TOWARDS VALIDATION AND VERIFICATION OF AUTONOMOUS VISION-BASED NAVIGATION FOR INTERPLANETARY SPACECRAFT

Eleonora Andreis[(1)], **Paolo Panicucci**[(2)], **Fabio Ornati**[(3)], **Francesco Topputo**[(4)]

[(1)]*Ph.D. Student, Politecnico di Milano, Via La Masa, 34, eleonora.andreis@polimi.it*
[(2)]*Assistant Professor, Politecnico di Milano, Via La Masa, 34, paolo.panicucci@polimi.it*
[(3)]*Ph.D. Student, Politecnico di Milano, Via La Masa, 34, fabio.ornati@polimi.it*
[(4)]*Full Professor, Politecnico di Milano, Via La Masa, 34, francesco.topputo@polimi.it*

ABSTRACT

In recent decades, there has been a significant increase in the number of spacecraft launched, allowing both private and public players to participate in space exploration. The traditional ground-based radiometric tracking is becoming outdated due to limitations in communication slots, high costs, and delays. Consequently, autonomous vision-based navigation is emerging as a crucial technology for present and future missions. During interplanetary cruises, the spacecraft can estimate its state by triangulating known planet positions within the Solar System. To ensure accurate and robust algorithms for interplanetary missions, ground testing campaigns are crucial. At this aim, this paper focuses on the validation of the vision-based navigation (VBN) algorithm through hardware-in-the-loop (HIL) simulation. Images are acquired on a simulated trajectory and spacecraft attitude profile using the RETINA test bench. These images are then processed sequentially by the vision-based navigation algorithm and the resulting spacecraft state estimates are compared against the ground truth to evaluate navigation accuracy. The validation process contributes to the advancement and implementation of autonomous navigation for interplanetary missions.

1 INTRODUCTION

As we approach a new era of interplanetary exploration and exploitation, adopting sustainable and efficient navigation methods becomes increasingly important. Traditional ground-based radiometric tracking is accurate and reliable, but it is a costly paradigm and heavily depends on limited resources such as ground stations and dedicated teams, making it unsustainable in the long term [1]. Therefore, enhancing the probe autonomy level is an urgent goal for future deep-space missions, which will prevent ground station saturation. In addition, autonomous guidance, navigation, and control (GNC) solutions onboard spacecraft can further reduce the space-entry price, opening deep-space exploration and exploitation to new and private players. Among the different navigation alternatives for spacecraft, e.g., autonomous X-ray pulsar-based, semi-autonomous radio-based navigation, and autonomous vision-based navigation (VBN), the latter is often preferred thanks to its reduced implementation cost, to its fully ground independence and its applicability on different spacecraft classes [2]. The VBN enables the probes to determine their position by observing celestial bodies on images taken by optical sensors, and it has already been adopted by various missions. In the interplanetary framework, VBN solutions have only undergone onboard testing, starting from the test campaign done during the Deep-Space 1 (DS1) mission in 1998 [3]. Nevertheless, in recent years, there has been a growing interest in VBN algorithms for interplanetary exploration, particularly for CubeSat missions

[4], as attested by the Miniaturized Asteroid Remote Geophysical Observer (M-ARGO) spacecraft, whose goal is to evaluate the feasibility of an onboard autonomous navigation algorithm during the interplanetary transfer towards the target asteroid [5, 6].
To validate the robustness and the accuracy of the VBN algorithms before the onboard adoption, it is necessary to perform on-ground testing campaigns, where the hardware characteristics and mission conditions are as truthful as possible to the one the spacecraft would encounter during an interplanetary mission. Gathering real interplanetary sky-field images to test the algorithm's performance is challenging. Available datasets are limited in number due to the low number of successful missions in the past. Therefore, high-fidelity rendering engines and hardware-in-the-loop (HIL) simulations are needed to generate image datasets. In this framework, this paper presents a validation and verification procedure of an autonomous vision-based navigation algorithm for interplanetary cruises through the exploitation of the RETINA optical facility.
The rest of the paper is structured as follows. Section 2 presents the deep-space rendering engine that has been developed within the EXTREMA project. Section 3 gives an overview of RETINA, the HIL testbench that is used for verification and validation, providing an overview of the geometrical and radiometric calibrations. Then, Section 4 illustrates the vision-based navigation algorithm providing details of the navigation filter implementation and of the image processing solution. Hardware-in-the-loop simulations are presented in Section 5 to underline the filter performance with different illumination and Moon appearance in the image. Finally, conclusions are gathered in Section 6.

2 DEEP-SPACE RENDERING ENGINE

VBN requires images for design, testing, and validation. Although images are available from past missions, they are generally constrained to mission observational geometry which limits their applicability. A solution to this drawback is to design rendering engines that provide images of the observed space scene by setting the scene configuration and the camera characteristics. Space rendering engines, such as SurRender [7], exist in the literature but they are generally proprietary software. Due to this limitation, a dedicated deep-space rendering engine is developed within the EXTREMA project. The deep-space rendering engine is an enhanced version of the one presented in Bella et al. [8]. The simulator renders a deep-space image by taking in input the spacecraft state, the celestial object states, and the camera characteristics. The simulator is designed to take into account light propagation and light aberrations phenomena. The current version of the engine can work in two modes: software simulations and hardware-in-the-loop (HIL) simulations.
In the software simulation mode, the camera is modeled within the rendering engine by taking into account the camera Point Spread Function, the conversion from photon to electron, and the sensor reading. Noises associated with this process are also included, such as the shot noise, the amplifier noise, the reset noise, the pattern noise, and the discretization noise. The rending engine can also take into account the streaks created by celestial objects in the image due to high spacecraft angular velocities. The output of the rendering is an image as it would be taken by the modeled sensor in deep space and whose intensity is expressed in pixel digital count according to the camera bit depth.
On the contrary, the HIL simulations mode foresees the rendering of deep-space scenes to project them on the microdisplay of the RETINA HIL facility (see Section 3). When performing this task particular care must be taken to ensure that the camera is correctly stimulated when mounted on the facility. First, the image displayed on the screen must have the screen resolution and the camera FOV, implying that the rendering camera intrinsic matrix must be selected to enforce these constraints. Second, the rendered scene must be warped to compensate for the distortion and errors induced by the facility. This is possible by estimating the calibration and by using this information to compensate for these effects at the rendering level (see Section 3 for details about the geometrical calibration).

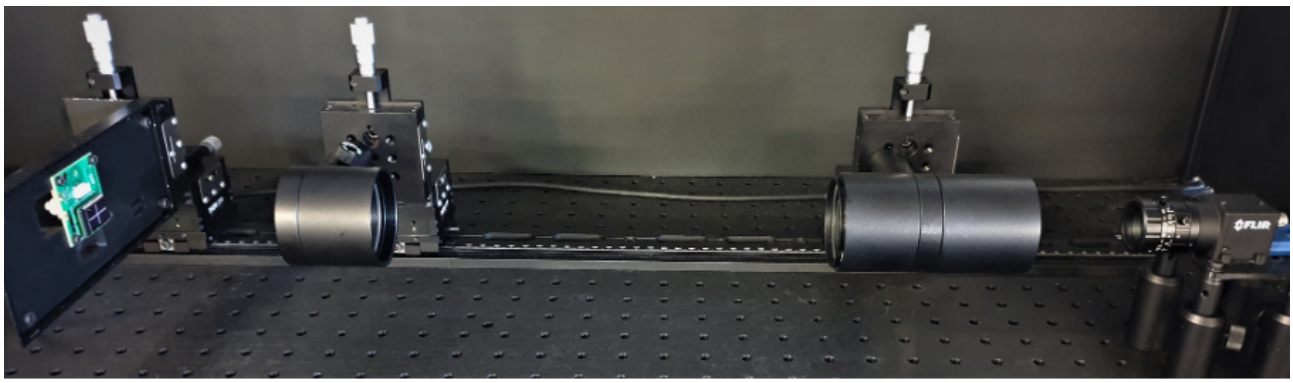


Figure 1: A picture of RETINA

Third, it is worth noting that the screen image pixel digital count depends on the screen radiometric setting, i.e., the quantity of light emitted by the screen substrate and the fraction of this light arriving at the camera pupil. Therefore, it is not possible to predict in advance at the rendering level which is the digital count of each pixel at the screen without performing the facility radiometric calibration (see Section 3 for details about the radiometric calibration). To decouple the problem the deep-space rendering engine generates an image whose pixels contain the irradiance observed by a camera in deep space. Then the irradiance is converted to the correct screen pixel digital count by knowing the photometric response of the facility.
It is important to recall that both modes are crucial for the correct validation procedure as the first is used for VBN design and validation in software-in-the-loop simulations, while the second one is used for VBN validation in hardware-in-the-loop simulations.

3 RETINA: LOW-ABERRATION HARDWARE-IN-THE-LOOP FACILITY FOR VISION-BASED NAVIGATION

A major concern once a VBN algorithm is designed is how to prove its robustness to hardware error and off-nominal performance. To perform verification and validation, in the space sectors it is possible to use real images from past missions, high-fidelity rendering engines images, or ground-based hardware-in-the-loop (HIL) facility simulations that reproduce space-borne observational conditions [9]. To assess the preliminary performance of the algorithm, a dedicated simulation framework can be implemented to prove the feasibility and to tune the algorithm parameters [10, 11]. As it is hard to correctly model the camera sensor and its source of error in detail, it is useful to directly test the developed algorithm in HIL test benches to correctly assess the algorithm performance with a real sensor without the need for any modeling effort. Thus, a HIL optical facility was developed: RETINA (Realistic Experimental faciliTy for vIsion-based NAvigation). The main goal of RETINA is to provide the VBN algorithm with an image as it would be acquired from the same hardware in space.
RETINA, shown in Figure 1, is composed of a high-resolution miniaturized screen, a lens system that collimates the screen light, and an optical camera consistent with CubeSats hardware. The camera, the system of lenses, and the screen are mounted on an optical breadboard and placed in a dark box to avoid light contamination. The lens system is designed to provide collimation for variable camera focal lengths, sensor sizes, and F-numbers by ensuring a low level of chromatic and achromatic aberrations at the camera pupil. The screen is a high-resolution microdisplay selected to reduce facility size and to have fine control of the screen pixel irradiance. The RETINA components are mounted on optical stages to ensure their precise alignment and centering.
Before using RETINA, two main tasks must be performed which ensures the correct use of the fa-

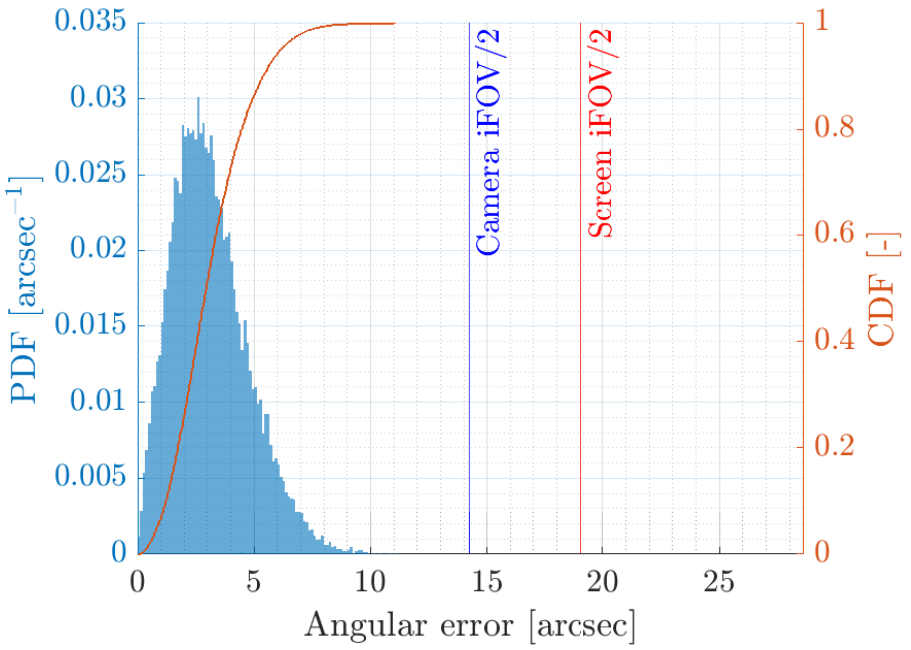
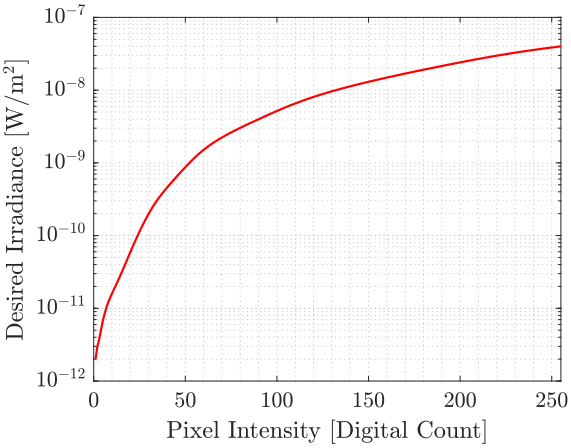


Figure 2: CDF and PDF of the angular errors between the LoSes displayed on the screen and the LoSes detected at the camera after calibration.

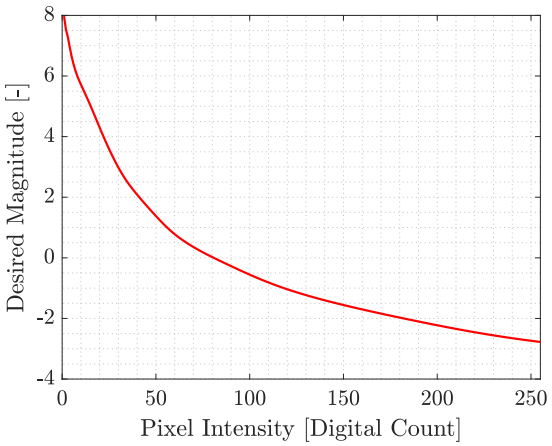
cility: geometrical and radiometric calibration. These tasks are necessary to ensure that the celestial body displayed on the screen is consistent with what is observed from the camera on orbit both geometrically and radiometrically.

On the one hand, the geometric calibration ensures that each object in the field of view is observed as it would be observed in orbit (see Panicucci and Topputo [9] for mathematical demonstration). This is necessary to correct for residual errors in hardware components' alignment and centering and to correct for lens assembly distortions. The calibration is obtained as in Panicucci and Topputo [9] by projecting several patterns on the screen and acquiring images with the camera. By knowing the location of points in the screen images and by associating the correct point in the camera image, it is possible to minimize the error between the two LoSes. This process provides a distortion model of RETINA. To correctly compensate for these RETINA distortions, the inverse of the distortion model is applied to the LoSes in the Deep-Space rendering engine, and the distorted image is projected on the screen. As the applied distortion is the inverse of RETINA one, the image taken by the camera is not affected by the HIL errors. An example of the Cumulative Distribution Function (CDF) and the Probability Distribution Function (PDF) of the error between the LoSes of the simulator and the LoSes in the camera image after the calibration are reported in Figure 2, showing that star and planet centroids have an angular error below 10 arcsec in more than 99% of the projected points.

On the other hand, the radiometric calibration determines the mapping between the desired irradiance detected at the camera pupil and the digital count of the screen pixel. The screen was selected among the ones available on the market to have a low pixel luminance and fine controllability of the radiometric response. Moreover, the selected display has high contrast between dark and illuminated pixels leading to correctly displaying the dark deep-space background. Thanks to these characteristics, the screen radiometric settings can be tuned to let the display work in the desired illuminance range at the camera pupil. To do so, a series of pixels has been illuminated at screen level by increasing the pattern digital count. A powermeter was placed at the collimator exit surface to compute the irradiance received by the camera. The obtained mapping is shown in Figure 3a. The irradiance was then converted to celestial object absolute magnitude as shown in Figure 3b to obtain the facility radiometric



(a) The mapping between the desired irradiance at the camera pupil and the screen pixel digital count.



(b) The mapping between the desired magnitude at the camera pupil and the screen pixel digital count.

Figure 3: The radiometric calibration of RETINA.

simulation range. The facility can cover a wide range of targets from faint stars to brighter celestial objects.

4 INTERPLANETARY AUTONOMOUS OPTICAL NAVIGATION PROBLEM

In deep space, the probe can estimate its position by observing the external environment, from which it can acquire information about the location of moving celestial bodies, e.g., planets and asteroids. Since these beacons appear as unresolved on images acquired during an interplanetary transfer, i.e., their light falls within a single pixel, their line-of-sight (LoS) directions are the only exploitable information to estimate autonomously the probe state on board. If two or more beacons' LoS directions are available, these latter can be used to statically triangulate the spacecraft's position. Otherwise, a dynamic estimator is needed. Since this work is framed inside the EXTREMA project [12], whose goal is to enable nanosatellites to travel autonomously in deep space, the CubeSat application is investigated. CubeSats differ from standard spacecraft for their low production cost, due to the adoption of commercial-off-the-shelf components, and their miniaturized dimensions. Following this paradigm, the on-board optical sensor has to be low-cost and miniaturized, yielding limited performances in terms of celestial object detectability. Usually, a miniaturized optical sensor has a threshold magnitude of 6. Therefore, only major celestial bodies (i.e., planets) can be exploited to navigate, whereas asteroids are too faint to be observed [13]. Due to the sparsity of the planets' position, it is unlikely to find more than one observable celestial body in the image. Therefore, to localize itself the probe tracks the first planet for a time interval, and then perform a slew maneuver to point to the successive planet. As measurements are not simultaneous, an extended Kalman filter fed by asynchronous planet measurements is exploited.
It is worth noting that the influence of light effects, such as light-time and light aberration, on celestial bodies' observations plays a crucial role in deep space [14, 15]. Therefore, when estimating the probe state, it is essential to correct these effects to avoid biases in the estimation.

4.1 Optical Navigation Filter for Interplanetary Cruise

A non-dimensionalized EKF is adopted to implement the optical autonomous navigation strategy because of its better numerical stability and computational performance [15]. The implemented scheme

is reported in Table 1, where all the terms are already non-dimensionalized following the approach discussed in Andreis et al. [15].

Table 1: Filtering Strategy

System State Space	$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}(t), t) + \boldsymbol{w}$
	$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k) + \boldsymbol{\nu}_k$
	$\dot{\boldsymbol{P}} = \boldsymbol{F}\boldsymbol{P} + \boldsymbol{P}\boldsymbol{F}^\top + \boldsymbol{Q}$
Propagation Block	$\boldsymbol{x}_{p_k} = \boldsymbol{x}_{c_{k-1}} + \int_{t_{k-1}}^{t_k} \boldsymbol{f}(\boldsymbol{x}(t), t)\mathrm{d}t$
	$\boldsymbol{P}_{p_k} = \boldsymbol{P}_{c_{k-1}} + \int_{t_{k-1}}^{t_k} \dot{\boldsymbol{P}}\mathrm{d}t$
Correction Block	$\boldsymbol{K}_k = \boldsymbol{P}_{p_k}\boldsymbol{H}_k^\top(\boldsymbol{H}_k\boldsymbol{P}_{p_k}\boldsymbol{H}_k^\top + \boldsymbol{R}_k)^{-1}$
	$\boldsymbol{x}_{c_k} = \boldsymbol{x}_{p_k} + \boldsymbol{K}_k[\boldsymbol{y}_k - \boldsymbol{h}(\boldsymbol{x}_{p_k})]$
	$\boldsymbol{P}_{c_k} = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H}_k)\boldsymbol{P}_{p_k}(\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H}_k)^\top + \boldsymbol{K}_k\boldsymbol{R}_k\boldsymbol{K}_k^\top$

In Table 1, $\boldsymbol{x}_{p_k}$ is the predicted state vector with error covariance matrix $\boldsymbol{P}_{p_k}$ at epoch t_k, $\boldsymbol{f}$ the dynamics model equation with Jacobian $\boldsymbol{F}$, $\boldsymbol{K}_k$ the Kalman gain, $\boldsymbol{x}_{c_k}$ the corrected state vector with error covariance matrix $\boldsymbol{P}_{c_k}$, $\boldsymbol{h}$ the measurement model equation with Jacobian $\boldsymbol{H}$, and $\boldsymbol{y}_k$ the external observation. In the following sections, the image processing procedure to extract the planet observables from deep-space images, and the dynamic and measurement models are further detailed.

4.1.1 Dynamics Model

The dynamics model is an extension of the one detailed in Andreis et al. [15]. The state vector $\boldsymbol{x}$ is defined as

$$\boldsymbol{x}(t) = [\boldsymbol{r}(t), \boldsymbol{v}(t), \boldsymbol{\eta}(t)]^\top \tag{1}$$

where $\boldsymbol{r}$ and $\boldsymbol{v}$ are the inertial probe position and velocity, respectively, and $\boldsymbol{\eta}$ is a vector of Gauss–Markow (GM) processes accounting for unmodeled terms: a 3-dimensional residual accelerations $\boldsymbol{\eta}_{\mathrm{R}}$ and the stochastic component of the Solar Radiation Pressure (SRP) $\boldsymbol{\eta}_{\mathrm{SRP}}$; that is, $\boldsymbol{\eta} = [\boldsymbol{\eta}_{\mathrm{R}}, \boldsymbol{\eta}_{\mathrm{SRP}}]^\top$. In this work, the VBN algorithm is tested on a ballistic trajectory, thus, the GM processes related to the off-nominal behavior of the low-thrust engine are not considered. The process is modeled using the following equation of motion

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), t) + \boldsymbol{w} \tag{2}$$

where $\boldsymbol{f}$ is the vector field embedding the deterministic part, while $\boldsymbol{w}$ is the process white noise:

$$\dot{\boldsymbol{x}}(t) = \underbrace{\begin{bmatrix} \boldsymbol{v} \\ \boldsymbol{a}_{\mathrm{Sun}} + \boldsymbol{a}_{\mathrm{SRP}} + \boldsymbol{a}_{\mathrm{pl_i}} \\ -\xi\boldsymbol{\eta}_{\mathrm{R}} \\ -\xi\boldsymbol{\eta}_{\mathrm{SRP}} \end{bmatrix}}_{\boldsymbol{f}} + \underbrace{\begin{bmatrix} \boldsymbol{0}_{3x1} \\ \boldsymbol{\eta}_{\mathrm{R}} + \boldsymbol{\eta}_{\mathrm{SRP}} \\ \boldsymbol{w}_{\mathrm{R}} \\ \boldsymbol{w}_{\mathrm{SRP}} \end{bmatrix}}_{\boldsymbol{w}} \tag{3}$$

and

$$\boldsymbol{a}_{\mathrm{Sun}} = -\mu_{\mathrm{Sun}}\frac{\boldsymbol{r}}{r^3} \tag{4}$$

$$\boldsymbol{a}_{\mathrm{SRP}} = C_{\mathrm{R}}\frac{P_0 R_0^2}{c}\frac{A_{\mathrm{s}}}{m_{\mathrm{s}}}\frac{\boldsymbol{r}}{r^3} \tag{5}$$

$$\boldsymbol{a}_{\mathrm{pl_i}} = \mu_i\left(\frac{\boldsymbol{r}_{\mathrm{pl_i}} - \boldsymbol{r}}{||\boldsymbol{r}_{\mathrm{pl_i}} - \boldsymbol{r}||^3} - \frac{\boldsymbol{r}_{\mathrm{pl_i}}}{||\boldsymbol{r}_{\mathrm{pl_i}}||^3}\right) \tag{6}$$

The terms that describe the SRP are [16]: C_{R} the coefficient of reflection, P_0 the solar power, R_0 the Sun radius, A_{s} the cross-section area of the probe, and m_{s} its mass. The third-body perturbations of the Earth-Moon barycenter, Mars, and Jupiter are included. In the Langevin equations that govern the GM processes the coefficient ξ defines the reciprocal of the correlation time, while $\boldsymbol{w}_{\mathrm{R}}$ and $\boldsymbol{w}_{\mathrm{SRP}}$ are the process noises of the GM parameters with σ_{R} and σ_{SRP} standard deviations, respectively [17]. The process noise covariance matrix is $\boldsymbol{Q}$:

$$\boldsymbol{Q} = \mathrm{diag}(\boldsymbol{0}_{3x3}, \boldsymbol{Q}_a, \boldsymbol{Q}_{\mathrm{R}}, \boldsymbol{Q}_{\mathrm{SRP}}) \tag{7}$$

with $\boldsymbol{Q}_{\mathrm{R}} = \sigma_{\mathrm{R}}^2 \boldsymbol{I}_{3x3}$, $\boldsymbol{Q}_{\mathrm{SRP}} = \sigma_{\mathrm{SRP}}^2 \boldsymbol{I}_{3x3}$, and $\boldsymbol{Q}_{\mathrm{a}} = (\boldsymbol{Q}_{\mathrm{R}} + \boldsymbol{Q}_{\mathrm{SRP}})/(2\xi)$.

4.1.2 Measurement Model

The exploited measurement model represents the planet position projection affected by the light-time and light-aberration effects in pixel coordinates in the 2D camera reference frame:

$$^{\mathbb{C}}_{h}\boldsymbol{r}_{\mathrm{pl}} = \boldsymbol{h}(\boldsymbol{x}_k) = \boldsymbol{K}_{\mathbf{cam}} \boldsymbol{A} \boldsymbol{l}^{\mathrm{aberr}}_{\mathrm{pl/sc}} \tag{8}$$

where

$$\boldsymbol{l}^{\mathrm{aberr}}_{\mathrm{pl/sc}} = \boldsymbol{l}_{\mathrm{pl/sc}} + \boldsymbol{l}_{\mathrm{pl/sc}} \times \left(\boldsymbol{\beta}_{\mathrm{sc}} \times \boldsymbol{l}_{\mathrm{pl/sc}}\right) \tag{9}$$

$\boldsymbol{\beta}_{\mathrm{sc}} = \dfrac{\boldsymbol{v}}{c}$, and $\boldsymbol{l}_{\mathrm{pl/sc}}$ is the planet LoS direction from the spacecraft position at time t, i.e., the epoch at which the light is received by the spacecraft, to the planet position at time τ, i.e., the epoch at which the light is emitted by the planet, with $\tau < t$. Thus:

$$\boldsymbol{l}_{\mathrm{pl/sc}} = \frac{\left(\boldsymbol{r}_{\mathrm{pl}}(t-\Delta t) - \boldsymbol{r}(t)\right)^{\top} \left(\boldsymbol{r}_{\mathrm{pl}}(t-\Delta t) - \boldsymbol{r}(t)\right)}{\left|\left|\left(\boldsymbol{r}_{\mathrm{pl}}(t-\Delta t) - \boldsymbol{r}(t)\right)^{\top} \left(\boldsymbol{r}_{\mathrm{pl}}(t-\Delta t) - \boldsymbol{r}(t)\right)\right|\right|} \tag{10}$$

where $\Delta t = t - \tau$.

4.1.3 External Observations Acquisition

The observable used to correct the probe's state estimation in the navigation filter is the projection of the planet position $^{\mathbb{C}}\boldsymbol{r}_{\mathrm{pl}_0}$ onto the 2D camera reference frame $\mathbb{C}$. To extract this information from the deep-space image, an image processing (IP) algorithm specifically designed for interplanetary cruises is employed [11, 18]. The main objective of this procedure is to identify the centroid in the image that corresponds to the planet projection and to obtain information about the planet's location in the image. The IP pipeline follows a three-step process: 1) Determining the probe's attitude, 2) Correcting the light aberration of the stars' centroids, and 3) Identifying the planets in the image. The first step is performed differently if the planet is acquired for the first time or not. In the former situation, a lost-in-space (LIS) algorithm is applied to determine the spacecraft attitude without any previous knowledge of the probe pose. In particular, the pyramid star identification algorithm [19] bulked up by a RANSAC procedure to reject outliers is exploited. When the planet is not observed for the first time, a recursive attitude determination method can be applied to recover the probe orientation since the information about the estimated attitude at the previous step is available. In detail, by knowing the previous attitude estimation, the LoS directions in the inertial reference frame $\mathcal{N}$ of the four corners of the image are determined. At this point, a check is performed to identify which stars of the onboard catalog are contained inside the space region delimited by the image, and the position projections in the 2D camera reference frame of the stars confined in the image are evaluated. Eventually, the centroids of the bright objects in the image are associated with the closest cataloged stars' position

projections. A RANSAC algorithm is always applied at the end of the procedure to reject possible false matches. When the recursive attitude determination fails, the spacecraft orientation at the following image acquisition will be determined again with the LIS method. Vice versa, when the LIS algorithm succeeds in the determination of the probe orientation, in the following image acquisition the recursive attitude determination algorithm will be adopted. Fig. 4 shows the workflow of the first step of the image processing algorithm. The second step of the IP consists of the correction of the

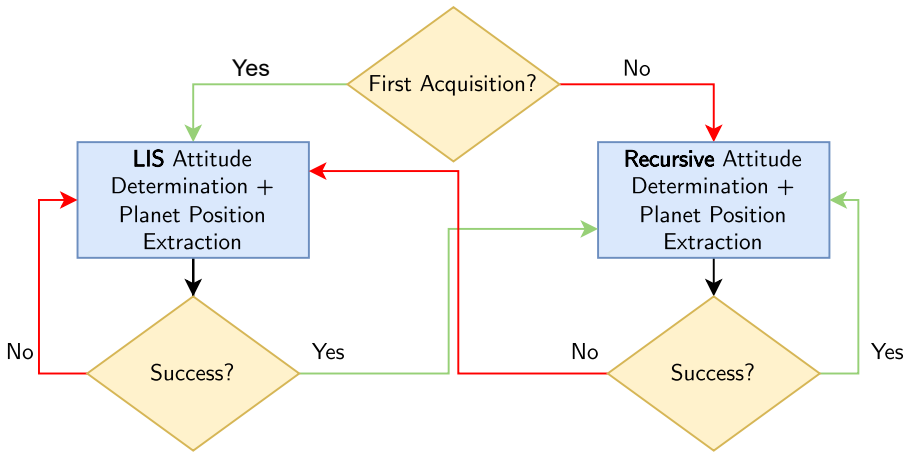


Figure 4: Image Processing Workflow

shift caused by the light aberration on the stars' centroids. The procedure applied to correct the effect is the one applied to planets in Raymond Karimi and Mortari [14]. Eventually, the probe orientation is recomputed with the corrected stars' centroids.

The final step is the identification of the planets in the image and the extraction of their position projections, which is used to correct the state estimation within the filter. The recognition is performed through the evaluation of the statistical momenta associated with the planet position projection, which defines the Gaussian probability to find the planet in that portion of the image. At first, the expected position projection of the observed planet is evaluated as: ${}^{\mathbb{C}}_{h}\mathbf{r}_{\mathrm{pl}_0} = \boldsymbol{K}_{\mathrm{cam}}\boldsymbol{A}({}^{\mathcal{N}}\mathbf{r}_{\mathrm{pl}} - {}^{\mathcal{N}}\mathbf{r}_{\mathrm{p}})$ where ${}^{\mathcal{N}}\mathbf{r}_{\mathrm{p}}$ is the predicted probe position in $\mathcal{N}$, ${}^{\mathcal{N}}\mathbf{r}_{\mathrm{pl}}$ the planet position in $\mathcal{N}$, $\boldsymbol{K}_{\mathrm{cam}}$ the camera calibration matrix, and $\boldsymbol{A}$ the estimated attitude. If ${}^{\mathbb{C}}\boldsymbol{r}_{\mathrm{pl}_0}$ falls within the boundaries of the image, its associated uncertainty ellipse can be defined. This latter depends on the uncertainties of the spacecraft pose and planet position and is centered in ${}^{\mathbb{C}}\mathbf{r}_{\mathrm{pl}_0}$. The ellipse of ${}^{\mathbb{C}}\mathbf{r}_{\mathrm{pl}_0}$ represents the area of the image where the planet is most likely to be found within a 3σ probability. The spike contained in the 3σ ellipse is identified as the planet position projection ${}^{\mathbb{C}}\mathbf{r}_{\mathrm{pl}}$. If multiple spikes are located within this ellipse, the object closest to the expected planet position is considered to be the planet itself, as it is most likely the true projected position of the planet.

If the IP algorithm fails, two procedures are implemented depending on the failure type: 1) when the attitude is not determined or the planet measurement is not extracted from the image, the state vector and its error covariance matrix are simply propagated until the next step; 2) when the planet is wrongly determined, a course outlier detection method is applied to reject false positives [20]. In particular, when the absolute value of the innovation term is greater than $k\sqrt{\boldsymbol{M}_{ii}}$ with $\boldsymbol{M} = \boldsymbol{H}_k\boldsymbol{P}_{p_k}\boldsymbol{H}_k^{\top} + \boldsymbol{R}_k$ and $k = 4$, the innovation term is set to zero, and the correction of the propagated state estimation is not performed. It is, indeed, preferred to keep an old but good prediction not to worsen the estimation.

5 NUMERICAL RESULTS

5.1 Navigation Cycle

During the interplanetary cruise, a combination of observation windows and only-propagation windows is employed. In the observation windows, the probe tracks asynchronously two planets whose measurements are adopted to correct the state vector estimation. The optimal couple of planets to observe is determined by following the strategy presented in Franzese and Topputo [21]. In the only-propagation windows, the state vector is only propagated and not corrected. During a navigation leg, firstly, the probe observes the first planet of the optimal pair for one hour with a tracking frequency of 0.01 Hz, then executes a slew maneuver to orient itself toward the second planet for 30 minutes, and eventually proceeds to track the second planet for another hour. At the end of the acquisition of the second planet, the state vector estimation is propagated for 10 days before starting the next navigation leg. Note that during the slew maneuver, no observations are acquired from the external environment, thus it is an only-propagation window. Fig. 5 shows a graphical representation of one navigation leg.

Figure 5: Navigation Leg

5.2 Simulation Settings

In this work, the proposed autonomous optical navigation filter is tested on an interplanetary high-fidelity ballistic trajectory between Earth–Mars [22]. Fig. 6 shows the analyzed leg of the nominal probe trajectory lasting 100 days. The initial standard deviations of the state are reported in Table 2.

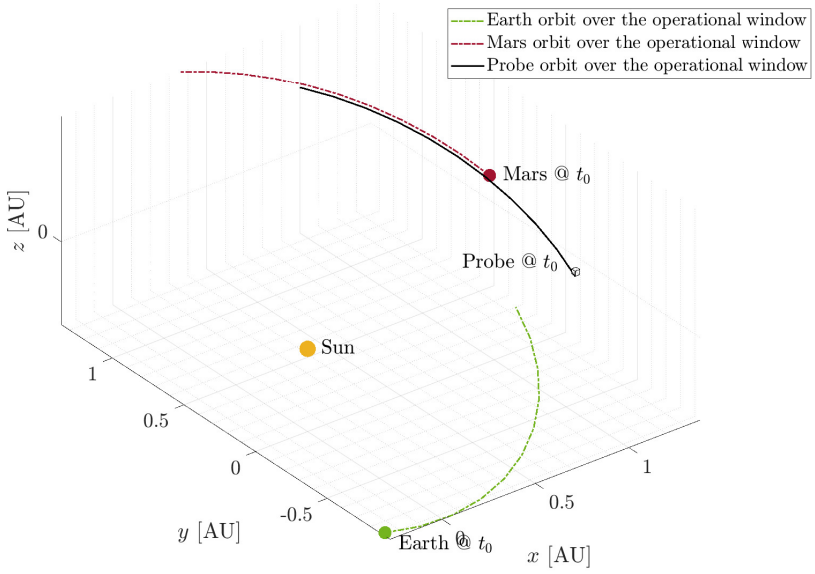


Figure 6: Ballistic Interplanetary Reference Trajectory

Table 2: Accuracy of the state components at t_0

σ_{r} [km]	σ_{v} [km/s]	σ_{SRP} [km/s^2]	σ_{R} [km/s^2]
10^4	10^{-1}	5×10^{-10}	5×10^{-10}

The standard deviation of the measurement error is set to $\sigma_{\mathrm{str}} = 0.55$ px by taking the upper bound error of the calibration procedure.
Eventually, only planets whose apparent magnitude is lower than 7 and whose Solar Exclusion Angle (SAE) is greater than 20° are assumed to be visible by the camera, therefore, only these are considered available for the optimal beacon selection process.

5.3 Simulation Results

A Monte Carlo run of 100 samples is performed. The image acquisition and the state estimation steps are asynchronously implemented. At first, the deep-space images are generated by exploiting the sky-field rendering engine with as input the true probe pose history, then, they are displayed on the RETINA's micro screen and acquired by the camera. In a second time, the state estimator is implemented and fed by the deep-space images previously acquired. The real-time processor-in-the-loop simulation will be the object of future investigations.
Figs. 7 and 8 show the position and velocity error profiles and covariance bounds in the J2000 ecliptic reference frame over the analyzed trajectory leg. At the end of the considered operational window of 100 days, the filter estimates the position and the velocity of the spacecraft with a 3σ accuracy of 5000 km and 1.6 m/s.
The achieved accuracy is lower than the one obtained without RETINA in the loop. In particular, in Andreis et al. [10], where the filter is fed by images fully and only generated by the sky-field rendering engine, the position, and velocity of the spacecraft have been estimated with an accuracy of about 1000 km and 0.4 m/s. The decrease in the filter performances is due to the increment of the measurement error which goes from a value of 0.1 pixels in Andreis et al. [10] to a value of 0.5 pixels when the optical facility is in the simulation loop. This measurement error accretion is caused by the presence of residual calibration errors at the image acquisition level. Indeed, note that in Fig. 2 the centroids acquired during the calibration procedure can reach an error of 0.33 pixels.
Eventually, to remark on the paramount importance of including light-effect compensations inside the filtering scheme, Figs. 9 and 9 show the estimator performances when these effects are not taken into account. Note that the position and velocity errors do not converge to zero when light-effects compensations are not included in the filtering scheme, yielding a biased estimation of the probe state.

6 CONCLUSION

In this paper, the performances of an optical autonomous navigation filter suited for deep-space cruises are tested and validated through the exploitation of the RETINA optical facility, where the hardware characteristics and mission conditions are representative of the ones the spacecraft would encounter during an interplanetary mission.
Preparation activities have to be performed before starting the validation of the VBN algorithms. These include the RETINA setting up through its geometrical and radiometric calibration and the development of a high-fidelity rendering engine to generate realistic sky-field images to display on the RETINA screen.
In this work, a trajectory leg between Earth and Mars is considered as a test case. At the end of the analyzed flying window, the filter estimates the spacecraft position and velocity with a 3σ accuracy of

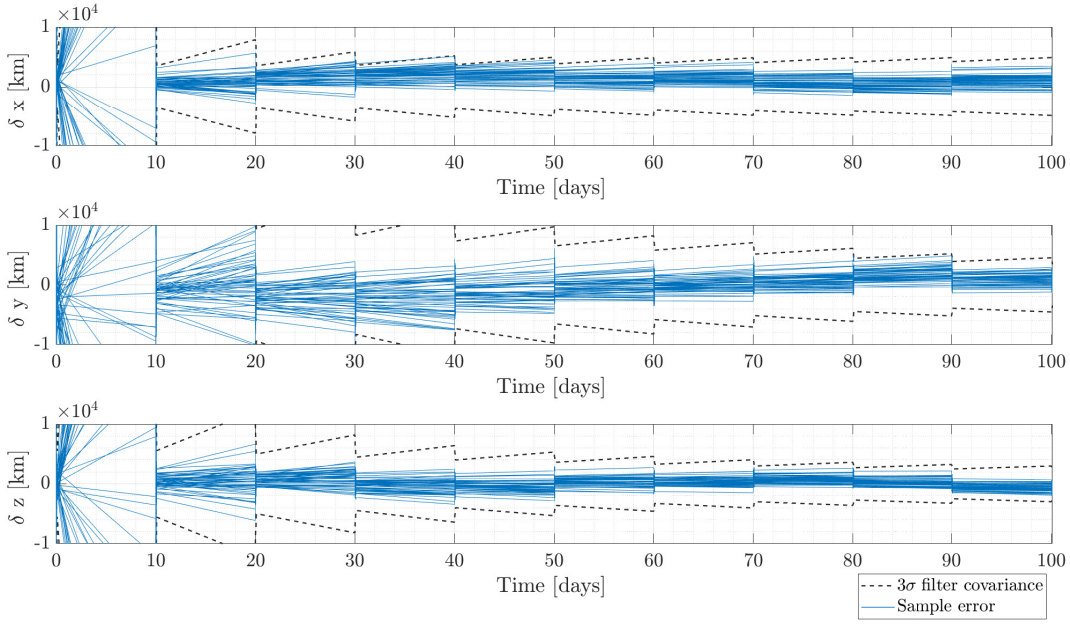


Figure 7: Estimated errors for each position component with related 3σ bounds. For clarity sake, only 50 out of 100 Monte Carlo sample histories are reported.

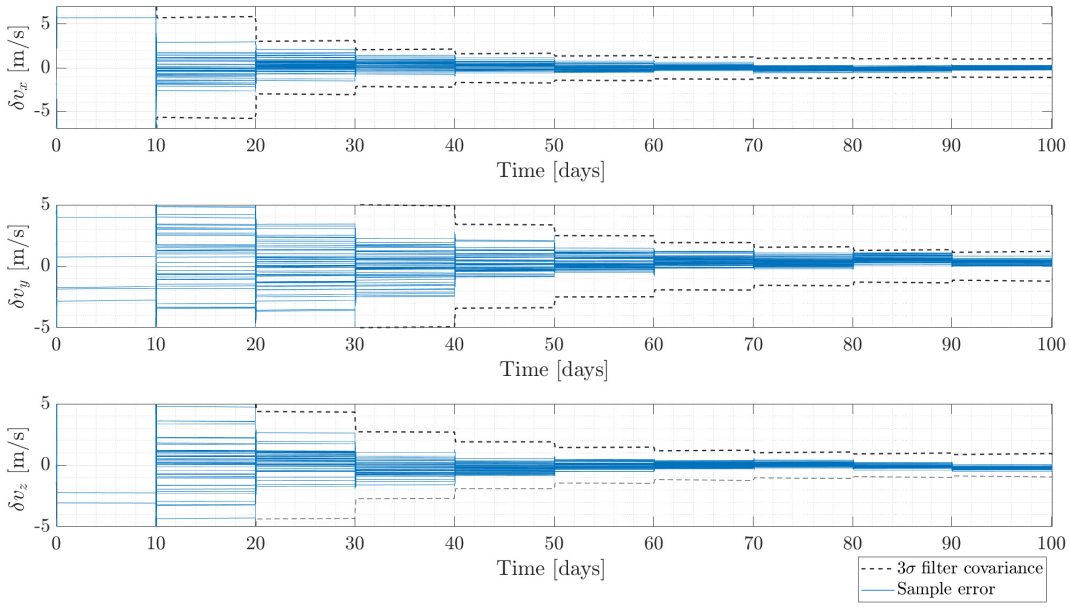


Figure 8: Estimated errors for each velocity component with related 3σ bounds. For clarity sake, only 50 out of 100 Monte Carlo sample histories are reported

5000 km and 1.6 m/s, respectively. These performances are strictly related to the measurement errors affecting the centroids extracted from the RETINA images, which are partially due to the calibration performance in the compensation of the distortions introduced by the hardware.
Future analysis should test the performance of the navigation filter through real-time camera- and processor-in-the-loop simulations, where the deployment of the algorithm on a processor representative of a miniaturized onboard computer such as a Raspberry Pi will be performed.

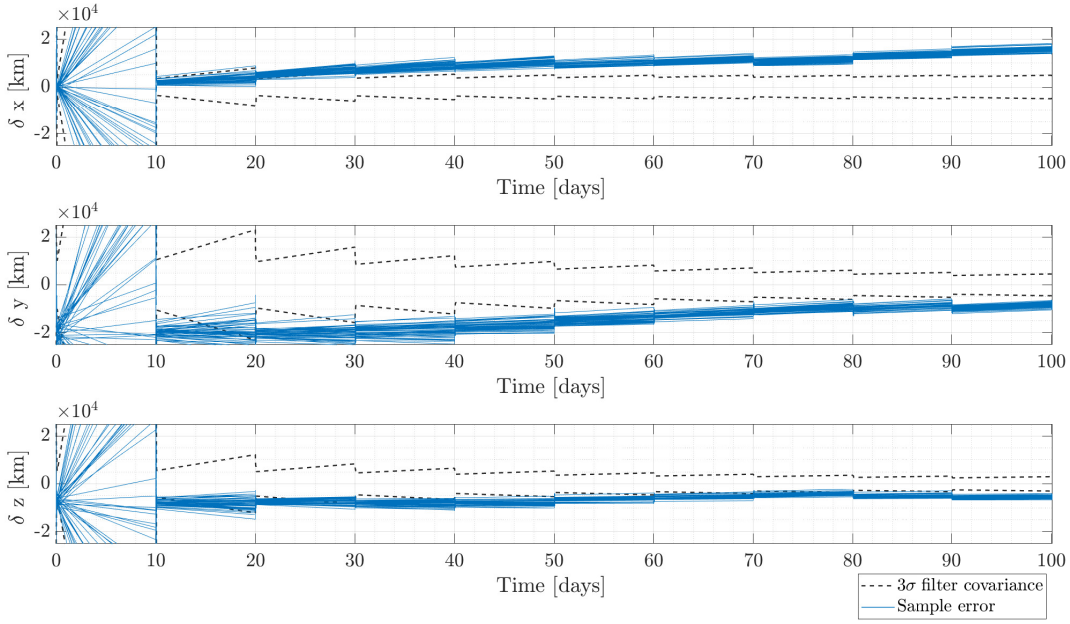


Figure 9: Estimated position error profiles with related 3σ bounds when no light-effect compensations are implemented. For clarity sake, only 50 Monte Carlo samples are reported.

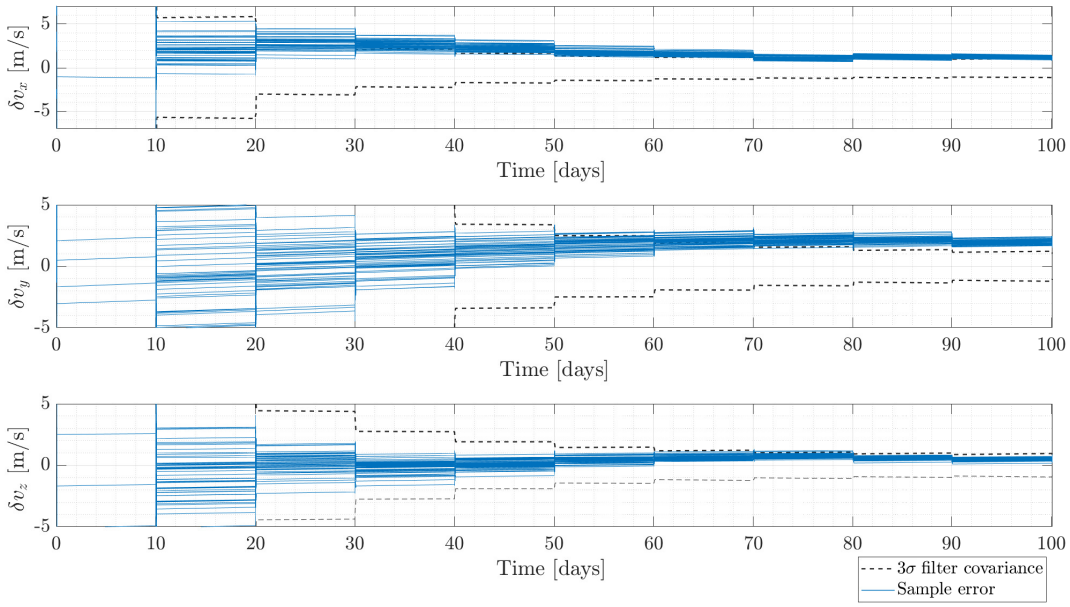


Figure 10: Estimated velocity error profiles with related 3σ bounds when no light-effect compensations are implemented. For clarity sake, only 50 Monte Carlo samples are reported.

ACKNOWLEDGMENTS

This research is part of EXTREMA, a project that has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant Agreement No. 864697). The authors are grateful to Dr. Gianmario Merisio for the computation of the high-fidelity trajectory.

REFERENCES

[1] S. M. Lichten, D. S. Abraham, Belinda Arroyo, Sami W. Asmar, Julia Bell, and Charles D. Edwards. Allocation of deep space network ground system tracking and communications assets during the 2020-2021 timeframe of the "mars armada". In *2018 SpaceOps Conference*, 2018. doi: 10.2514/6.2018-2502.

[2] Stefano Speretta, Angelo Cervone, Prem Sundaramoorthy, Ron Noomen, Samiksha Mestry, Ana Cipriano, Francesco Topputo, James Biggs, Pierluigi Di Lizia, Mauro Massari, et al. Lumio: an autonomous cubesat for lunar exploration. *Space operations: inspiring Humankind's future*, pages 103–134, 2019.

[3] Shyam Bhaskaran. Autonomous navigation for deep space missions. In *SpaceOps 2012*, page 1267135. 2012.

[4] Erdem Turan, Stefano Speretta, and Eberhard Gill. Autonomous navigation for deep space small satellites: Scientific and technological advances. *Acta Astronautica*, 2022.

[5] V Franzese, F Topputo, F Ankersen, and R Walker. Deep-space optical navigation for m-argo mission. *The Journal of the Astronautical Sciences*, 68(4):1034–1055, 2021.

[6] Francesco Topputo, Yang Wang, Carmine Giordano, Vittorio Franzese, Hannah Goldberg, Franco Perez-Lissi, and Roger Walker. Envelop of reachable asteroids by m-argo cubesat. *Advances in Space Research*, 67(12):4193–4221, 2021.

[7] Jérémy Lebreton et al. Image Simulation for Space Applications with the SurRender Software. In *Proceedings of the 11th International ESA Conference on Guidance, Navigation & Control Systems, 22 - 25 June 2021, Virtual*, 2021.

[8] S. A. Bella, E. Andreis, V. Franzese, P. Panicucci, and F. Topputo. Line-of-sight extraction algorithm for deep-space autonomous navigation. In *2021 AAS/AIAA Astrodynamics Specialist Conference*, 08 2021. URL `http://hdl.handle.net/11311/1183096`.

[9] Paolo Panicucci and Francesco Topputo. The tinyv3rse hardware-in-the-loop vision-based navigation facility. *Sensors*, 22(23):9333, 2022.

[10] E Andreis, P Panicucci, V Franzese, F Topputo, et al. A vision-based navigation algorithm for autonomous deep-space cruise. In *3rd Space Imaging Workshop*, pages 1–10, 2022.

[11] Eleonora Andreis, Paolo Panicucci, and Francesco Topputo. An image processing pipeline for autonomous deep-space optical navigation. *arXiv preprint arXiv:2302.06918*, 2023.

[12] Gianfranco Di Domenico, Eleonora Andreis, Andrea Carlo Morelli, Gianmario Merisio, Vittorio Franzese, Carmine Giordano, Alessandro Morselli, Paolo Panicucci, Fabio Ferrari, and Francesco Topputo. The erc-funded extrema project: Achieving self-driving interplanetary cubesats. In *Modeling and Optimization in Space Engineering: New Concepts and Approaches*, pages 167–199. Springer, 2022.

[13] Vittorio Franzese and Francesco Topputo. Celestial bodies far-range detection with deep-space cubesats. *Sensors*, 23(9):4544, 2023.

[14] Reza Raymond Karimi and Daniele Mortari. Interplanetary autonomous navigation using visible planets. *Journal of Guidance, Control, and Dynamics*, 38(6):1151–1156, 2015.

[15] Eleonora Andreis, Vittorio Franzese, and Francesco Topputo. Onboard orbit determination for deep-space cubesats. *Journal of guidance, control, and dynamics*, pages 1–14, 2022.

[16] Isabelle Jean, Alfred Ng, and Arun K. Misra. Impact of solar radiation pressure modeling on orbital dynamics in the vicinity of binary asteroids. *Acta Astronautica*, 165:167–183, 2019. ISSN 0094-5765. doi: https://doi.org/10.1016/j.actaastro.2019.09.003.

[17] C. N. D'Souza J. R. Carpenter. Navigation Filter Best Practices. Technical Report 20180003657, NASA, 04 2018.

[18] E Andreis, P Panicucci, V Franzese, and 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*, pages 1–19, 2022.

[19] Daniele Mortari, Malak A Samaan, Christian Bruccoleri, and John L Junkins. The pyramid star identification technique. *Navigation*, 51(3):171–183, 2004. doi: https://doi.org/10.1002/j.2161-4296.2004.tb00349.x.

[20] Hancong Liu, Sirish Shah, and Wei Jiang. On-line outlier detection and data cleaning. *Computers & chemical engineering*, 28(9):1635–1647, 2004.

[21] V Franzese and F Topputo. Optimal beacons selection for deep-space optical navigation. *The Journal of the Astronautical Sciences*, 67(4):1775–1792, 2020.

[22] Gianmario Merisio, Francesco Topputo, et al. Characterization of ballistic capture corridors aiming at autonomous ballistic capture at mars. In *2021 AAS/AIAA Astrodynamics Specialist Conference*, pages 1–21, 2022.