5×10^{-7}
pitch angle velocity/(deg/s)	5×10^{-7}	5×10^{-7}
roll angle velocity/(deg/s)	5×10^{-7}	5×10^{-7}

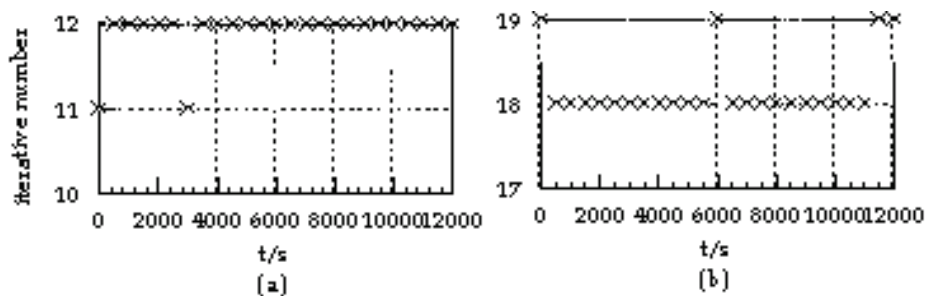Table 1. The Initial parameters of the simulations

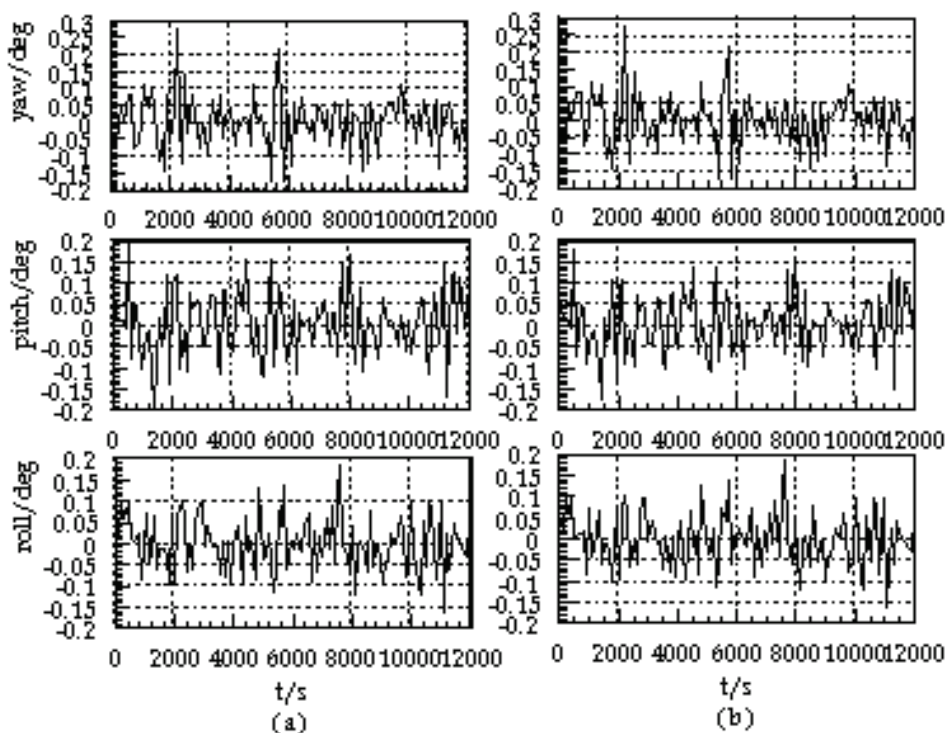Fig. 8. Iterative number of the algorithm based-on quaternion

Fig. 9. Relative attitude errors of the algorithm based-on quaternion
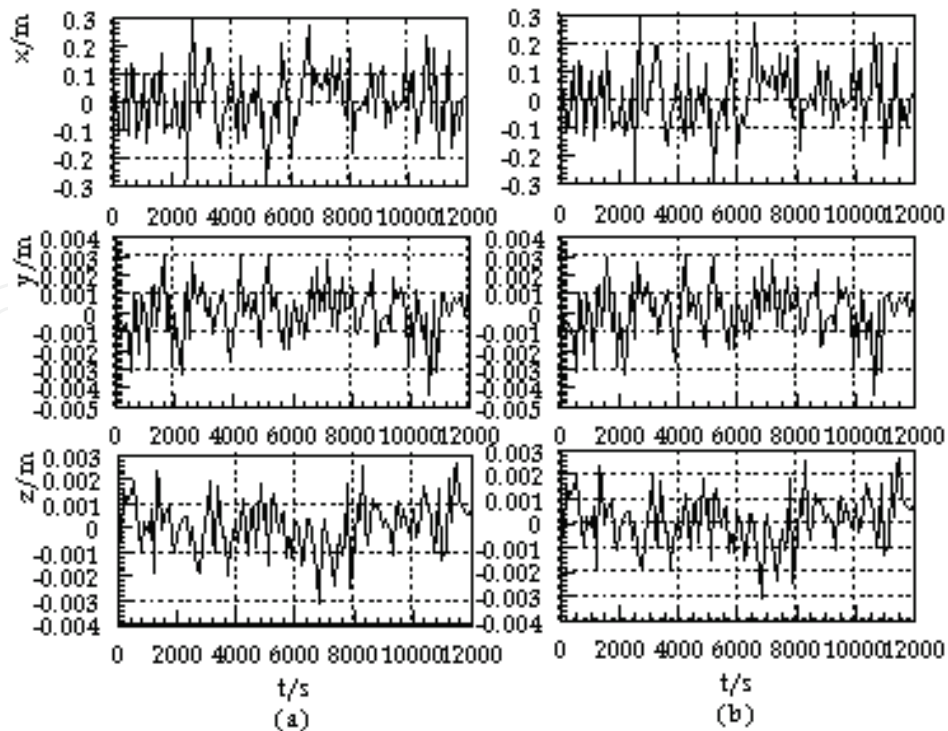
Fig. 10. Relative position errors of the algorithm based-on quaternion

From the simulation results, we can see that both two methods of (a) and (b) can get the high and similar precision results, whereas the iterative number of (a) is 11-12, and the iterative number of (b) is 18-19, moreover the consumable times of (a) is about half of the times of (b). All these show that the algorithm (a) is better than (b).

3.2 Static forecast algorithm based-on Rodrigues

As mentioned as **3.1** section, the algorithm based on quaternion is better than Hall's, because the Jacobi matrix of this method is lower than Hall's. But there is redundance value by using quaternion to represent the attitude. Rodrigues has three parameters to describe the attitude with no redundance variable. In this section we will discuss the static forecast algorithm by using Rodrigues.

3.2.1 Representation of 3D vector transformation by Rodrigues

Considering that $\mathbf{x}$ stands for 3D vector, and $\mathbf{x}'$ is a 3D vector from $\mathbf{x}$ by transformation matrix $\mathbf{R}$, this transformation can be represented as

$$\mathbf{x}' = \mathbf{R}\mathbf{x} \tag{30}$$

Where $\mathbf{R} = \dfrac{1}{\Delta} \begin{bmatrix} 1+a^2-b^2-c^2 & 2(ab+c) & 2(ac-b) \\ 2(ab-c) & 1-a^2+b^2-c^2 & 2(bc+a) \\ 2(ac+b) & 2(bc-a) & 1-a^2-b^2+c^2 \end{bmatrix} = \begin{bmatrix} l_{11} & l_{12} & l_{13} \\ l_{21} & l_{22} & l_{23} \\ l_{31} & l_{32} & l_{33} \end{bmatrix}$,

$$\Delta = |\mathbf{I} + \mathbf{S}| = 1 + a^2 + b^2 + c^2 .$$

3.2.2 The relative position and pose model based-on Rodrigues

In eq. (20), there are six absolute parameters: three attitude parameters and three translation parameters. In order to reduce the calculation parameters, Rodrigues is applied here. Let

$$\left.\begin{array}{l}\overline{X} = l_{11}x_W + l_{12}y_W + l_{13}z_W + t_1 = (1 + a^2 - b^2 - c^2)X_W + 2(ab+c)Y_W + 2(ac-b)Z_W + (1 + a^2 + b^2 + c^2)t_1 \\ \overline{Y} = l_{21}x_W + l_{22}y_W + l_{23}z_W + t_2 = 2(ab-c)X_W + (1 - a^2 + b^2 - c^2)Y_W + 2(bc+a)Z_W + (1 + a^2 + b^2 + c^2)t_2 \\ \overline{Z} = l_{31}x_W + l_{32}y_W + l_{33}z_W + t_3 = 2(ac+b)X_W + 2(bc-a)Y_W + (1 - a^2 - b^2 + c^2)Z_W + (1 + a^2 + b^2 + c^2)t_3\end{array}\right\} \quad (31)$$

Then formula (20) can be rewritten as

$$\left.\begin{array}{l}x_i = f\dfrac{\overline{X}}{\overline{Z}} \\ y_i = f\dfrac{\overline{Y}}{\overline{Z}}\end{array}\right\} \quad (32)$$

Obviously, eq. (32) are nonlinear equations, then the linearisations are accomplisationed by expanding the function in a Taylor series to the first order (linear term) as,

$$\left.\begin{array}{l}x = Rodrx_0 + \Delta Rodrx \\ y = Rodry_0 + \Delta Rodry\end{array}\right\} \quad (33)$$

Where $Rodrx_0, Rodry_0$ are the results of entered the initial value of a, b, c into eq. (32). $\Delta Rodrx, \Delta Rodry$ are calculated as follows,

$$\Delta Rodrx = \frac{\partial x}{\partial a}\Delta a + \frac{\partial x}{\partial b}\Delta b + \frac{\partial x}{\partial c}\Delta c + \frac{\partial x}{\partial t_1}\Delta t_1 + \frac{\partial x}{\partial t_2}\Delta t_2 + \frac{\partial x}{\partial t_3}\Delta t_3 \quad (34)$$

$$\Delta Rodry = \frac{\partial y}{\partial a}\Delta a + \frac{\partial y}{\partial b}\Delta b + \frac{\partial y}{\partial c}\Delta c + \frac{\partial y}{\partial t_1}\Delta t_1 + \frac{\partial y}{\partial t_2}\Delta t_2 + \frac{\partial y}{\partial t_3}\Delta t_3 \quad (35)$$

In eq. (34) and (35), $\Delta a, \Delta b, \Delta c, \Delta t_1, \Delta t_2, \Delta t_3$ are the corrections of the initial value of a, b, c, $\frac{\partial x}{\partial a}, \frac{\partial x}{\partial b}, \frac{\partial x}{\partial c}, \frac{\partial y}{\partial a}, \frac{\partial y}{\partial b}, \frac{\partial y}{\partial c}, \frac{\partial x}{\partial t_1}, \frac{\partial x}{\partial t_2}, \frac{\partial x}{\partial t_3}, \frac{\partial y}{\partial t_1}, \frac{\partial y}{\partial t_2}, \frac{\partial y}{\partial t_3}$ are partial derivatives.

The iterative calculation above will be continued until the corrections less than the threshold values.

When observation point number $n > 4$, the results will be calculated by using least squares method. According to literature (Z.G. Zhu, 1995), the results of least squares method can be get from formula (12) as follow,

$$\Delta \mathbf{X} = -(\mathbf{A}^{\mathrm{T}}\mathbf{P}\mathbf{A})^- \mathbf{A}^{\mathrm{T}}\mathbf{P}\mathbf{l} \quad (36)$$

Where $\Delta \mathbf{X} = \left[\Delta a, \Delta b, \Delta c, \Delta t_1, \Delta t_2, \Delta t_3\right]^{\mathrm{T}}$; $\mathbf{A}$ is the coefficient matrix of $\mathbf{X}$ coefficients of formula (34) and (35). $\mathbf{P}$ stands for weight matrix;
$\mathbf{l} = [((Rodrx_0)_1 \quad (Rodry_0)_1 \quad \cdots \quad (Rodrx_0)_n \quad (Rodry_0)_n)^{\mathrm{T}} \quad -(x_1 \quad y_1 \quad \cdots \quad x_n \quad y_n)^{\mathrm{T}}]^T$; n is the number of observation.

3.2.3 Relative navigation based on Rodrigues and spacecraft orbit & attitude information

Considering S_a is the body frame of spacecraft A , S_P is the body frame of spacecraft P , the relation of S_a , S_P and inertial frame S_i can be represented by using Rodrigues as

$$\left.\begin{array}{ccc} S_i & \underline{\mathbf{Rodr}_{Pi}} & S_P \\ S_i & \underline{\mathbf{Rodr}_{Ai}} & S_A \\ S_P & \underline{\mathbf{Rodr}_{AP}} & S_A \end{array}\right\} \tag{37}$$

Thus relative attitude of spacecraft A and P can be described as

$$\mathbf{Rodr}_{AP} = \mathbf{Rodr}_{iP} * \mathbf{Rodr}_{Ai} \tag{38}$$

Where * is Rodrigues multiplication sign.

Hereto, as section **3.1**, the relative attitude and position parameters between objective spacecraft P and camera frame are calculated. Then let these parameters as the initial value of eq. (36). And then the relative attitude and position between active spacecraft A and objective spacecraft P can be determined quickly.

3.2.4 Simulations and analyses

On the basis of the algorithm based-on Rodrigues above, considering the simulation conditions as **3.1.5** section, we can get the rusults as Fig. 11-Fig. 13. In Fig. 11-Fig. 13, (a) stands for the results based on spacecraft orbit & attitude information, (b) stands for the results based on optional value.
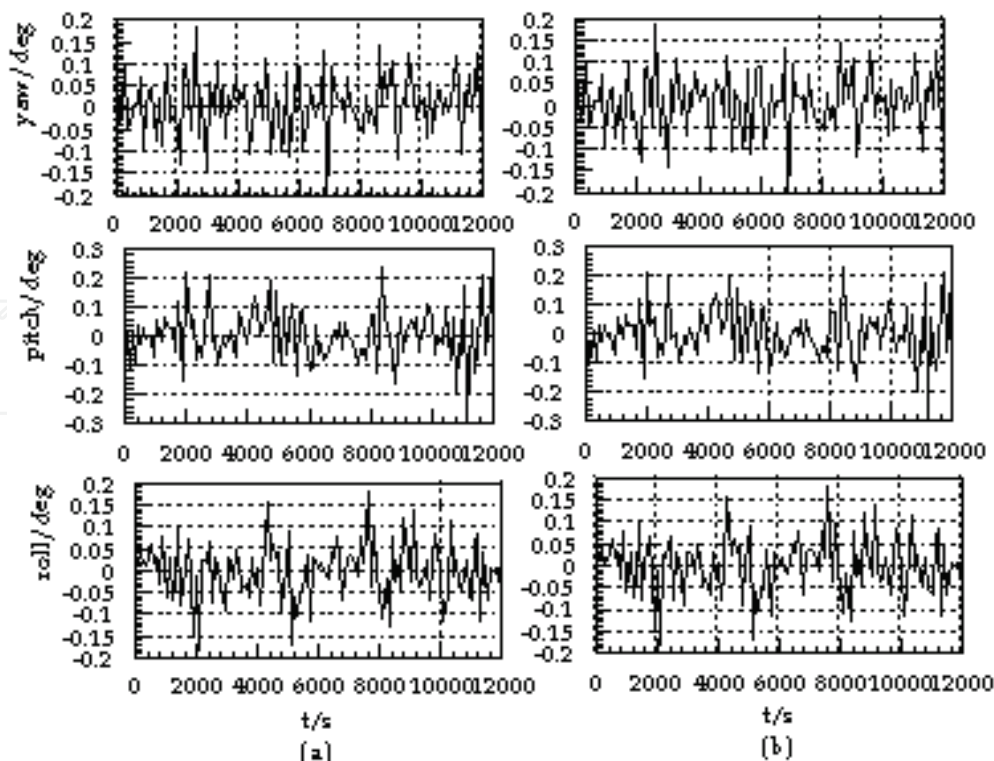

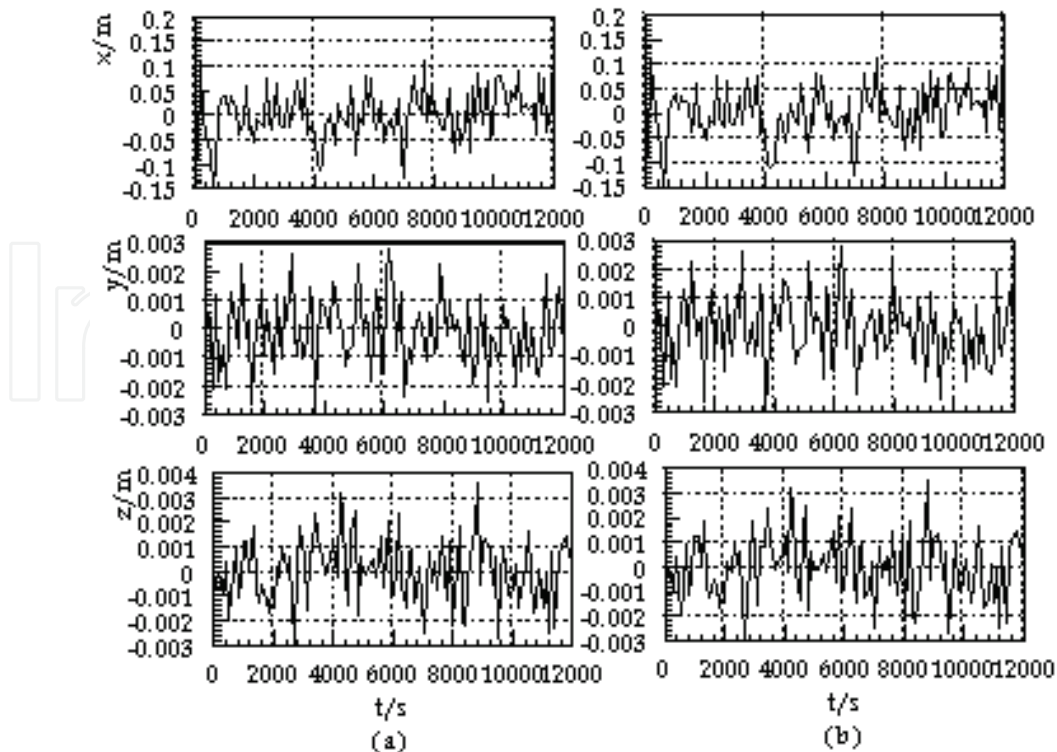Fig. 11. Relative attitude errors of the algorithm based-on Rodrigues

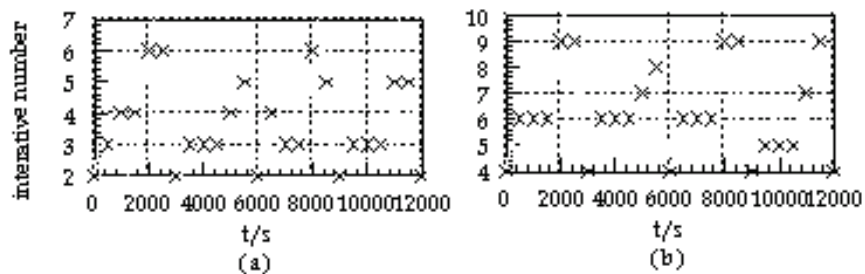Fig. 12. Relative position errors of the algorithm based-on Rodrigues

Fig. 13. Iterative number of calculation of the algorithm based-on Rodrigues

The results of simulation are calculated by using the computer of HP Pavilion Intel (R), Pentium (R) 4, CPU 3.06GHz, 512 MB, the consumable times of method (a) and (b) are 3281 ms and 6344 ms respectively.

From the simulation results, we can see that both two methods of (a) and (b) can get the high and similar precision results, whereas the iterative number of (a) is 1-5, and the iterative number of (b) is 4-9, moreover the consumable times of (a) is about half of the times of (b). All these show that the algorithm (a) is better than (b).

4. Pose and motion and estimation for spacecraft

4.1 Autonomous relative navigation for spacecraft based-on Quaternion and EKF (QEKF)

It is an innovative way to solve some difficult space problems by distributed spacecraft system, which depends on the collaboration each satellite of the system. And these difficult spacecraft problems always can not be solved by one satellite alone. Recent years, many researches about distributed spacecraft system have been developed. And considerable

progresses have been made in space exploration, earth observation and military domain etc (Graeme B. Shaw, 1998; Dr. Kim Luu, 1999; RF Antenna C. Sabol, 1999; H. P. Xu , 2006). But autonomous relative navigation, which is one of key technologies for distributed spacecraft system, and the relative theories need to be studied yet.

In this section, we first introduce how to select the state variable and build the state equations according to C-W equation and quaternion differential equation; Then how to build observation equation according to con-line equation of vision navigation and the state variable is described.

4.1.1 State equation of QEKF

To solve the dynamic estimation problem based on EKF, the state equation must be built. And how to select state variable is introduced here firstly. Since the filter computation time is proportional to the number of state variables, fewer variables are desirable. Based on the approach given by T. J. Broida (1990), J. S. Goddard (1997), Daniël François Malan (2004), thirteen variables are used. In these applications, angle velocity vector is regarded as constant. But in the application of relative navigation for spacecraft, relative angle velocity vector $(\boldsymbol{\omega}_{AP})_b$ is a variable. In this chapter, we can estimate $(\boldsymbol{\omega}_{AP})_b$ in advance, and look it as an input variable of time t . In this way, the number of state variables is reduced, but also the practical problem is solved well. Thus the state variables are three relative position variables $(\Delta x_{PA-O'}, \Delta y_{PA-O'}, \Delta z_{PA-O'})^T$, three relative velocity variables $(\Delta Vx, \Delta Vy, \Delta Vz)^T$ and four relative attitude variables $\Delta \mathbf{Q} = (\Delta q_0, \Delta q_1, \Delta q_2, \Delta q_3)^T$. Where the vector $(\Delta x_{PA-O'}, \Delta y_{PA-O'}, \Delta z_{PA-O'}, \Delta Vx, \Delta Vy, \Delta Vz)^T$ is defined in orbital frame of objective spacecraft, vector $\Delta \mathbf{Q}$ is defined in body frame of active spacecraft A . However, vision relative navigation is based on camera frame, and we must build the relationship of the camera frame, orbital frame and body frame each other. This will be studied in section three.

As mentioned above, the state variable assignment is

$$\mathbf{S} = (\Delta x_{PA-O'}, \Delta y_{PA-O'}, \Delta z_{PA-O'}, \Delta Vx, \Delta Vy, \Delta Vz, \Delta q_0, \Delta q_1, \Delta q_2, \Delta q_3)^T \tag{39}$$

Without disturbances, according to literature (Y. L. Xiao, 2003), the nonlinear continuos equation of $\mathbf{S}$ is

$$f(\mathbf{S}) = \frac{1}{2}\begin{bmatrix} 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4\Omega & 0 & 0 & 0 & 0 \\ 0 & -2\Omega^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 6\Omega^2 & -4\Omega & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\omega_x & -\omega_y & -\omega_z \\ 0 & 0 & 0 & 0 & 0 & 0 & \omega_x & 0 & \omega_z & -\omega_y \\ 0 & 0 & 0 & 0 & 0 & 0 & \omega_y & -\omega_z & 0 & \omega_x \\ 0 & 0 & 0 & 0 & 0 & 0 & \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix}\begin{bmatrix} \Delta x_{PA-O'} \\ \Delta y_{PA-O'} \\ \Delta z_{PA-O'} \\ \Delta Vx \\ \Delta Vy \\ \Delta Vz \\ \Delta q_0 \\ \Delta q_1 \\ \Delta q_2 \\ \Delta q_3 \end{bmatrix} \tag{40}$$

Where Ω is the angle velocity of objective satellite P, $(\omega_x, \omega_y, \omega_z)^T$ is the vector $(\boldsymbol{\omega}_{AP})_b$. From eq. (40), we can get the linearization matrix $\boldsymbol{\Phi}$.

4.1.2 Observation equation of QEKF

From equations (20) and (39), we can see that the state variables we selected are not related with image coordinate directly. So we must build the relationship between them before observation equation is built. Hereinafter, we will talk about the transformation of $(\Delta x_{PA-O'}, \Delta y_{PA-O'}, \Delta z_{PA-O'})^T$ firstly, and then will discuss the transformation of $\Delta \boldsymbol{Q}$.

4.1.2.1 The transformation of $(\Delta x_{PA-O'}, \Delta y_{PA-O'}, \Delta z_{PA-O'})^T$

a. Transform $(\Delta x_{PA-O'}, \Delta y_{PA-O'}, \Delta z_{PA-O'})^T$ from orbital frame defined in objective spacecraft P to inertial frame defined in active spacecraft A as follows

$$\begin{bmatrix} \Delta x_{AP-i} \\ \Delta y_{AP-i} \\ \Delta z_{AP-i} \end{bmatrix} = -\begin{bmatrix} \Delta x_{PA-i} \\ \Delta y_{PA-i} \\ \Delta z_{PA-i} \end{bmatrix} = -\mathbf{L}_{P-iO}\mathbf{L}_{P-OO'}\begin{bmatrix} \Delta x_{PA-O'} \\ \Delta x_{PA-O'} \\ \Delta x_{PA-O'} \end{bmatrix} \tag{41}$$

Where $\mathbf{L}_{P-iO}$ is the transition matrix from orbital frame defined in objective spacecraft P to its inertial frame; $\mathbf{L}_{P-OO'}$ is the transition matrix from second orbital frame defined in objective spacecraft P to its geocentric orbital frame.

b. Transform $(\Delta x_{AP-i} \quad \Delta y_{AP-i} \quad \Delta z_{AP-i})^T$ form inertial frame defined in active spacecraft A to its body frame by the formulation

$$\begin{bmatrix} \Delta x_{AP-b} \\ \Delta y_{AP-b} \\ \Delta z_{AP-b} \end{bmatrix} = \mathbf{L}_{A-bO'}\mathbf{L}_{A-O'O}\mathbf{L}_{A-Oi}\begin{bmatrix} \Delta x_{AP-i} \\ \Delta y_{AP-i} \\ \Delta z_{AP-i} \end{bmatrix} \tag{42}$$

Where $\mathbf{L}_{A-bO'}$ is the transition matrix from second orbital frame defined in active spacecraft A to its body frame; $\mathbf{L}_{A-O'O}$ is the transition matrix from geocentric orbital frame in active spacecraft A to its second orbital frame; $\mathbf{L}_{A-Oi}$ is the transition matrix from inertial frame defined in active spacecraft A to its geocentric orbital frame.

c. Transform $(\Delta x_{AP-b} \quad \Delta y_{AP-b} \quad \Delta z_{AP-b})^T$ from body frame defined in active spacecraft A to camera frame by the formulation as follows

$$\begin{bmatrix} \Delta x_{AP-C} \\ \Delta y_{AP-C} \\ \Delta z_{AP-C} \end{bmatrix} = \mathbf{M}\begin{bmatrix} \Delta x_{AP-b} \\ \Delta y_{AP-b} \\ \Delta z_{AP-b} \end{bmatrix} + \mathbf{T} \tag{43}$$

Where $\mathbf{M}$ is the attitude transition matrix, and $\mathbf{T}$ is the transition matrix from body frame defined in active spacecraft A to camera frame. They can be designed or measured.

Thus the relative relationship between active spacecraft A and objective spacecraft P is built, we can described this relation by the formulation as follows

$$\begin{bmatrix} \Delta x_{AP-C} \\ \Delta y_{AP-C} \\ \Delta z_{AP-C} \end{bmatrix} = -\mathbf{M}\mathbf{L}_{A-bO'}\mathbf{L}_{A-O'O}\mathbf{L}_{A-Oi}\mathbf{L}_{P-iO}\mathbf{L}_{P-OO'} \begin{bmatrix} \Delta x_{PA-O'} \\ \Delta y_{PA-O'} \\ \Delta z_{PA-O'} \end{bmatrix} + \mathbf{T} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}\begin{bmatrix} \Delta x_{PA-O'} \\ \Delta y_{PA-O'} \\ \Delta z_{PA-O'} \end{bmatrix} + \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix} \quad (44)$$

4.1.2.2 The transformation of $\Delta \mathbf{Q}$

We can translate the relative attitude $\mathbf{M}$ into quaternion $\mathbf{Q}_{Cb}$ form according to the transformation between cosine matrix and quaternion.

$$\mathbf{Q}_{Cb} = (q_{Cb-0} \quad q_{Cb-1} \quad q_{Cb-2} \quad q_{Cb-3})^T \quad (45)$$

And we can transform $\Delta \mathbf{Q}$ form body frame defined in active spacecraft A to camera frame by $\mathbf{Q}_{Cb}$ as follows

$$\mathbf{Q_{CP}} = (q_{CP-0} \quad q_{CP-1} \quad q_{CP-2} \quad q_{CP-3})^T = \Delta\mathbf{Q} \circ \mathbf{Q_{Cb}} \quad (46)$$

Substitute $(\Delta x_{CP}, \Delta x_{CP}, \Delta x_{CP})^T$ of eq. (44), $\mathbf{Q}_{CP}$ of eq. (46) for $t_i(i=1,2,3)$ and $a_{ij}(i=1,2,3; j=1,2,3)$ of eq. (20) in turn, observation equation about state variable $\mathbf{S}$ can be represented as

$$\begin{cases} x_i = f\dfrac{l_{11}X_{W_i} + l_{12}Y_{W_i} + l_{13}Z_{W_i} + \Delta x_{AP-C}}{l_{31}X_{W_i} + l_{32}Y_{W_i} + l_{33}Z_{W_i} + \Delta z_{AP-C}} \\ \\ y_i = f\dfrac{l_{21}X_{W_i} + l_{22}Y_{W_i} + l_{23}Z_{W_i} + \Delta y_{AP-C}}{l_{31}X_{W_i} + l_{32}Y_{W_i} + l_{33}Z_{W_i} + \Delta z_{AP-C}} \end{cases} \quad (47)$$

Where

$$\left.\begin{aligned} l_{11} &= q_{CP-0}^2 + q_{CP-1}^2 - q_{CP-2}^2 - q_{CP-3}^2 \\ l_{12} &= 2(q_{CP-1}q_{CP-2} + q_{CP-0}q_{CP-3}) \\ l_{13} &= 2(q_{CP-3}q_{CP-1} - q_{CP-0}q_{CP-2}) \\ l_{21} &= 2(q_{CP-1}q_{CP-2} - q_{CP-0}q_{CP-3}) \\ l_{22} &= q_{CP-0}^2 - q_{CP-1}^2 + q_{CP-2}^2 - q_{CP-3}^2 \\ l_{23} &= 2(q_{CP-2}q_{CP-3} + q_{CP-0}q_{CP-1}) \\ l_{31} &= 2(q_{CP-3}q_{CP-1} + q_{CP-0}q_{CP-2}) \\ l_{32} &= 2(q_{CP-2}q_{CP-3} - q_{CP-0}q_{CP-1}) \\ l_{33} &= q_{CP-0}^2 - q_{CP-1}^2 - q_{CP-2}^2 + q_{CP-3}^2 \end{aligned}\right\} \quad (48)$$

Let

$$\begin{cases} \overline{X} = l_{11}X_{W_i} + l_{12}Y_{W_i} + l_{13}Z_{W_i} + r_{11}\Delta x_{PA-O'} + r_{12}\Delta y_{PA-O'} + r_{13}\Delta z_{PA-O'} + T_1 \\ \overline{Y} = l_{21}X_{W_i} + l_{22}Y_{W_i} + l_{23}Z_{W_i} + r_{21}\Delta x_{PA-O'} + r_{22}\Delta y_{PA-O'} + r_{23}\Delta z_{PA-O'} + T_2 \\ \overline{Z} = l_{31}X_{W_i} + l_{32}Y_{W_i} + l_{33}Z_{W_i} + r_{31}\Delta x_{PA-O'} + r_{32}\Delta y_{PA-O'} + r_{33}\Delta z_{PA-O'} + T_3 \end{cases} \quad (49)$$

Then the formulation (47) can be simplified as

$$\begin{cases} x_i = f\dfrac{\overline{X}}{\overline{Z}} \\[2ex] y_i = f\dfrac{\overline{Y}}{\overline{Z}} \end{cases} \tag{50}$$

Obviously, the equations of eq. (50) are nonlinear about $\mathbf{S}$. And they must be linearized in visual relative navigation estimation based-on EKF. On the basis of the theory of EKF, we can get the linearization matrix $\mathbf{H}_k$ of observation equation.

$$\mathbf{H}_k(\widehat{\mathbf{S}}_k(-)) = \left.\frac{\partial h(\mathbf{S})}{\partial \mathbf{S}}\right|_{\mathbf{S}=\widehat{\mathbf{S}}_k(-)} \tag{51}$$

Where $\mathbf{H}_k$ is a $2i \times n$ matrix, i is the number of feature point, n is the number of state variable.

4.1.3 Simulations and analyses

On the basis of the algorithm above, let the camera focus $f = 50$mm, the objective spacecraft P is a $4.0m \times 4.0m \times 4.0m$ cube, and its body frame coordinates of feature points are respectively $\{-2.25,-8.5,2.25\}$, $\{-2.25,-7.8,-2.25\}$, $\{-2.25,9,2.25\}$, $\{-2.25,7.8,-2.25\}$, $\{-2.25,6.0,-2.25\}$, $\{1.25,7.6,2.25\}$, $\{2.25,-9,2.25\}$, $\{2.25,8.9,2.25\}$. Table 2 lists the initial parameters of the simulations. The initial variance covariance matrix of $\begin{bmatrix} \Delta x & \Delta y & \Delta z \end{bmatrix}^T$ is $0.4\mathbf{I}_{3\times3}$, the variance covariance matrix of $\begin{bmatrix} Vx & Vy & Vz \end{bmatrix}^T$ is $2\times10^{-4}\mathbf{I}_{3\times3}$, the variance covariance matrix of $\Delta\mathbf{Q}$ is $1\times10^{-7}\mathbf{I}_{4\times4}$ ($\mathbf{I}$ is identity matrix). The simulation time is three periods. Fig. 14 is the simulation results.

	spacecraft A	spacecraft P	Initial parameters	
ascending node/deg	0	0	Δ x/m	0.0
inclination of orbit/deg	57.5005	57.5	Δ y/m	0.0
argument of perigee/deg	0	0	Δ z/m	35.25502
excentricity	0.000005	0.0	Vx/(m/s)	0.07518633
semi-major axis/km	7051.000	7051.000	Vy/(m/s)	-0.06561435
time of perigee passage/s	0	0	Vz/(m/s)	0.0
yaw/deg	0.5	0.4	Δ q$_0$	1
pitch/deg	0.2	0.2	Δ q$_1$	0
roll/deg	0.5	0.5	Δ q$_2$	0
yaw angle velocity/(deg/s)	5×10^{-7}	5×10^{-7}	Δ q$_3$	0
pitch angle velocity/(deg/s)	5×10^{-7}	5×10^{-7}		
roll angle velocity/(deg/s)	5×10^{-7}	5×10^{-7}		

Table 2. The initial parameters of the simulations

It is not intuitionistic to represent the attitude results by quaternion, yet the attitude results are described as their Euler form.
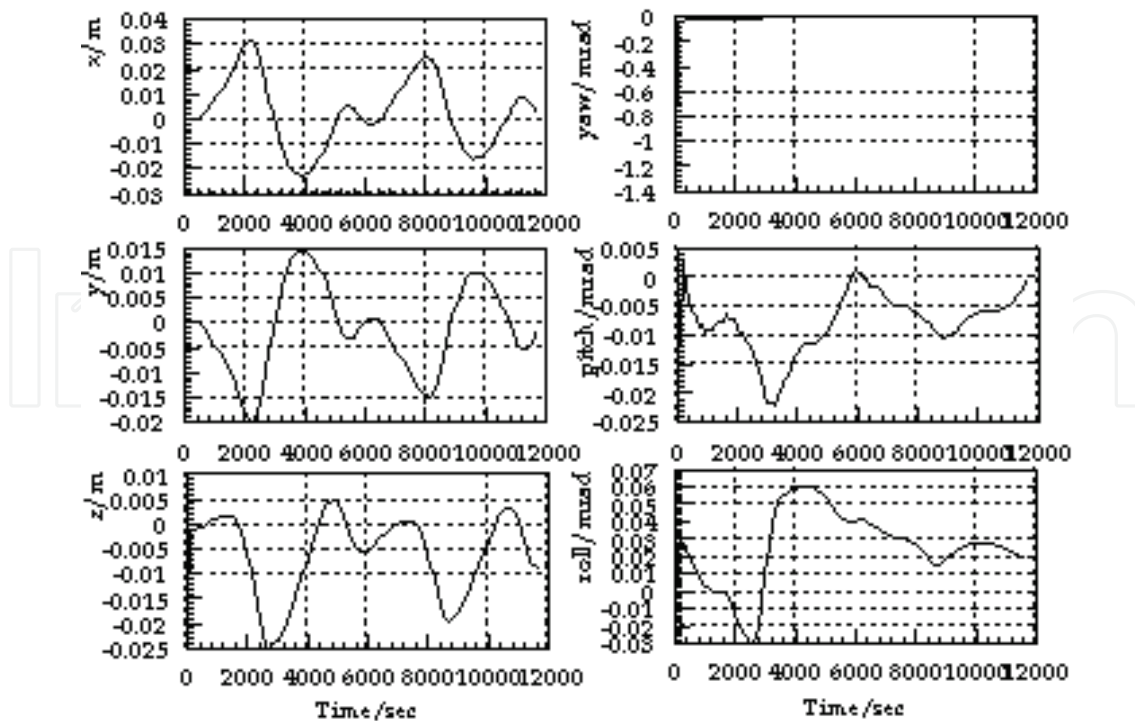
Fig. 14. Relative position and attitude errors of QEKF

From Fig. 14, we can see that this algorithm is convergent. And Fig. 14 shows the relative position x error is within -0.03m to 0.04m; y error is within -0.02m to 0.015m; z error is within -0.025m to 0.01m; relative yaw angle error is about -1.4 to zero mrad when the simulation time is about 400 seconds before, but after 400 seconds relative yaw angle error tend to zero; relative pitch error is within -0.025 to 0.005 mrad; relative roll angle error is within -0.03 mrad to 0.07 mrad.

4.2 Autonomous relative navigation for spacecraft based-on Rodrigues and EKF (REKF)

4.2.1 State equation of REKF

For Rodrigues with no redundance valvate to represent the attitude. So here we use Rodrigues $\mathbf{\Phi}_{AP-b} = \begin{pmatrix} \Phi_{AP-b-1} & \Phi_{AP-b-2} & \Phi_{AP-b-3} \end{pmatrix}^T$ instead of quaternion. As section **4.1,** we can select the state variable assignment as eq. (52).

$$f(\mathbf{S}) = \frac{1}{2} \begin{bmatrix} 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4\Omega & 0 & 0 & 0 \\ 0 & -2\Omega^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 6\Omega^2 & -4\Omega & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & Rodr_{11} & Rodr_{12} & Rodr_{13} \\ 0 & 0 & 0 & 0 & 0 & 0 & Rodr_{21} & Rodr_{22} & Rodr_{23} \\ 0 & 0 & 0 & 0 & 0 & 0 & Rodr_{31} & Rodr_{32} & Rodr_{33} \end{bmatrix} \begin{bmatrix} \Delta x_{PA-O'} \\ \Delta y_{PA-O'} \\ \Delta z_{PA-O'} \\ \Delta Vx \\ \Delta Vy \\ \Delta Vz \\ \Delta a \\ \Delta b \\ \Delta c \end{bmatrix} \quad (52)$$

Where Ω is the angle velocity of objective satellite P, $Rodr_{11} = 2\Delta a\omega_x + \Delta b\omega_y + \Delta c\omega_z$, $Rodr_{12} = \Delta a\omega_y + \omega_z$, $Rodr_{13} = -\omega_y + \Delta a\omega_z$, $Rodr_{21} = \Delta b\omega_x - \omega_z$, $Rodr_{22} = \Delta a\omega_x + 2\Delta b\omega_y + \Delta c\omega_z$, $Rodr_{23} = \omega_x + \Delta b\omega_z$, $Rodr_{31} = \Delta c\omega_x + \omega_y$, $Rodr_{32} = -\omega_x + \Delta c\omega_y$, $Rodr_{33} = \Delta a\omega_x + \Delta b\omega_y + 2\Delta c\omega_z$, $(\omega_x, \omega_y, \omega_z)^T$ is the vector $(\boldsymbol{\omega}_{AP})_b$.

From eq. (52), we can get the linearization matrix $\boldsymbol{\Phi}$.

4.2.2 Observation equation of REKF

The structural observation equation of REKF is simular to the QEKF's. Here we mainly discuss the transformation of $\boldsymbol{\Phi}_{AP-b} = \begin{bmatrix} \Phi_{AP-b-1} & \Phi_{AP-b-2} & \Phi_{AP-b-3} \end{bmatrix}^T$. Transform $\mathbf{M}$ of eq. (44) into Rodrigues $\boldsymbol{\Phi}_{Cb} = \begin{bmatrix} \Phi_{Cb-1} & \Phi_{Cb-2} & \Phi_{Cb-3} \end{bmatrix}^T$, transform $\boldsymbol{\Phi}_{AP-b}$ form body frame of active spacecraft to the camera frame as follows

$$\boldsymbol{\Phi}_{CP} = \boldsymbol{\Phi}_{AP-b} * \boldsymbol{\Phi}_{Cb} \tag{53}$$

Substitute $(\Delta x_{CP}, \Delta x_{CP}, \Delta x_{CP})^T$ of eq. (44), $\boldsymbol{\Phi}_{CP}$ of eq. (53) for $t_i (i=1,2,3)$ and $a_{ij} (i=1,2,3; j=1,2,3)$ of eq. (20) in turn, observation equation about state variable $\mathbf{S}$ can be represented as

$$\begin{cases} x_i = f \dfrac{l_{11}X_{W_i} + l_{12}Y_{W_i} + l_{13}Z_{W_i} + \Delta x_{AP-C}}{l_{31}X_{W_i} + l_{32}Y_{W_i} + l_{33}Z_{W_i} + \Delta z_{AP-C}} \\[3mm] y_i = f \dfrac{l_{21}X_{W_i} + l_{22}Y_{W_i} + l_{23}Z_{W_i} + \Delta y_{AP-C}}{l_{31}X_{W_i} + l_{32}Y_{W_i} + l_{33}Z_{W_i} + \Delta z_{AP-C}} \end{cases} \tag{54}$$

Where

$$\left. \begin{aligned} l_{11} &= (1 + \Phi_{CP-1}^2 - \Phi_{CP-2}^2 - \Phi_{CP-3}^2) / (1 + \Phi^2) \\ l_{12} &= 2(\Phi_{CP-1}\Phi_{CP-2} + \Phi_{CP-3}) / (1 + \Phi^2) \\ l_{13} &= 2(\Phi_{CP-1}\Phi_{CP-3} - \Phi_{CP-2}) / (1 + \Phi^2) \\ l_{21} &= 2(\Phi_{CP-1}\Phi_{CP-2} - \Phi_{CP-3}) / (1 + \Phi^2) \\ l_{22} &= (1 - \Phi_{CP-1}^2 + \Phi_{CP-2}^2 - \Phi_{CP-3}^2) / (1 + \Phi^2) \\ l_{23} &= 2(\Phi_{CP-2}\Phi_{CP-3} + \Phi_{CP-1}) / (1 + \Phi^2) \\ l_{31} &= 2(\Phi_{CP-1}\Phi_{CP-3} + \Phi_{CP-2}) / (1 + \Phi^2) \\ l_{32} &= 2(\Phi_{CP-2}\Phi_{CP-3} - \Phi_{CP-1}) / (1 + \Phi^2) \\ l_{33} &= (1 - \Phi_{CP-1}^2 - \Phi_{CP-2}^2 + \Phi_{CP-3}^2) / (1 + \Phi^2) \end{aligned} \right\} \tag{55}$$

The linear method of eq. (54) is simular to **4.1.2.2** section.

4.2.3 Simulations and analyses

On the basis of the theories of REKF, considering the simulation conditions as **4.1.3**, and let $\boldsymbol{\Phi}_{AP-b} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$, the initial variance covariance matrix of $\boldsymbol{\Phi}_{AP-b}$ is $1 \times 10^{-7} \mathbf{I}_{3\times3}$, we get the simulation results as Fig. 15.

From Fig. 15, we can see that this algorithm is convergent. And Fig. 15 shows the relative position x error is within -0.02m to 0.03m; y error is within -0.015m to 0.01m; z error is within -0.03m to 0.01m; relative yaw angle error is about -0.7 to zero mrad when the simulation time is about 400 seconds before, but after 400 seconds relative yaw angle error tend to zero; relative pitch error is about -0.16 mrad when the simulation time is about 400 seconds before, but after 400 seconds relative pitch error is within -0.08 to 0.02 mrad; relative roll angle error is abot -0.12 mrad when the simulation time is about 400 seconds before, but after 400 seconds relative roll error is within -0.04 to 0.04 mrad.
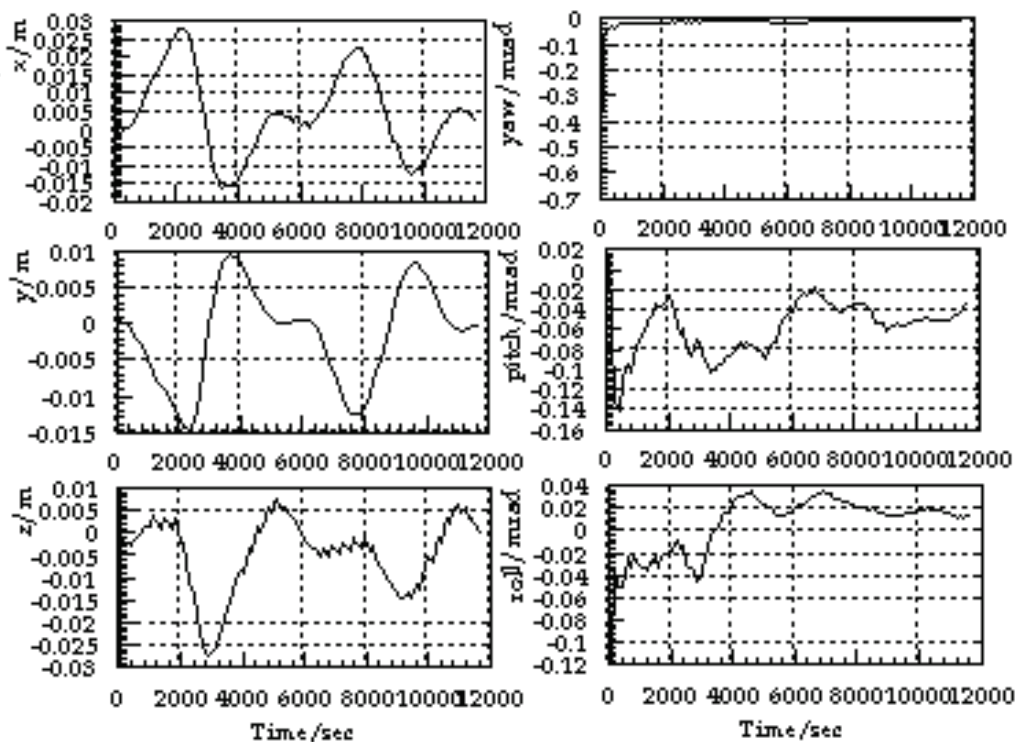
Fig. 15. Relative position and attitude errors of REKF

4.3 Autonomous relative navigation for spacecraft based-on Dual Quaternion and EKF (DQEKF)

4.3.1 How to depict a 3-D line and its transformation with dual quaternion

According to geometric algebra theory (D. Hestenes, G. Sobczyk,1984), dual quaternion can be found in special 4D even subalgebra of $\Xi_{3,0,1}$ and is spanned by the following basis

$$\underbrace{1}_{scalar} \quad \underbrace{\mathbf{e}_2\mathbf{e}_3, \mathbf{e}_3\mathbf{e}_1, \mathbf{e}_1\mathbf{e}_2, \mathbf{e}_4\mathbf{e}_1, \mathbf{e}_4\mathbf{e}_2, \mathbf{e}_4\mathbf{e}_3}_{6 bivectors} \quad \underset{unit \quad pseudoscalar}{\mathbf{I}} \tag{56}$$

Since a rigid motion consists of the transformations rotation and translation according to Euler theorem, a simple rotor in its Euler representation for a rotation by an angle θ

$$\mathbf{R} = q_0 + q_1\mathbf{e}_2\mathbf{e}_3 + q_2\mathbf{e}_3\mathbf{e}_1 + q_3\mathbf{e}_1\mathbf{e}_2 = q_0 + \mathbf{q} = \cos(\theta/2) + \sin(\theta/2)\mathbf{n} = q_0 + q_s\mathbf{n} \tag{57}$$

Where $\mathbf{n}$ is the unit 3D bivector of the rotation-axis spanned by the bivector basis $\mathbf{e}_2\mathbf{e}_3$, $\mathbf{e}_3\mathbf{e}_1$, $\mathbf{e}_1\mathbf{e}_2$ and $q_0, q_s (s = 1, 2, 3) \in R$.

In $\Xi_{3,0,1}$, a translation $\mathbf{t}$ is represented by a spinor $\mathbf{T}$. Thus applying $\mathbf{T}$ from the left and its conjugated from the right to the $\mathbf{R}$ in eq. (57), we can get the modified rotor

$$\mathbf{M} = \mathbf{TRT}^* = (1 + \mathbf{I}\frac{\mathbf{t}}{2})(q_0 + \mathbf{q})(1 - \mathbf{I}\frac{\mathbf{t}}{2}) = q_0 + \mathbf{q} + \mathbf{I}(\mathbf{q} \wedge \mathbf{t}) \tag{58}$$

As we know a 3D line $\mathbf{l}_a$ can be represented by PlÜcker coordinate

$$\breve{\mathbf{l}}_a = \mathbf{l}_a + \varepsilon\mathbf{m}_a \tag{59}$$

Hence the Euclidean transformation of the 3D line $\mathbf{l}_a$ by the modified rotor $\mathbf{M}$ can be represented

$$\breve{\mathbf{l}}_b = \mathbf{l}_b + \varepsilon\mathbf{m}_b = \mathbf{M}\breve{\mathbf{l}}_a\mathbf{M}^* \tag{60}$$

4.3.2 State equation of DQEKF
The structural state equation of DQEKF is simular similar to the QEKF's. It is unnecessary to go into details.

4.3.3 Observation equation of DQEKF
On the basis of **4.3.1**, here we will build the observation equations according to the observation valuable. Fig.16 shows the relation of projective line and projective plane with space object frame.
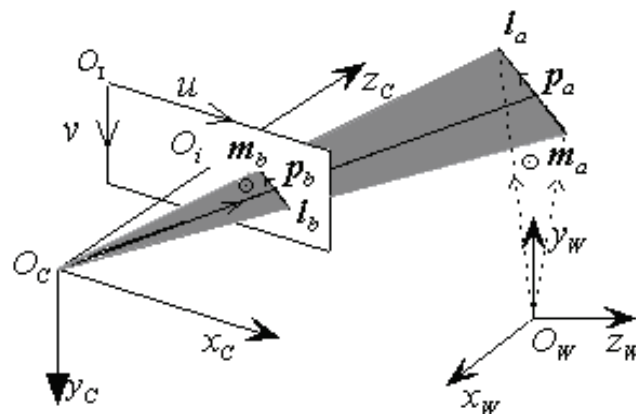
Fig. 16. Relation of projective line and projective plane with space object frame

$O_W - x_W y_W z_W$ stands for object frame, i.e. body frame of passive spacecraft in this paper; $O_C - x_C y_C z_C$ stands for camera frame; $O_i - x_i y_i$ stands for image frame. In Fig.16, $\mathbf{l}_a$ stands for a 3D object line and its projective line is $\mathbf{l}_b$. We can represent the transformation between $\mathbf{l}_a$ and $\mathbf{l}_b$ as formulation (60). But how to calculate the translator $\mathbf{M}$ using image coordinates and object coordinates? In order to process the data simply, first we give a definition as follow.

Definition: Feature line point is the intersection point of the projective line $\mathbf{l}_b$ and a perpendicular line passing though the origin O_i, it is unique.

In camera frame, the projective plane, which is shown in grey, can be represented as

$$m_{bx}x_C + m_{by}y_C + m_{bz}z_C = 0 \tag{61}$$

Where $(m_{bx} \quad m_{by} \quad m_{bz})^T = \mathbf{m}_b$.

From Fig.16, we can see that the projective line $\mathbf{l}_b$ lies in either projective plane or image plane. When $z_C = f$ (f is the focus of the camera), the equation of the projective line $\mathbf{l}_b$ lies in image plane can be described as

$$m_{bx}x_i + m_{by}y_i + m_{bz}f = 0 \tag{62}$$

Thus we can get the vector $\mathbf{m}_{bP}$ containing the feature line point of $\mathbf{l}_b$ normal to the projective plane

$$\mathbf{m}_{bP} = \frac{f}{\sqrt{m_{bx}^2 + m_{by}^2}}[m_{bx} \quad m_{by} \quad m_{bz}]^T \tag{63}$$

Then the feature line point of $\mathbf{l}_b$ coordinates is described as

$$\begin{cases} x_{iP} = -f\dfrac{m_{bx}m_{bz}}{m_{bx}^2 + m_{by}^2} \\[2mm] y_{iP} = -f\dfrac{m_{by}m_{bz}}{m_{bx}^2 + m_{by}^2} \end{cases} \tag{64}$$

$\mathbf{m}_b$ can be calculated according to literature (LI K. Z, 2007).

Obviously, eq. (64) are nonlinear equations about $\mathbf{S}$. And they must be linearized in visual relative navigation estimation based-on EKF. The partial differential equations to eq. (64) about $\mathbf{S}$ can be represented as follows

$$\left.\begin{aligned} \frac{\partial x_{iP}}{\partial \mathbf{S}} &= \frac{\partial x_{iP}}{\partial m_{bx}}\frac{\partial m_{bx}}{\partial \mathbf{S}} + \frac{\partial x_{iP}}{\partial m_{by}}\frac{\partial m_{by}}{\partial \mathbf{S}} + \frac{\partial x_{iP}}{\partial m_{bz}}\frac{\partial m_{bz}}{\partial \mathbf{S}} \\ \frac{\partial y_{iP}}{\partial \mathbf{S}} &= \frac{\partial y_{iP}}{\partial m_{bx}}\frac{\partial m_{bx}}{\partial \mathbf{S}} + \frac{\partial y_{iP}}{\partial m_{by}}\frac{\partial m_{by}}{\partial \mathbf{S}} + \frac{\partial y_{iP}}{\partial m_{bz}}\frac{\partial m_{bz}}{\partial \mathbf{S}} \end{aligned}\right\} \tag{65}$$

From eq. (57)-(60), we can get the equation as follows

$$\mathbf{l}_b + \varepsilon\mathbf{m}_b = (\mathbf{q} + \varepsilon\frac{\mathbf{t}}{2}\mathbf{q})(\mathbf{l}_a + \varepsilon\mathbf{m}_a)(\mathbf{q}^* + \varepsilon\frac{1}{2}\mathbf{q}^*\mathbf{t}^*) = \mathbf{q}\mathbf{l}_a\mathbf{q}^* + \varepsilon(\frac{1}{2}\mathbf{q}\mathbf{l}_a\mathbf{q}^*\mathbf{t}^* + \mathbf{q}\mathbf{m}_a\mathbf{q}^* + \frac{1}{2}\mathbf{t}\mathbf{q}\mathbf{l}_a\mathbf{q}^*) \tag{66}$$

And from eq. (66), we can get

$$\mathbf{m}_b = \frac{1}{2}\mathbf{q}\mathbf{l}_a\mathbf{q}^*\mathbf{t}^* + \mathbf{q}\mathbf{m}_a\mathbf{q}^* + \frac{1}{2}\mathbf{t}\mathbf{q}\mathbf{l}_a\mathbf{q}^* \tag{67}$$

According to the theories of quaternion, we can described the eq. (67) as follows

$$\mathbf{m}_b = \frac{1}{2}\bar{\mathbf{M}}_{\mathbf{t}^*}\overset{+}{\mathbf{M}}_{\mathbf{q}}\bar{\mathbf{M}}_{\mathbf{q}^*}\mathbf{l}_a + \overset{+}{\mathbf{M}}_{\mathbf{q}}\bar{\mathbf{M}}_{\mathbf{q}^*}\mathbf{m}_a + \frac{1}{2}\overset{+}{\mathbf{M}}_{\mathbf{t}}\overset{+}{\mathbf{M}}_{\mathbf{q}}\bar{\mathbf{M}}_{\mathbf{q}^*}\mathbf{l}_a \tag{68}$$

$$\text{Let } \mathbf{R_M} = \overset{+}{\mathbf{M}_\mathbf{q}} \overset{-}{\mathbf{M}_{\mathbf{q}^*}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 + q_0q_3) & 2(q_3q_1 - q_0q_2) \\ 0 & 2(q_1q_2 - q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 + q_0q_1) \\ 0 & 2(q_3q_1 + q_0q_2) & 2(q_2q_3 - q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \quad (69)$$

$$\text{Where } \begin{bmatrix} q_0 & q_1 & q_2 & q_3 \end{bmatrix}^T = \begin{bmatrix} q_{CP-0} & q_{CP-1} & q_{CP-2} & q_{CP-3} \end{bmatrix}^T,$$

$$\mathbf{M_{t,t^*}} = 2 \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -t_3 & t_2 \\ 0 & t_3 & 0 & -t_1 \\ 0 & -t_2 & t_1 & 0 \end{bmatrix} \quad (70)$$

Eq. (68) can be simplified as follows

$$\mathbf{m}_b = \mathbf{R_M}\mathbf{m}_a + \frac{1}{2}\mathbf{M_{t,t^*}}\mathbf{R_M}\mathbf{l}_a \quad (71)$$

So the partial differential equations of $\mathbf{S}$ can be calculated simply from eq. (71).

4.3.4 Simulations and analyses
On the basis of the theories of DQEKF, considering the simulation conditions as **4.1.3** section, we can get the results as Fig. 17.
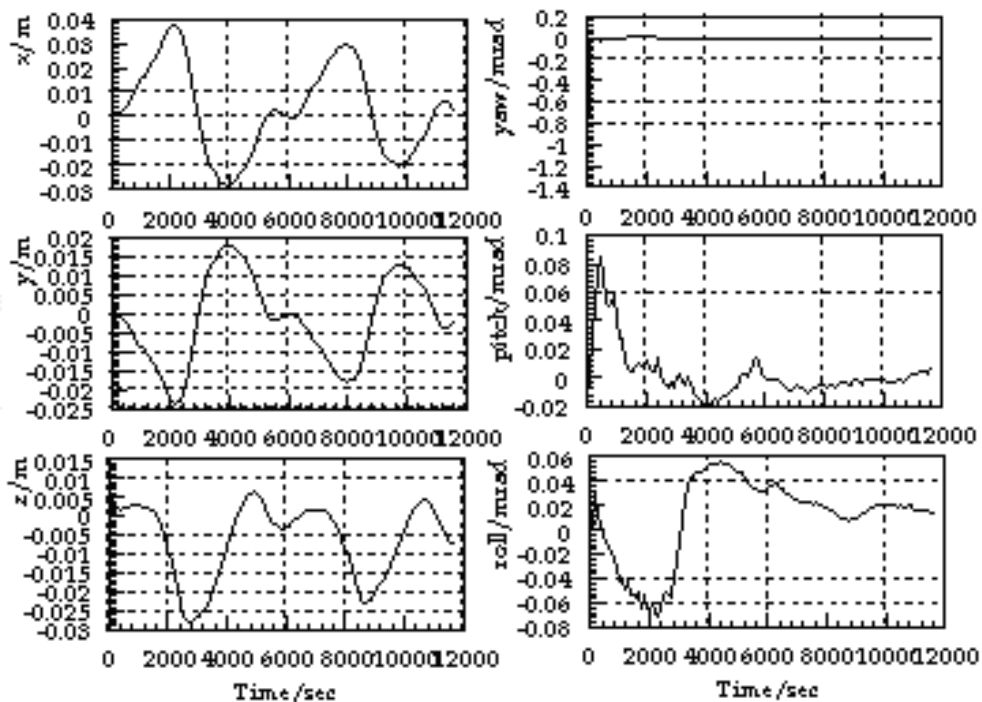
Fig. 17. Relative position and attitude errors of DQEKF

From Fig. 17, we can see that this algorithm is convergent. And Fig. 17 shows the relative position x error is within -0.03m to 0.04m; y error is within -0.025m to 0.02m; z error is within -0.03m to 0.015m; relative yaw angle error is about -1.4 to zero mrad when the simulation time is about 400 seconds before, but after 400 seconds relative yaw angle error tend to zero; relative pitch error is about 0.1 mrad when the simulation time is about 400 seconds before, but after 400 seconds relative pitch error is within -0.02 to 0.04 mrad; relative roll angle error is within -0.08 to 0.06 mrad.

4.4 Autonomous relative navigation for spacecraft based-on Dual Rodrigues-Quaternion and EKF (DRQEKF)
4.4.1 State equation of DRQEKF

Compare with section **4.3**, here we use Rodrigues $\mathbf{\Phi}_{AP-b} = \left(\Phi_{AP-b-1} \quad \Phi_{AP-b-2} \quad \Phi_{AP-b-3} \right)^{T}$ instead of quaternion. Thus the structural state equation of DRQEKF is simular similar to section **4.2.1**. It is unnecessary to go into details.

4.4.2 Observation equation of DRQEKF
For attitude matrix has a unique attribute, $\mathbf{R}_{M}$ of eq. (69) can also be represented as

$$\mathbf{R_{M}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & l_{11} & l_{12} & l_{13} \\ 0 & l_{21} & l_{22} & l_{23} \\ 0 & l_{31} & l_{32} & l_{33} \end{bmatrix} \tag{72}$$

$l_{ij}(i=1,2,3; j=1,2,3)$ of eq. (72) equal to l_{ij} of eq. (55).

So the partial differential equations to eq. (71) about $\Phi_{AP-b-i}(i=1,2,3)$ can be represented as follows

$$\frac{\partial \mathbf{m}_{b}}{\partial \Phi_{AP-b-i}} = \frac{\partial \mathbf{R_{M}}}{\partial \Phi_{AP-b-i}} \mathbf{m}_{a} + \frac{1}{2} \mathbf{M}_{t,t^{*}} \frac{\partial \mathbf{R_{M}}}{\partial \Phi_{AP-b-i}} \mathbf{l}_{a}, (i=1,2,3) \tag{73}$$

The other partial differential of the state valuable parameters can be deduced as section **4.3.3**.

4.4.3 Simulations and analyses
On the basis of the theories of DRQEKF, considering the simulation conditions as **4.1.3** and **4.1.3** sections, we can get the results as Fig. 18.
From Fig. 18, we can see that this algorithm is convergent. And Fig. 18 shows the relative position x error is within -0.02m to 0.03m; y error is within -0.02m to 0.01m; z error is within -0.03m to 0.01m; relative yaw angle error is about -0.5 to zero mrad when the simulation time is about 400 seconds before, but after 400 seconds relative yaw angle error tend to zero; relative pitch error is about -0.1 mrad when the simulation time is about 400 seconds before, but after 400 seconds relative pitch error is within -0.08 to 0.02 mrad; relative roll angle error is within -0.08 to 0.08 mrad.
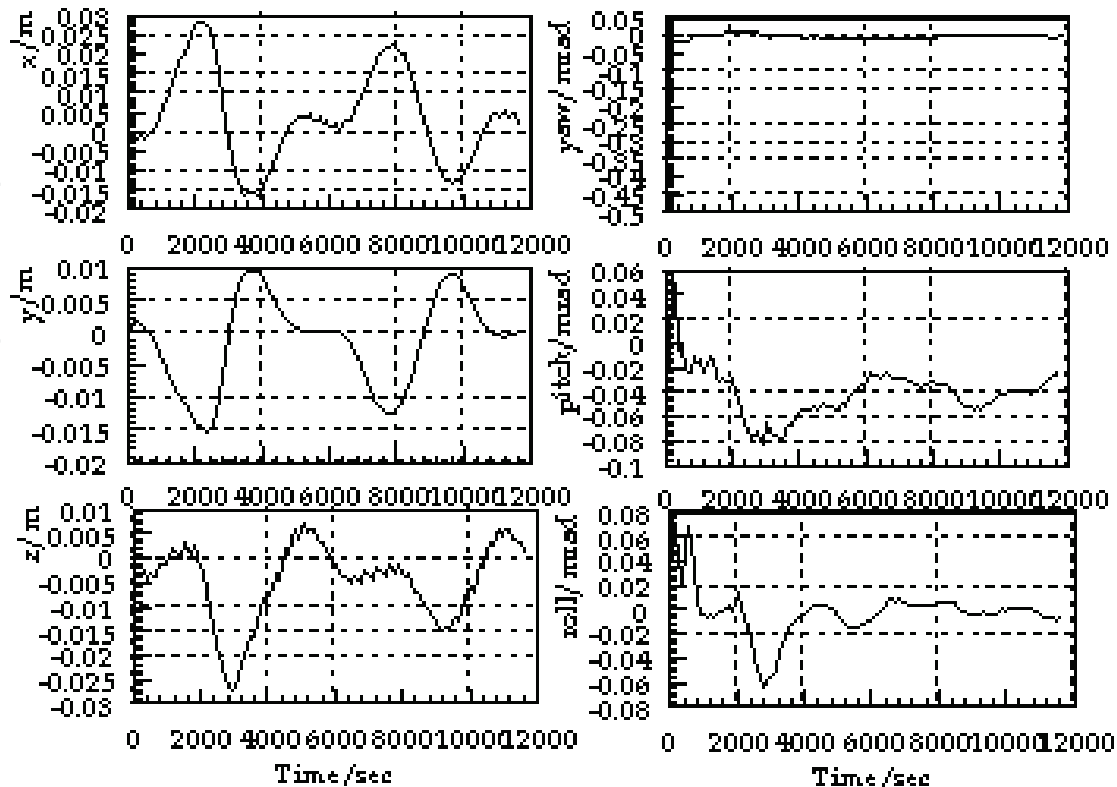
Fig. 18. Relative position and attitude errors of DRQEKF

5. Conclusion

In section **2**, a feature extraction algorithm based-on dynamically structured element is proposed. The simulation results show that this algorithm is an accurate valid method in feature extraction for distributed spacecraft system.

In section **3**, we propose the relative position and pose algorithms based on quaternion/Rodrigues and the orbit & attitude information of the spacecrafts. The algorithms reduces the Jacobian matrix rank by using quaternion/Rodrigues. The simulations show that the iterative numbers of this algorithm are reduced when the orbit & attitude information of the spacecrafts has been used. The algorithm based-on Rodrigues is better than quaternion's when attitude angles are smaller. This thought is valuable for the similar applications based on least squares method.

In section 4, four pose and motion estimation algorithms are proposed. And they can meet the high requirement of spacecraft. QEKF and REKF are based-on point observations, and DQEKF and DRQEKF are based-on line observations. The results of simulation are calculated by using the computer of HP Pavilion Intel (R), Pentium (R) 4, CPU 3.06GHz, 512 MB, the consumable times of method QEKF, REKF, DQEKF and DRQEKF are 6688 ms, 4187 ms, 12437 ms and 7141 ms respectively. So if the observations are points, REKF is proposed when the attitude angles are smaller; if the observations are lines, DRQEKF is proposed when the attitude angles are smaller.

In our future work, the theories in SE reconstruction and relative position and attitude combination will be further studied, and the corresponding simulations about them will be also researched. And we will consider the disturbance factor of satellites in order to improve the practicability of the static forecast and the pose and motion estimation algorithms.

6. Acknowledgment

7. References

D.Mehrholz, L.Leushacke, W.Flury etc (2002). Detecting, tracking and imaging space debris. *ESA bulletin*, February 2002, 109, pp: 128-134

Ben Iannotta (2002). DART Aims at Space Rendezvous. *Aerospace America*, March 2005, pp. 26-30

Richard P. Kornfeld, Robert L (2002). Bunker, Gordy C. Cucullu, et al. New Millennium ST6 Autonomous Rendezvous Experiment (ARX). IEEEAC paper #1114, Updated September 19, 2002, pp. 1-12

LiYingju，LiHuifeng，NingWei (2006). Rendezvous and Docking Technologies－Present and Future. *Sciencepaper Online, http://www.paper.edu.cn*. pp: 1-5

Michael A. Dornheim (2006). Orbital Express to Test Full Autonomy for On-Orbit Service. *http://www.aviationweek.com/aw/generic/story_generic.jsp?channel=awst&id=news/aw06 0506p1.xml]*

Joseph W. Evans, Tom A. Mulder (2006). Autonomous Rendezvous Guidance and Navigation for Orbital Express and Beyond. *16th AAS/AIAA Space Flight Mechanics Conference*, Tampa, Florida, January 22-26, 2006, Paper AAS 06-194

Richard T. Howard, Andrew F. Heaton, Robin M. Pinson etc (2008). The Advanced Video Guidance Sensor: Orbital Express and the Next Generation. *http://ntrs.nasa.gov /archive/nasa/casi.ntrs.nasa.gov /20080015656_2008015631.pdf*, pp : 1-8

Isao Kawano, Masaaki Mokuno, Toru Kasai, Takashi Suzuki (1999).Result and Evaluation of Autonomous Rendezvous Docking Experiment of ETS-VII. AlAA-99-4073

Yoshihiko Torano (2010). H-IIA Launch Vehicle. *http://www.jaxa.jp/pr/brochure/pdf/01 /rocket03.pdf*, pp :1-4

Gianni Casonato, Giovanni B. Palmerini (2004). Visual Techniques Applied to the ATV/ISS Rendez-Vous Monitoring. *IEEE Aerospace Conference Proceedings*, pp. 613-625

Wang Guangjun, Tian Jinwen, Liu jian (2004). Infrared small objects detection based on local entropy method, *Infrared and laser engineering*, Vol.29 No.4, pp. 26-29

Li Guokuan, Peng Jiaxion (2000). Infrared Imaging Dim Target Detection Based on Wavelet Transformation, *J. Huazhong Univ. of Sci. & Tech.*, 28 (5), pp. 69-71

H. P. Xu , J. Chen, B. F. Wang , Y. Q. Zhou (2006). Quick method of distributed small satellite synthetic aperture radar single-look complex image simulation. *Journal of Beijing University of Aeronautics and Astronautics*, pp : 445-449

Richard Alan Peters Ⅱ (1995). A new algorithm for image noise reduction using mathematical morphology. *IEEE Translation on Imaging Processing*, 4 (3): pp : 554-568

Joonki Paik, Cheolha P.Lee, and Mongi A. Abidi (2002). *Image processing-based mine detection techniques: A review. Subsurface Sensing Technologies and Applications*, 3 (3): pp :203-252

Ulisses Braga-Neto, Manish Choudhary and Johan Goutsias (2003). Automatic target detection and tracking in forward–looking infrared image sequences using morphological connected operators. *Journal of Electronic Imaging*, 3: pp : 1-22.

Cui Yi (2002). Imaging processing and analysis-mathematical morphology methods and applications. Beijing : *Science Press*, pp : 1-157

Y. L. Xiao (2003). Foundation of Flight Dynamics-Modeling of Aerospace Vehicle Motion. *BeiHang University Press*, pp : 1-80

Schwab A. L (2002). Quaternions, Finite Rotation and Euler Parameters [EB/OL]. *http://tam.cornell.edu/~als93/quaternion.pdf*, pp :1-4

Z.G. Zhu, H. Sun, B. G. Cui (1995). Foundation of Photogrammetry. *Mapping Press of China*, 185-190 (In Chinese)

S. D. Ma, Z. Y. Zhang (1998). Computer Vision－Foundation of Calculation and Algorithm. *Science Press*, PP : 20-36 (In Chinese)

G. J. Zhang (2005). Machine Vision. *Science Press of China*, pp : 14-32 (In Chinese)

Marc Pollefeys (2002). Visual 3D Modeling from Images. *http://www.cs.unc.edu/~marc /tutorial.pdf*, pp :8-100

Graeme B. Shaw (1998). The Generalized information network analysis methodology for distributed satellite systems [Ph.D Thesus]. *MIT*, pp: 12-58

Dr. Kim Luu, Mr. Maurice Martin, Dr. Mike Stallard, et al. (1999). University Nanosatellite Distributed Satellite Capabilities to support TechSat 21. *AIAA/USU SMALL SATELLITE CONFERENCE, LOGAN UT*, pp : 23-26

RF Antenna C. Sabol, R. Burns, and C. McLaughlin (1999). Formation Flying Design and Evolution. *AAS/AIAA Spaceflight Mechanics Meeting*, Breckenridge, CO, 7-10 February 1999, AAS paper, pp : 99-121

T. J. Broida, S.Chandrashekhar, R. Chaellappa (1990). Recursive 3-D motion estimation from a monocular image sequence. *IEEE Transactions on Aerospace and Electronic Systems*, 26 (4), pp : 639-656

J. S. Goddard (1997). Pose and Motion Estimation from Vision Using Dual Quaternion-Based Extended Kalman Filtering [Ph. D Thesis]. *The University of Tennessee*, Knoxville, pp : 20-60

Daniël François Malan (2004). 3D Tracking between Satellites using Monocular Computer Vision [Master Thesis]. Department of Electrical & Electronic Engineering, *University of Stellenbosch*, pp : 35-88

D. Hestenes, G. Sobczyk (1984). Clifford Algebra to Geometric calculus. D. *Reidel Publishing Company*, Holland, pp: 45-78

LI K. Z., Yuan J. P, Yue X. K. , FANG Q. (2007). Autonomous Navigation Algorithm for
       Spacecrafts Based-on Dual Quaternion, *Second International Conference on Space
       Information Technology*, Proc. SPIE 6795, 3K1-3K6

Advances in Spacecraft Technologies

Edited by Dr Jason Hall

The development and launch of the first artificial satellite Sputnik more than five decades ago propelled both the scientific and engineering communities to new heights as they worked together to develop novel solutions to the challenges of spacecraft system design. This symbiotic relationship has brought significant technological advances that have enabled the design of systems that can withstand the rigors of space while providing valuable space-based services. With its 26 chapters divided into three sections, this book brings together critical contributions from renowned international researchers to provide an outstanding survey of recent advances in spacecraft technologies. The first section includes nine chapters that focus on innovative hardware technologies while the next section is comprised of seven chapters that center on cutting-edge state estimation techniques. The final section contains eleven chapters that present a series of novel control methods for spacecraft orbit and attitude control.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

INTECH
open science | open minds