



POLITECNICO
MILANO 1863

RE.PUBLIC@POLIMI

Research Publications at Politecnico di Milano

This is the accepted version of:

M. Pugliatti, P. Panicucci, V. Franzese, F. Topputo
Tinyv3rse: the Dart Vision-Based Navigation Test-Bench
in: AIAA Scitech 2022 Forum, AIAA, 2022, ISBN: 9781624106316, p. 1-13, AIAA 2022-1193
[AIAA Scitech 2022 Forum, San Diego, CA, USA & Virtual Conference, 3-7 Jan. 2022]
doi:10.2514/6.2022-1193

The final publication is available at <https://doi.org/10.2514/6.2022-1193>

When citing this work, cite the original published paper.

Permanent link to this version

<http://hdl.handle.net/11311/1196040>

TinyV3RSE: The DART Vision-Based Navigation Test Bench

Mattia Pugliatti*, Paolo Panicucci†, Vittorio Franzese‡, and Francesco Topputo§
Politecnico di Milano, Via La Masa 34, 20156, Milan, Italy

Image processing and vision-based navigation algorithms require images for design, testing, and validation. For space exploration purposes, it is complex if not impossible to retrieve realistic images. To mitigate this, two approaches can be used: high-fidelity rendering of celestial bodies or hardware-in-the-loop testing. In this work, we focus on the latter by elaborating on the design, implementation, validation, and calibration of a vision-based navigation test bench called TinyV3RSE. The design of such facility has been a collaborative effort at the Deep-space Astrodynamics Research & Technology (DART) group, which will benefit from its usage in various projects and missions in which is involved. In this work, for the first time, we present the facility design, the current calibration procedure, and also some preliminary results. These are focused on the image processing in a small-body mission and on the performance of a traditional and well-known optical navigation algorithm about the Moon.

I. Introduction

SOLAR System exploration is booming. The recent discovery of ocean worlds, the scientific and economic interest for small bodies, and the future human and robotic exploration of the main ones are pushing for numerous mission concepts aimed at the massive exploration and exploitation of the Solar System [1]. The increasing number of space missions, also promoted by deep-space small satellites, raises issues in how to operate them. First, traditional ground-based control centers and tracking facilities (e.g., the Deep Space Network – DSN) have limited number of communications slots. Second, the impetus for compelling science implies riskier operations, such as touch-and-go and landing on small bodies, or real-time and accurate state estimation, which is currently limited by the delayed DSN communications and cut-off time. Third, the current approach to spacecraft navigation implies high costs due to human-in-the-loop intervention that could become a critical driver with the increase of mission number.

Autonomous vision-based navigation (VBN) and robust Image Processing (IP) are crucial to overcome these limitations. This is because of the capability of such algorithms to provide near real-time information to navigation filters on-board. Among all the navigation sensors available on the market, cameras are usually preferred as they are light, compact and low power demanding when compared to other advanced navigation sensors, such as LIDARs. For these reasons the use of passive cameras, in combination with IP algorithms, provides compelling navigation performances with light and cost-effective hardware.

IP and VBN require images for design, testing, and validation. In exploration missions, it is complex if not impossible to gather images to perform such studies during the design phase of the algorithms. This is due to several reasons. First, even though the number of interplanetary missions is increasing, few missions are flown every year. This implies limited publicly available databases to be exploited, which could lead to limited testing capabilities. Second, archived images are associated with a spacecraft state estimate from the orbit determination solution. This, however, is affected by navigation errors that make the true spacecraft state not available if not with uncertainty associated with it. Thus, the VBN state estimate cannot be compared against the true spacecraft state and image renderings could lead to rendering errors [2]. Furthermore, available datasets are associated with specific missions. This implies that images are directly constrained to the flown trajectory and designed spacecraft. Given these constraints, it is not possible to change illumination conditions, camera parameters, and the dynamical context.

As space missions data are affected by navigation errors, two approaches are usually used to address the testing of VBN algorithms: high-fidelity rendering simulations or hardware-in-the-loop (HIL) testing. High-fidelity rendering engines, such as ESA's PANGU [3] or Airbus Defence & Space's SurRender [4], enable reproducing the data received at the processors by simulating not only the light propagation from the source to the sensor, but also the camera electronics

*PhD Candidate, Department of Aerospace Science and Technology, mattia.pugliatti@polimi.it

†Postdoctoral Research Fellow, Department of Aerospace Science and Technology, paolo.panicucci@polimi.it

‡Postdoctoral Research Fellow, Department of Aerospace Science and Technology, vittorio.franzese@polimi.it

§Full Professor, Department of Aerospace Science and Technology, topputo.francesco@polimi.it

[5]. On the contrary, HIL test benches rely on the manufactured sensor or a flight model in a laboratory environment to characterize the performances without any sensor simulation.

The former solution is generally more convenient as a broad series of sensors can be implemented without any additional expense in terms of equipment. Moreover, it enables the true spacecraft trajectory to be simulated without any constraints on the camera position, as is the case with robotic arms. A major drawback of this approach is the validation of the rendering simulator in reproducing the real world. Validation procedures could last years, if not decades, which lead to costs growth. Moreover, additional issues of this approach are the software licenses. Some of them, as PANGU and SurRender, are not open-source, while others like Blender* or POVray† are not specifically designed for the rendering of celestial objects nor for engineer applications. The latter solution requires reproducing the mission dynamical environment and a trustworthy representation of the observed scene to generate images affected by error. The HIL test bench is generally more expensive - at least in terms of equipment - as the representative dynamics or the observed scene requires expensive hardware as robotic arms or terrains analogs. Moreover, the use of real sensors and simulation hardware implies efforts in calibration which are not needed for high-fidelity rendering engines since the sensor is simulated numerically. Despite these limitations, HIL test benches are generally preferred as the hardware used within the simulation loop is more representative of the mission scenario.

Historically VBN test benches have been designed on optical tables to characterize star trackers performances [6–11] or to simulate close proximity or landing operations in highly-non-linear dynamics exploiting robotic arms [12–15]. Only recently, work has been done to design and build test benches that aim to validate VBN algorithms and their integration with cameras [16]. This work presents TinyV3RSE, the vision-based navigation test bench designed and built at the Deep-space Astrodynamics Research & Technology ‡ (DART) group. TinyV3RSE stands for Tiny Versatile 3D Reality Simulation Environment and it is used to support the design, validation, and testing of IP and VBN algorithms. This facility has been designed as a collaborative effort at the DART group and will play a pivotal role in the current and future projects in which the group is involved.

The rest of the paper is organized as follows. Section II provides a description of the facility design. Then, Section III illustrates the TinyV3RSE calibration procedure. Section IV reports some preliminary results that have been obtained through imaging in the facility. Finally, Section V summarizes the conclusions of this work.

II. TinyV3RSE design

A. Design drivers

TinyV3RSE is composed of three main elements: a screen, a collimator, and a camera, positioned as in Figure 4. When the light emitted by the screen passes through the collimating lens, it respects the thin lens equation. Thus:

$$\frac{1}{f_{\text{coll}}} = \frac{1}{d_r} + \frac{1}{d_i} \quad (1)$$

where f_{coll} is the collimating lens focal length, d_i is the distance between the collimating lens and the image, and d_r is the distance between the collimating lens and the object, i.e., the screen. Recall that f_{coll} is positive for converging lenses and negative for diverging ones. Moreover, note that d_r is positive when it is placed on the left side of the lens and negative otherwise. Finally d_i is positive when the image is generated on the right side of the lens, i.e. a real image is formed, and negative otherwise, i.e., a virtual image is formed. Figure 1 shows the geometrical configuration under study, which in this case generates a virtual image.

Equation 1 can be rewritten to explicitly compute the image distance as:

$$d_i = \left(\frac{d_r}{d_r - f_{\text{coll}}} \right) f_{\text{coll}} \quad (2)$$

This shows that the object to be observed, i.e., the screen, must be placed at the focal length distance. The perfect design choice would be that the screen image fully fits the camera field of view (FOV). To simplify the design procedure, the vertical FOV is considered since it is smaller than the horizontal one. Under the assumption of perfect components' alignment, the problem can be studied as outlined in Figure 2. Thanks to basic geometrical relationships, it is easy to show that:

*<https://www.blender.org/>, last accessed 20th November 2021.

†<http://www.povray.org/>, last accessed 20th November 2021.

‡<https://dart.polimi.it/>, last time accessed: November 5th 2021

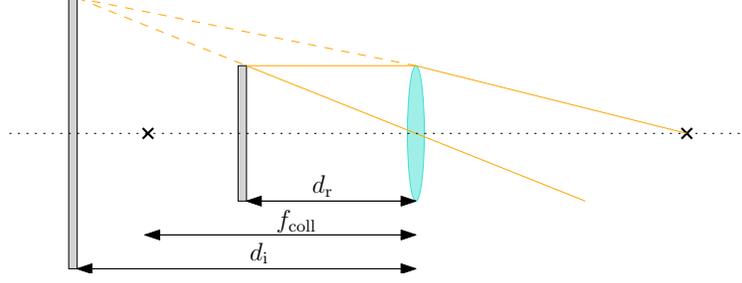


Fig. 1 Geometrical configuration for the lens equation of the collimator in the case considered in TinyV3RSE .

$$\tan\left(\frac{\theta}{2}\right) = \frac{h_s}{2f_{\text{coll}}} \quad (3)$$

where θ is the camera FOV, h_s is the vertical screen size, and $\theta_1 = \theta_2 = \frac{\theta}{2} = \frac{FOV}{2}$. Equation 3 links the three components of TinyV3RSE , showing that their design choice is not arbitrary.

Note that, because of the collimation, the distance between the camera and the collimating lens d_{cam} is not a design parameter that depends on the collimator focal length. This parameter is important to determine the diameter of the collimating lens [6]. To avoid that the camera observes outside of the collimating lens, the following relation has to be satisfied:

$$R_{\text{coll}} \leq R_{\text{cam}} + d_{\text{cam}} \tan\left(\frac{\theta}{2}\right) \quad (4)$$

where R_{coll} is the collimating lens radius and R_{cam} is the camera lens objective radius. Moreover, to ensure to work in paraxial area of the collimating lens, i.e., where the thin lens equation hypothesis holds, d_{cam} must be chosen as small as possible.

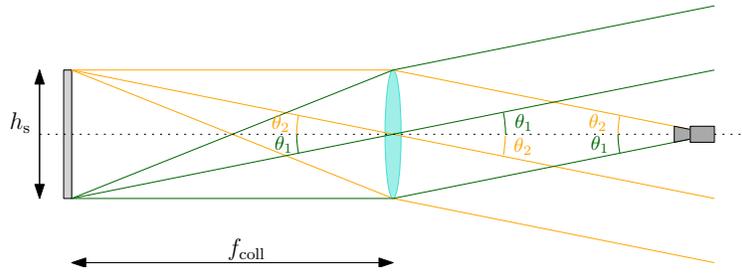


Fig. 2 Optical configuration of the components of the TinyV3RSE facility.

B. Components description

TinyV3RSE is composed of three main modules mounted on an optical table:

- 1) **The camera**, rigidly mounted on its mechanical support which enables vertical translation, pitch, and yaw mechanical adjustments;
- 2) **The high-resolution screen**, whose orientation is set to ensure that the screen and the optical plane of the camera are parallel;
- 3) **The collimator**, which ensures that the light coming from the screen and entering the camera is simulated as coming from infinity (or from a very high distance). The collimator is mounted on an optical support that can rotate, change in elevation, and can be finely adjusted laterally and transversely.

These three modules are visible from the CAD model in Figure 3 and from the top view of the TinyV3RSE facility in Figure 4, while their relationships and functional connections are illustrated in Figure 6. Each module is now described in detail.

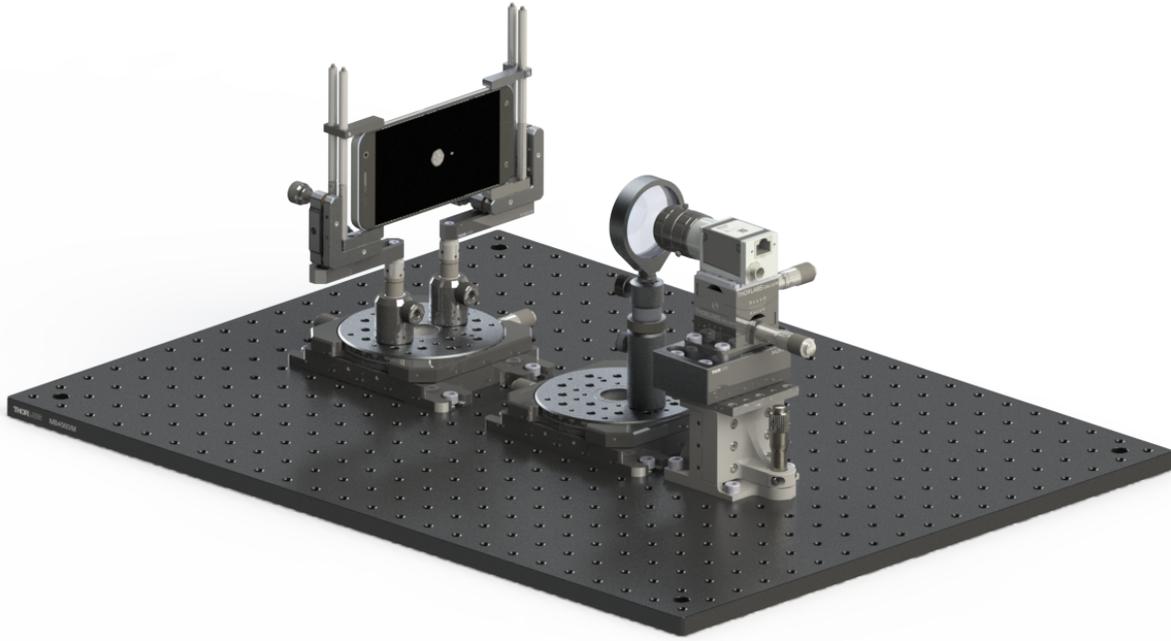


Fig. 3 CAD model of TinyV3RSE

The camera currently in use is a Basler acA1300-22gm (CS-Mount)[§] with a 12mm C series fixed focal length lens[¶]. The key data-sheet characteristics of the camera assembly are a focal length of 12 mm, a resolution of 1280 pixels \times 960 pixels, a pixel size of $3.75 \mu\text{m} \times 3.75 \mu\text{m}$, and a sensor size of 4.9 mm \times 3.6 mm. The camera FOV is $22.6^\circ \times 17^\circ$. The camera is mounted on a dedicated assembly which is composed of three parts. The first one is a vertical translation stage ensuring vertical control of the assembly. The second one is a goniometer enabling pitch and roll. The third and last part is a custom mounting adapter to interface between the camera and the optical assembly. In Figure 5 it is possible to see a close-up view of the camera assembly pointed towards the collimator. As explained in Section II.A, the distance between the collimator and the camera is kept as small as possible.

The screen is represented by a Galaxy S7 smartphone^{||} with a resolution of 2560 pixels \times 1440 pixels, a pixel size of $44.1 \mu\text{m} \times 44.1 \mu\text{m}$ and a screen size of 112.9 mm \times 63.5 mm. As for the camera and collimator, the screen is mounted on a dedicated assembly composed of two parts. The first one is a translational stage enabling movements of the screen on a plane parallel to the optical one. The second one is a screen holding mechanism that enables re-orientation by changing 4 pins disposed close to the screen's corners. The choice to use a commercial smartphone as a screen presents several advantages. First, in terms of the sizing of the facility, having a compact high-resolution screen makes it possible to position it within a limited distance from the camera-collimator assemblies, thus ultimately ensuring a compact facility. This is an advantage both in terms of laboratory space but also in terms of portability and eventual external testing since the optical test bench could easily be moved as carry-on luggage to a different location. Having a smartphone as a screen is also simple to set up the interfaces with the server. The smartphone is therefore a commercial, hence low-cost solution, which also possesses interesting properties in terms of image contrast. The screen used is an OLED one, which does not suffer from screen bleeding phenomena typical of Liquid Crystal Displays (LCDs) and exhibits a high contrast between inactive and active pixels. This is of particular interest in the rendering of the pitch-black background of a celestial scene, before considering camera noise. The smartphone as a screen solution exhibits also a drawback, which is given by the screen resolution. Some facilities designed in the past seems to abide by an empirical sampling law for which each pixel of the sensor is to be stimulated at least by 4 pixels of the screen (or 1:2 if considered linear) [17, 18]. This is to ensure a continuous representation of the environment to the sensor and to

[§] Camera data-sheet, last time accessed: November 5th 2021

[¶] Lens data-sheet, last time accessed: November 5th 2021

^{||} Screen data-sheet, last time accessed: November 5th 2021

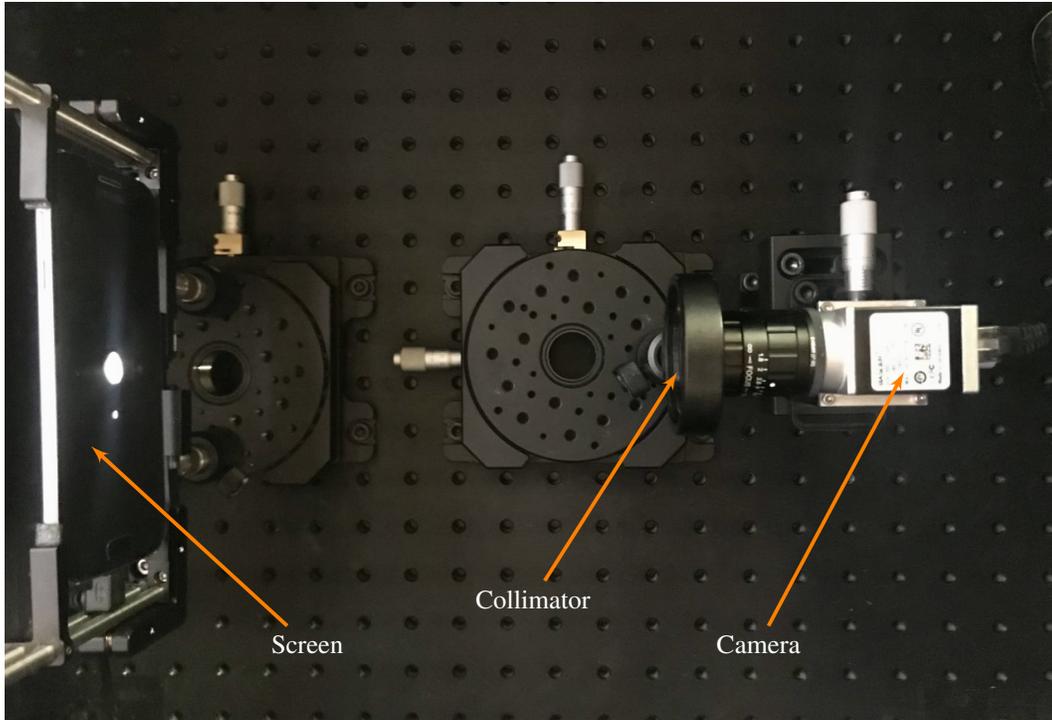


Fig. 4 Top view of the TinyV3RSE facility.

satisfy Nyquist sampling theorem. However, at the time of the facility design, a higher resolution screen than the one considered has not been identified from existing commercial smartphones. The current setup has roughly a 1:1.43 ratio between sensor and screen pixels. It is also observed that this phenomenon did not seem to have played an important disturbance on the performance of the algorithms tested so far.

Finally, the collimator used is a 2" diameter N-BK7 plano-convex lens (AR Coating: 350 - 700 nm)** with a focal length of 200 mm. The collimator is mounted on a dedicated assembly which is composed of a roto-translational stage and a post holder which is used to gain vertical alignment between the collimator and the camera.

By considering the screen and camera characteristics and using Equation 3, the camera should be placed at 211.7 mm in order to perfectly fit the screen vertical dimension with the vertical length of the camera's FOV. Because of that, the collimator has been chosen with a trade-off study among the plano-convex lenses available as off-the-shelf components. The selected one has been chosen to maximize the observed portion of the screen while avoiding vignetting. Note that, when an image is displayed on the screen, it has the size of 112.9 mm × 63.5 mm. By taking out the calculation with the camera FOV and a collimating distance of 200 mm under the hypothesis of perfectly aligned optical components, a coarse estimation give that only 80 mm × 60 mm of the screen is covered by the facility camera FOV. Thus the image taken by the facility camera is just a portion of the image displayed on the screen. This is important to be considered when operating and calibrating TinyV3RSE .

The functional architecture of the facility is illustrated in Figure 6. The screen and server are directly connected to a power outlet while the camera is exchanging data and power via a Power over Ethernet cable (PoE). Virtual scenes are rendered on the server with Blender or PovRay (depending on the user preference) and then sent to the screen via a Wi-Fi or USB connection. The server is also responsible for activating the camera, receiving and eventually processing the images obtained by the same. The screen, collimator, and camera are all enclosed in a box that is closed during the collection of the images. This is done to ensure that proper illumination conditions are met and that no light artifacts are generated on the screen due to the external conditions (such as reflections, external lighting, shadows of personnel working next to the facility).

In Figure 7 the functional workflow used to generate images with the facility is illustrated. The starting point is the simulated world, in which the physical and geometrical properties of the celestial bodies of interest are simulated in a

**Collimator data-sheet, last time accessed: November 5th 2021

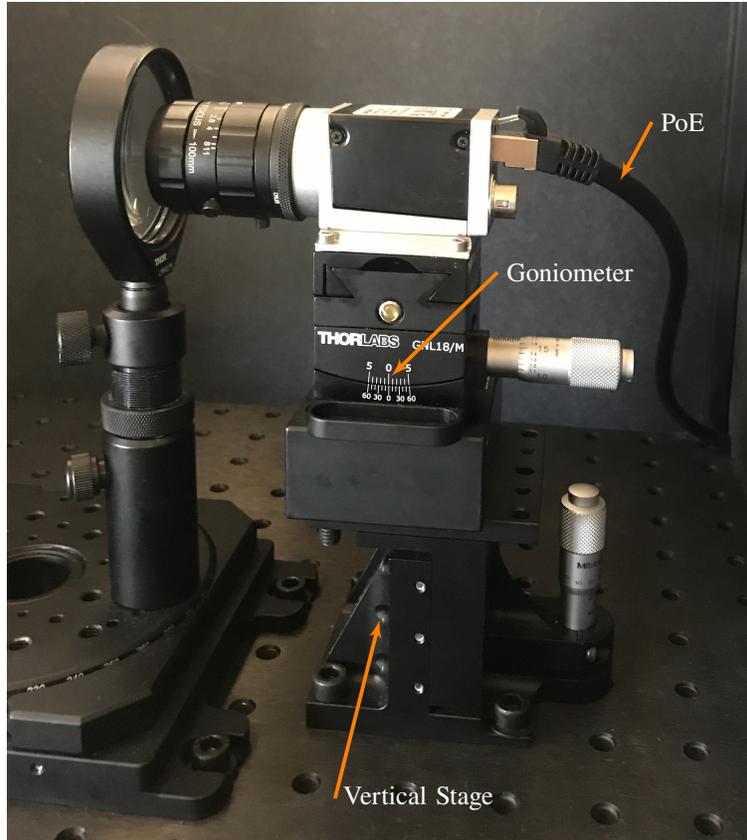


Fig. 5 Close-up view of the camera and collimator assemblies.

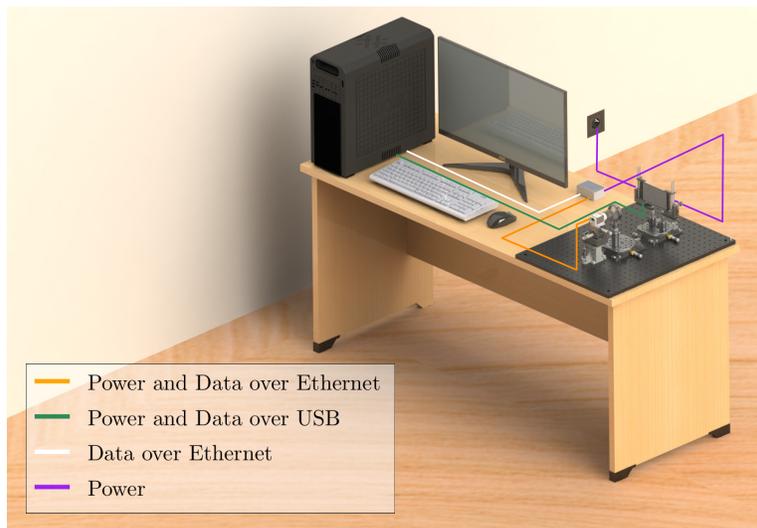


Fig. 6 Functional architecture of the TinyV3RSE facility.

virtual environment. Choosing a rendering software to do so is convenient since it enables sampling of such a virtual environment assuming a certain camera model positioned from a specific point of view. A rendering of a scene can thus be seen as a sampling of this simulated synthetic environment through the physical model of the camera. As it is possible to see from Figure 7, this is done twice for any given camera position: The first time to generate an image

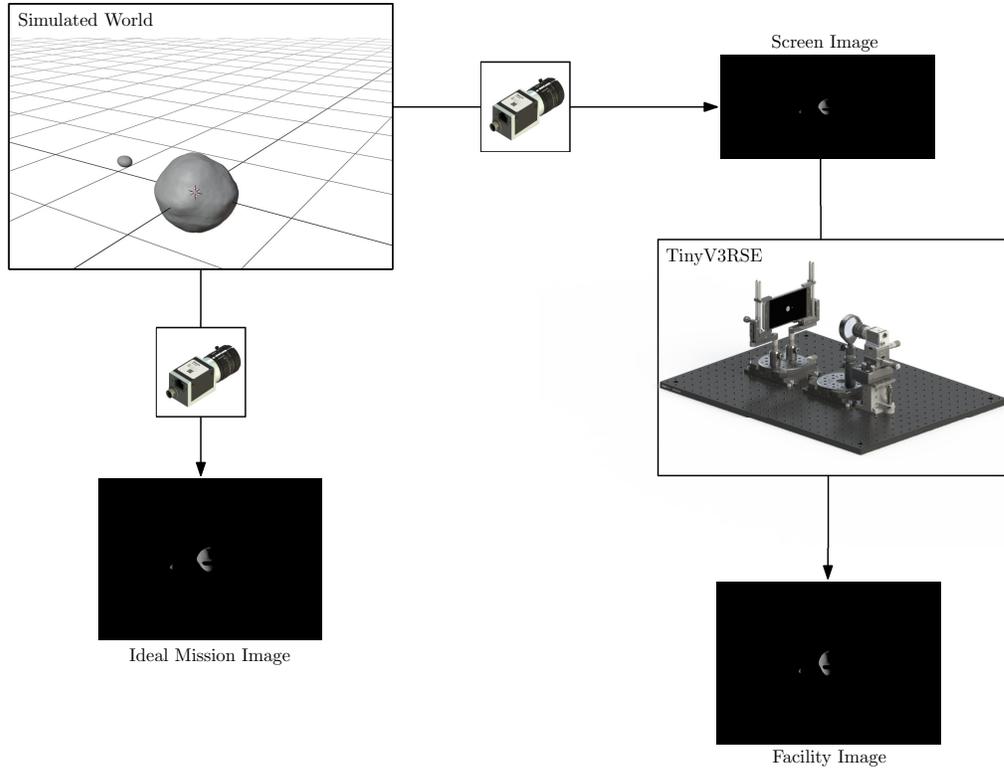


Fig. 7 Functional work-flow in TinyV3RSE

representative of what the mission camera would be seeing and the second time to get the scene to be projected on the screen at the proper resolution. These are respectively the "Ideal Mission Image" and "Screen Image" illustrated in Figure 7. Once the screen image is projected, it will then stimulate the collimator and camera, which will capture this scene with the real sensor properly positioned, given that a successful calibration ensures the correct alignment of all the components of TinyV3RSE. A final step is required to transform the image captured with the sensor in the facility to an equivalent version of the one captured in the virtual environment. This step is fundamental since, apart from calibration errors, these two images should be geometrically equivalent, yet photometrically different. The synthetic one has been generated with an ideal camera model, with no noise and perfect environmental conditions while the image from the facility encompasses noise and all phenomena typical of a sensor reading. The difference between these two images also represents the same domain gap between real and synthetic images which TinyV3RSE aims to reproduce for validation. A previous version of the facility presented in this work is also illustrated in [19]. In such a version, all the design choices are illustrated in detail, the interested reader is directed towards it.

III. Geometrical Calibration

Before using TinyV3RSE, a geometrical calibration procedure is necessary. This section describes such procedure. First, the calibration is required to find the intrinsic camera matrix of the equivalent pinhole camera model for the camera mounted in the facility [20]. Second, the calibration is necessary to take into account the lenses distortion. When the light emitted by the display passes through the collimator and the optical head lenses, it is distorted by the lenses. Thus, the image projected into the screen is warped by the lenses apparatus, therefore changing the observed image. The distortion is a major factor when a series of lenses is used. A previous work Tang et al. [21] introduces various mathematical representations to model the optical distortion introduced by lenses. In this work, the radial and tangential distortion model [22] is used for its physical interpretation. Other representations can be used to capture distortion losing the physical interpretation, such as in [16] and Samaan et al. [7]. Lastly, the geometrical calibration is necessary to estimate the misalignment among the components in the facility. Large angular errors would cause wrong functioning of IP and VBN algorithms to be tested, which could invalidate their performances and comparability. These

problems are solved in a sequential procedure. First, the camera mounted on the facility is calibrated with the algorithm proposed in Zhang [23] to find the equivalent pinhole camera model. Then, the alignment of the screen with respect to the camera assembly is estimated by displaying on the screen a series of checkerboards with different orientations, as shown in Figure 8. The procedure followed is similar to the one presented in Samaan et al. [7].

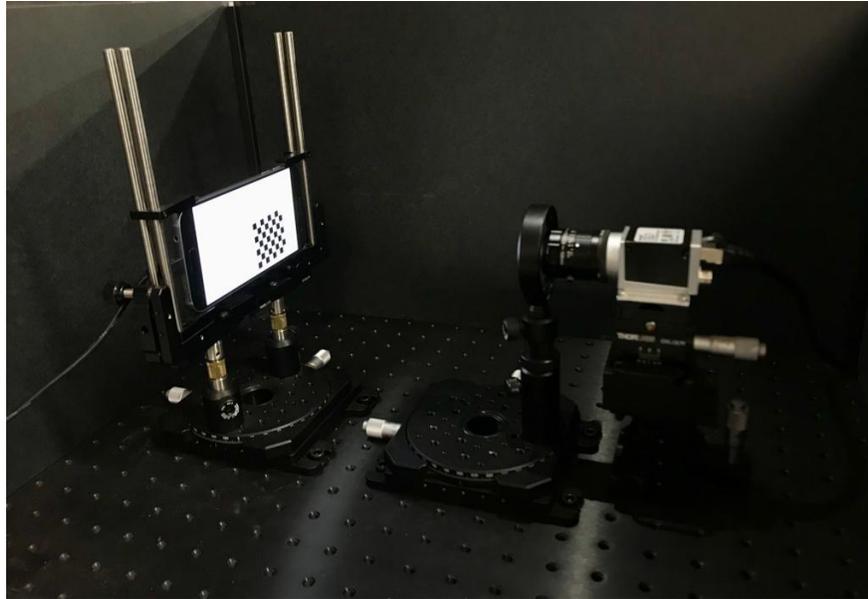


Fig. 8 A picture of the TinyV3RSE test bench during calibration while using a checkerboard pattern.

At the end of the procedure, the camera intrinsic matrix, the radial, and tangential distortion coefficients, and the facility misalignment are all estimated. This helps in converting the IP output to a consistent measurement as it would have been seen from the real camera without alignment errors from the correct view. The error obtained with this procedure is shown in Figure 9 in terms of root mean squared error (RMSE).

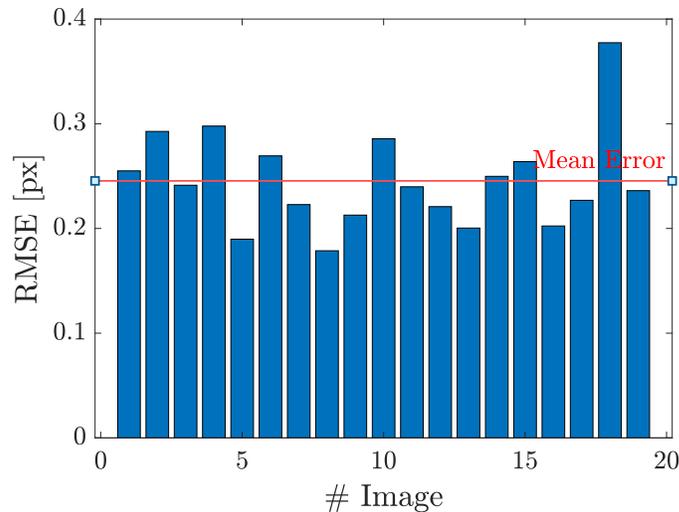


Fig. 9 Pixel reprojection error after the calibration procedure.

IV. Preliminary results

In this Section, preliminary results that have been obtained by the DART group using the TinyV3RSE facility are illustrated. These results demonstrate the type of analysis that can be performed with the use of such facility within the framework of validation and verification of IP and VBN algorithms.

A. Small bodies case

In the work presented in [24], a Convolutional Neural Network (CNN) is designed for object classification. The purpose is to develop a generalized classification algorithm that can be used to classify the shape of an unknown small body into one of eight different archetypal shapes. The performances of the CNN are compared with those of other 3 explicit features-based algorithms. The work shows how the CNN is capable of greatly outperform all other methods considered, as well as to generalize both in terms of illumination conditions and small body shapes never seen during training. All techniques used have been applied to binary images.

In this work, we present a brief continuation of the analysis illustrated in [24] with a performance assessment of the CNN on images of the test set generated using TinyV3RSE. The CNN works in inference on those images and no additional training is performed on the network illustrated in [24]. This analysis serves to illustrate the robustness region of the CNN method and to demonstrate an exemplar use of the facility.

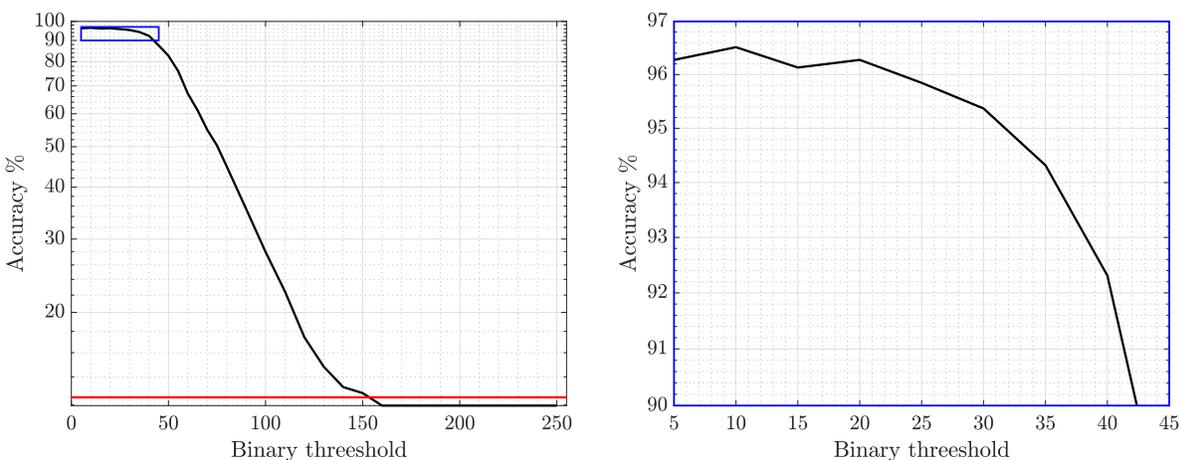


Fig. 10 Performance of the CNN used for classification as function of the binary threshold used on the images acquired in the TinyV3RSE facility. The red line represents the accuracy of a random guess at the classification task.

The accuracy of the CNN reported on the test set in [24] is 98.52%. In Figure 10, it is possible to see the accuracy of the same network on the same images obtained in TinyV3RSE across several values of the binary threshold used. It is possible to observe the existence of an interval for the binary threshold between 5 and 42 in which the CNN retains high accuracy ($> 90\%$) but suffers a small drop in peak performances (from 98.52% with synthetic images to 96.45% with TinyV3RSE images). The robustness of the CNN in this interval is possible since the network in [24] has been trained on binary images to properly handle real case scenarios. This choice seems to be supported by the results presented here, which introduce noise elements from the camera side. Indeed, the introduction of noise from a real camera and by the TinyV3RSE setup is not having a large impact on the CNN performance, given that these factors are properly handled by the binarization procedure used (Otsu in this case). When considering a binary threshold above 42, the performance steeply degrades. Starting from a value of 150, the CNN is less accurate than a random guess and thus completely unreliable for inference. To provide some context, in Figure 11 it is possible to observe the effect of the binary threshold on the silhouette of a small body at 4 different values on the same image taken in the facility. Thus, there seems to be a domain gap between synthetic and real images, but in this case, it seems to be handled well given the fact that binary images have been used to train the CNN.

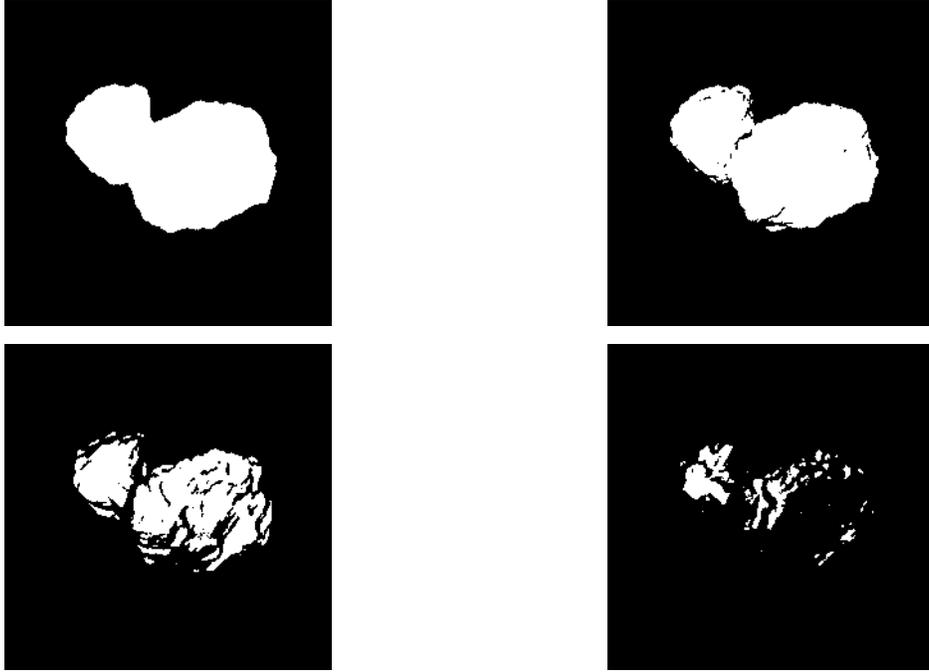


Fig. 11 Examples of binarized images acquired in the facility and processed with different values of binary thresholds: from top to bottom, left to right 5, 50, 100, 130.

B. Moon case

In the work presented in [19] a preliminary analysis of a VBN algorithm is illustrated for a spacecraft around the Moon adopting the full-disk navigation method [25]. The same face of the moon is rendered at varying distances from 5000 km to 100000 km, every 500 km. Some intermediate steps of this algorithm are illustrated in Figure 12. In the same figure it is possible to see the synthetic capture of the Moon as well as the one seen by the camera in the facility. It is also possible to see an example of horizon fit and ellipse fit with respect to the Moon. The error on the position estimate obtained with these images is illustrated in Figure 13, decomposed in the 3 axes of the camera frame. The x and y are the ones co-planar with the image plane, while the z -axis represents the bore-sight direction. The error is computed as relative percentage error, as the ratio between the absolute positioning error and the true range from the Moon, expressed as a percentage error. By inspection of Figure 13, it is possible to see that that error remains small in the x and y axes, never exceeding 0.1%, while it grows in the radial direction (z), as expected in optical navigation. This gradually increases from roughly 18000 km onward, reaching a maximum value of about 4% roughly at 100000 km.

This effect is caused by the natural behavior of optical navigation methods and uncertainty in calibration procedures. In general, it is natural for an optical navigation algorithm to be less precise in the depth axis with respect to the other ones because the optical information does not lie on the radial direction. To compensate for this, an altimeter could be used to provide more precise estimates on the z -axis. However, analysis of the VBN algorithm applied to the synthetic images has shown this effect should be smaller than the one recorded on real images. The second contribution is due to the calibration. The convergence of the calibration algorithm to the principal point of the camera can not be exact. Thus, some errors in the calibration procedure are affecting the intrinsic camera matrix and thus the solution provided by the VBN algorithm.

V. Conclusions and future works

In this work we have presented for the first time the design of the DART's vision-based navigation test bench, TinyV3RSE, a hardware-in-the-loop test bench used by the DART group for the development, validation, and testing of IP and VBN algorithms. The design and calibration procedures adopted within TinyV3RSE have been briefly illustrated as well its exploitation for IP and VBN algorithms applied to small bodies and in the lunar environment. The results presented here are preliminary, future works are planned to further extend the analysis performed with TinyV3RSE

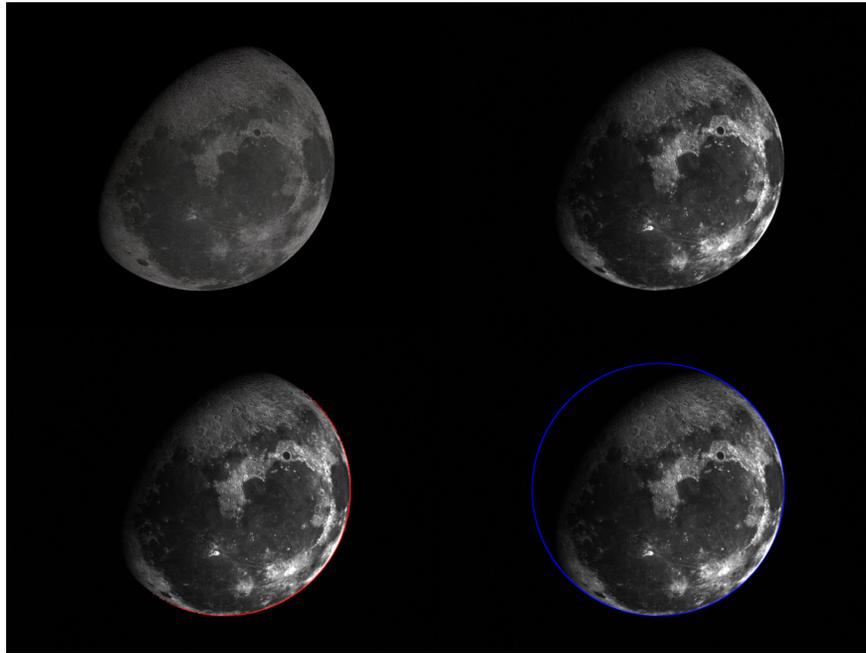


Fig. 12 From left to right, top to bottom: original image of the moon projected to screen, image captured by the camera in the TinyV3RSE facility, identification of the lit horizon (red), identification of the ellipse fit (blue).

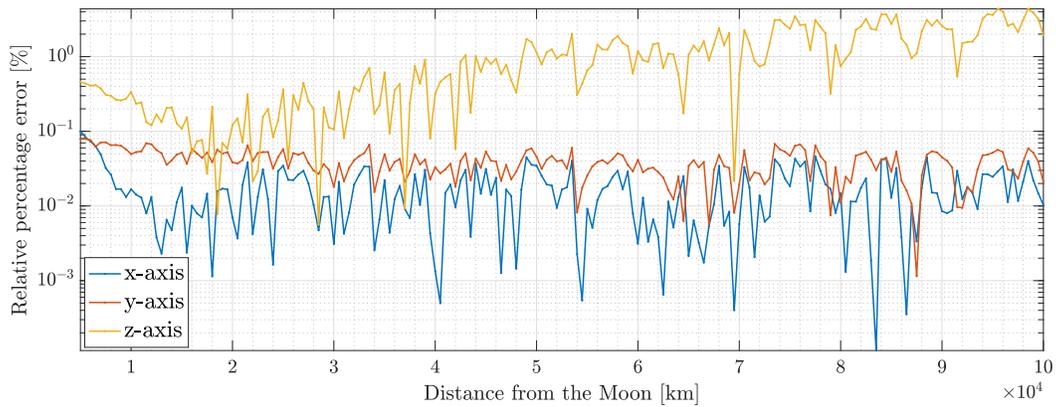


Fig. 13 Relative positioning error per axis of the VBN algorithm used in the moon case at varying distances.

as well as to further refine the design of the optomechanical components (especially relative to the screen-support mounting) and overall calibration procedure. The facility will be exploited to assess the VBN and IP performances of on-going space missions [25–27] and projects of the DART’s group.

Acknowledgments

The authors would like to acknowledge the funding within 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). M.P. and F.T. would like to acknowledge the funding received from the European Union’s Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie (Grant agreement No. 813644). The authors would also like to acknowledge Nicolò Vattai for the preliminary design of the test bench.

References

- [1] Quadrelli, M. B., Wood, L. J., Riedel, J. E., McHenry, M. C., Aung, M., Cangahuala, L. A., Volpe, R. A., Beauchamp, P. M., and Cutts, J. A., “Guidance, Navigation, and Control Technology Assessment for Future Planetary Science Missions,” *Journal of Guidance, Control, and Dynamics*, Vol. 38, No. 7, 2015, pp. 1165–1186. <https://doi.org/10.2514/1.g000525>.
- [2] Belgacem, I., Jonniaux, G., and Schmidt, F., “Image processing for precise geometry determination,” *Planetary and Space Science*, Vol. 193, 2020, p. 105081. <https://doi.org/10.1016/j.pss.2020.105081>.
- [3] Rowell, N., Parkes, S., Dunstan, M., and Dubois-Matra, O., “PANGU: Virtual spacecraft image generation,” *5th Int. Conf. on Astrodynamics Tools and Techniques, ICATT*, 2012.
- [4] Lebreton, J., Brochard, R., Baudry, M., Jonniaux, G., Salah, A. H., Kanani, K., Le Goff, M., Masson, A., Ollagnier, N., Panicucci, A., Paolo ad Proag, and Robin, C., “Image Simulation for Space Applications with the SurRender Software,” *Proceedings of the 11th International ESA Conference on Guidance, Navigation & Control Systems, 22 - 25 June 2021, Virtual*, 2021.
- [5] Jonniaux, G., and Gherardi, D., “Development, tests and results of onboard image processing for JUICE,” *Proceedings of the 11th International ESA Conference on Guidance, Navigation & Control Systems, 22 - 25 June 2021, Virtual*, 2021.
- [6] Rufino, G., and Moccia, A., “Laboratory test system for performance evaluation of advanced star sensors,” *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 2, 2002, pp. 200–208. <https://doi.org/10.2514/2.4888>.
- [7] Samaan, M. A., Steffes, S. R., and Theil, S., “Star tracker real-time hardware in the loop testing using optical star simulator,” *Spaceflight Mechanics*, Vol. 140, 2011.
- [8] Boone, B., Bruzzi, J., Dellinger, W., Kluga, B., and Strobehn, K., “Optical simulator and testbed for spacecraft star tracker development,” *Optical Modeling and Performance Predictions II*, Vol. 5867, International Society for Optics and Photonics, 2005, p. 586711. <https://doi.org/10.1117/12.619133>.
- [9] Nardino, V., Guzzi, D., Burrelli, M., Cecchi, M., Cecchi, T., Corti, F., Corti, M., Franci, E., Guidotti, G., Pippi, I., et al., “MINISTAR: a miniaturized device for the test of star trackers,” *International Conference on Space Optics—ICSO 2018*, Vol. 11180, International Society for Optics and Photonics, 2019, p. 111807V.
- [10] Rufino, G., Accardo, D., Grassi, M., Fasano, G., Renga, A., and Tancredi, U., “Real-time hardware-in-the-loop tests of star tracker algorithms,” *International Journal of Aerospace Engineering*, Vol. 2013, 2013, pp. 1–13. <https://doi.org/10.1155/2013/505720>.
- [11] Filipe, N., Jones-Wilson, L., Mohan, S., Lo, K., and Jones-Wilson, W., “Miniaturized star tracker stimulator for closed-loop testing of cubesats,” *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 12, 2017, pp. 3239–3246. <https://doi.org/10.2514/1.g002794>.
- [12] Volpe, R., Sabatini, M., Palmerini, G., and Mora, D., “Testing and Validation of an Image-Based, Pose and Shape Reconstruction Algorithm for Didymos Mission,” *Aerotecnica Missili & Spazio*, 2020, pp. 17–32. <https://doi.org/10.1007/s42496-020-00034-6>.
- [13] Petit, A., Marchand, E., and Kanani, K., “Vision-based space autonomous rendezvous: A case study,” *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2011, pp. 619–624.
- [14] Zwick, M., Huertas, I., Gerdes, L., and Ortega, G., “ORGL–ESA’S test facility for approach and contact operations in orbital and planetary environments,” 2018.

- [15] Brannan, J., Scott, N., and Carignan, C., “Robot Servicer Interaction with a Satellite During Capture,” *International Symposium on Artificial Intelligence, Robotics and Automation in Space (iSAIRAS)*, 2018.
- [16] Beierle, C., and D’Amico, S., “Variable-magnification optical stimulator for training and validation of spaceborne vision-based navigation,” *Journal of Spacecraft and Rockets*, Vol. 56, No. 4, 2019, pp. 1060–1072. <https://doi.org/10.2514/1.a34337>.
- [17] Pellacani, A., Graziano, M., Fittock, M., Gil, J., and Carnelli, I., “HERA vision based GNC and autonomy,” 2019. <https://doi.org/10.13009/EUCASS2019-39>.
- [18] Holt, G. N., D’Souza, C. N., and Saley, D. W., “Orion optical navigation progress toward exploration mission 1,” *2018 Space Flight Mechanics Meeting*, 2018, p. 1978.
- [19] Vattai, N., “Development and validation of a horizon-based optical navigation test facility,” Master’s thesis, Politecnico di Milano, 12 2019. Accessible from: <https://www.politesi.polimi.it/handle/10589/151672>.
- [20] Andrew, A. M., “Multiple view geometry in computer vision,” *Kybernetes*, 2001.
- [21] Tang, Z., von Gioi, R. G., Monasse, P., and Morel, J.-M., “A precision analysis of camera distortion models,” *IEEE Transactions on Image Processing*, Vol. 26, No. 6, 2017, pp. 2694–2704.
- [22] Heikkila, J., and Silvén, O., “A four-step camera calibration procedure with implicit image correction,” *Proceedings of IEEE computer society conference on computer vision and pattern recognition*, IEEE, 1997, pp. 1106–1112.
- [23] Zhang, Z., “A flexible new technique for camera calibration,” *IEEE Transactions on pattern analysis and machine intelligence*, Vol. 22, No. 11, 2000, pp. 1330–1334. <https://doi.org/10.1109/34.888718>.
- [24] Pugliatti, M., and Topputo, F., “Small-Body Shape Recognition with Convolutional Neural Network and Comparison with Explicit Features Based Methods,” *AAS/AIAA Astrodynamics Specialist Conference, Lake Tahoe*, 2020.
- [25] Franzese, V., Di Lizia, P., and Topputo, F., “Autonomous Optical Navigation for the Lunar Meteoroid Impacts Observer,” *Journal of Guidance, Control, and Dynamics*, Vol. 42, No. 7, 2019, pp. 1579–1586. <https://doi.org/10.2514/1.g003999>.
- [26] Ferrari, F., Franzese, V., Pugliatti, M., Giordano, C., and Topputo, F., “Preliminary mission profile of Hera’s Milani CubeSat,” *Advances in Space Research*, Vol. 67, No. 6, 2021, pp. 2010–2029. <https://doi.org/10.1016/j.asr.2020.12.034>.
- [27] Topputo, F., Wang, Y., Giordano, C., Franzese, V., Goldberg, H., Perez-Lissi, F., and Walker, R., “Envelop of reachable asteroids by M-ARGO CubeSat,” *Advances in Space Research*, 2021, pp. 4193–4221. <https://doi.org/10.1016/j.asr.2021.02.031>.