

# A High-accuracy Extraction Algorithm of Planet Centroid Image in Deep-space Autonomous Optical Navigation

Siliang Du<sup>1,2</sup>, Mi Wang<sup>2</sup>, Xiao Chen<sup>3</sup>, Shenghui Fang<sup>1</sup> and Hongbo Su<sup>4</sup>

<sup>1</sup>(School of Remote Sensing and Information Engineering, Wuhan University, Wuhan, 430079, P.R. China)

<sup>2</sup>(State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan, Hubei 430079, China)

<sup>3</sup>(Shanghai Institute of Satellite Engineering, Shanghai 200240, China)

<sup>4</sup>(Department of Civil, Environmental and Geomatics Engineering, Florida Atlantic University, 33431 USA)

(E-mail: [wangmi@whu.edu.cn](mailto:wangmi@whu.edu.cn))

A planet centroid is an important observable object in autonomous optical navigation. A high-accuracy algorithm is presented to extract the planet centroid from its raw image. First, we proposed a planet segmentation algorithm to segment the planet image block to eliminate noise and to reduce the computation load. Second, we developed an effective algorithm based on Prewitt-Zernike moments to detect sub-pixel real edges by determining possible edges with the Prewitt operator, removing pseudo-edges in backlit shady areas, and relocating real edges to a sub-pixel accuracy in the Zernike moments. Third, we proposed an elliptical model to fit sub-pixel edge points. Finally, we verified the performance of this algorithm against real images from the Cassini-Huygens mission and against synthetic simulated images. Simulation results showed that the accuracy of the planet centroid is up to 0·3 pixels and that of the line-of-sight vector is at  $2\cdot1 \times 10^{-5}$  rad.

## KEY WORDS

1. Autonomous optical navigation.      2. Planet segmentation.      3. Zernike moments.      4. Planet centroid.

Submitted: 11 March 2015. Accepted: 7 November 2015. First published online: 23 December 2015.

1. INTRODUCTION. Traditional spacecraft navigation uses Earth-based resources and has proved to be very reliable and robust in many successful space missions (Iess et al., 2009). However, this technology requires continuous observation, and since the maximum signal delay can be up to six hours in a solar exploration, it cannot meet the needs of real time navigation, especially during the planetary, rendezvous, and landing phases. Meanwhile, the cost of Earth-based navigation is very high, lagging

behind the navigation requirement of controlling deep space exploration missions in real time; a less expensive and more reliable alternative is emerging in autonomous optical navigation technology (Betto et al., 2006).

Autonomous optical navigation has the advantages of independence, high accuracy, and real-time performance, and has become a hotspot for research in recent years (Yu et al., 2014; Ning and Fang, 2009; Ma et al., 2013). Autonomous optical navigation reduces the complexity of operation and mission costs, simplifies the Earth-based support system, and greatly enhances the efficiency of deep space exploration. Even in conditions where the detector and ground communications are interrupted, it is still capable of completing orbit determination, orbit keeping, and attitude control. At present, the feasibility of optical navigation has been preliminarily validated in many deep-space missions. In the Voyager missions, key techniques for deep space navigation, including the celestial centroid-extracting algorithm, were developed (Rudd et al., 1997). Later, optical navigation was successfully conducted in the Galileo spacecraft's approach to and crossing of an asteroid. The first full application of autonomous navigation technology, though, was not realised until the Deep Space 1 mission. The main idea of autonomous optical navigation in this mission was in determining the craft's position and velocity via the optical measurement of an asteroid and background stars. In addition, the European Space Agency (ESA) used Smart-1 to verify an autonomous navigation system in deep space missions. The ESA test used an AMIE camera to take pictures of certain celestial bodies or navigation stars so as to determine the line-of-sight direction of the celestial body or astral, and then calculated the orbit of the probe (Marini et al., 2002; Rathsman et al., 2005).

With rapid technological developments of high-resolution cameras, image size has become increasingly larger, as has the complexity of image processing algorithms. It is challenging to develop image-processing algorithms to meet the demands of real-time high-accuracy navigation. Traditional image processing algorithms cannot be applied directly by an autonomous navigation system because of the low on board computation capacity and the particularity of deep space (Owen, 2011). Recently, papers have been published on the subject of autonomous optical navigation; most of them are about navigation measurement modelling and navigation filters, and few focus on image processing and extraction of navigation observables (Giannitrapani et al., 2011; Thompson et al., 2012; Shuang and Cui, 2008). Owen (2011) pointed out that useful navigation observables include the apparent diameter and centroid of the planet, the line-of-sight vector, and the angle between the horizon and the reference star. (Christian and Lightseys, 2012) used binaryzation, dilation, erosion, and image closing, analysed the interconnected components and then fitted an ellipse to the planet in order to extract the planet centroid. Li et al. (2013) presented a detailed procedure for the extraction of navigation observables, used the Canny operator to extract planet edges, and then fitted the planet ellipse with edge points. The accuracies achieved by these methods are able to meet the requirements of deep-space autonomous navigation. However, higher accuracy navigation observables are needed to achieve a more reliable optical navigation, and less computation in the image-processing algorithm is necessary to implement a faster optical navigation.

In this study, we developed a simple and effective image-processing algorithm for extracting a planet centroid. The planet centroid is an important parameter with which the spacecraft state can be estimated after the navigation filter and system dynamics equations are established. It is assumed that the perspective projection of a

target planet body forms an ellipse on the image plane. In the remainder of this paper, Section 2 briefly presents the principle of autonomous optical navigation and the image processing algorithms. Section 3 presents a method for segmenting the planet image block from the whole image so as to eliminate noise and to reduce the computation burden. Section 4 introduces an algorithm to extract sub-pixel real edge points. Section 5 describes the method used for ellipse fitting based on a set of real edge points and the centroid computation formula. Section 6 validates the algorithm with experiments for detection accuracy. Section 7 gives the conclusions.

**2. PRINCIPLE OF AUTONOMOUS OPTICAL NAVIGATION.** Here, autonomous optical navigation is based on the extraction of the planet centroid from the planet images captured by image sensors. The principles of autonomous optical navigation are described as follows.

The position of a planet in celestial coordinate system  $OXYZ$  is described by right ascension  $\alpha$  and declination  $\delta$  (Figure 1). The planet unit vector  $e_I$  in an inertial frame can be expressed as:

$$e_I = \begin{bmatrix} \cos \alpha \cos \delta \\ \sin \alpha \cos \delta \\ \sin \delta \end{bmatrix} \quad (1)$$

The planet image captured by a detector (usually a Charge-Coupled Device (CCD) or Complementary Metal-Oxide Semiconductor (CMOS) array in this case) can be described by a set of  $[x_p, y_p]$  in the image plane frame  $Oxy$  (Figure 2). According to the geometric relations shown in Figure 2, the line-of-sight direction to an object is given by the following equation (Christian and Lightsey, 2012):

$$e_c = \frac{1}{\sqrt{x_p^2 + y_p^2 + f^2}} \begin{bmatrix} -x_p \\ -y_p \\ f \end{bmatrix} \quad (2)$$

where  $f$  is the camera focal length and  $e_c$  is the unit vector of the line-of-sight described in the camera coordinate system  $O'x_cy_cz_c$ . The line-of-sight vector is then rotated from the camera frame to the inertial frame:

$$e_I = T_B^I T_C^B e_C \quad (3)$$

in which  $T_C^B$  rotates a vector from the camera frame to the spacecraft body frame, and  $T_B^I$  rotates a vector from the spacecraft body frame to the inertial frame. The transformation matrices  $T_C^B$  and  $T_B^I$  include information on the attitude and position of the spacecraft. The unit vector  $e_I$  can be obtained by referring to the ephemeris. Because the camera's focal length  $f$  is known to be fixed for a known camera, the main challenge to acquiring the line-of-sight unit vector  $e_c$  is to define the centroid  $(x, y)$  of the planet (Owen et al., 2008).

**3. PLANET SEGMENTATION.** On a deep-space image, a planet accounts for a part of the whole image, as does the noise of other stars and objects. To process only a planet image block can save computation load and time.

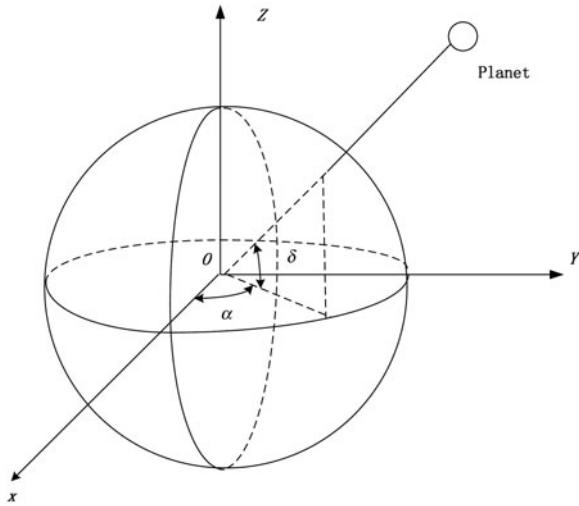


Figure 1. The planet coordinates in celestial coordinate system.

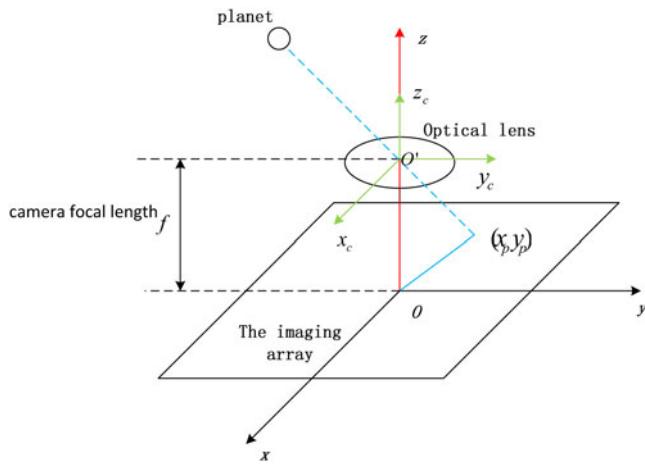


Figure 2. Geometry of planet imaging.

First, conducting threshold segmentation to simplify a raw image into a binary image helps speed up the subsequent image processing. Assuming that a threshold value  $\tau$  is selected, creating a binary image would check the pixel value at each pixel point. Each pixel in intensity above  $\tau$  is set to 255 (white) and below  $\tau$  is set to 0 (black) (Javidi et al., 1995). The pixels with a value of 255 are considered to be in the “foreground” in which the planet information is contained. Figure 3(b) shows the binary image of the planet Mars with a threshold value  $\tau = 30$ .

Second, morphological opening is operated to remove the noise of secondary stars and objects. This operation uses erosion first and then dilation (Haralick et al., 1987).

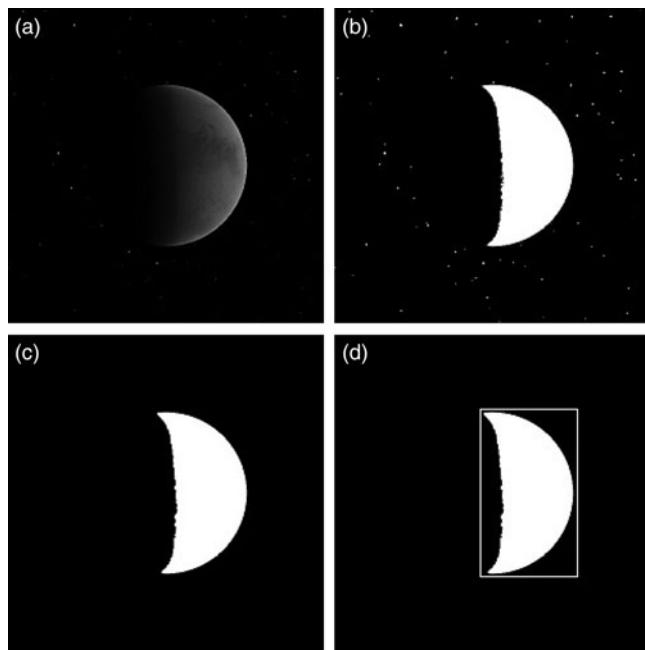


Figure 3. Image of Mars and the results of a planet segmentation (a) Image of Mars; (b) Mars' binary image; (c) background stars removed by an opening operator; (d) foreground Mars' area found in the whole image.

In general, an opening operation can smooth the outline of an object in the image, and it eliminates ziggags. When applied in deep space images, small stars/objects can be removed effectively, and the edges of a planet become smooth.

Using a structure element  $S$ , one can carry out the opening on image  $A$  according to Kemper (1998):

$$A \circ S = (A \ominus S) \oplus S \quad (4)$$

where  $\circ$  is for the opening operator,  $\ominus$  is the erosion operator, and  $\oplus$  is the dilation operator.

**Figure 3(c)** shows the opening operator's result. It can be seen that the background stars have been cleared up. **Figure 4** is the local planetary binary image resulting from an opening operator. **Figure 4(a)** shows noise in the background at the edge of the planet. **Figure 4(b)** shows the result of the opening operation, in which the noise is cleaned out and the edges become smooth.

Third, a planet image block is segmented from the whole image. As shown in **Figure 3(c)**, noise has been cleaned up on the image. The segmentation includes the following steps.

- (1) To find the start line  $x_1$  and end line  $x_2$  with pixel value 255 at direction  $x$  in the whole image;
- (2) To find the start column  $y_1$  and end column  $y_2$  with pixel value 255 at direction  $y$  in the whole image;

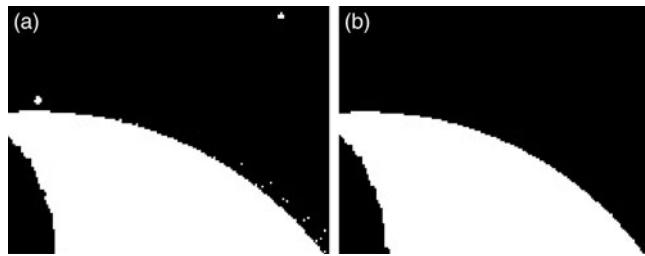


Figure 4. Local planetary binary image and the result of the opening operation (a) local planetary binary image (b) the result of the opening operator.

- (3) The area surrounded by point  $(x_1, y_1)$  and point  $(x_2, y_2)$  is defined as the planet image block represented by the white frame in Figure 3(d).

Now, a planet image block is successfully segmented. Next, edge detection will be applied on just the block. All unwanted information is discarded, and the image processing load and time can be greatly reduced.

**4. SUB-PIXEL EDGE DETECTION.** Edge detection employs local operators to compute approximately the first derivative of grey level gradient on an image in the spatial domain. Locations in the local maximum of the first derivative will be considered to be the edge points. Ideally, an edge detection algorithm will be able to determine all edges and edge points accurately in the position. At present, many such algorithms have been developed at the pixel or sub-pixel level. Pixel-level edge detections, including Prewitt, Sobel, Laplacian of Gaussian (LOG), and Canny operators are capable of detecting edges quickly but with low accuracy (Ding and Goshtasby, 2001; Wang 2007; Hou and Wei, 2002). Li et al. (2013) detected the edges at pixel level with the Canny operators. All the edges may be detected in this way, but the position accuracy is low. Sub-pixel level methods, such as interpolation and the Zernike method, can enhance the accuracy but require a longer runtime (Wee and Paramesran, 2007). Ying-Dong et al. (2005) propose an approach combining the pixel-level method with the sub-pixel level method to detect an object edge using the Sobel-Zernike moments operator.

We propose an algorithm of Prewitt-Zernike moments edge detection and apply it in planet-containing images. The Prewitt-Zernike moments edge detection is an algorithm that first detects possible edges with the Prewitt operator and then relocates them with possible edges to sub-pixel accuracy determined by the Zernike moments. To save computation while retaining real edges, and before possible edges detected by the Prewitt operator are relocated to sub-pixel accuracy by Zernike moments, pseudo-edges in the backlit shade areas are removed. Our proposed edge detection algorithm consists of three steps. First, the Prewitt operator is used to detect approximately all possible edge points. Second, pseudo-edges in the backlit shade areas are removed. Third, the Zernike moments operator is used to relocate the real edge points to sub-pixel accuracy.

**4.1. Prewitt Edge Detection.** A Prewitt operator detects edges by calculating partial derivatives in  $3 \times 3$  neighbourhoods and by smoothing out noise. It is relatively simpler than other operators such as the LOG operator, Canny operator, and so on. In

our example, the planet image block is very simple ([Figure 5\(a\)](#)), in which the Prewitt operator was used to detect edges. The pre-processing steps (the procedure listed in Section 3) eliminate noise and segment the planet image block. After pre-processing, the Prewitt operator is used to detect the edge. The partial derivatives in horizontal direction  $F_h$  and vertical direction  $F_v$  are given according to Kuo et al. ([1997](#)):

$$\begin{aligned} F_h(x, y) = & f(x+1, y-1) + f(x+1, y) + f(x+1, y+1) \\ & - f(x-1, y-1) - f(x-1, y) - f(x-1, y+1) \end{aligned} \quad (5)$$

$$\begin{aligned} F_v(x, y) = & f(x-1, y+1) + f(x, y+1) + f(x+1, y+1) \\ & - f(x-1, y-1) - f(x, y-1) - f(x+1, y-1) \end{aligned} \quad (6)$$

Then, the edge gradient magnitude  $M(x, y)$  and direction  $\theta(x, y)$  are determined as follows:

$$M(x, y) = \sqrt{F_h^2(x, y) + F_v^2(x, y)} \quad (7)$$

$$\theta(x, y) = \arctan(F_v(x, y)/F_h(x, y)) \quad (8)$$

Then the threshold value  $t$  is selected, if it is satisfied that

$$M(x, y) > t \quad (9)$$

Hereafter, a point means an edge point. All the points must be marked. [Figure 5\(b\)](#) shows the result from Prewitt edge detection, which obviously includes the pseudo-edges. The next section describes removal of these pseudo-edges.

**4.2. Pseudo-edges removal.** Pseudo-edges usually occur at the border between a sunlight face and backlit face (Thompson et al., [2012](#)). In order to extract a planet centroid to high accuracy, real edges must be distinguished from pseudo-edges. Pseudo-edges can be detected by calculating the angle between the solar direction and the gradient direction of edges. In fact, the angle between the gradient direction of edges and the solar direction is within  $90^\circ$ . The main difficulty is to determine the direction to the sun, which may be measured directly by an instrument such as a Sun sensor. Given the planet's edge gradient vector  $g$  and the direction  $n$  to the Sun, the following equation can be satisfied (Li et al., [2013](#)):

$$\frac{g \cdot n}{|g||n|} > 0 \quad (10)$$

which can be simplified as:

$$g \cdot n > 0 \quad (11)$$

[Figure 5\(c\)](#) shows successful removal of pseudo-edges; the green arrows stand for the edge gradient vectors and the red ones are the solar light direction.

**4.3. Sub-pixel edges relocated.** In Section 4.2, real edges are detected to just pixel-level accuracy. In order to reach a higher accuracy of planet centroid, a Zernike moments operator is used to relocate edges to sub-pixel accuracy among all the real edge points.

Edges of a continuous ideal image can be seen in a step model as shown in [Figure 6](#), in which  $h$  is the background grey level,  $k$  is the step height,  $l$  is the perpendicular

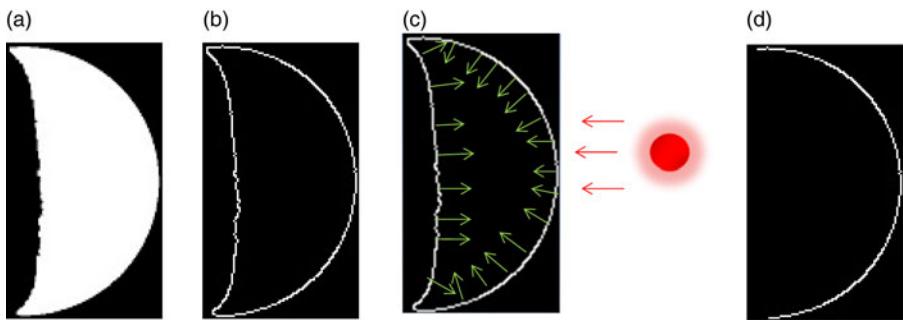


Figure 5. Edge detection algorithm applied to planet image block (a) planet image block; (b) result by Prewitt operator; (c) pseudo-edges removal; (d) relocating the edges with sub-pixel accuracy.

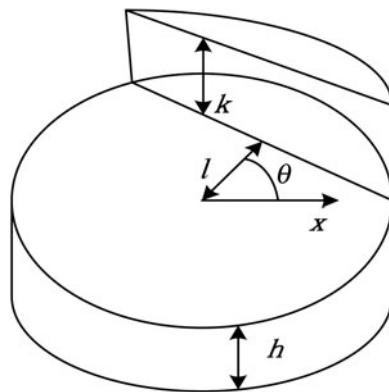


Figure 6. The two-dimensional sub-pixel step-model.

distance between the actual edge and the centre, and  $\theta$  is the angle between the edge and horizontal line and satisfied as  $\theta \in (-\frac{\pi}{2}, \frac{\pi}{2})$ .

Zernike moments for an image  $f(x, y)$  are defined as per Ghosal and Mehrotra (1993):

$$A_{nm} = \frac{n+1}{\pi} \iint_{x^2+y^2 \leq 1} f(x, y) V_{nm}^*(\rho, \theta) dx dy \quad (12)$$

where  $(n+1)/\pi$  is a normalisation factor. In discrete form,  $A_{nm}$  can be expressed as

$$A_{nm} = \sum_x \sum_y f(x, y) V_{nm}^*(\rho, \theta), \quad x^2 + y^2 \leq 1 \quad (13)$$

Equation (13) shows that in a discrete image, the neighbourhood of an image point should be mapped onto the interior of a unit circle for evaluating Zernike moments  $A_{nm}$  of the point. The complex polynomials  $V_{nm}(\rho, \theta)$  can be expressed in polar

coordinates as

$$V_{nm}(\rho, \theta) = R_{nm}(\rho)e^{jm\theta} \quad (14)$$

where  $V_{nm}^*$  and  $V_{nm}$  are conjugate and  $R_{nm}(\rho, \theta)$  is a radial polynomial defined as

$$R_{nm}(\rho) = \sum \frac{(-1)^s(n-s)!}{s!(\frac{n+|m|}{2}-s)!} \frac{\rho^{n-2s}}{(\frac{n-|m|}{2}-s)!} \quad (15)$$

If the image is rotated by an angle  $\varphi$  counterclockwise, the Zernike moments of the rotated image would be

$$A_{nm} = A'_{nm}e^{jm\varphi} \quad (16)$$

Therefore, Zernike moments require a phase shift on rotation while their magnitudes remain constant. For calculating edge parameters  $l$  and  $\theta$ , the two masks  $A_{11}$  and  $A_{20}$  should be deduced. According to Equations (14) and (15), the orthogonal complex polynomials can be written as:

$$V_{11} = x + jy, V_{20} = 2x^2 + 2y^2 - 1 \quad (17)$$

The unit circle in Figure 7 is divided into  $5 \times 5$  grids, and masks are calculated when integrating  $V_{11}^*$ ,  $V_{20}^*$  on a dashed area of every grid (Ying-Dong et al., 2005).

Convolving these masks with the image points can obtain Zernike moments as shown in Figures 8 and 9. The relationship between the Zernike moments of the original image  $A_{11}$ ,  $A_{20}$  and rotated image  $A'_{11}$ ,  $A'_{20}$  can be given as

$$A'_{11} = A_{11}e^{j\varphi} \quad (18)$$

$$A'_{20} = A_{20} \quad (19)$$

The following equations can be deduced in the theory of Zernike moments (Ghosal and Mehrotra, 1993):

$$A'_{11} = \iint_{x^2+y^2 \leq 1} f'(x, y)(x - jy)dxdy = \frac{2k(1-l^2)^{3/2}}{3} \quad (20)$$

$$A'_{20} = \iint_{x^2+y^2 \leq 1} f'(x, y)(2x^2 + 2y^2 - 1)dxdy = \frac{2kl(1-l^2)^{3/2}}{3} \quad (21)$$

where  $f'(x, y)$  is the rotated image. When the edge is rotated by an angle  $-\theta$ , it could be aligned in parallel to the  $y$ -axis so that  $\iint_{x^2+y^2 \leq 1} f'(x, y)y dxdy = 0$ , and it becomes the imaginary component of  $A'_{11}$  and can be expressed as

$$\text{Im}[A'_{11}] = \sin(\theta)\text{Re}[A_{11}] - \cos(\theta)\text{Im}[A_{11}] = 0 \quad (22)$$

So,

$$\theta = \tan^{-1}\left(\frac{\text{Im}[A_{11}]}{\text{Re}[A_{11}]}\right) \quad (23)$$

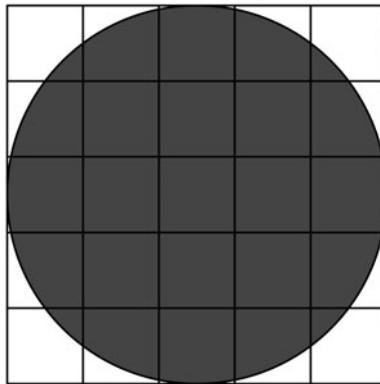


Figure 7. Circular kernel defined for a  $5 \times 5$  pixel area.

-0.0147+0.0147j	-0.0469+0.0933j	0.0000+0.0640j	0.0469+0.0933j	0.0147+0.0147j
-0.0933+0.0147j	-0.0640+0.0640j	0.0000+0.0640j	0.0640+0.0640j	0.0933+0.0469j
-0.1253+j0	-0.0640+j0	0.0000+j0	0.0640+j0	0.1253+j0
-0.0933-0.0469j	-0.0640-0.0640j	0.0000-0.0640j	0.0640-0.0640j	0.0933-0.0469
-0.0147-0.0147j	-0.0469-0.0933j	0.0000-0.1253j	0.0469-0.0933j	0.0147-0.0147j

Figure 8. Mask of  $A_{11}$ .

0.0177	0.0595	0.0507	0.0595	0.0177
0.0595	-0.0492	-0.1004	-0.0492	0.0595
0.0507	-0.1004	-0.1516	-0.1004	0.0507
0.0595	-0.0492	-0.1004	-0.0492	0.0595
0.0177	0.0595	0.0507	0.0595	0.0177

Figure 9. Mask of  $A_{20}$ .

Solving Equations (20) and (21), the edge parameter  $l$  can be given as:

$$l = \frac{A_{20}}{A'_{11}} \quad (24)$$

And the sub-pixel edge detection formula is given as:

$$\begin{bmatrix} x_s \\ y_s \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} + l \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} \quad (25)$$

where  $(x_s, y_s)$  is the sub-pixel point of the edge.

5. ELLIPSE FITTING. The perspective projection of a sphere or ellipsoid forms an ellipse on an image plane. Once candidate edge points are found, an ellipse must be fitted to this data set. The details of this ellipse fitting process are discussed below.

5.1. *Robust model fitting.* The first useful ellipse fitting is a direct least-square algorithm developed by Fitzgibbon et al. (1999). Any conic section may be described by the following implicit quadratic equation:

$$F(a, x_i) = a^T x_i = Ax_i^2 + Bx_iy_i + Cy_i^2 + Dx_i + Ey_i + F = 0 \quad (26)$$

where  $[x_i, y_i]$  is a point on the conic section,  $a = [A \ B \ C \ D \ E \ F]^T$ , and  $x_i = [x_i^2 \ x_iy_i \ y_i^2 \ x_i \ y_i \ 1]^T$ . The conic will be an ellipse if the coefficients of the equation satisfy  $4AC - B^2 > 0$ . The form allows the constants to be scaled arbitrarily such that the ellipse inequality constraint may be rewritten as an equality constraint:

$$4AC - B^2 = 1 \quad (27)$$

In general, a point detected by the edge detection algorithm mentioned in Section 4 does not lie exactly on the ellipse, and  $F(a, x_i) \neq 0$  because of imaging noise. Therefore, Fitzgibbon et al. (1999) proposed an optimisation method using the square of model-fit residuals:

$$\min J = \sum_{i=1}^n [F(a, x_i)]^2 = a^T G^T G a \quad (28)$$

where  $G = [x_1 \ x_2 \dots x_n]^T$ , the equality constraint of Equation (28) can be rewritten in vector form

$$4AC - B^2 = a^T \begin{bmatrix} 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} a = a^T H a = 1 \quad (29)$$

If the equality of Equation (29) is adjoined to an objective function with a Lagrange multiplier, then

$$\min I = a^T G^T G a + \lambda(1 - a^T H a) \quad (30)$$

The solution to this optimisation problem is the following rank-deficient generalised eigenvalue problem

$$(G^T G)a = \lambda H a \quad (31)$$

Substituting this solution into the objective function shows that

$$J = a^T G^T G a = \lambda a^T H a = \lambda \quad (32)$$

which means that the optimal solution occurs at the minimal positive eigenvalue (Fitzgibbon et al., 1999). The solution to Equation (32) yields exactly a positive eigenvalue. This positive eigenvalue corresponds to the eigenvector that is the solution to  $a$ . As noise may exist at the edge points, it is difficult to determine exact analytic solutions directly. Christian and Lightsey (2012) used the M-Estimator Sample Consensus

(MSAC) algorithm to eliminate outliers in the edge points set using least-square ellipse-fitting and found that it was a robust method for fitting the elliptical model to the edge points. We adapted the MSAC algorithm to eliminate outliers so as to fit the elliptical model at the edge points.

**5.2. Centroid calculation.** Transformation from coefficients of an implicit conic equation into the standard ellipse parameters is a well-known classical geometry and given by:

$$x = \frac{2CD - BE}{B^2 - 4AC} \quad y = \frac{2AE - BD}{B^2 - 4AC} \quad (33)$$

$$a = \sqrt{\frac{2[AE^2 + CD^2 - BDE + F(B^2 - 4AC)]}{(B^2 - 4AC)[\sqrt{(A - C)^2 + B^2} - A - C]}} \quad (34)$$

$$b = \sqrt{\frac{2[AE^2 + CD^2 - BDE + F(B^2 - 4AC)]}{(B^2 - 4AC)[- \sqrt{(A - C)^2 + B^2} - A - C]}} \quad (35)$$

$$\phi = \begin{cases} 0 & B = 0 \text{ and } A < C \\ \frac{\pi}{2} & B = 0 \text{ and } A > C \\ \frac{1}{2}\cot^{-1}\left(\frac{A - C}{B}\right) & B \neq 0 \text{ and } A < C \\ \frac{\pi}{2} + \frac{1}{2}\cot^{-1}\left(\frac{A - C}{B}\right) & B \neq 0 \text{ and } A > C \end{cases} \quad (36)$$

where  $(x, y)$  is the coordinate of the ellipse centre,  $a$  and  $b$  stand for the semi-major axis and semi-minor axis of the ellipse; and  $\phi$  is the angle from the  $x$ -axis to the major ellipse axis. Therefore, an ellipse can be well described with the above five parameters. With these parameters, we can easily compute the line-of-sight vector from the spacecraft to the centroid of a target celestial body with Equation (2).

**6. EXPERIMENTS.** Experiments were performed to validate the detection accuracy for the algorithm. First, a computer-generated standard circle with noise was used to test the accuracy of the edge-detecting algorithm in Prewitt-Zernike moments. Second, real images obtained from deep-space missions were used for the validation. Lastly, the synthetic simulated images were applied to test the accuracy of the image-processing algorithm.

The experiments are shown in Figure 10(a). The size of the image is  $256 \times 256$  in pixels, the centre of the circle is at  $(128.0, 128.0)$ , and the radius is  $100.0$  pixels. To robustly withstand noise, Gaussian white noise with variance  $\sigma = 0.01$  was added to the image. We adapted the method of Li et al. (2013), in which the Canny operator was used to locate the edge points, then used the sub-pixel edge detection methods of Polynomial interpolation and the Gauss surface fitting technique to relocate the sub-pixel edge points for comparing (Ye et al., 2005; Hermosilla et al., 2008). In addition, our method was applied to locate the edge points, including Prewitt-Zernike moments.

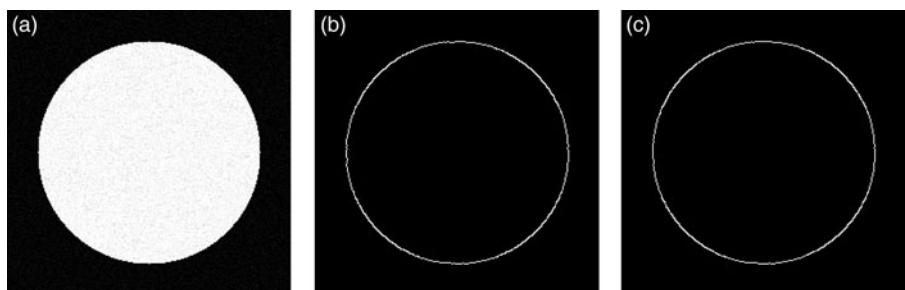


Figure 10. A standard circle with noise and the results of edge detection: (a) Standard circle with noise; (b) Edge detection result by method of Li et al. (2013); (c) Edge detection by the method proposed in this paper.

Figure 11 shows the absolute errors of edge locations detected in Prewitt-Zernike moments, the Polynomial interpolation method, and the Gauss surface fitting technique. The Root Mean Square (RMS) of edge detection in the Prewitt-Zernike moments is 0.14 pixels, while the RMS of the Polynomial interpolation method and Gauss surface fitting technique are 0.29 and 0.23 pixels respectively. The test result shows the Prewitt-Zernike moments have a higher accuracy than the other sub-pixel edge detection methods of Polynomial interpolation and Gauss surface fitting technique, and is more robust against Gaussian white noise. Although a sub-pixel edge detection in Prewitt-Zernike moments can reach a higher accuracy, the computation load of Prewitt-Zernike moments is still great. In order to cope with the problem, we segmented the planet image block from the whole image first, processed the edge detecting algorithm on the block only, removed fake edge points, and relocated the real edge with Prewitt-Zernike moments at sub-pixel accuracy. In this process, unnecessary sub-pixel relocating operations are reduced, focusing only on the real edges with the sub-pixel relocating algorithm in Prewitt-Zernike moments. Therefore, a more accurate planet centroid can be calculated.

In order to verify the effectiveness of the algorithm proposed, we used real raw image data that came from the Cassini-Huygens mission. The image was taken in visible light with the Cassini spacecraft wide-angle camera on 9 October 2008, at approximately 83 000 km (52 000 miles) from Enceladus, the sixth largest moon of Saturn, as seen in Figure 12(a). First, we isolated the moon image blocks in the whole images adapting the segmentation algorithm. As seen in Figure 12(b), the image block of the moon has been successfully shown as a white rectangle. After that, the edge detection method described in Section 4 was used to extract the sub-pixel edge points of the moon. The pseudo-edges at the backlit face of the moon were removed by calculating the angle between the solar direction and the gradient direction of edges. The solar direction was determined by a sun sensor. However, due to possible lack of data from the sensor, the solar direction has to be judged by visual inspection. As seen in Figure 12(c), the real edges of the moon were successfully detected. Finally, ellipses were fitted to the moon. As the true coordinate of the moon centroid in the real raw image was unknown, the accuracy could not be evaluated. In the next experiment, synthetic simulated images close to the real flyby raw images are utilised to verify the accuracy of the image-processing algorithm.

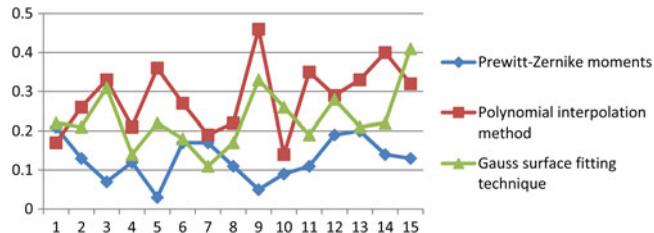


Figure 11. The absolute errors of edge location for standard circle image.

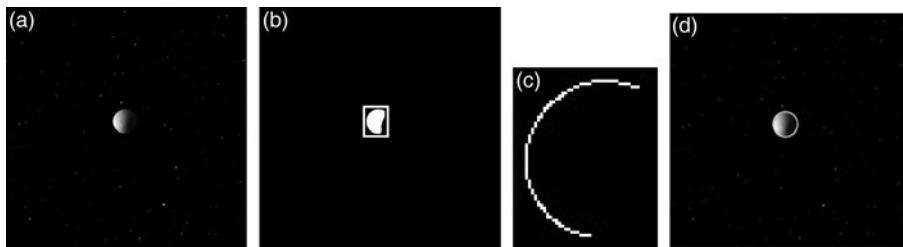


Figure 12. Planet centroid extracting algorithm applied to real raw image from Cassini-Huygens mission (a) Raw image of Enceladus; (b) Segmentation; (c) Extract of the real sub-pixel edges; (d) Earth with best ellipse.

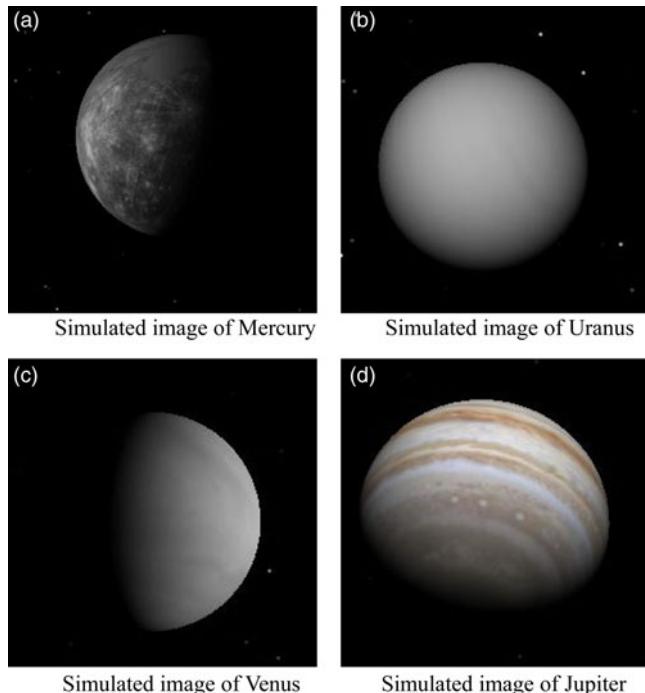


Figure 13. Planet images simulated by Celestial software.

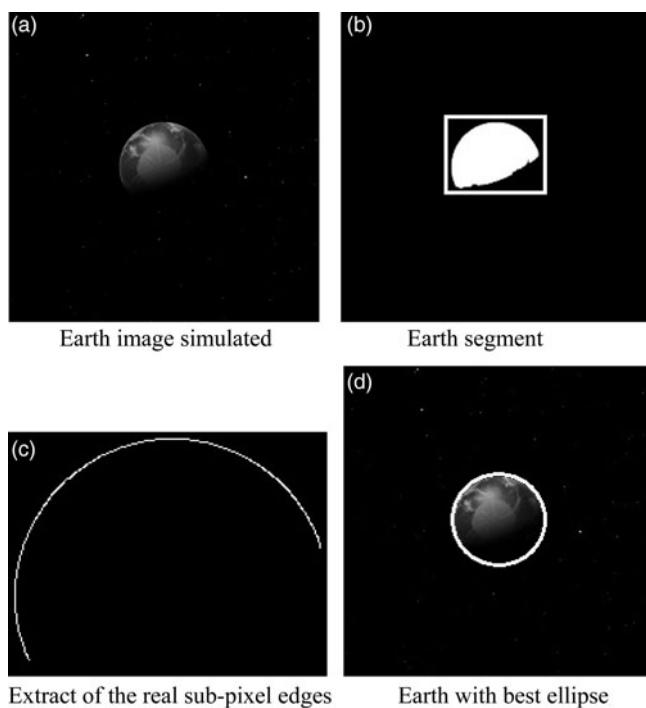


Figure 14. Simulating the Earth image with the planet centroid extracting algorithm.

A synthetic simulated image is simulated by Celestia software, open-source software for astronomical works supported by NASA Technology. The outer space environment, the atmosphere of the planet surface, and the texture of the planet simulated with the software are very similar to the corresponding real ones (Celestia, 2015). In order to test the accuracy of the planet centroid, five images of Mercury, Uranus, Venus, Jupiter, and Earth were simulated as shown in Figures 13(a), (b), (c), (d) and 14(a), respectively. The complete image processing algorithms developed in this study were applied to the five planets at different sun azimuth angles, surface textures, and lighting and atmospheric conditions. The Earth image was used as an example to display the procedure of the algorithm we proposed.

As indicated in Figure 14, the planet image block of the Earth is successfully segmented, the pseudo-edges removed, the real edges detected, the edge points relocated to sub-pixel accuracy, a best ellipse fitted to the planet, and finally, the centroid coordinate of the planet obtained with Equation (33). As the planet centroid in the simulated image can be positioned accurately in advance, errors of planet centroid can be calculated easily. The edge points were detected by Prewitt-Zernike moments, the Polynomial interpolation method, the Gauss surface fitting technique, and then were fitted to an ellipse. The errors that occurred in the three methods are shown in Figure 15.

Figure 15 shows that the maximum errors of the three methods were 0.35, 0.66 and 0.56 pixels, and the RMS were 0.29, 0.51 and 0.43 pixels, respectively. The method in Prewitt-Zernike moments is more accurate and stable than those in the Polynomial

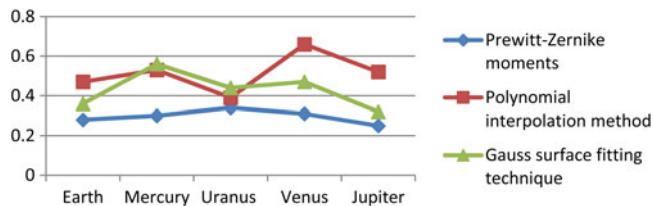


Figure 15. Extracting errors of planet centroid.

interpolation and Gauss surface fitting. Assuming the focal length of camera  $f = 200$  mm, the angle of view field  $\theta = 3.5^\circ$ , and scale factor  $K_x = K_y = 83.8$  pixel/mm, then the maximum observation error of the line-of-sight vector from camera to planet centroid in our proposed method is  $2.1 \times 10^{-5}$  rad, which easily meets the requirements of deep-space autonomous navigation.

**7. CONCLUSION.** To meet the high accuracy and fast processing demands of deep-space autonomous navigation, a new image-processing algorithm is proposed to extract the centroid of a planet. In particular, a simple and effective planet segmentation algorithm was developed to eliminate noise and reduce computation load. Meanwhile, a sub-pixel edge-detecting algorithm based on the Prewitt-Zernike moments is developed to detect edge points of planets at the sub-pixel level. Both real flyby images from the Voyager mission and corresponding simulated images were used to verify the performance of the algorithm. Simulating a standard circle with noise shows that the precision of the proposed Prewitt-Zernike moments operators achieved 0.14 pixels in edge location accuracy. In addition, experiments on a real raw image proved that the algorithm is effective, as the planet was successfully segmented from the real raw image and an ellipse was well fitted to the planet. In addition, experiments on planet image simulations indicate that the accuracy of the planet centroid reached as high as 0.3 pixels and the accuracy of the line-of-sight vector to  $2.1 \times 10^{-5}$  rad. Therefore, with the higher-accuracy positioning for a planet centroid, the reliability of deep-space autonomous optical navigation will be improved.

#### ACKNOWLEDGEMENTS

The work described in this paper was supported by: the National Basic Research Program of China 973 Program (Grant NO. 2014CB744201), National Natural Science Foundation of China (NSFC) (Grant NO.41371430 and 91438111). The author fully appreciates their financial support.

#### REFERENCES

- Betto, M., Rgensen, J.L.J., Rgensen, P.S.J. and Denver, T. (2006). Advanced stellar compass deep space navigation, ground testing results. *Acta Astronautica*, **59**(8–11), 1020–1028.
- Celestia. (2015). The Celestia Motherlode, <http://www.celestiamotherlode.net/>.
- Christian, J.A. and Lightsey, E.G. (2012). Onboard Image-Processing Algorithm for a Spacecraft Optical Navigation Sensor System. *Journal of Spacecraft and Rockets*, **49**(2), 337–352.
- Ding, L.J. and Goshtasby, A. (2001). On the Canny edge detector. *Pattern Recognition*, **34**(3), 721–725.

- Fitzgibbon, A., Pilu, M. and Fisher, R.B. (1999.) Direct least square fitting of ellipses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **21**(5), 476–480.
- Ghosal, S. and Mehrotra, R. (1993). Orthogonal moment operators for subpixel edge detection, *Pattern Recognition*, **26**(2), 295–306.
- Giannitrapani, A., Ceccarelli, N., Scortecci, F. and Garulli, A. (2011). Comparison of EKF and UKF for Spacecraft Localization via Angle Measurements. *IEEE Transactions on Aerospace and Electronic Systems*, **47**(1), 75–84.
- Haralick, R.M., Sternberg, S.R. and Zhuang, X. (1987). Image analysis using mathematical morphology. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **9**(4), 532–550.
- Hermosilla, T., Bermejo, E., Balaguer, A. and Ruiz, L.A. (2008). Non-linear fourth-order image interpolation for subpixel edge detection and localization. *Image and Vision Computing*, **26**(9), 1240–1248.
- Hou, Z.J. and Wei, G.W. (2002). A new approach to edge detection. *Pattern Recognition*, **35**(7), 1559–1570.
- Iess, L., Asmar, S. and Tortora, P. (2009). MORE: An advanced tracking experiment for the exploration of Mercury with the mission BepiColombo. *Acta Astronautica*, **65**(5–6), 666–675.
- Javidi, B., Li, J., Fazlollahi, A.H. and Horner, J. (1995). Binary nonlinear joint transform correlator performance with different thresholding methods under unknown illumination conditions. *Applied Optics*, **34**(5), 886–896.
- Kemper, P.J. (1998). Mathematical morphology enhancement of maximum entropy thresholding for small targets, *Bellingham: Spie-Int Soc Optical Engineering*, 84–91.
- Kuo, Y.H., Lee, C.S. and Liu, C.C. (1997). A new fuzzy edge detection method for image enhancement. *Proceedings Of The Sixth IEEE International Conference on Fuzzy Systems, Vols I - III*, 1069–1074.
- Li, S., Lu, R.K., Zhang, L. and Peng, Y.M. (2013). Image Processing Algorithms For Deep-Space Autonomous Optical Navigation. *Journal of Navigation*, **66**(4), 605–623.
- Ma, X., Fang, J. and Ning, X. (2013). An overview of the autonomous navigation for a gravity-assist interplanetary spacecraft. *Progress in Aerospace Sciences*, **63**(0), 56–66.
- Marini, A.E., Racca, G.D. and Foing, B.H. (2002). SMART-1 technology preparation for future planetary missions. *Advances in Space Research*, **30**(8), 1895–1900.
- Ning, X. and Fang, J. (2009). A new autonomous celestial navigation method for the lunar rover. *Robotics and Autonomous Systems*, **57**(1), 48–54.
- Owen, W.M. (2011). *Methods Of Optical Navigation, San Diego: Advances in the Astronautical Sciences*, **140**, 1635–1653.
- Owen, W.M., Duxbury, T.C., Acton, C.H., Synnott, S.P., Riedel, J.E. and Bhaskaran, S. (2008). A Brief History of Optical Navigation At JPL, *San Diego: American Astronautical Society*, **131**, 329–348.
- Rathsman, P., Kugelberg, J., Bodin, P., Racca, G.D., Foing, B. and Stagnaro, L. (2005). SMART-1: Development and lessons learnt. *Acta Astronautica*, **57**(2–8), 455–468.
- Rudd, R.P., Hall, J.C. and Spradlin, G.L. (1997). The voyager interstellar mission. *Acta Astronautica*, **40**(2–8), 383–396.
- Shuang, L. and Cui, P.Y. (2008). Landmark tracking based autonomous navigation schemes for landing spacecraft on asteroids. *Acta Astronautica*, **62**(6–7), 391–403.
- Thompson, D.R., Bunte, M., Castano, R., Chien, S. and Greeley, R. (2012). Image processing onboard spacecraft for autonomous plume detection. *Planetary and Space Science*, **62**(1), 153–159.
- Wang, X., (2007). Laplacian operator-based edge detectors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **29**(5), 881–886.
- Wee, C. and Paramesran, R. (2007). On the computational aspects of Zernike moments. *Image and Vision Computing*, **25**(6), 967–980.
- Ye, J., Fu, G. and Poudel, U.P. (2005). High-accuracy edge detection with Blurred Edge Model. *Image and Vision Computing*, **23**(5), 453–467.
- Ying-Dong, Q., Cheng-Song, C., San-Ben, C. and Jin-Quan, L. (2005). A fast subpixel edge detection method using Sobel-Zernike moments operator. *Image and Vision Computing*, **23**(1), 11–17.
- Yu, M., Cui, H. and Tian, Y. (2014). A new approach based on crater detection and matching for visual navigation in planetary landing. *Advances in Space Research*, **53**(12), 1810–21.