Acta Astronautica 213 (2023) 177–196

Contents lists available at ScienceDirect

Acta Astronautica

journal homepage: www.elsevier.com/locate/actaastro

Research paper

Vision-based estimation of small body rotational state☆

Paolo Panicucci [a,b,c,1,*], Jérémy Lebreton [c], Roland Brochard [c], Emmanuel Zenou [a], Michel Delpech [b]

[a] ISAE-SUPAERO, Université de Toulouse, 10 Avenue Edouard Belin, 31400, Toulouse, France
[b] Future Mission Engineering Department, CNES, 18 Avenue Edouard Belin, 31400, Toulouse, France
[c] Sensor Processing Chain Department, Airbus Defence & Space, 31 Rue des Cosmonautes, 31400, Toulouse, France

ARTICLE INFO

Keywords:
Small body exploration
Vision-based estimation
Geometrical computer vision
Autonomous vision-based characterization

ABSTRACT

The heterogeneity of the small body population complicates the prediction of small body properties before the spacecraft's arrival. In the context of autonomous small body exploration, it is crucial to develop algorithms that estimate the small body characteristics before orbit insertion and close proximity operations. This paper develops a vision-based estimation of the small-body rotational state (i.e., the center of rotation and rotation axis direction) during the approach phase. In this mission phase, the spacecraft observes the rotating celestial body and tracks features in images. As feature tracks are the projection of the landmarks' circular movement, the possible rotation axes are computed. Then, the rotation axis solution is chosen among the possible candidates by exploiting feature motion and a heuristic approach. Finally, the center of rotation is estimated from the center of brightness. The algorithm is tested on more than 800 test cases with two different asteroids (i.e., Bennu and Itokawa), three different lighting conditions, and more than 100 different rotation axis orientations. Each test case is composed of about 250 synthetic images of the asteroid which are used to track features and determine the rotational state. Results show that the error between the true rotation axis and its estimation is below 10° for 80% of the considered test cases, implying that the proposed algorithm is a suitable method for autonomous small body characterization.

1. Introduction

The heterogeneity of small body characteristics is a crucial factor to consider during mission preparation and in-orbit operations. In particular, the limited knowledge of small bodies' properties imposes numerous challenges during mission design and operations. Shape, rotational state, and geophysical characteristics can strongly change depending on the small body under study and it is hard to predict the exact small body properties before the encounter. When a small body mission is planned, an observation campaign is performed to bound the characteristics of the small body under study and to estimate the range of uncertainty of these quantities [1]. In particular, preliminary shape, pole orientation, and rotation period can be deduced from light-curve inversion [2,3] and, if feasible, radar campaign [4]. The rotational period has shown high agreement with the light-curve inversion procedure and it can be estimated from on-board during far approach [5,6]. Rotation pole, shape, and gravity field estimation is a more complex problem and ground-based observations are often inaccurate.

On the one hand, the rotation pole orientation is directly coupled, through the rotational equations, to the inertial tensor which is deduced from the shape. Even though small bodies, in particular asteroids, are mainly principal axis rotators [7], fine shape estimation remains deeply coupled with the rotation pole inertial orientation. Current techniques to solve this problem rely on stereophotoclinometry or its variations [8–10]. On the other hand, the gravity field can be derived from the shape under the assumption of constant density [11–14] or gathered by solving the orbit determination problem by processing long trajectory arcs [15,16]. Unfortunately, these techniques strongly rely on human intervention to control the solution convergence and validate the output shape. Moreover, the required computational time makes it unsuitable for onboard applications. It is worth noting that all these techniques rely on communications with the ground implying radio signal delays and high costs.

To overcome these limitations, vision-based systems have been gaining attention as a cost-effective and accurate solution because they

☆ Preliminary results of this work were presented in "Panicucci, P., Autonomous vision-based navigation and shape reconstruction of an unknown asteroid during approach phase, Ph.D. thesis, ISAE-SUPAERO, 30 March 2021. URL http://www.theses.fr/2021ESAE0011.

* Correspondence to: Dipartimento di Scienze e Tecnologie Aerospaziali, Politecnico di Milano, Italy.
E-mail address: paolo.panicucci@polimi.it (P. Panicucci).

[1] Currently Assistant Professor at Dipartimento di Scienze e Tecnologie Aerospaziali, Politecnico di Milano.

https://doi.org/10.1016/j.actaastro.2023.08.046
Received 27 February 2023; Received in revised form 4 August 2023; Accepted 31 August 2023
Available online 9 September 2023

can provide real-time information to the spacecraft with limited impact on system budgets. In this context, it is crucial to design autonomous vision-based algorithms capable of supporting small-body parameter estimation without ground communications. The rotation pole, the shape, and the gravity field are of primary importance to allow the GNC architecture to perform critical mission phases like orbit insertion and close proximity characterization [17]. These quantities are strongly correlated and their onboard estimation is difficult to be performed independently. The gravity field depends on the small body shape and its density distribution. Therefore it can be derived from the shape under the assumption of constant density [11–14] or estimated during the localization procedure [18]. The shape can be deduced by shape from motion algorithms [6,19] or by methods solving for the SLAM (Simultaneous Localization And Mapping) problem [20–22]. The rotational state initial estimation is required for two main reasons. First, it is necessary to define a relative reference frame which implies the estimation of a point and three vectors. Second, the knowledge – even if coarse – of the small body rotational dynamics can help the convergence of the relative localization as proposed in Panicucci [20]. The rotational state can be determined implicitly by solving the SLAM problem as the spacecraft poses, i.e., positions and orientations, are estimated in the small-body-fixed reference frame. Alternatively, it can be determined during close approach when the small body is resolved and image processing algorithms can extract information about the small body rotation in time. Bandyonadhyay et al. [6] proposes a method to determine the rotation state of the small body by simultaneously optimizing the shape and the rotation state. The proposed method accurately estimates the pole orientation and the small body shape, but it requires a Monte-Carlo optimization which increases the computational burden. Bissonnette et al. [23] studies the estimation of the pole orientation from matched features. This work studies the possibility of detecting ellipses in the image and reconstructing the landmark's circles movements in 3D. In recent year several algorithms have been developed in the spacecraft close proximity community to solve for the angular velocity and inertia tensor f an unknown target. The pioneering work of Masutani et al. [24] studied the dimensionless inertia estimation from frame-to-frame tracking by exploiting the analytical solution of the Euler equations. Despite the angular velocity was not the focus of the estimation, angular velocity data were derived from noiseless synthetic data and used to determine the inertial tensor of a torque-free angular motion. Augenstein and Rock [25] proposes a Bayesian filtering approach to estimate the chaser transnational state, the target rotation state and its inertia parameter ratio. Despite the. Padial et al. [26] extended Augenstein and Rock [25]'s work by merging range measurements and vision data to solve the scale factor. Range measurements are also used in Lichter and Dubowsky [27] where the proposed methodology determines the angular state and the inertia matrix of a uncooperative target from range measurements filtered in a Kalman filter. Also Hillenbrand and Lampariello [28] uses range measurements with iteratively-reweighted-least-squares to determine the target rotational state and its inertia parameters. Results show agreement when measurements were available, but the target rotational state diverges quite rapidly, implying a coarse estimation of the rotational parameters. A purely vision-based approach was used by Tweddle et al. [29] by exploiting bundle adjustment in a graph-based SLAM framework to solve for the rotational and translational evolution of two uncooperative spacecrafts. Results show agreement with testing conducted on the ISS, leading also to a correct estimation of the target inertia ratios. A different approach is used in Setterfield et al. [30] where the polhode is analyze to estimate and predict the target rotational state and parameters. The previous references mainly focus on rotational state and inertia ratio estimation for artificial satellite close proximity, where the target undergoes high and variable rotation rate, the observational period is short, and the observed spacecraft is an human-made object. On the contrary, the algorithm presented in this work studies small body applications which differs in several aspect to human-made uncooperative target. First the rotation axis is generally fixed and the rotational period is long. This implies long tracking windows which are challenging for feature tracking and matching algorithms. Second, the observed body is composed of rocky and dusty terrain implying different performance on image processing performance.

Therefore, in the context of making space exploration more autonomous and avoid to perform long-lasting ground-based characterization before launch, the current study aims to develop an algorithm that estimate the rotation state of an asteroid during the approach phase. The main motivation is to avoid relying to ground-based observations and to purely rely on images obtained during the preliminary characterization phase. It is worth to underline the paradigm shift between the this approach and the ground-based lightcurve inversion. Lightcurve inversion processes photometric data from ground-based observations which can span up to several years [31]. Data are validated and chosen by ground-based operators who plan carefully the observations schedule to improve the output accuracy and remove outliers caused by corrupted observation. The rotation pole ambiguity (see Sections 6.3.2 and 6.4.2 for details) is solved by exploiting multiple observational geometry and long-lasting observations, even thought this is a complex task for low-inclination asteroid orbit [31]. Moreover, despite pole orientation estimation seems to provide valuable and reliable results for near-Earth asteroids, Hanuš et al. [32] shows that pole orientation estimation errors can be considerably higher (i.e., 10 degrees of standard deviation for the pole latitude and 5 degree for the pole longitude). Finally, the computational burden of the lightcurve inversion take often dozens of hours of computing time on a desktop computer [31]. On the contrary, the proposed paradigm wants to exploit short observation timeframe during in-situ exploration to determine the pole orientation. The proposed approach works with hour-lasting observations (i.e., the observation time is comparable with the asteroid rational period) by exploiting geometrical information from image processing and computer vision algorithms developed for autonomous and on-board applications.

In this context, this paper develops an algorithm to autonomously determine the small body rotational state during close approach without the need of performing the localization task starting from Bissonnette et al. [23]'s work. The proposed approach also estimates the small-body-fixed reference frame origin leading to the definition of the small-body-fixed reference frame. This task is crucial to be performed before the localization task because it enables changing the navigation from the inertial reference frame to the small-body-fixed reference frame. By tracking points extracted from small body images taken during the approach, it is possible to retrieve the circular movement of the landmark in the inertial frame. Indeed, the trajectory of each 3D landmark is a circle having as center a point on the rotation axis and lying on a plane defined by the same axis. By assuming known the rotational period, the feature circular movements provide an estimate of all the possible rotation axes. Multiple solutions are present because the camera cannot retrieve the movement of the feature along its boresight. The correct one is determined by using a heuristic approach. The algorithm is tested through numerical simulation with synthetic images in the loop for a wide range of observation geometries and two different asteroids, i.e., Bennu and Itokawa. Finally, numerical results are presented and the algorithm's performance and limitations are discussed.

2. Notation

In this paper the following notation is used:

- 3D vectors are in lower case bold text, such as $\boldsymbol{r}$, and 2D vectors are in upper case bold, such as $\boldsymbol{R}$.
- Matrices are in plain text in brackets, such as $[A]$
- Vector initialization are performed with parenthesis, such as $\boldsymbol{b} = \left(\boldsymbol{a}^T\ \boldsymbol{a}^T\right)^T$

- $\mathcal{A} = \{a, \boldsymbol{a}_1, \boldsymbol{a}_2, \boldsymbol{a}_3\}$ is a 3D reference frame centered in a with axes $\boldsymbol{a}_1$, $\boldsymbol{a}_2$, and $\boldsymbol{a}_3$. All the reference frame are right-handed.
- $\mathbb{A} = \{A, \boldsymbol{A}_1, \boldsymbol{A}_2\}$ is a 2D reference frame. This reference frame is centered in A with axes $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$.
- The vector $\boldsymbol{r}$ expressed in the $\mathcal{S}$ reference frame is denoted ${}^{\mathcal{S}}\boldsymbol{r}$
- The homography matrix from $\mathbb{S}$ to $\mathbb{C}$ is $[\mathbb{CS}]$.
- The vector $\boldsymbol{R}$ expressed in the $\mathbb{S}$ reference frame is denoted ${}^{\mathbb{S}}\boldsymbol{R}$
- The angular velocity of reference frame $\mathcal{B}$ with respect to reference frame $\mathcal{N}$ is labeled $\boldsymbol{\omega}_{\mathcal{B}/\mathcal{N}}$

3. Problem statement and algorithm overview

During close approach to a small body, the spacecraft moves at low velocity with respect to its target and acquires images of the small body to perform characterization. The proposed algorithm is conceived to be used during the early characterization phase during which the shape and the rotation state are coarsely estimated before performing orbit insertion [33]. Note that the small body gravity field is negligible in the dynamics as the early characterization is performed during the approach, thus the spacecraft motion is fully determined by the Sun gravity and deep-space perturbations. The spacecraft observes with an onboard camera the small body which is resolved in the image and rotates around its rotation axis which is assumed fixed in the inertial reference frame. Note that the small body must be resolved to enable the possibility of tracking features on its surface. Moreover, to ensure image processing accuracy and precision during tracking, it is convenient to process images where the small body occupies a large portion of the field of view (e.g., hundreds of pixels), such that features can be correctly identified and followed.

The observation geometry is depicted in Fig. 1 where the main geometrical entities are defined. In particular:

- The approach angle is the angle between the small body rotation axis and the approach direction, i.e., the camera-small-body direction.
- The illumination angle or phase angle is defined as the angle between the approach direction and the Sun-small-body direction.
- The obliquity, also known as the axial tilt, is the angle between the small-body rotation axis and the small-body orbital plane.

Note that the approach angle defines which hemisphere of the small body is observed during the approach. Moreover, the illumination angle is an indicator of possible shadows due to the terminator line and self-shadowing. A more detailed analysis of shadow generation can be found in Panicucci et al. [19]. Finally, the obliquity defines which hemisphere is illuminated at approach time, thus which is the observable part of the small body with a vision-based sensor. The combination of these three angles defines the appearance of the small body in the image, which influences the performance of the image processing and vision-based system under study.

Note that the movement of the spacecraft in the small-body-fixed reference frame can be decomposed in the motion of the spacecraft in the inertial reference frame and the motion due to the small-body rotational dynamics. As the spacecraft's inertial trajectory is usually dynamically slower than the rotational dynamics of the small body, the motion of the latter in the images is mainly due to its rotation. When this is not the case as in fast fly bys, an estimation of the spacecraft inertial poses can be exploited to reconstruct the inertial epipolar geometry between the different poses and correct the images for the inertial spacecraft rototranslational motion. From images of the rotating small body, it is possible to extract and follow 2D features which represent the projection of surface 3D landmarks. In the inertial reference frame landmarks, which are points anchored to the small body, rotate according to the small-body rotational dynamics. When considering a principal axis rotator, the landmark trajectory in the inertial space is circular. This movement is projected as a conic or as a single line in the camera frame (see Section 6.1 for a more rigorous analysis). By fitting the conics from the observed small body feature tracks, it is possible to reconstruct the circle traced by the landmarks. This procedure provides a direct estimation of all possible solutions for the orientation of the small body rotation axis. As outlined in detail in Section 6, some solutions can be pruned by exploiting the information about feature movement in the images. Others cannot be removed without information about the landmarks' movement in the boresight direction (which is not observable), thus a heuristic approach is exploited to determine the correct solution.

The algorithm is composed as follows:

1. The small-body-fixed reference frame origin is estimated from the first image to define the small-body-fixed reference frame origin as outlined in Section 5.
2. Images are processed sequentially to identify in the image the projection of the circle associated with the landmark movement. Several features are extracted which implies that several landmarks are observed.
3. The algorithm decides whether to consider the conics as degenerated. This is a crucial algorithm step as the optimization procedure is different between the two cases (see Section 6 for more details).
4. The conics are fitted and the possible solutions for the rotation axis orientation are identified. This process is outlined for the not-degenerate case in Section 6.3 and for the degenerate case in Section 6.4.
5. The solution is identified by discarding unfeasible solutions. This process strongly depends on the degeneracy of the observed conics as outlined in Sections 6.3 and 6.4.

The main assumptions are:

1. The rotational period is known from light curves. This is a standard estimation from ground-based campaigns [16] or far-approach photometric studies [5,6].
2. The distance between the small body and the probe is known for the first camera view. This is required to break the scale ambiguity and to place the small body reference frame at the time of the first camera view. This could be considered a strong assumption but a wrong estimation of the distance of the first camera view, or equivalently the scale factor, would simply induce an overall scaling of the solution. The small-body-spacecraft distance can be determined by a preliminary scale estimation through ΔV ranging with circle or ellipsoid fitting [34,35].
3. The small body is observed for one rotation period to enable the spacecraft to observe long feature tracks needed to bound the rotation axis estimation.

Note that the knowledge of the distance between the small body and the probe only affects the determination of the small-body reference frame origin (see Section 4). The image processing pipeline developed in the following sections is not affected by the scale factor as the calculations are directly performed on image points, leading to be scale invariant. Indeed, the possible solution for the rotation axis are scale invariant as well because their calculation is computed from the eigenvectors which are known up to a scale factor (see Sections 6.3.2 and 6.4.2). It is worth noting that the small-body reference frame origin is used to discard unfeasible solutions for the not-degenerate case (see Section 6.3.3), but a change in the scale – due to an error in the asteroid–spacecraft range – would affect only the small body overall size without any impact in the pruning process. In other words, the spacecraft would estimate points moving on a larger small body if the range is larger than the truth. Otherwise, the spacecraft would estimate points moving a smaller celestial body.

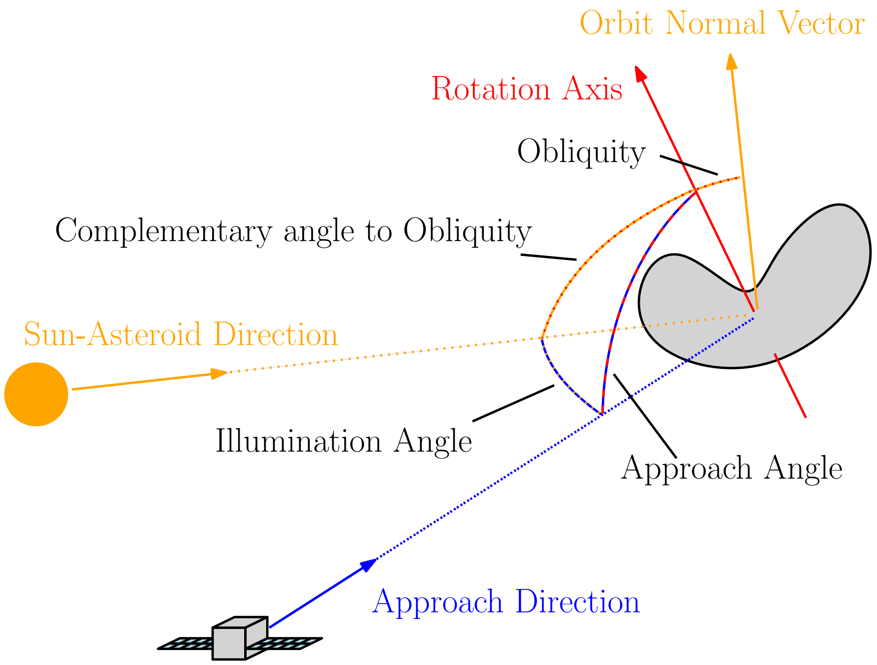


Fig. 1. The observational geometry during small body approach.

4. Determination of the small-body-fixed reference frame origin

In this section the estimation of an approximation of the rotation center is presented. From rigid body dynamics, the rotation axis passes through the small body barycenter, thus the rotation center is coincident with the barycenter. As the probe is not orbiting around the small body, the mass distribution, and thus the barycenter, cannot be determined. To do so, Bandyonadhyay et al. [6] proposes to determine the center of rotation as the average between a given image and the one with the asteroid rotated of 180 degrees. This approach can be useful for low illumination angle, where self-shadowing does not influence the center of brightness estimation. A different approach foresees the computation of the center of brightness and to apply a correction to estimate the projection of the center of mass [36,37]. For the sake of simplicity. the center of brightness is computed only for the first image in this work and no correction is applied. Only the first image is used as the distance to the small body is considered known only for that instant. Note that the center of brightness is usually close to the center of mass only for low illumination angles, as the center of brightness can shift considerably when the illumination angle increases. In the framework of this work, no correction is applied for the sake of simplicity and future work should investigate the benefit of adding the correction within the proposed algorithmic pipeline.

To compute the center of brightness, the first image is binarized by using Otsu's method [38] to identify the bright pixels. By labeling χ_1 the bright pixel map in the first image, the center of brightness ${}^{\mathbb{C}_1}\boldsymbol{R}_{\mathrm{CB}}$ expressed in the first image reference frame $\mathbb{C}_1$ is computed as

$$ {}^{\mathbb{C}_1}\boldsymbol{R}_{\mathrm{CB}} = \left(\frac{\sum_{P\in\chi_1} \boldsymbol{P}_x}{|\chi_1|}, \frac{\sum_{P\in\chi_1} \boldsymbol{P}_y}{|\chi_1|} \right)^T \tag{1} $$

where $\boldsymbol{P}$ is an arbitrary pixel in χ_1, the subscripts x and y label the x and y coordinates in the image, and $|\chi_1|$ is the area of χ_1.

By knowing the distance from the first camera pose to the small body, the rotation center $\boldsymbol{r}_{\mathrm{RC}}$ is computed by backprojecting in the 3D space the center of brightness:

$$ {}^{\mathcal{N}}\boldsymbol{r}_{\mathrm{RC}} = {}^{\mathcal{N}}\boldsymbol{r}_{\mathrm{SC}_1} + \frac{d_{\mathrm{SC}_1/\mathrm{SB}}}{\left\| [K]^{-1}\, {}^{\mathbb{C}_1}_{h}\boldsymbol{R}_{\mathrm{CB}} \right\|} [NC_1]\,[K]^{-1}\, {}^{\mathbb{C}_1}_{h}\boldsymbol{R}_{\mathrm{CB}} \tag{2} $$

where $[K]$ is the camera calibration matrix, $\boldsymbol{r}_{\mathrm{SC}_1}$ is the spacecraft position at the first image time, $d_{\mathrm{SC1}}/\mathrm{SB}$ is the small-body-spacecraft distance at the first image time, and $\left[NC_1\right]$ is the rotation matrix from the camera frame C_1 at the first image time to the inertial frame $\mathcal{N}$. Note that the camera calibration matrix can be computed from camera characteristics as follows:

$$ [K] = \begin{bmatrix} \frac{f}{s_x} & 0 & C_x \\ 0 & \frac{f}{s_y} & C_y \\ 0 & 0 & 1 \end{bmatrix} \tag{3} $$

where f if the camera focal length, s_x and s_y are the pixel physical size in x and y components, and $\boldsymbol{C} = \left(C_x, C_y\right)^T$ is the camera center in pixels.

The computation of ${}^{\mathcal{N}}\boldsymbol{r}_{\mathrm{RC}}$ defines the origin of the small-body-fixed reference frame $\mathcal{B}$. In order to fully define $\mathcal{B}$, the angular velocity $\boldsymbol{\omega}_{\mathcal{B}/\mathcal{N}}$ of the small-body-fixed reference frame $\mathcal{B}$ with respect to the inertial reference frame $\mathcal{N}$ must be characterized. As the rotational period is known, only the unit vector of the rotational axis $\hat{\boldsymbol{\omega}}_{\mathcal{B}/\mathcal{N}}$ must be estimated.

5. Feature extraction and tracking

5.1. Feature-based image processing

The first step of the algorithm is to process images to detect landmarks' movement. From subsequent images, information about the relative motion of the scene can be recovered. The apparent motion of the scene and objects in the image is called optical flow, which is caused by both the observer's and the object's motion. The optical flow estimation provides information about the relative movement between the observer and the scene.

A standard procedure to perform this task is to use a feature-based image processing algorithm. Features are usually 2D entities that are characterized by a location and a compact description of the feature information. As a consequence, two tasks are performed to identify a feature [39]: feature detection and feature description. Feature extraction deals with the problem of determining the feature locations in the image. Many algorithms (e.g., SURF [40], BRISK [41], ORB [42], SIFT [43]) are available to perform this task. The main difference is the type of structure (e.g., corners, edges, or blobs) they try to detect and the required computational effort. Feature description is designed to build a compact and informative descriptor of the information contained in the feature location neighborhood designed to be non-redundant and optimized for feature association. Feature detection algorithms often have an associated descriptor (e.g., SURF [40], BRISK [41], and KAZE [44]), but any combination of feature description and detection is possible. Once features are detected from different images, it is necessary to determine their path to gather the optical flow. This is a complex procedure as feature appearance can significantly change from different points of view and illumination conditions. Moreover, features can be shadowed during their motion which implies their loss in the image. This task can be fulfilled with two different approaches [39]: feature matching and feature tracking. On the one hand, feature matching exploits feature description to perform feature association between different images. A naive and widely-used method to perform this task is brute force matching where features in one image are associated according to the closest descriptor in the other image. As feature matching is based on feature description, it is usually exploited when images are generated from different observation geometries and when the motion is rapidly changing in time. On the other hand, feature tracking determines feature association by searching in the neighborhood of the feature location. A classical approach to perform feature tracking is Kanade–Lucas–Tomasi (KLT) algorithm [45,46]. The KLT tracker exploits the spatial gradient of image intensity to find the feature in the next image by Newton–Raphson gradient descent algorithm.

In this work SURF features are used to identify salient points in small body images. They are chosen as they are extracted faster than other features and are robust to camera rotation and translation. Feature association is then performed with the KLT algorithm as it does not require the use of time-consuming descriptor algorithms.

5.2. Kanade–Lucas–Tomasi optical flow tracker

In this section, an overview of the KLT algorithm is provided as depicted in Lucas and Kanade [45] and Tomasi and Kanade [46]. Let I and J be two different grayscale images. The aim is to find the displacement $\boldsymbol{D}$ of the feature $\boldsymbol{P}$ from I to J by minimizing the error between the two images. The error function ε between the two images is defined by computing the squared difference in intensity in a region of interest R which is normally considered a user-defined rectangle or square. As a consequence:

$$\varepsilon = \sum_{R} \left(I\left(\boldsymbol{P}+\boldsymbol{D}\right) - J\left(\boldsymbol{P}\right)\right)^2 \tag{4}$$

Image I can be linearly approximated:

$$\varepsilon \simeq \sum_{R} \left(I\left(\boldsymbol{P}\right) + \left[\frac{\partial I}{\partial \boldsymbol{D}}\right] \boldsymbol{D} - J\left(\boldsymbol{P}\right)\right)^2 \tag{5}$$

where the spatial partial derivative $\left[\frac{\partial I}{\partial \boldsymbol{D}}\right]$ can be computed by finite differences. The error can be thus minimized:

$$\begin{aligned}\left[\frac{\partial \varepsilon}{\partial \boldsymbol{D}}\right] \simeq & \frac{\partial}{\partial \boldsymbol{D}} \sum_{R} \left(I\left(\boldsymbol{P}\right) + \left[\frac{\partial I}{\partial \boldsymbol{D}}\right] \boldsymbol{D} - J\left(\boldsymbol{P}\right)\right)^2 \\ =& 2\sum_{R} \left(I\left(\boldsymbol{P}\right) + \left[\frac{\partial I}{\partial \boldsymbol{D}}\right] \boldsymbol{D} - J\left(\boldsymbol{P}\right)\right)^T \left[\frac{\partial I}{\partial \boldsymbol{D}}\right] = 0\end{aligned} \tag{6}$$

The displacement $\boldsymbol{D}$ is analytically computed:

$$\boldsymbol{D} = -\left(\sum_{R} \left[\frac{\partial I}{\partial \boldsymbol{D}}\right]^T \left[\frac{\partial I}{\partial \boldsymbol{D}}\right]\right)^{-1} \left(\sum_{R} \left[\frac{\partial I}{\partial \boldsymbol{D}}\right]^T \left(I\left(\boldsymbol{P}\right) - J\left(\boldsymbol{P}\right)\right)\right) \tag{7}$$

In the KLT algorithm, not all features are tracked as only a subset is informative for the tracker. Good features are defined internally from the tracking algorithm as the ones having the highest eigenvalues of the matrix $\left(\sum_R \left[\frac{\partial I}{\partial \boldsymbol{D}}\right]^T \left[\frac{\partial I}{\partial \boldsymbol{D}}\right]\right)$ [47]. This ensures that feature tracking is a well-conditioned process and that the algorithm is numerically stable. Moreover, to increase algorithm performance and to exploit different detail levels in the image a pyramidal scheme is often used [48].

In the present work the feature extraction and tracking are performed with Airbus Defence & Space software Themis [49]. Themis is an enhanced space-certified KLT tracker which compensates for features rotation, scaling, and translation. Moreover, it implements a pyramidal scheme and homography filtering to improve the algorithm's overall performance. The use of Themis is justified by the fact that it has been optimized for space application, and it has shown its embeddability in space-certified processors LEON4 [49].

The output of Themis is a series of feature tracks of different length. These tracks are exploited to fit the conic obtained from the circle projection in the image and to determine the rotation axis orientation.

6. Estimation of small body rotation axis

6.1. Projection of the circle

Before entering into the details of the algorithm fitting the conics generated by the projection of the tracked landmarks, it is worth analyzing how a 3D circle projects into a camera.

Let $\mathcal{P}$ the 3D reference centered in the circle origin and whose third axis oriented as the circle plane normal $\boldsymbol{n}_{\mathrm{cl}}$. The circle plane is the plane where the circle lies and where the first two axes of $\mathcal{P}$ lie. Moreover, let $\mathbb{P}$ be the 2D reference frame centered in circle center projected on the circle plane and whose axes are oriented as the projection on the circle plane of the first two axes of $\mathcal{P}$. In $\mathbb{P}$, the equation of the circle can be expressed as follows:

$${}^{\mathbb{P}}_{h}\boldsymbol{P}^T \begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & -R^2\end{bmatrix} {}^{\mathbb{P}}_{h}\boldsymbol{P} = {}^{\mathbb{P}}_{h}\boldsymbol{P}^T \left[C\right] {}^{\mathbb{P}}_{h}\boldsymbol{P} = 0 \tag{8}$$

where R is the circle radius and ${}^{\mathbb{P}}_{h}\boldsymbol{P}^T$ is a 2D homogeneous point expressed in $\mathbb{P}$ and belonging to the circle plane defined by $\boldsymbol{n}_{\mathrm{cl}}$. More details about homogeneous coordinates can be found in Hartley and Zisserman [50]. By denoting $\mathbb{C}$ the image plane reference, the mapping from the circle plane to the image plane is defined by the following homography [50]:

$$\left[\mathbb{CP}\right] = \left[K\right]\left[{}^{C}\boldsymbol{p}_1\ {}^{C}\boldsymbol{p}_2\ {}^{C}\boldsymbol{r}_{\mathrm{cl}}\right] \tag{9}$$

where ${}^{C}\boldsymbol{p}_i$ is the ith axis of $\mathcal{P}$ expressed in $\mathcal{C}$ and ${}^{C}\boldsymbol{r}_{\mathrm{cl}}$ is the vector from the camera to the circle origin in $\mathcal{C}$. Note that ${}^{C}\boldsymbol{p}_i$ is also the ith column of the rotation matrix $[CP]$ from $\mathcal{P}$ to the camera frame $\mathcal{C}$.

The circle as projected in the camera can be computed by noticing that:

$${}^{\mathbb{C}}_{h}\boldsymbol{P} = \left[\mathbb{CP}\right] {}^{\mathbb{P}}_{h}\boldsymbol{P} \tag{10}$$

Thus:

$${}^{\mathbb{C}}_{h}\boldsymbol{P}^T \left[\mathbb{CP}\right]^T \left[C\right] \left[\mathbb{CP}\right] {}^{\mathbb{P}}_{h}\boldsymbol{P} = {}^{\mathbb{C}}_{h}\boldsymbol{P}^T \left[E\right] {}^{\mathbb{C}}_{h}\boldsymbol{P} = 0 \tag{11}$$

where $[E]$ is the conic representing the projection of the circle in the image. If $[E]$ is full rank, the conic represents an ellipse, an hyperbola, or a parabola. Otherwise, the conic degenerates to a line. As the circle is not a degenerate conics and its matrix has full rank, it is worth noticing that the conics degenerate to a line only if the homography matrix has not full rank. This happens when the camera boresight is perpendicular to the circle plane normal, i.e., $\boldsymbol{p}_3^T \boldsymbol{r}_{\mathrm{cl}} = 0$.

To prove that the projection is a single line, it is worth demonstrating that the projection of $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ in the image are parallel. For the sake of simplicity, let $\hat{\boldsymbol{r}}_{\mathrm{cl}}$ be the unit vector pointing as $\boldsymbol{r}_{\mathrm{cl}}$. Note that $\hat{\boldsymbol{r}}_{\mathrm{cl}}$ is linearly dependent of $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$, thus $\hat{\boldsymbol{r}}_{\mathrm{cl}} = \hat{\boldsymbol{r}}_{\mathrm{cl}}^T \boldsymbol{p}_1 \boldsymbol{p}_1 + \hat{\boldsymbol{r}}_{\mathrm{cl}}^T \boldsymbol{p}_2 \boldsymbol{p}_2$. The vector perpendicular to the camera boresight unit vector are:

$$\boldsymbol{e}_1 = \boldsymbol{p}_1 - \hat{\boldsymbol{r}}_{\mathrm{cl}}^T \boldsymbol{p}_1 \hat{\boldsymbol{r}}_{\mathrm{cl}} \tag{12}$$

$$e_2 = p_2 - \hat{r}_{\mathrm{cl}}^T p_2 \hat{r}_{\mathrm{cl}} \tag{13}$$

Note that these vectors belongs to the circle plane and to the image plane. To prove that these vectors are parallel, it must be proven that $e_1 \times e_2 = 0$. Thus:

$$\begin{aligned} e_1 \times e_2 &= p_1 \times p_2 - \hat{r}_{\mathrm{cl}}^T p_2 p_1 \times \hat{r}_{\mathrm{cl}} - \hat{r}_{\mathrm{cl}}^T p_1 \hat{r}_{\mathrm{cl}} \times p_2 + \hat{r}_{\mathrm{cl}}^T p_1 \, \hat{r}_{\mathrm{cl}}^T p_2 \hat{r}_{\mathrm{cl}} \times \hat{r}_{\mathrm{cl}} \\ &= p_3 - \left(\hat{r}_{\mathrm{cl}}^T p_2\right)^2 p_3 - \left(\hat{r}_{\mathrm{cl}}^T p_1\right)^2 p_3 \\ &= \left(1 - \left(\hat{r}_{\mathrm{cl}}^T p_2\right)^2 - \left(\hat{r}_{\mathrm{cl}}^T p_1\right)^2\right) p_3 \\ &= \left(1 - \|\hat{r}_{\mathrm{cl}}\|^2\right) p_3 = 0 \end{aligned} \tag{14}$$

This implies that the degenerate conic is composed of a single line. Therefore either the circle is projected as a full rank conic or to a rank-1 degenerate conic. No other situations are possible. From an operative perspective, this suggests that the algorithm must be able to cope with full-rank conic estimation and with single line fitting.

When the conic is full rank, it is represented by a full rank matrix which is usually written as follows:

$$[E] = \begin{bmatrix} A & \frac{B}{2} & \frac{D}{2} \\ \frac{B}{2} & C & \frac{E}{2} \\ \frac{D}{2} & \frac{E}{2} & F \end{bmatrix} \tag{15}$$

Otherwise, when the conic degenerates, it is represented by the following equation [50]

$$[E] = M L^T + L M^T \tag{16}$$

where M and L are the two lines composing the degenerate conic. As the two lines coincide for the circle projection, $L = M$. Thus:

$$[E] = 2 L L^T \tag{17}$$

which implies that Eq. (11) can be rewritten as the classical line equation ${}^{\mathbb{C}}_{h}P^T L = 0$.

As multiple features are tracked by Themis, several 3D circles are projected in the image, so different ellipses or lines are present. Note that the 3D landmark movement is caused by the rotational dynamics of the small body. Therefore, landmarks move on 3D circles whose circle plane normal is the small body rotation axis. Thus, the 3D circles all have the same circle plane normal and have different origins lying on the rotation axis.

6.2. Detection of degenerate solutions

As shown in Section 6.1, the projection of the 3D circle, generated by the landmarks, is a series of ellipses when the camera boresight is not perpendicular to the circle plane normal. As the circle plane normal is the rotation axis direction for all the 3D circles, this situation happens when the rotation axis is tilted with respect to the camera boresight. Otherwise, the projection of the 3D circle degenerates into a sheaf of parallel lines. It is thus necessary to develop a method to detect whether the rotation axis is perpendicular to the camera boresight only using tracked features. To do so, it is proposed hereafter a detection algorithm to understand if the conic projection is degenerate.

Recall that a point lying on an ellipse can be represented as follows:

$$A\,\xi^2 + B\,\xi\eta + C\,\eta^2 + D\,\xi + E\,\eta + F = 0 \quad \text{with} \quad 4AC - B^2 > 0 \tag{18}$$

where (ξ, η) are Cartesian coordinates obtained from the image coordinates through normalization with respect to the image size. The normalization is required to let the following steps of the algorithm be independent with respect to the image size.

All the reprojected points of the same track must verify Eq. (18). Thus:

$$\begin{bmatrix} \xi_1^2 & \xi_1\eta_1 & \eta_1^2 & \xi_1 & \eta_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \xi_i^2 & \xi_i\eta_i & \eta_i^2 & \xi_i & \eta_i & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \xi_{N_j}^2 & \xi_{N_j}\eta_{N_j} & \eta_{N_j}^2 & \xi_{N_j} & \eta_{N_j} & 1 \end{bmatrix} \begin{pmatrix} A_j \\ B_j \\ C_j \\ D_j \\ E_j \\ F_j \end{pmatrix} = \left[\left[D_{2,j}\right] \; \left[D_{1,j}\right] \right] \begin{pmatrix} A_j \\ B_j \\ C_j \\ D_j \\ E_j \\ F_j \end{pmatrix} = \left[\mathbb{0}_{N_j \times 1}\right] \tag{19}$$

where N_j is the number of frames during which the 3D point is tracked for the jth conic, $\left[\mathbb{0}_{n\times m}\right]$ is the zero $n \times m$ matrix, the j subscript define the parameters associated with the jth conic, $\left[D_{2,j}\right] \in \mathbb{R}^{N_j \times 3}$ denotes the matrix associated with the second-order conic coefficients, and $\left[D_{1,j}\right] \in \mathbb{R}^{N_j \times 3}$ the matrix associated with the remaining-orders conics coefficients.

Recall that a point lying on a line can be represented as follows:

$$D\,\xi + E\,\eta + F = 0 \quad \text{with} \quad D^2 + E^2 = 1 \tag{20}$$

Moreover, the points of the jth line verify:

$$\begin{bmatrix} \xi_1 & \eta_1 & 1 \\ \vdots & \vdots & \vdots \\ \xi_i & \eta_i & 1 \\ \vdots & \vdots & \vdots \\ \xi_{N_j} & \eta_{N_j} & 1 \end{bmatrix} \begin{pmatrix} D_j \\ E_j \\ F_j \end{pmatrix} = \left[D_{1,j}\right] \begin{pmatrix} D_j \\ E_j \\ F_j \end{pmatrix} = \left[\mathbb{0}_{N_j \times 1}\right] \tag{21}$$

As the lines of the matrix in Eq. (21) are linearly dependent, the rank of the matrix $\left[D_{1,j}\right]$ is one. Due to tracking errors, the rank is not one. Nevertheless, being $\left[D_{1,j}\right]$ nearly singular, its condition number is high. Note that this is only true when the conics are degenerate. As $\left[D_{1,j}\right]$ has N_j lines, it is more convenient to work with $\left[S_{3,j}\right] = \left[D_{1,j}\right]^T \left[D_{1,j}\right] \in \mathbb{R}^{3\times3}$.

To understand if the family of conics degenerates, the following steps are applied:

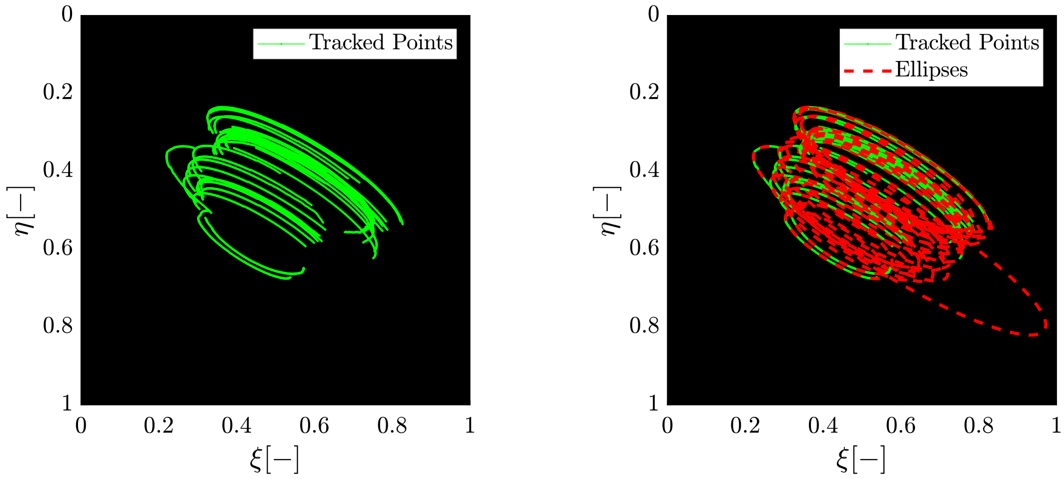
1. For each tracked feature, the track length is computed to store only the longest M_{curv} tracks. This step is necessary to avoid processing a high number of tracks which increases the numerical burden. Moreover, it is worth noting that long-tracked features are the ones that provide more information about the rotational state. M_{curv} is set to 50 in this work. The maximum number of curves number can be also set free and select a minimum tracking time to reject short tracks. The proposed approach is used in this work also to limit the computational burden.
2. For each tracked feature, the conditioning number of $\left[S_{3,j}\right]$ is computed. If the tracks conditioning number is greater than a threshold γ_{CN} for more than γ_{sheaf}, the conics are considered degenerate and processed as a sheaf of parallel lines. Otherwise, they are considered a family of ellipses. γ_{CN} and γ_{sheaf} are set to 10^5 and 0.8 respectively in this work. See Section 7.3 for comments about parameter selection.

This procedure detects autonomously whether the rotation axis is perpendicular to the camera boresight. As the 3D circle projection generates different conics, the estimation procedure is not identical between the two cases and two algorithms must be developed. First, in Section 6.3 the general case of a tilted axis with respect to the camera boresight is presented. Then, in Section 6.4 the algorithm when the two axes are perpendicular is explained.

6.3. Rotation axis tilted with respect to camera boresight

6.3.1. Ellipses fitting

If the rotation axis is tilted with respect to the camera boresight, the projection of the concentric circles is a family of ellipses. As the



(a) The tracked points

(b) The ellipse reconstruction with same orientation ϕ

Fig. 2. Example of the ellipse fitting from tracked features. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

3D circle origins belong to the same axis, the ellipse semi-major axes are all oriented in the same direction [23]. The optimization to gather the ellipses family exploits this property to compute a series of ellipses with the same orientation. This optimization technique is outlined in Bissonnette et al. [23] and reviewed hereafter.

Let M be the number of ellipses to be found. The orientation ϕ_j of the jth ellipse according to its Cartesian representation (see Eq. (18)) is:

$$\tan 2\phi_j = \frac{B_j}{C_j - A_j} \tag{22}$$

Eq. (22) shows that to obtain a family of ellipses with the same orientation, the fraction of the three Cartesian parameters must be the same for all the M ellipses. By introducing $k_j = C_j - A_j$, it is straightforward to impose that $B_j = B$ and $k_j = k\ \forall j \in [1, M]$ to implicitly verify the constraint in Eq. (22). Thus, the equation of the jth ellipses becomes [23]:

$$A_j\left(\xi^2+\eta^2\right) + B\xi\eta + k\eta^2 + D_j\,\xi + E_j\,\eta + F_j = 0 \tag{23}$$

By defining the vectors $\boldsymbol{a}_1 = \left(B, k, A_1, \ldots, A_M\right)^T$ and $\boldsymbol{a}_2 = \left(D_1, E_1, F_1, \ldots, D_M, E_M, F_M\right)^T$ and the matrices

$$[D_2] = \begin{bmatrix}
\xi_{1,1}\eta_{1,1} & \eta^2_{1,1} & \xi^2_{1,1}+\eta^2_{1,1} & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\xi_{N_1,1}\eta_{N_1,1} & \eta^2_{N_1,1} & \xi^2_{N_1,1}+\eta^2_{N_1,1} & 0 & 0 & 0 \\
\xi_{1,2}\eta_{1,2} & \eta^2_{1,2} & 0 & \xi^2_{1,2}+\eta^2_{1,2} & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\xi_{N_2,2}\eta_{N_2,2} & \eta^2_{N_2,2} & 0 & \xi^2_{N_2,2}+\eta^2_{N_2,2} & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
\xi_{1,M}\eta_{1,M} & \eta^2_{1,M} & 0 & 0 & 0 & \xi^2_{1,M}+\eta^2_{1,M} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\xi_{N_M,M}\eta_{N_M,M} & \eta^2_{N_M,M} & 0 & 0 & 0 & \xi^2_{N_M,M}+\eta^2_{N_M,M}
\end{bmatrix} \tag{24}$$

$$[D_1] = \begin{bmatrix}
\xi_{1,1} & \eta_{1,1} & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\xi_{N_1,1} & \eta_{N_1,1} & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \xi_{1,2} & \eta_{1,2} & 1 & 0 & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & \xi_{N_2,2} & \eta_{N_2,2} & 1 & 0 & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \xi_{1,M} & \eta_{1,M} & 1 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \xi_{N_M,M} & \eta_{N_M,M} & 1
\end{bmatrix} \tag{25}$$

where $\left(\xi_{i,j}, \eta_{i,j}\right)$ is the ith point of the jth ellipse in normalized coordinates, the problem is to be rewritten as:

$$\left[[D_2]\ \ [D_1]\right]\begin{pmatrix}\boldsymbol{a}_1\\ \boldsymbol{a}_2\end{pmatrix} = [D]\,\boldsymbol{a} = \left[\mathbb{0}_{(4M+2)\times 1}\right] \tag{26}$$

The solution of this homogeneous equation is found by minimizing $J = \boldsymbol{a}^T\,[D]^T\,[D]\,\boldsymbol{a}$ subject to $\boldsymbol{a}^T\boldsymbol{a} = 1$.

This optimization can be solved by solving the rank-deficient generalized eigenvalue system associated with the problem [51]. To have a numerically-stable computation the algorithm proposed by Halır and Flusser [52] is used to compute the solution. By defining $\left[S_1\right] = \left[D_1\right]^T\left[D_1\right]$, $\left[S_2\right] = \left[D_2\right]^T\left[D_1\right]$ and $\left[S_3\right] = \left[D_2\right]^T\left[D_2\right]$, the solution for $\boldsymbol{a}_1$ is the eigenvector associated with the smallest positive eigenvalue of $\left(\left[S_1\right] - \left[S_2\right]\left[S_3\right]^{-1}\left[S_2\right]^T\right)$. The solution for $\boldsymbol{a}_2$ is the computed as $\boldsymbol{a}_2 = -\left[S_3\right]^{-1}\left[S_2\right]^T\boldsymbol{a}_1$.

This provides a family of conics all with the same orientation. Note that the conics are not necessary ellipses as no constraint has been imposed on the Cartesian coefficients to verify it. Thus, for each conic, the eccentricity is computed and, if greater or equal to 1, the conic is discarded. In Figs. 2(a) and 2(b) the different optimization outputs for the main steps are shown.

6.3.2. Computation of the rotation axis candidates

Once the ellipses family is found, the rotation axis can be reconstructed by knowing that a 2D ellipse in the image is the projection of a 3D circle [53]. From a single ellipse, an elliptic cone, i.e. a cone with an elliptical cross-section, is generated from backprojection. If a sheaf of planes is intersected with the cone, only two planes generate circles when intersected [53]. This geometrical construction is shown in Fig. 3. In the figure, the red ellipse is backprojected in the 3D space generating the green opaque cone. Among all the possible directions only two of them are generating a circle on the cone when intersecting it. The two circle solutions are shown in purple and blue. The solid part of the circle is the one visible from the reader's perspective and not covered by the elliptic cone, while the dashed part is the one hidden by the cone to the reader. The blue and purple lines represent the perpendicular directions to the circles. The green dotted line represents the camera boresight which intersects the image plane in the green dot.

From these 2 planes, 4 unit normal vectors can be defined. It is important to note that both the positive and the negative normal vectors define the same plane. But, if the goal is to gather the rotation axis, the two normal vectors are associated with opposite rotation

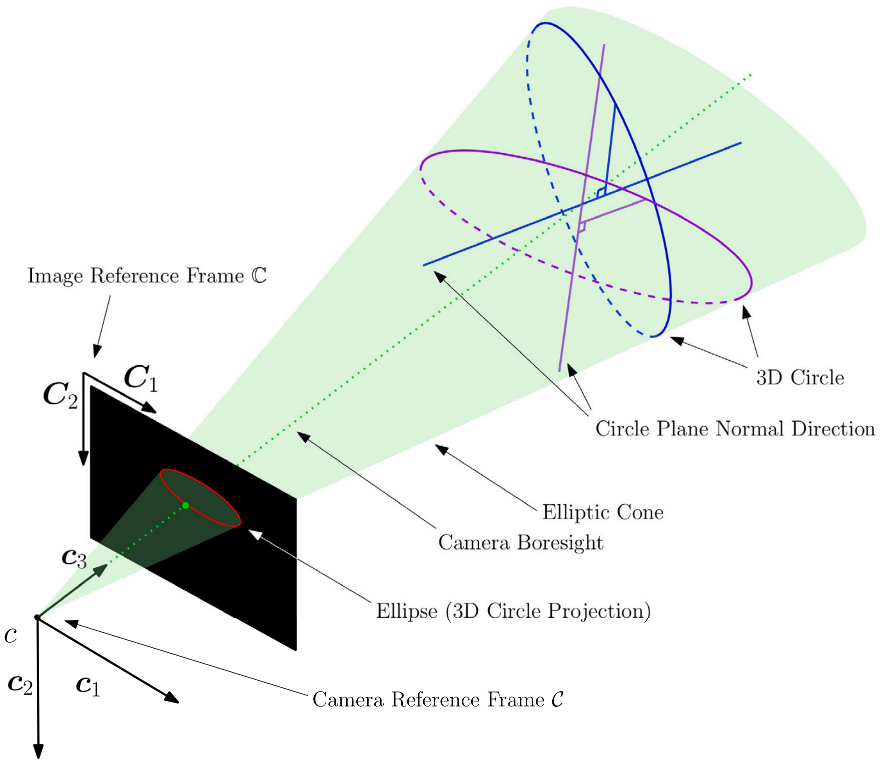


Image Reference Frame ℂ
C_1
C_2
c_3
C
c_2
c_1
3D Circle
Circle Plane Normal Direction
Elliptic Cone
Camera Boresight
Ellipse (3D Circle Projection)
Camera Reference Frame $\mathcal{C}$

Fig. 3. The geometrical construction generating the two circles.

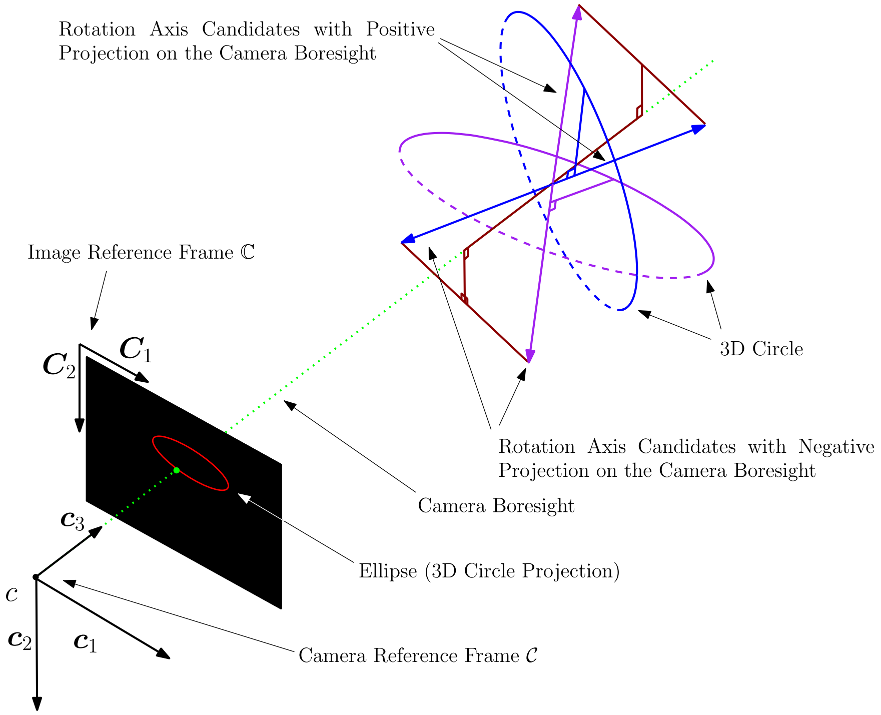

Rotation Axis Candidates with Positive Projection on the Camera Boresight
Image Reference Frame ℂ
C_1
C_2
3D Circle
Rotation Axis Candidates with Negative Projection on the Camera Boresight
Camera Boresight
c_3
Ellipse (3D Circle Projection)
C
c_2
c_1
Camera Reference Frame $\mathcal{C}$

Fig. 4. The four normal vector solutions from the circle computation.

directions on the same plane. Therefore, the correct normal vector must be identified.

The first step is to compute the four unit normal vectors. By knowing that the matrix $[E]$ (see Eq. (15)) represents an ellipse which is projection of a 3D circle, all the unit normal vectors generating the conics in the image can be extracted [53]. Note that the matrix $[E]$ is composed of the ellipse Cartesian parameters in pixel units. Thus the parameters found in the previous step must be rescaled according to the normalization procedure for consistency.

Let λ_i and $\boldsymbol{u}_i$ $\forall i \in [1, 3]$ be the eigenvalues and eigenvectors of $[E]$. If $\det[E] < 0$, the eigenvalues are ordered in such a way that $\lambda_3 < 0 < \lambda_1 \leq \lambda_2$. The four solutions for the circle plane normal are

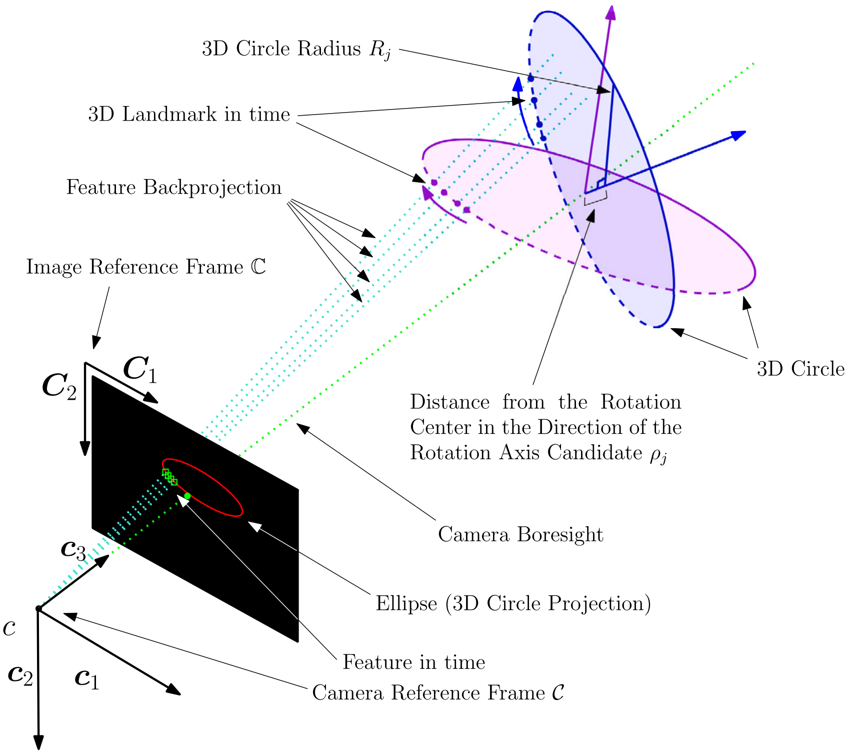

3D Circle Radius R_j
3D Landmark in time
Feature Backprojection
Image Reference Frame $\mathbb{C}$
C_2
C_1
c_3
C
c_2
c_1
3D Circle
Distance from the Rotation Center in the Direction of the Rotation Axis Candidate ρ_j
Camera Boresight
Ellipse (3D Circle Projection)
Feature in time
Camera Reference Frame $\mathcal{C}$

Fig. 5. The estimation of the circles from the two possible rotation axis candidates.

computed as [53]:

$$\hat{\boldsymbol{v}} = \pm\frac{\boldsymbol{v}}{\|\boldsymbol{v}\|} \quad \text{where} \quad \boldsymbol{v} = \sqrt{\frac{\lambda_2 - \lambda_1}{\lambda_2 - \lambda_3}}\boldsymbol{u}_2 \pm \sqrt{\frac{\lambda_1 - \lambda_3}{\lambda_2 - \lambda_3}}\boldsymbol{u}_3 \tag{27}$$

To estimate the four solutions, the two vectors $\boldsymbol{v}$ are computed for each ellipse and their mean is calculated to minimize numerical errors. It is worth noting that the proposed method is not robust with respect to outliers. RANSAC could be exploited to identify the correct orientation among a given orientation set as in Andreis et al. [54].

In Fig. 4 the geometry of the normal vectors reconstruction is shown. The four solutions are depicted in two different colors according to the generating circle plane. Moreover, their projections on the $\mathcal{C}_1$ reference frame are shown in brown. Note that they are paired in two different ways. First, each pair has the same projection on the camera boresight direction. Second, each pair has the same module in the camera plane, but opposite direction. Recall that the solutions with opposite directions but associated with the same plane denote the same circle, but swept in opposite directions.

6.3.3. Rotation axis pruning and selection

As the correct rotation axis must be found among the four solutions, a procedure is outlined hereafter to identify the correct normal vector among the 4 different possibilities. The main idea is to remove two spurious solutions among the four by exploiting how features move in time. The estimation of the correct rotation axis between the two remaining solutions is a more complex task as they are both acceptable with respect to the camera projection and both respect the features' motion. The idea behind the correct identification of the solution is the selection based on a heuristic, using statistical facts or a priori information. Two heuristics are investigated:

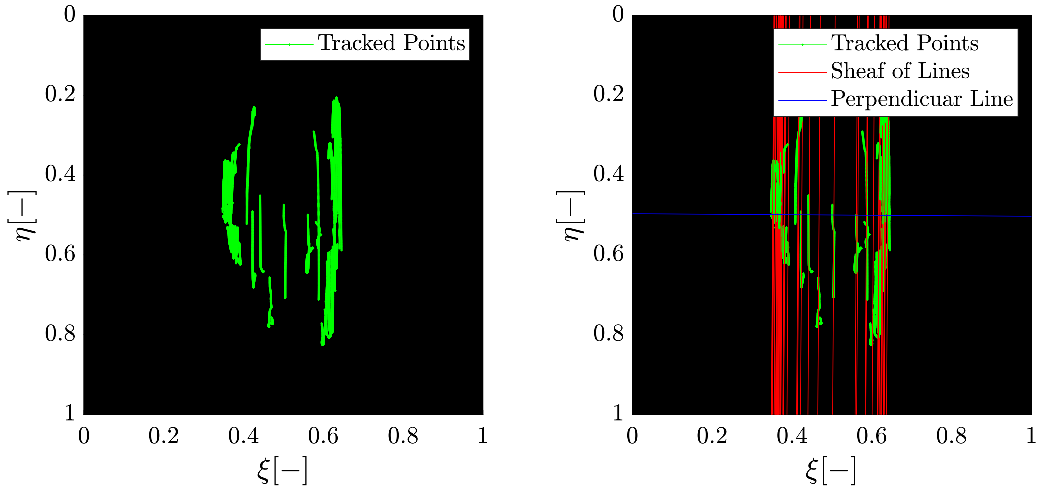
1. A coarse guess of the rotation axis direction is available on board from ground-based observation. The correct solution is identified as the closest one to the onboard guess.
2. It is remarked that, even though the observed small body can have concavities, the majority of the initialized 3D points during tracking lie between the camera and the estimated rotation center. To have statistical relevance, this reasoning must be applied to a high number of 3D points. As a consequence all 3D points that are tracked more than a given percentage of frame γ_{heur} over one rotational period are considered. In this work, γ_{heur} is set to be 0.2. Once it is understood which is the solution that initializes the majority of the point closer to the camera, that solution is the correct one [23]. Note that increasing γ_{heur} would imply to consider longer tracks, thus less point. On the contrary, decreasing γ_{heur} would lead to more statistical relevance at the cost of more computational burden. The selected value is selected as a trade off between these two factors, leading to satisfying results.

The first step is to exploit the feature rotation as seen in the image to remove two spurious solutions. Except tracking errors, matched points rotate all in the same direction. This rotation identifies uniquely the projection of the rotation axis on the camera boresight. Thus, by computing the feature rotation, it is possible to identify the two correct rotation axis candidates.

To avoid selecting the wrong solutions due to tracking errors, the rotation direction of each track is computed and the rotation axis candidates not consistent with the average feature rotation are discarded. The rotation direction $\overline{\boldsymbol{v}}_{i,j}$ between two successive frames is computed as:

$${}^{C}\overline{\boldsymbol{v}}_{i,j} = \left({}_{h}\Xi_{i,j} - {}_{h}\Xi_{c_j}\right) \times \left({}_{h}\Xi_{i+1,j} - {}_{h}\Xi_{c_j}\right) \tag{28}$$

where ${}_{h}\Xi_{i,j} = \left(\xi_{i,j}, \eta_{i,j}\right)^T$ is the jth tracked feature location in the ith image in homogeneous normalized coordinates and ${}_{h}\Xi_{c_j} = \left(\xi_{c_j}, \eta_{c_j}\right)^T$ is the jth ellipse center in homogeneous normalized coordinates. The vector $\overline{\boldsymbol{v}}_{i,j}$ provides information about the rotation direction of the jth feature between the ith image and the $(i+1)$th image. By counting the features moving clockwise and the ones moving counterclockwise, it is possible to understand which is the average feature motion. It is thus



(a) The tracked points

(b) The lines reconstruction

Fig. 6. The perpendicular-axis case reconstruction. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

useful to define the clockwise index J_{cw}:

$$J_{\text{cw}} = \sum_{j=1}^{M}\sum_{i=1}^{N_j-1} J_{\text{cw}_{i,j}} \quad \text{where} \quad J_{\text{cw}_{i,j}} = \begin{cases} 1 & \text{if } {}^{C}\boldsymbol{v}^{T}_{\text{calc}_{i,j}}\boldsymbol{c}_3 > 0 \\ -1 & \text{otherwise} \end{cases} \tag{29}$$

where $\boldsymbol{c}_3$ is the camera boresight unit vector. According to J_{cw} definition, features move clockwise if $J_{\text{cw}} > 0$ and counterclockwise otherwise. The two correct solutions are selected as the ones providing the same rotation direction when projected on the camera boresight.

At this stage, two solutions are still feasible and the correct one must be determined. To do so, two heuristic approaches are investigated. If a coarse first guess is available on board, the correct solution is selected as the one with the smallest angular error between the two. Otherwise, the second heuristic is exploited: the correct axis is the one initializing the majority of the points between the camera and the rotation center estimated in Section 4. It is worth noting that the circle plane has not been estimated yet as the direction of the circle plane normal defines a sheaf of parallel planes where the circle could lie. It is thus necessary to select a point to uniquely define where the circle plane is. By knowing the axis direction and a point on the axis, i.e., the previously-calculated rotation center $\boldsymbol{r}_{\text{RC}}$, the circle can be fully reconstructed [55]. The equation of the jth 3D circle is written as:

$$U(u,v)\rho_j^2 + V(u,v)\rho_j + W(u,v) - R_j^2 = 0 \tag{30}$$

where ρ_j is the distance between the rotation center $\boldsymbol{r}_{\text{RC}}$ and jth circle center in the direction of the rotation axis, R_j is the jth circle radius, and $U(u,v)$, $V(u,v)$ and $W(u,v)$ are computed as in Appendix. The parameter ρ_j uniquely defines the plane in the sheaf, whereas R_j uniquely identifies the circle size on that plane. Note that the pixel coordinates (u,v) are used for this optimization step.

This geometrical situation is depicted in Fig. 5 for two possible circle solutions colored according to the respective rotation axis. In the figure, empty green squares in the image are the tracked features and the turquoise dotted lines represent their backprojections. These backprojections intersect the two 3D circles in the full dots colored according to the intersected circle color code. The rotation direction is shown on each circle to provide information about the beginning and the end of the 3D landmark movement. Note that the rotation center is the intersection of the two possible rotation axes. Moreover, 3D circles parameters R_j and ρ_j are shown only for the blue circle for the sake of completeness. Furthermore, it is worth noting that all the colored full dots on the circle are associated with one 3D point at different times during tracking.

To determine the parameters R_j and ρ_j, the following cost function is minimized:

$$J_{\circ,j} = \sum_{i=1}^{N_j}\left(U(u_i,v_i)\rho_j^2 + V(u_i,v_i)\rho_j + W(u_i,v_i) - R_j^2\right)^2 \tag{31}$$

Once this procedure is performed for all tracks, an estimate of the parameters R_j and ρ_j $\forall j$ is determined. This enables the determination of the plane where the 3D circle lies and the 3D circle radius. It is now necessary to understand where the landmark has been detected at feature initialization. This is performed by computing the initialization angular position β_j of the 3D landmark on the jth circle at the beginning of the observational period.

Let $\mathcal{F}_{\hat{\boldsymbol{v}}_i}$ be the reference frame centered in the rotation center $\boldsymbol{r}_{\text{RC}}$ and with the vertical axis directed as $\hat{\boldsymbol{v}}_i$, i.e., one of the remaining solution for the rotation axis. The other two unit vectors are arbitrary. The position of the jth 3D landmark ${}^{\mathcal{F}_{\hat{\boldsymbol{v}}_i}}\boldsymbol{r}_{\text{LM}_j}$ in the $\mathcal{F}_{\hat{\boldsymbol{v}}_i}$ frame at time t is:

$${}^{\mathcal{F}_{\hat{\boldsymbol{v}}_i}}\boldsymbol{r}_{\text{LM}_j} = \begin{pmatrix} R_j\cos\left(\beta_j + \dfrac{2\pi}{T_{\text{rot}}}\Delta t\right) \\ R_j\sin\left(\beta_j + \dfrac{2\pi}{T_{\text{rot}}}\Delta t\right) \\ \rho_j \end{pmatrix} \tag{32}$$

where $\Delta t = t - t_{\text{init}}$ is the interval between t and the initialization time t_{init} and T_{rot} is the small body rotational period. By starting from $\beta_j = 0$, the jth 3D landmark is projected in the image for all the available tracking times and the reprojection error is minimized to find β_j. Note that this optimization step could be easily removed by simple backprojection of the initial feature location and the intersection of the feature ray with the 3D circle plane, as proposed by Bissonnette et al. [23]. This solution has not been investigated in the present paper, but it could be a valuable option to reduce computational costs.

These last two steps, i.e., the determination of the 3D circle and the angular position at the beginning of the tracking, are computed for all ellipses and the two remaining feasible orientation axis solutions. Then, by counting how many landmarks are initialized between the camera and the rotation center, it is possible to select the rotation axis solution as the one providing the higher number of point initialization closer to the camera.

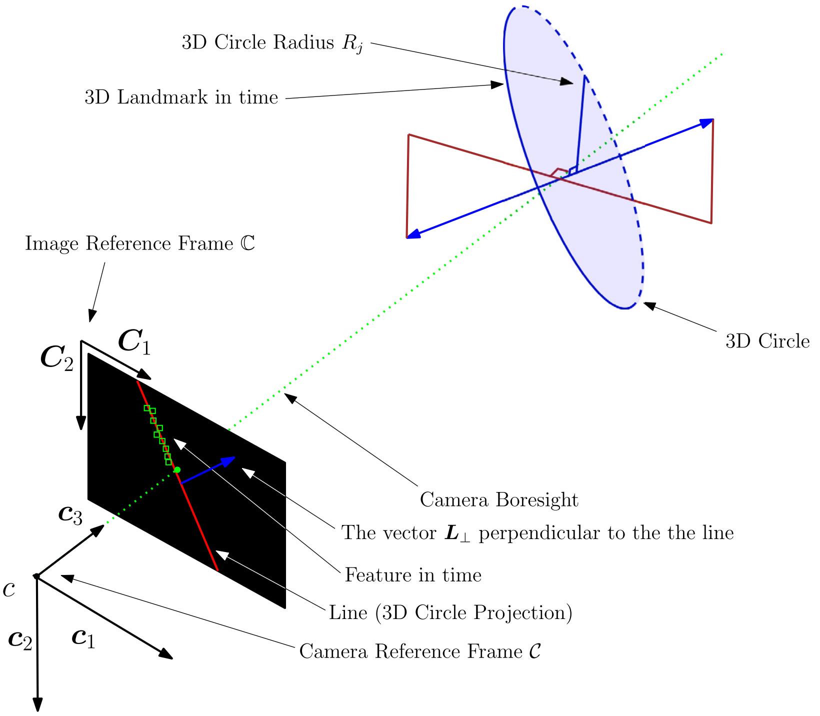

3D Circle Radius R_j
3D Landmark in time
Image Reference Frame $\mathbb{C}$
C_2
C_1
3D Circle
Camera Boresight
c_3
The vector $L_\perp$ perpendicular to the the line
Feature in time
C
Line (3D Circle Projection)
c_2
c_1
Camera Reference Frame $\mathcal{C}$

Fig. 7. The estimation of the rotation axis for the perpendicular-axis case.

6.4. Rotation axis perpendicular to camera boresight

6.4.1. Lines fitting

If the rotation axis is detected to be perpendicular to the camera boresight, the 3D circle projections degenerate into a sheaf of parallel lines as outlined in Section 6.1. It is worth noting that the tracked features do not form perfect lines because of tracking errors. Moreover, since the projection of the rotation axis on the camera boresight is null, only two rotation axis candidates are possible.

In this section, the same notation of Section 6.3 is used for sake of simplicity. Let M be the number of lines to be estimated. A sheaf of parallel lines is characterized by having the same orientation among all the lines. The equation of the jth line is given by Eq. (20). Note that the parameters D and E are constant for all the lines in the sheaf by construction.

Thus, $(\xi_{i,j}, \eta_{i,j})$, i.e., the jth feature location in the ith image, must verify the line equation in Eq. (20). By defining the parameter vector $\boldsymbol{a} = \left(D,\, E, F_1,\, \dots,\, F_M\right)^T$ and the matrix $\left[D_{\text{sheaf}}\right]$:

$$\left[D_{\text{sheaf}}\right] = \begin{bmatrix} \xi_{1,1} & \eta_{1,1} & 1 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \xi_{N_1,1} & \eta_{N_1,1} & 1 & 0 & 0 & 0 \\ \xi_{1,2} & \eta_{1,2} & 0 & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \xi_{N_2,2} & \eta_{N_2,2} & 0 & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \xi_{1,M} & \eta_{1,M} & 0 & 0 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \xi_{N_M,M} & \eta_{N_M,M} & 0 & 0 & 0 & 1 \end{bmatrix} \tag{33}$$

the sheaf fitting problem is rewritten as

$$\left[D_{\text{sheaf}}\right] \boldsymbol{a} = \left[\mathbb{0}_{M\times 1}\right] \tag{34}$$

It is solved by solving through Singular Value Decomposition [51]. An example of this procedure is reported in Figs. 6(a) and 6(b)

6.4.2. Computation of the rotation axis candidates

As lines are the degenerate projection of the planes where the 3D circles lie, the rotation axis projection on the image is given by the line perpendicular to the sheaf of parallel lines and passing through the rotation center. This geometrical configuration is shown in Fig. 6(b). Therefore, the 2D vector perpendicular to the sheaf must be backprojected in the 3D space as shown in Fig. 7. Let the vectors $\boldsymbol{L}_\perp$ and $\boldsymbol{L}_\parallel$ be respectively the normal and parallel vectors to the sheaf. These vectors can be computed from the line equation after proper rescaling of the equation in pixel coordinates. Therefore, the two solutions for the rotation axis are:

$$\hat{\boldsymbol{v}} = \pm \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|} \quad \text{where} \quad \boldsymbol{v} = \boldsymbol{r}_{\text{SC}_1} \times \left([K]^{-1}\,{}_h\boldsymbol{L}_\parallel\right) \tag{35}$$

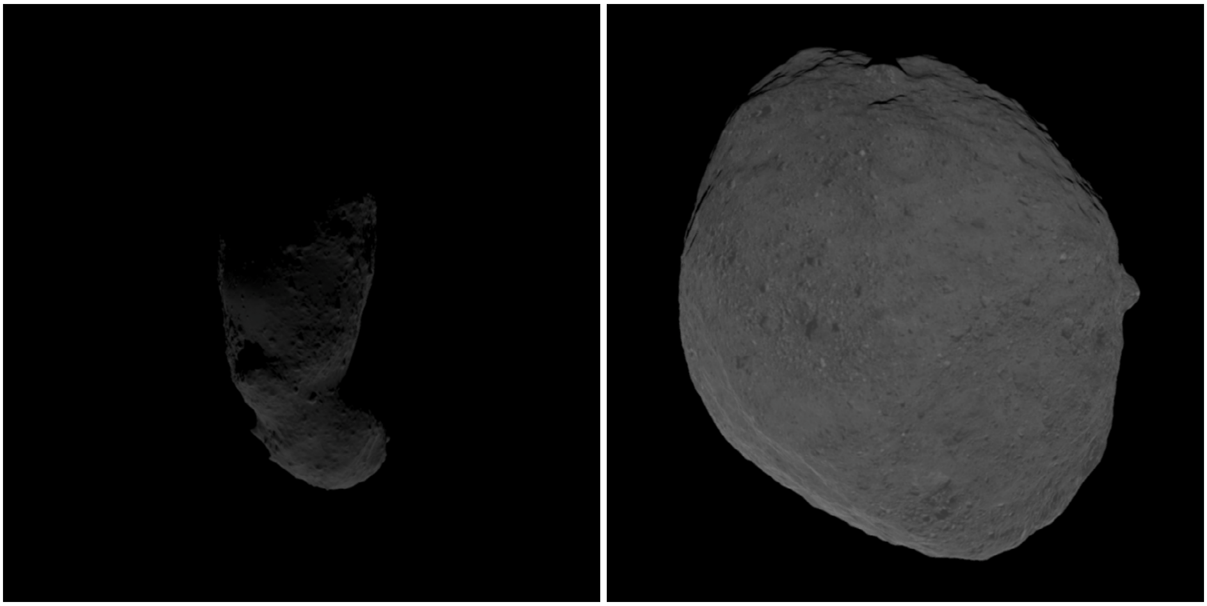
6.4.3. Rotation axis pruning and selection

To exclude the spurious solution, a heuristic similar to the one in the previous section is introduced. If a coarse guess exists, the solution is chosen to be the closest one to the initial guess. Otherwise, it is possible to determine which is the correct solution by determining how features move in the image on average.

Let $\mathcal{L} = \left\{l, \boldsymbol{l}_1, \boldsymbol{l}_2, \boldsymbol{l}_3\right\}$ be a reference frame centered in the image center and with unit vectors $\boldsymbol{l}_1 = \left(\boldsymbol{l}_\perp, 0\right)^T$, $\boldsymbol{l}_2 = \left(\boldsymbol{l}_\parallel, 0\right)^T$, and the perpendicular to define a right-handed reference frame. Let “in front” and “behind” be the subsets of the image defined by an observer looking in the direction of $\boldsymbol{l}_\parallel$ while standing on the image. If the rotation axis is projected as $\boldsymbol{l}_\perp$, the majority of the 3D points are moving from “behind” to “in front”; from “in front” to “behind” if projected as $-\boldsymbol{l}_\perp$. Indeed, as the majority of the 3D landmarks are initialized between the camera and the rotation center, the feature mean movement respects this motion. As in the non-degenerate conic case, shorter tracks are considered with a parameter $\gamma_{\text{heur}} = 0.2$ to increase the statistical relevance of the heuristic.

The movement vector $\boldsymbol{m}_{i,j}$ in the direction of $\boldsymbol{l}_2$ of jth feature in the ith image is computed as

$$\boldsymbol{m}_{i,j} = \left({}_h\boldsymbol{\Xi}_{i+1,j} - {}_h\boldsymbol{\Xi}_{i,j}\right)^T \boldsymbol{l}_2\, \boldsymbol{l}_2 \tag{36}$$



(a) Itokawa with illumination angle of 75°

(b) Bennu with illumination angle of 10°

Fig. 8. Examples of synthetic images for Bennu and Itokawa with different illumination angles.

Table 1
Numerical values for the camera parameters.

Camera	FoV[a] [°]	f[b] [mm]	$s_x = s_y$[c] [μm]	Image size [px]	$\boldsymbol{C}$[d] [px]
OSIRIS-REx's MapCam	3.99	125	8.5	1024 × 1024	$(512, 512)^T$
Hayabusa's AMICA	5.8	121	12	1024 × 1024	$(512, 512)^T$

[a] FoV is the camera field of view.

[b] f is the camera focal length.

[c] s_x and s_y are the pixel physical size in x and y components.

[d] $\boldsymbol{C}$ is the camera center.

Finally, the mean movement direction index J_{dir} is computed as:

$$J_{\text{dir}} = \sum_{j=1}^{M} \sum_{i=1}^{N_j-1} J_{\text{dir}_{i,j}} \quad \text{where} \quad J_{\text{dir}_{i,j}} = \begin{cases} 1 & \text{if} \quad \boldsymbol{l}_3^T \left(\boldsymbol{l}_1 \times \boldsymbol{m}_{i,j} \right) > 0 \\ -1 & \text{otherwise} \end{cases} \tag{37}$$

The value of J_{dir} gives information about whether the landmarks move from "in front" to "behind" or the other way around. If $J_{\text{dir}} > 0$, the rotation axis is projected in the image in the same direction of $\boldsymbol{l}_{\perp}$; the opposite otherwise. Thus, the sign of J_{dir} uniquely defines the rotation axis orientation in the inertial reference frame.

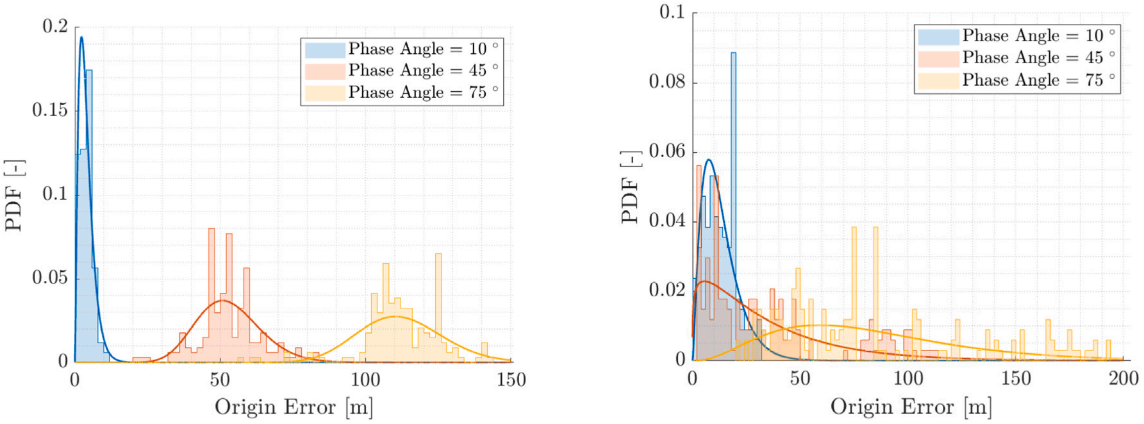
7. Numerical results

7.1. Test cases description

To test the proposed algorithm, numerical simulations have been performed. In the simulations, the spacecraft is in close approach to the small body. The considered scenarios foresee a spacecraft trajectory to simulate the approach to asteroids Bennu or Itokawa. The spacecraft starts observing the small body from about 9 km (i.e., about 60 asteroid radii for Itokawa and 36 asteroid radii for Bennu), implying that the asteroid gravity is negligible. The spacecraft velocity points towards the asteroid and its magnitude are selected to be consistent with approach velocity from SPICE kernels. As the probe is not under the influence of the small body gravity field, the trajectory has been simulated with a dynamics dominated by Solar Radiation Pressure (SRP) in the Sun-small-body rotating frame. Note that the spacecraft is orbiting the Sun but the swept arc of the conic is so short that it can be approximated by a straight line, so as to ignore the Sun gravity in the equations of motion. The small body rotates around an inertially-fixed axis with constant angular velocity. The spacecraft orientation in the inertial frame $\mathcal{N}$ is considered known and the camera points to the small body center of mass. More details about the dynamical model and the simulation set-up can be found in Section 3.6 of Panicucci [20].

The mapping camera, like Hayabusa's AMICA [56] or OSIRIS-REx's MapCam [57] (see Table 1), is constantly observing the small body. The two cameras are selected among the OSIRIS-Rex and Hayabusa's instruments because their characteristics ensure to observe respectively Bennu and Itokawa letting the asteroid occupy hundreds of pixels at the approach distance. Images are simulated with the SurRender software [58]. Small-body shapes, Bennu and Itokawa in this study, are taken from the Planetary Data System - Small Body Node by downloading the highest resolution polyhedral shape model. The shape model is input to SurRender together with the small-body bulk optical properties, like albedo, reflectivity, and diffusivity, and assuming the Hapke BRDF (Bidirectional Reflectance Distribution Function) [59] to obtain high-fidelity simulated images. A texture is added to the small body to take into account the small-body surface albedo. The camera optics is represented with a Gaussian Point Spread Function (PSF) and a pin-hole projection model. The KLT tracker Themis processes images every 60 s for Bennu and 180 s for Itokawa to account for realistic image processing operational constrains. Numerical values, including 3D meshes, are taken from operational missions scenario [56,57,60–62]. Note that in this study no real images are considered because of unavailability of finding frequent and successive images to provide to the KLT algorithm.

Two main test cases are presented hereafter: Bennu and Itokawa. To study the performance of the proposed algorithm the approach direction is kept constant and 3 different illumination angles are considered: 10°, 45°, and 75°. After having defined the Sun-small-body



(a) Bennu

(b) Itokawa

Fig. 9. Probability Density Function (PDF) of the origin estimation error. The distributions in the figures are the best-fit Gamma distribution. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

and the approach directions, the rotation axis must be defined to fully determine the observational geometry during the approach (see Fig. 1). To characterize the algorithm performance, the rotation axis is varied over the entire celestial sphere by changing the right ascension α and the declination δ with respect to the approach direction. Note that $\alpha = 0°$ and $\delta = 0°$ means that the rotation axis is pointing towards the spacecraft during the approach. Moreover, all cases with $\alpha = 90°$ or $\delta = 90°$ imply that the rotation axis is perpendicular to the approach direction. The discretization of the (α, δ) grid is 30° as each point of the grid implies the rendering of 250 images. By considering that only one case is simulated for $\delta = \pm 90$ as α is not defined and that $\delta = \pm 180$ are the same orientation for all α, 134 simulations are performed per illumination angle and asteroid leading to 804 simulations and more than 200,000 images. This rendering and simulation effort has been performed to push towards the validation and the performance assessment of the algorithm when illumination changes, shape varies, or the rotation axis orientation is unforeseeable. Examples of synthetic images for both asteroids with different illumination angles are shown in Figs. 8(a) and 8(b) for the sake of completeness.

7.2. Origin estimation

The first analyzed results deal with the small-body-fixed reference frame estimation. To assess the performance in estimating the origin, the origin error is defined as the norm between the estimated origin and the true one.

Note that the origin error is mainly influenced by the center of brightness computation, as the distance to the small body is considered know (see Section 3). The probability density functions (PDFs) of the origin error are reported in Figs. 9(a) and 9(b) for different values of the illumination angle. Recall that Bennu and Itokawa have a best-fitting-ellipse semiaxes of 252 m × 246 m × 228.3 m and 267.5 m × 147 m × 104.5 m, respectively. As known from the literature [35–37], the center of brightness deviates from its true value when the illumination angle increases. Thus its backprojection deviates from the true origin accordingly. For very high illumination angle the performance degrades and the origin is estimated to be far from the true value. This is even more important for an highly concave and oblate body as Itokawa. It is worth noting that this error can influence the rotation axis estimation as the origin determination is necessary for the determination of the 3D circle parameter. Despite this behavior, the rotation axis estimation shows good performance for the majority of the simulated scenarios, implying that the origin estimation play a minor role. For the sake of completeness, Table 2 shows the mean and the standard deviation of the best Gamma distribution fitting the origin estimation error data.

Table 2
Mean and standard deviation of the best-fit Gamma distribution of the origin estimation error.

Test-case	Bennu			Itokawa		
Illumination angle [°]	10	45	75	10	45	75
Mean [m]	6.38	121.05	214.22	13.12	34.07	82.98
Variance [m]	3.93	53.09	112.18	73.05	975.18	1998.88

The Gamma distribution is preferred with respect to other distribution as the PDF support is limited to the positive real numbers. The means and the standard deviations confirm the previous interpretation of the error. The performance degrades when increasing the illumination angle as shown by the errors means, but this effect is more relevant for the Itokawa test-case where the variability of the results due to selfshadowing implies higher standard deviations with respect to the Bennu test-case. As stated in Section 4, algorithms were designed in the past to compensate for this effect implying a possible reduction of the origin error and its influence on the rotation axis estimation.

7.3. Detection of degenerate solutions

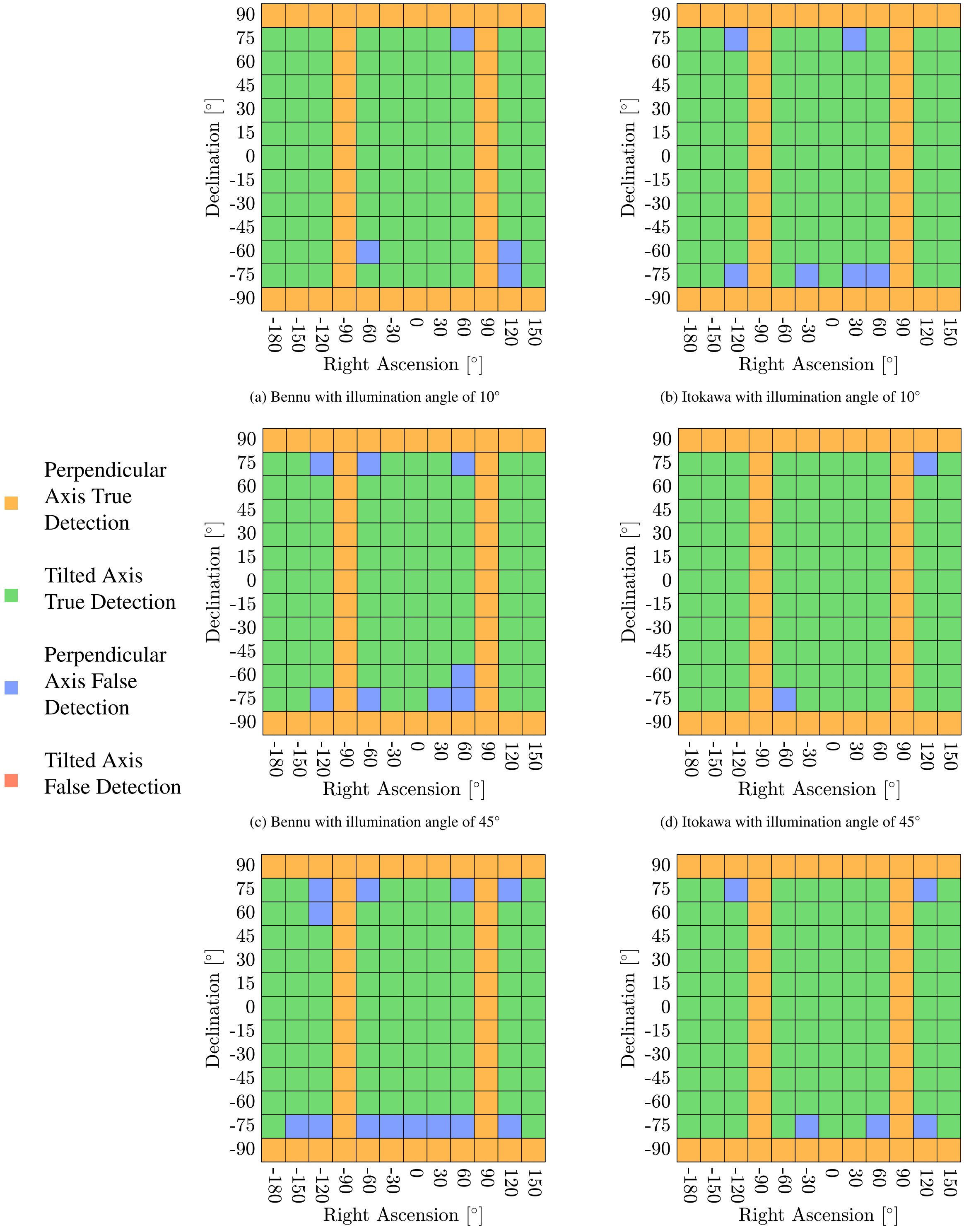
In this section, the results of the degenerate solutions detection are presented. To evaluate the performance of the detection, note that two classes are present: “Tilted Rotational Axis” and “Perpendicular Rotational Axis”. For each class it is possible to define precision Pr, recall Re, and the F-score F_1:

$$Pr = \frac{TP}{TP + FP} \tag{38}$$

$$Re = \frac{TP}{TP + FN} \tag{39}$$

$$F_1 = \frac{2Pr\,Re}{Re + Pr} \tag{40}$$

where TP is the true positive number, FP is the false positive number, and FN is the false negative number. These metrics are selected as precision Pr provides information about how many detections within a class are correct, recall Re states how many samples are correctly detected in the considered class, and the F-score F_1 is a mix of precision and recall which is high when both are high. For the two considered classes, the considered metrics are reported in Table 3. The precision of the “Tilted Rotation Axis” class is 1 which implies that all the solutions labeled within the class belong to this class. Its recall is always above 0.85 which implies that more than 85% of the tilted axis cases are identified correctly. Figs. 10(a)–10(f) show the detection results by



(a) Bennu with illumination angle of 10°

(b) Itokawa with illumination angle of 10°

(c) Bennu with illumination angle of 45°

(d) Itokawa with illumination angle of 45°

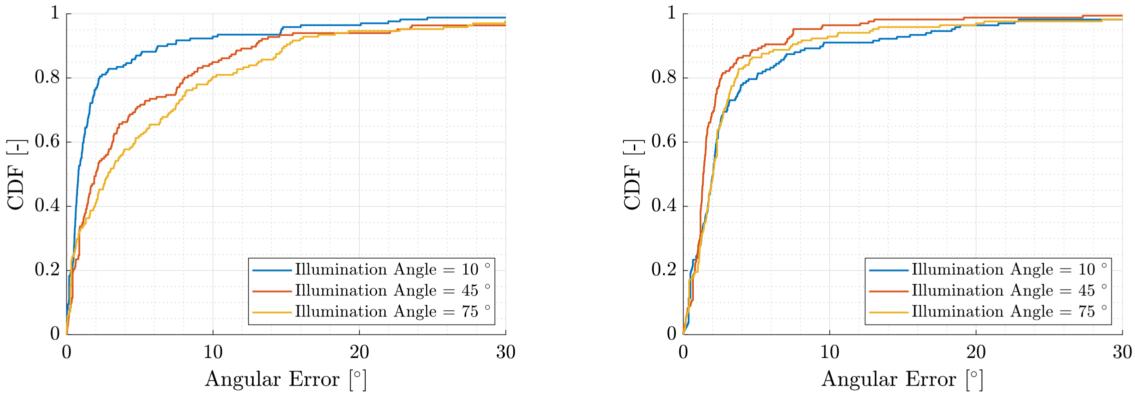
(e) Bennu with illumination angle of 75°

(f) Itokawa with illumination angle of 75°

Fig. 10. Degenerate solutions detection performance varying the rotation axis orientation with respect to the camera boresight. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

varying the right ascension and declination. It is worth noting that no false detection of the tilted rotation axis is present. Finally, the F-score for the "Tilted Rotation Axis" is high implying that a good balance between precision and recall is achieved.

Regarding the "Perpendicular Rotation Axis" class, its recall is 1 which means that all the perpendicular rotation axis are correctly identified. The precision is lower implying that some of the found solutions do not belong to the class. By looking at Figs. 10(a)–10(f), all the false



(a) Bennu

(b) Itokawa

Fig. 11. Zoom of the CDFs of the rotation axis estimation error with the heuristic based on onboard guess. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 3
Metrics for the degenerate solution detection.

Test-case	Bennu			Itokawa		
Illumination angle [°]	10	45	75	10	45	75
Tilted Rotation Axis precision	1	1	1	1	1	1
Tilted Rotation Axis recall	0.9636	0.9272	0.8818	0.9454	0.9818	0.9545
Tilted Rotation Axis F-score	0.9814	0.9622	0.9372	0.9719	0.9908	0.9767
Perpendicular Rotation Axis precision	0.92	0.8518	0.7796	0.8846	0.9583	0.9019
Perpendicular Rotation Axis recall	1	1	1	1	1	1
Perpendicular Rotation Axis F-score	0.9583	0.92	0.876	0.9387	0.9787	0.9484

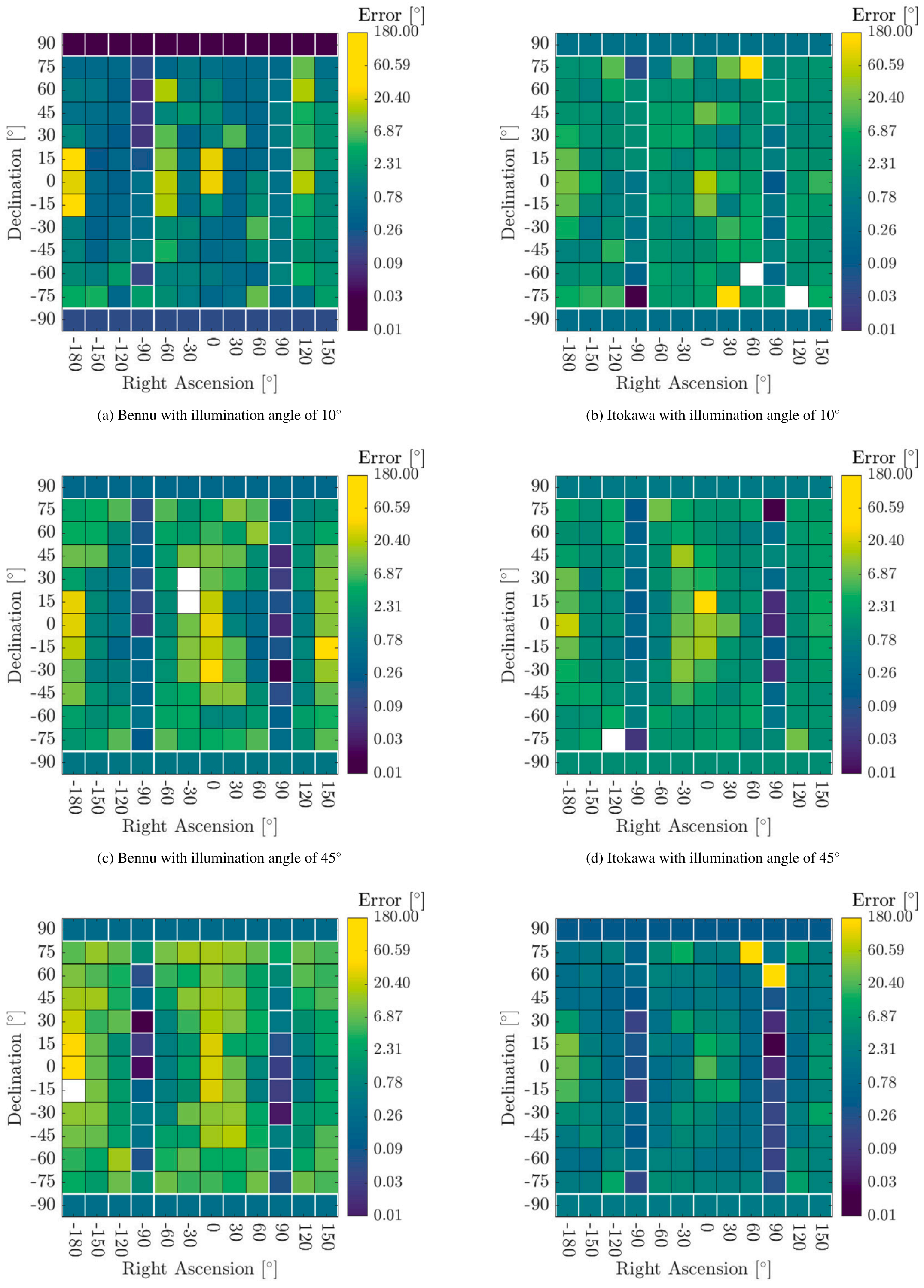
detections are close to the perpendicular case in the right-ascension-declination plane. This is because tracks are almost linear and the algorithm classifies them as lines. This problem could be avoided by increasing the conditioning number parameter γ_{CN}, but this could cause the presence of false detection of perpendicular rotation axis as tracks could be detected as ellipses due to tracking errors. Otherwise, γ_{sheaf} could be increased to have a more stringent selection .when labeling a curve as a line. Nevertheless, the proposed numerical values show good performance in terms of precision and recall, implying that the detection of the degenerate solution is performed correctly. The F-score is high also for this class which implies a good balance between precision and recall.

Note that results are similar for both asteroids meaning that performance is similar despite the different onboard cameras. It is worth noting that the Bennu test case has more false detection of the perpendicular axis which is probably due to the low concavity of the asteroid (see Figs. 10(a)–10(f) and Table 3). Indeed, when the body is convex, it is harder to detect tracks that deviate from a line for high approach angles. Moreover, the performance is not affected by the illumination angle variation for the Itokawa test case. This is not true for the Bennu test case where the performance metrics decrease with the illumination angle increase. This is mainly due to the difference in shadowing. The performance of the Itokawa test case is mainly influenced by self-shadowing, whereas the tracking performance is governed by the terminator line shadows for the Bennu test case. As self-shadowing is present even for low illumination angles, the tracks have generally the same length for all the simulations in the Itokawa test case. This is not happening in the Bennu test case where features are initiated close to the terminator line and are tracked for more time when the illumination angle is low.

7.4. Rotation axis estimation

The last result to be analyzed is the performance of the rotation axis estimation. As explained in Section 7.1, the rotation axis is varied by defining a spaced grid in the right-ascension-declination plane to cover all the geometrical configurations that the probe could encounter during the approach. To assess the algorithm performance, the angular error with respect to the true rotation axis is used as the performance index. Moreover, two different simulations are reported hereafter in accord with the two heuristics proposed in Section 6: the final selection based on onboard guess or the one based on the point initialization.

First, the results for the rotation axis estimation based on the available onboard guess are analyzed. This means that the final rotation axis is selected as the closest one to a coarse estimation of the rotation axis available on board. The coarse first guess is simulated in this study by applying a rotation of 20 degrees to the true axis in a random direction. A compact visualization of the results is reported in the Cumulative Density Function (CDFs) in Figs. 11(a) and 11(b) for various illumination angles and the two studied asteroids. Note that the CDFs are cropped at 30° to have a detailed look close to the low error solutions. In both cases, the angular error is below 10° for 80% of the cases and it is below 20° for 90% of the cases. This implies that the proposed method estimates correctly the 4 solutions and identifies the correct one. Note that the illumination does not seem to play a major role for the Itokawa test case (see Fig. 11(b)), while the performance degrades degrade for the Bennu test case when the illumination angle increase (see Fig. 11(a)). As stated in Section 7.3, this is due to the different shadows present in the two test cases. A more detailed view of the results is reported in Figs. 12(a)–12(f) where the angular error is shown in the right-ascension-declination plane. In the figures, the sectors with the white contours are associated with rotation axes perpendicular to the camera boresight. Moreover, the white sectors label the solutions where the algorithm has not converged. This is mainly



(a) Bennu with illumination angle of 10°

(b) Itokawa with illumination angle of 10°

(c) Bennu with illumination angle of 45°

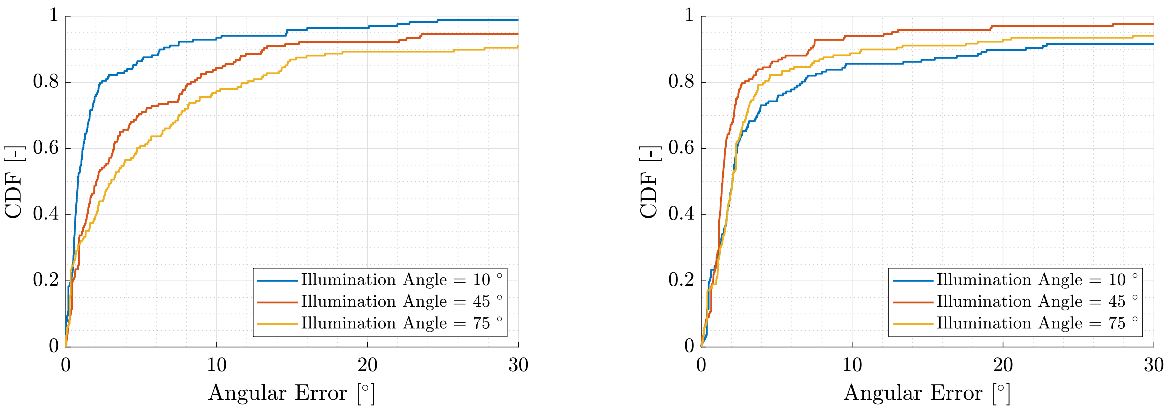
(d) Itokawa with illumination angle of 45°

(e) Bennu with illumination angle of 75°

(f) Itokawa with illumination angle of 75°

Fig. 12. Rotation axis estimation performance with the heuristic based on onboard guess. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

due to the impossibility of detecting a series of ellipses all with the same orientation leading to the detection of a conic family with an eccentricity greater than 1. Finally, the yellow sectors are associated with very high errors which are due to wrong detection among the four solutions. It is worth noting that the perpendicular cases have generally low errors and higher errors are present when the rotation axis is parallel to the approach direction. Note that the wrong classifications



(a) Bennu

(b) Itokawa

Fig. 13. Zoom of the CDFs of the rotation axis estimation error with the heuristic based on landmark initialization. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

outlined in 7.3 do not show high errors because they occur when the rotation axis is almost perpendicular.

Second, the results for the rotation axis estimation based on the landmark initialization are studied. Results are similar with respect to the previous ones with small performance degradation, mainly because the correct axis is selected less often. The performance reduction is anyway low as shown in Figs. 13(a) and 13(b). Indeed, the angular error is below 10° for 75% of the samples for all asteroids and illumination angles and it is below 20° for 85% of the cases. Note that the performance reduction is not due to a change in the angular error, which is the same as the tracked features are the same, but to the higher number of samples in the distribution tails. This is visible in Figs. 14(a)–14(f) where the angular error is shown in the right-ascension-declination plane. The number of yellow sectors increases because the heuristic fails in detecting the correct solution more often than the onboard guess heuristic. However, it is worth noting that this heuristic does not rely on a previously-computed guess and the rotation axis estimation is performed fully autonomously from images.

8. Conclusion

This paper outlines an algorithm to autonomously determine the rotational state of a small body during the approach phase. Under the assumption of knowing the rotational period, the algorithm determines the rotation axis orientation of a small body exploiting images during the approach phase. The spacecraft camera acquires images of an unknown small body and extracts features from these images. The long-tracked features are the projection of 3D landmarks whose movement is due to the small body rotation. By exploiting the 2D feature tracks, the algorithm fits a family of conics which are the projection of the landmark 3D circle. By backprojecting the found conics in 3D, the algorithm finds all the possible solutions for the small body rotation axis. The correct one is selected by ensuring coherence with the movement of the features in the image and by exploiting a heuristic approach leading to identifying the right rotation axis. Moreover, the center of rotation is determined from the first image by backprojecting the small body center of brightness.

Numerical simulations are performed with a concave and a convex asteroid over a wide range of illumination angles and pole orientations. The simulations underline that the algorithm can efficiently detect the case in which the rotation axis is perpendicular to the spacecraft approach direction. Moreover, the rotation axis estimation is performed with limited error in most cases. Indeed the error between the true rotation axis and its estimation is below 10° for 80% of the considered test cases. The computed rotation axis is a valuable first guess of the rotation axis to enable spacecraft localization around an unknown small body. The proposed rotation axis estimation and the origin determination are valuable tools to define the small-body-fixed reference frame during the approach phase. This is an important and preliminary task to be fulfilled before performing insertion maneuvers, shape determination, or small-body-fixed localization.

Declaration of competing interest

The authors declare to have no conflict of interest.

Acknowledgments

This research received funding from the CNES, France, the ISAE-SUPAERO and Airbus Defence & Space, France under the doctoral contract CNES-2879. P. P. would like to thank Felice Piccolo for the discussions and advises.

Appendix. The computation of $U(u,v)$, $V(u,v)$ and $W(u,v)$

The functions $U(u,v)$, $V(u,v)$ and $W(u,v)$ introduced in Section 6.3.3 are computed as follows:

$$U(u,v) = U_1^2(u,v) + W_1^2(u,v) \tag{A.1}$$

$$V(u,v) = 2U_1(u,v)\,V_1(u,v) + 2W_1(u,v)\,Y_1(u,v) \tag{A.2}$$

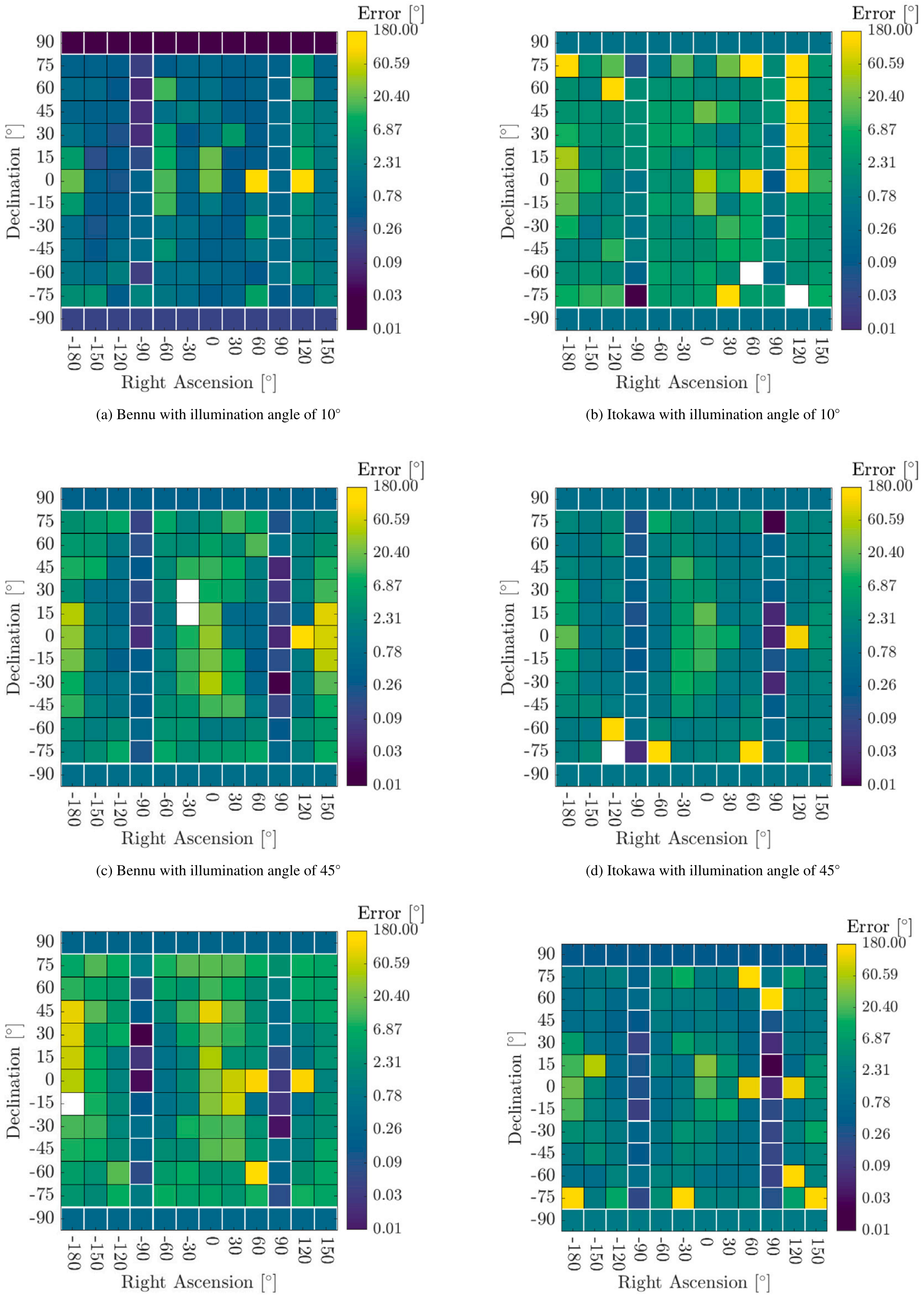
$$W(u,v) = V_1^2(u,v) + Y_1^2(u,v) \tag{A.3}$$

$$U_1(u,v) = U_{1,1}(u,v)\,W_1(u,v) + V_{1,1}(u,v) \tag{A.4}$$

$$V_1(u,v) = U_{1,1}(u,v)\,Y_1(u,v) + W_{1,1}(u,v) \tag{A.5}$$

where (u,v) are pixel coordinates. By defining the projection matrix $[P]$ between the 3D world homogeneous coordinates to the 2D homogeneous image coordinates, the needed quantities are computed:

$$W_1(u,v) = \frac{\left([P]_{2,1} - [P]_{3,1}\,v\right)V_{1,1} + [P]_{2,3} - [P]_{3,3}\,v}{\left([P]_{3,1}\,v - [P]_{2,1}\right)U_{1,1} + [P]_{3,2}\,v - [P]_{2,2}} \tag{A.6}$$



(a) Bennu with illumination angle of 10°

(b) Itokawa with illumination angle of 10°

(c) Bennu with illumination angle of 45°

(d) Itokawa with illumination angle of 45°

(e) Bennu with illumination angle of 75°

(f) Itokawa with illumination angle of 75°

Fig. 14. Rotation axis estimation performance with the heuristic based on landmark initialization. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

$$Y_1(u,v) = \frac{\left([P]_{2,1} - [P]_{3,1}\, v\right) W_{1,1} + [P]_{2,4} - [P]_{3,4}\, v}{\left([P]_{3,1}\, v - [P]_{2,1}\right) U_{1,1} + [P]_{3,2}\, v - [P]_{2,2}} \tag{A.7}$$

$$U_{1,1}(u,v) = \frac{[P]_{1,2} - [P]_{3,2}\, u}{[P]_{3,1}\, u - [P]_{1,1}} \tag{A.8}$$

$$V_{1,1}(u,v) = \frac{[P]_{1,3} - [P]_{3,3}\, u}{[P]_{3,1}\, u - [P]_{1,1}} \tag{A.9}$$

$$W_{1,1}(u,v) = \frac{[P]_{1,4} - [P]_{3,4}\, u}{[P]_{3,1}\, u - [P]_{1,1}} \tag{A.10}$$

where $[P]_{k,l}$ is the kth row and lth column element of the matrix.

References

[1] J. Melosh, A. Harris, B.L. Ida, L. McFadden, M. Mommert, G. Rieke, A. Rivkin, D.J. Scheeres, E.F. Tedesco, Finding Hazardous Asteroids Using Infrared and Visible Wavelength Telescopes, The National Academies Press, 2019, http://dx.doi.org/10.17226/25476.

[2] M. Kaasalainen, J. Torppa, Optimization methods for asteroid lightcurve inversion: I. shape determination, Icarus 153 (1) (2001) 24–36, http://dx.doi.org/10.1006/icar.2001.6673.

[3] M. Kaasalainen, J. Torppa, K. Muinonen, Optimization methods for asteroid lightcurve inversion: II. The complete inverse problem, Icarus 153 (1) (2001) 37–51, http://dx.doi.org/10.1006/icar.2001.6674.

[4] S.J. Ostro, R.S. Hudson, L.A.M. Benner, J.D. Giorgini, C. Magri, J.-L. Margot, M.C. Nolan, Asteroid radar astronomy, in: Asteroids III, 2002, pp. 151–168.

[5] F. Castellini, D. Antal-Wokes, R. Pardo de Santayana, K. Vantournhout, Far approach optical navigation and comet photometry for the rosetta mission, in: Proceedings of the 25th International Symposium on Space Flight Dynamics (ISSFD'15), 2015.

[6] S. Bandyonadhyay, I. Nesnas, S. Bhaskaran, B. Hockman, B. Morrell, Silhouette-Based 3D Shape Reconstruction of a Small Body from a Spacecraft, in: 2019 IEEE Aerospace Conference, IEEE, 2019, pp. 1–13, http://dx.doi.org/10.1109/AERO.2019.8741753.

[7] D.J. Scheeres, Orbital Motion in Strongly Perturbed Environments: Applications to Asteroid, Comet and Planetary Satellite Orbiters, Springer, 2016, http://dx.doi.org/10.1007/978-3-642-03256-1.

[8] R.W. Gaskell, O.S. Barnouin-Jha, D.J. Scheeres, T. Mukai, N. Hirata, S. Abe, J. Saito, M. Ishiguro, T. Kubota, T. Hashimoto, J. Kawaguchi, M. Yoshikawa, K. Shirakawa, T. Kominato, Landmark navigation studies and target characterization in the hayabusa encounter with itokawa, in: AIAA/AAS Astrodynamics Specialist Conference and Exhibit, 2006, p. 6660, http://dx.doi.org/10.2514/6.2006-6660.

[9] R.W. Gaskell, O.S. Barnouin-Jha, D.J. Scheeres, A.S. Konopliv, T. Mukai, S. Abe, J. Saito, M. Ishiguro, T. Kubota, T. Hashimoto, J. Kawaguchi, M. Yoshikawa, K. Shirakawa, T. Kominato, N. Hirata, H. Demura, Characterizing and navigating small bodies with imaging data, Meteorit. Planet. Sci. 43 (6) (2008) 1049–1061, http://dx.doi.org/10.1111/j.1945-5100.2008.tb00692.x.

[10] C. Capanna, G. Gesquière, L. Jorda, P. Lamy, D. Vibert, Three-dimensional reconstruction using multiresolution photoclinometry by deformation, Vis. Comput. 29 (6–8) (2013) http://dx.doi.org/10.1007/s00371-013-0821-5.

[11] R.A. Werner, D.J. Scheeres, Exterior gravitation of a polyhedron derived and compared with harmonic and mascon gravitation representations of asteroid 4769 castalia, Celestial Mech. Dynam. Astronom. 65 (3) (1996) 313–344, http://dx.doi.org/10.1007/BF00053511.

[12] P. Panicucci, B. Bercovici, E. Zenou, J. McMahon, M. Delpech, J. Lebreton, K. Kanani, Uncertainties in the gravity spherical harmonics coefficients arising from a stochastic polyhedral shape, Celestial Mech. Dynam. Astronom. 132 (4) (2020) 1–27, http://dx.doi.org/10.1007/s10569-020-09962-8.

[13] B. Bercovici, P. Panicucci, J. McMahon, Analytical shape uncertainties in the polyhedron gravity model, Celestial Mech. Dynam. Astronom. 132 (5) (2020) 1–32, http://dx.doi.org/10.1007/s10569-020-09967-3.

[14] R.A. Werner, Spherical harmonic coefficients for the potential of a constant-density polyhedron, Comput. Geosci. 23 (10) (1997) 1071–1077, http://dx.doi.org/10.1016/S0098-3004(97)00110-6.

[15] J.W. McMahon, D.J. Scheeres, S.G. Hesar, D. Farnocchia, S. Chesley, D. Lauretta, The OSIRIS-REx radio science experiment at bennu, Space Sci. Rev. 214 (2018) 1–41, http://dx.doi.org/10.1007/s11214-018-0480-y.

[16] L. Jorda, R. Gaskell, C. Capanna, S. Hviid, P. Lamy, J. Ďurech, G. Faury, O. Groussin, P. Gutiérrez, C. Jackman, S.J. Keihn, H.U. Keller, J. Knollenberg, E. Kührt, S. Marchi, S. Mottola, E. Palmer, F.P. Schloerb, H. Sierks, J.-B. Vincent, M.F. A'Hearn, C. Barbier, R. Rodrigo, D. Koschny, H. Rickman, M.A. Barucci, J.L. Bertaux, I. Bertini, G. Cremonese, V. Da Deppo, B. Davidsson, S. Debei, M. De Cecco, S. Fornasier, M. Fulle, C. Gütter, W.-H. Ip, J.R. Kramm, M. Küppers, L.M. Lara, M. Lazzarin, J.J. Lopez Moreno, F. Marzari, G. Naletto, N. Oklay, N. Thomas, C. Tubiana, K.-P. Wenzel, The global shape, density and rotation of comet 67P/Churyumov-Gerasimenko from preperihelion rosetta/OSIRIS observations, Icarus 277 (2016) 257–278, http://dx.doi.org/10.1016/j.icarus.2016.05.002.

[17] D.J. Scheeres, J.W. McMahon, Autonomous architectures for small body exploration, in: Proceedings of the 29th AAS/AIAA Astrodynamics Specialist Conference, 2019.

[18] N. Stacey, S. D'Amico, Robust autonomous spacecraft navigation and environment characterization, 2022, Authorea Preprints.

[19] P. Panicucci, J. Lebreton, R. Brochard, E. Zenou, M. Delpech, Shadow-robust silhouette reconstruction for small-body applications, J. Spacecr. Rockets (2023) 1–17, http://dx.doi.org/10.2514/1.A35444.

[20] P. Panicucci, Autonomous Vision-Based Navigation and Shape Reconstruction of an Unknown Asteroid During Approach Phase (Ph.D. thesis), ISAE-SUPAERO, 2021, URL http://www.theses.fr/2021ESAE0011.

[21] M. Dor, T. Driver, K. Getzandanner, P. Tsiotras, AstroSLAM: Autonomous monocular navigation in the vicinity of a celestial small body–theory and experiments, 2022, arXiv preprint arXiv:2212.00350.

[22] J. Villa, J.W. McMahon, I.A. Nesnas, Point cloud visual SLAM for autonomous navigation and mapping around small celestial bodies, in: Space Imaging Workshop, 2022, 2022.

[23] V. Bissonnette, M. Dor, E. Zenou, S. Lizy-Destrez, Vision-Based Autonomous Characterization and Navigation for Asteroids, Technical report, ISAE-SUPAERO, 2015.

[24] Y. Masutani, T. Iwatsu, F. Miyazaki, Motion estimation of unknown rigid body under no external forces and moments, in: Proceedings of the 1994 IEEE International Conference on Robotics and Automation, IEEE, 1994, pp. 1066–1072, http://dx.doi.org/10.1109/ROBOT.1994.351227.

[25] S. Augenstein, S.M. Rock, Improved frame-to-frame pose tracking during vision-only SLAM/SFM with a tumbling target, in: 2011 IEEE International Conference on Robotics and Automation, IEEE, 2011, pp. 3131–3138, http://dx.doi.org/10.1109/ICRA.2011.5980232.

[26] J. Padial, M. Hammond, S. Augenstein, S.M. Rock, Tumbling target reconstruction and pose estimation through fusion of monocular vision and sparse-pattern range data, in: 2012 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), IEEE, 2012, pp. 419–425, http://dx.doi.org/10.1109/MFI.2012.6343026.

[27] M.D. Lichter, S. Dubowsky, State, shape, and parameter estimation of space objects from range images, in: IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004, Vol. 3, IEEE, 2004, pp. 2974–2979, http://dx.doi.org/10.1109/ROBOT.2004.1307513.

[28] U. Hillenbrand, R. Lampariello, Motion and parameter estimation of a free-floating space object from range data for motion prediction, in: Proceedings of I-SAIRAS, 2005.

[29] B.E. Tweddle, A. Saenz-Otero, J.J. Leonard, D.W. Miller, Factor graph modeling of rigid-body dynamics for localization, mapping, and parameter estimation of a spinning object in space, J. Field Robotics 32 (6) (2015) 897–933, http://dx.doi.org/10.1002/rob.21548.

[30] T.P. Setterfield, D.W. Miller, A. Saenz-Otero, E. Frazzoli, J.J. Leonard, Inertial properties estimation of a passive on-orbit object using polhode analysis, J. Guid. Control Dyn. 41 (10) (2018) 2214–2231, http://dx.doi.org/10.2514/1.G003394.

[31] K. Muinonen, J. Torppa, X.-B. Wang, A. Cellino, A. Penttilä, Asteroid lightcurve inversion with Bayesian inference, Astron. Astrophys. 642 (2020) A138, http://dx.doi.org/10.1051/0004-6361/202038036.

[32] J. Hanuš, J. Ďurech, M. Brož, B.D. Warner, F. Pilcher, R. Stephens, J. Oey, L. Bernasconi, S. Casulli, R. Behrend, D. Polishook, T. Henych, M. Lehký, F. Yoshida, T. Ito, A study of asteroid pole-latitude distribution based on an extended set of shape models derived by the lightcurve inversion method, Astron. Astrophys. 530 (2011) A134, http://dx.doi.org/10.1051/0004-6361/201116738.

[33] D.J. Scheeres, B.G. Williams, W.E. Bollman, R.P. Davis, C.E. Helfrich, S.P. Synnott, D.K. Yeomans, Navigation for low-cost missions to small solar-system bodies, Acta Astronaut. 35 (1995) 211–220, http://dx.doi.org/10.1016/0094-5765(94)00186-P.

[34] S. Takahashi, D.J. Scheeres, Autonomous exploration of a small near-earth asteroid. J. Guidance Control Dyn. 44 (4) 701–718.

[35] C.A. Wright, A.J. Liounis, B.W. Ashman, Optical navigation algorithm performance, 2018.

[36] S. Bhaskaran, J.E. Riedel, S.P. Synnott, Autonomous nucleus tracking for comet/asteroid encounters: the stardust example, in: 1998 IEEE Aerospace Conference Proceedings (Cat. No. 98TH8339), 2, IEEE, 1998, pp. 353–365, http://dx.doi.org/10.1109/AERO.1998.687921.

[37] M. Pugliatti, V. Franzese, F. Topputo, Data-driven image processing for onboard optical navigation around a binary asteroid, J. Spacecr. Rockets 59 (3) (2022) 943–959, http://dx.doi.org/10.2514/1.A35213.

[38] N. Otsu, A threshold selection method from gray-level histograms, IEEE Trans. Syst. Man Cybern. 9 (1) (1979) 62–66, http://dx.doi.org/10.1109/TSMC.1979.4310076.

[39] R. Szeliski, Computer Vision: Algorithms and Applications, Springer Nature, 2022.

[40] H. Bay, T. Tuytelaars, L. Van Gool, SURF: Speeded Up Robust Features, in: European Conference on Computer Vision, Springer, 2006, pp. 404–417, http://dx.doi.org/10.1007/11744023_32.

[41] S. Leutenegger, M. Chli, R.Y. Siegwart, BRISK: Binary robust invariant scalable keypoints, in: 2011 International Conference on Computer Vision, IEEE, 2011, pp. 2548–2555, http://dx.doi.org/10.1109/ICCV.2011.6126542.

[42] E. Rublee, V. Rabaud, K. Konolige, G. Bradski, ORB: An efficient alternative to SIFT or SURF, in: 2011 International Conference on Computer Vision, IEEE, 2011, pp. 2564–2571, http://dx.doi.org/10.1109/ICCV.2011.6126544.

[43] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110, http://dx.doi.org/10.1023/B:VISI.0000029664.99615.94.

[44] P.F. Alcantarilla, A. Bartoli, A.J. Davison, KAZE features, in: Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VI 12, Springer, 2012, pp. 214–227, http://dx.doi.org/10.1007/978-3-642-33783-3_16.

[45] B.D. Lucas, T. Kanade, An iterative image registration technique with an application to stereo vision, 1981.

[46] C. Tomasi, T. Kanade, Shape and motion from image streams under orthography: a factorization method, Int. J. Comput. Vis. 9 (2) (1992) 137–154, http://dx.doi.org/10.1073/pnas.90.21.9795.

[47] J. Shi, C. Tomasi, Good features to track, in: 1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 1994, pp. 593–600, http://dx.doi.org/10.1109/CVPR.1994.323794.

[48] J.-Y. Bouguet, Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm, Intel Corp. 5 (1–10) (2001) 4.

[49] P. Duteïs, R. Brochard, S. Tiberio, D. Djafari-Rouhani, M. Sanchez-Gestido, GENEVIS: Generic vision-based navigation technology building blocks for space applications, in: Proceedings of the GR740 User Day, 2019.

[50] R.I. Hartley, A. Zisserman, Multiple View Geometry in Computer Vision, Cambridge University Press, 2004, http://dx.doi.org/10.1017/CBO9780511811685.

[51] A. Fitzgibbon, M. Pilu, R.B. Fisher, Direct least square fitting of ellipses, IEEE Trans. Pattern Anal. Mach. Intell. 21 (5) (1999) 476–480, http://dx.doi.org/10.1109/34.765658.

[52] R. Halır, J. Flusser, Numerically stable direct least squares fitting of ellipses, in: Proc. 6th International Conference in Central Europe on Computer Graphics and Visualization. WSCG, Vol. 98, 1998, pp. 125–132.

[53] K. Kanatani, N. Ohta, Automatic detection of circular objects by ellipse growing, Int. J. Image Graph. 4 (01) (2004) 35–50.

[54] E. Andreis, P. Panicucci, V. Franzese, F. Topputo, A robust image processing pipeline for planets line-of-sign extraction for deep-space autonomous cubesats navigation, in: 44th AAS Guidance, Navigation and Control Conference, 2022, pp. 1–19.

[55] V. Fremont, R. Chellali, Turntable-based 3D object reconstruction, in: IEEE Conference on Cybernetics and Intelligent Systems, 2004, Vol. 2, IEEE, 2004, pp. 1277–1282, http://dx.doi.org/10.1109/ICCIS.2004.1460775.

[56] M. Ishiguro, R. Nakamura, D.J. Tholen, N. Hirata, H. Demura, E. Nemoto, A.M. Nakamura, Y. Higuchi, A. Sogame, A. Yamamoto, K. Kitazato, Y. Yokota, T. Kubota, T. Hashimoto, J. Saito, The Hayabusa spacecraft asteroid multi-band imaging camera (AMICA), Icarus 207 (2) (2010) 714–731, http://dx.doi.org/10.1016/j.icarus.2009.12.035.

[57] B. Rizk, C.Y. Drouet d'Aubigny, D. Golish, C. Fellows, C. Merrill, P. Smith, M.S. Walker, J.E. Hendershot, J. Hancock, S.H. Bailey, D.N. DellaGiustina, D.S. Lauretta, R. Tanner, M. Williams, K. Harshman, M. Fitwgibbon, W. Vertw, J. Chen, T. Connors, D. Hamara, A. Dowd, A. Lowman, M. Dubin, R. Burt, M. Whiteley, M. Watson, T. McMahon, M. Ward, D. Booher, M. Read, B. Williams, M. Hunten, E. Little, T. Saltzman, D. Alfred, S. O'Dougherty, M. Walthall, K. Kenagy, S. Peterson, B. Crowther, M.L. Morgan, C. Castle, R. Dominguez, M. Sullivan, OCAMS: the OSIRIS-REx camera suite, Space Sci. Rev. 214 (1) (2018) 26, http://dx.doi.org/10.1007/s11214-017-0460-7.

[58] J. Lebreton, R. Brochard, M. Baudry, G. Jonniaux, A.H. Salah, K. Kanani, M. Le Le Goff, A. Masson, N. Ollagnier, P. Panicucci, A. Proag, C. Robin, Image simulation for space applications with the SurRender software, 2021, arXiv preprint arXiv:2106.11322.

[59] B. Hapke, Theory of Reflectance and Emittance Spectroscopy, Cambridge University Press, 2012, http://dx.doi.org/10.1017/CBO9781139025683.

[60] H. Demura, S. Kobayashi, E. Nemoto, N. Matsumoto, M. Furuya, A. Yukishita, N. Muranaka, H. Morita, K. Shirakawa, M. Maruya, H. Ohyama, M. Uo, T. Kubota, T. Hashimoto, J. Kawaguchi, A. Fujiwara, J. Saito, S. Sasaki, H. Miyamato, N. Hirata, Pole and global shape of 25143 Itokawa, Science 312 (5778) (2006) 1347–1349, http://dx.doi.org/10.1126/science.1126574.

[61] O.S. Barnouin, M.G. Daly, E.E. Palmer, R.W. Gaskell, J.R. Weirich, C.L. Johnson, M.M. Al Asad, J.H. Roberts, M.E. Perry, H.C.M. Susorney, R.T. Daly, E.B. Bierhaus, J.A. Seabrook, R.C. Espiritu, A.H. Nair, L. Nguyen, G.A. Neumann, C.M. Ernst, W.V. Boynton, M.C. Nolan, C.D. Adam, M.C. Moreau, B. Rizk, C.Y. Drouet d'Aubigny, E.R. Jawin, K.J. Walsh, P. Michel, S.R. Schwartw, R.-L. Ballouz, E.M. Mazarico, D.J. Scheeres, J.W. McMahon, W.F. Bottke, S. Sugita, N. Hirata, N. Hirata, S.-i. Watanabe, K.N. Burke, D.N. DellaGiustina, C.A. Bennett, D.S. Lauretta, The OSIRIS-REx Team, Shape of (101955) Bennu indicative of a rubble pile with internal stiffness, Nat. Geosci. 12 (4) (2019) 247–252, http://dx.doi.org/10.1038/s41561-019-0330-x.

[62] D.S. Lauretta, D.N. DellaGiustina, C.A. Bennett, D.R. Golish, K.J. Becker, S.S. Balram-Knutson, O.S. Barnouin, T.L. Becker, W.F. Bottke, W.V. Boynton, H. Campins, B.E. Clark, H.C. Connolly Jr., C.Y. Drouet d'Aubigny, J.P. Dworkin, J.P. Emery, H.L. Enos, V.E. Hamilton, C.W. Hergenrother, E.S. Howell, M.R.M. Izawa, H.H. Kaplan, M.C. Nolan, B. Rizk, H.L. Roper, D.J. Scheeres, P.H. Smith, K.J. Walsh, C.W.V. Wolner, The OSIRIS-REx Team, The unexpected surface of asteroid (101955) Bennu, Nature 568 (7750) (2019) 55–60, http://dx.doi.org/10.1038/s41586-019-1033-6.