Politecnico di Milano
Department of Aerospace Science and Technology
Doctoral Programme in Aerospace Engineering

Spacecraft Relative Navigation with Electro-Optical Sensors around Uncooperative Targets

Doctoral Dissertation of:
Margherita Piccinin

Supervisor:
Prof. Michèle Lavagna

Tutor:
Prof. Giuseppe Gibertini

Coordinator:
Prof. Pierangelo Masarati

Year 2022 – Cycle XXXV

Abstract

Spacecraft relative navigation is a crucial and yet challenging capability for missions ranging from science and exploration to on-orbit-servicing and active debris removal. Traditional approaches vastly rely on the target's cooperation or on the ground acquisition of knowledge of the target's characteristics, then exploited on-board. Today, the achievement of an autonomous navigation is becoming a primary task, enabling new complex mission objectives that require low latencies and thus can not completely rely on ground commands, as is typically done. To accomplish this task, future missions will embark sensors suites composed by multiple Electro-Optical instruments. In fact, in the last decade, the technological development of space sensors as uncooled microbolometers, multispectral cameras and laser scanners, made them portable also on small platforms and apt to be employed for navigation purposes, as opposed to complex scientific instruments. Differentiating the input measurements to navigation not only would raise robustness and accuracy, which are keys for autonomy, but would also increase flexibility, allowing to widen the operational ranges.

This research work focuses on the exploitation of Electro-Optical sensors for enhancing autonomous relative navigation in proximity of an uncooperative target. Different types of sensors are examined, probing the advantages and limitations of their stand-alone employment and proposing methods to engage them in synergy for the navigation. Monocular cameras working in the visible band are considered at first, being the most common solution adopted for relative navigation. To explore benefits and boundaries of such approach,

particularly challenging applications are considered, as of use to understand when other sensors may come into place. As a result, multispectral navigation is proposed to overcome visibility and illumination-related issues proper of the visible band, identifying from analyses the Long Wave Infrared Band as interesting for a multispectral navigation. Secondly, the navigation performance achievable with currently available thermal infrared cameras is studied. The operative conditions in which thermal-based navigation is advantageous with respect to visible imaging are defined and a multi-modal navigation is proposed. Moreover, visible-infrared images fusion is implemented and compared with the two stand-alone techniques of visible and infrared images. Finally, classical and Deep Learning methods based on LiDAR scanners are developed for pose estimation, offering structured information of the target geometry. The possibility of an integration with a classical vision-based approach is also proposed. The research work successfully finds key elements and guidelines for an exploitation of the different kind of sensors in various scenarios. The proposed methods can increase accuracy, robustness or enable operations for new mission concepts.

The Thesis also poses its accent on the verification and testing of the proposed navigation techniques. In particular, the topic of generating representative synthetic data and of their validation with real or experimental data is argued for the three sensors. The lack of prior tools is mostly challenging for thermal cameras, therefore an innovative pipeline for generating synthetic thermal images is proposed and implemented. Experimental activities including calibration of sensors in GNC facilities are carried-out. Such activities allowed tests of the navigation algorithms, comparing the performance achieved with respect to synthetic data.

This Thesis offers an overview and initial framework for the concurrent exploitation of visible cameras, thermal cameras and LiDAR sensors, making a step forward to overcome major challenges of proximity relative navigation with uncooperative objects.

Ringraziamenti

Vorrei ringraziare in primo luogo la Prof.ssa Michèle Lavagna, grazie alla quale ho avuto la possibilità di intraprendere il percorso di dottorato e di lavorare su progetti interessanti e variegati. Grazie per aver sempre riposto fiducia nelle mie capacità e potenzialità, proponendomi degli obiettivi stimolanti, che hanno consentito una crescita professionale ed umana altrimenti impossibile.

Un grazie speciale anche a Roberto, Uli e Klaus per aver supervisionato il mio periodo al DLR, interessandosi al mio lavoro con costanza: avete arricchito l'esperienza al RMC Institute, rendendola preziosa per la mia vita professionale.

Ringrazio poi i miei colleghi ed amici del team ASTRA, con cui ho condiviso la quotidianità al DAER. *Compagno Vanni*, abbiamo iniziato questa avventura del dottorato assieme. Grazie per aver creato con i tuoi modi pacati un'armonia nel gruppo che ci ha fatto lavorare con il sorriso. Capa, ti ringrazio per avermi accolta in ASTRA. La tua gentilezza ed educazione mi hanno fatta sentire fin da subito meno sola in un ambiente nuovo. Ste, in questi anni ci siamo spesso ritrovati a lavorare fianco a fianco. Ho sempre trovato in te una persona affidabile, seria e disponibile. Grazie per il tuo ottimismo ed entusiasmo contagiosi. Andre, abbiamo saputo abbattere la barriera della diversità e delle apparenze. Grazie per essere un punto fermo, sei er più *macho* de tutti.

Un grazie agli amici di sempre, che mi hanno dimostrato il loro affetto e vicinanza in questi anni di dottorato, a Laura, Giulia C., Elisa, Giulia I., alla Simo, a Marta e ai *Magnifici 7*. Grazie per saper mantenere e rinnovare sempre la nostra amicizia. Grazie anche alle nuove amicizie di questi anni, agli *Scappati di casa* per le cene improvvisate e alle ragazze della *Tara* per le Cene Eleganti!

Grazie a tutte le persone che hanno reso speciale la mia permanenza a Monaco, grazie agli amici del DLR, a Marzia, Seba, Fede e ai *Vonderianos*.

Ringrazio infine mamma, papà e Pier, grazie per essere una casa sempre sicura per me e allo stesso tempo sostenere il mio volo, grazie per tutto quanto.

Table of Contents

List of Figures

List of Tables

List of Acronyms

ADR	Active Debris Removal.
AI	Artificial Intelligence.
BRDF	Bidirectional Reflectance Distribution Function.
CD	Chamfer Distance.
CT	Calibration Target.
DEM	Digital Elevation Model.
DL	Deep Learning.
DN	Digital Number.
EMD	Earth Mover's Distance.
EO	Electro-Optical.
FF	Formation Flying.
FoV	Field of View.
FPA	Focal Plane Array.
FQI	Feature Quality Index.
GNC	Guidance Navigation and Control.
GPS	Global Positioning System.
GUI	Graphical User Interface.
HIL	Hardware-In-the-Loop.

ICP	Iterative Closest Point.
IP	Image Processing.
LiDAR	Light Detection And Ranging.
LO	LiDAR Odometry.
LoS	Line of Sight.
LWIR	Long Wave InfraRed.
MLI	Multi Layer Insulation.
MLP	Multi Layer Perceptron.
NETD	Noise Equivalent Temperature Difference.
NIR	Near InfraRed.
OOS	On-Orbit Servicing.
PCN	Point Completion Network.
Ph.D.	Philosophiae Doctor.
PIL	Processor-In-the-Loop.
PnP	Perspective-n-Point.
PSF	Point Spread Function.
RF	Radio Frequency.
RMSE	Root Mean Square Error.
ROI	Region Of Interest.
SLAM	Simultaneous Localization And Mapping.
SNR	Signal to Noise Ratio.
TIR	Thermal InfraRed.
TM	Template Matching.
ToF	Time-of-Flight.
TRL	Technology Readiness Level.
VIS	Visible.
VO	Visual Odometry.

CHAPTER 1

Introduction

For several space missions, the task of relative navigation is of vital importance for the fulfillment of the proposed objectives. Relative navigation in space has already been performed in past decades, but, due to its inherent complexity, it still requires the massive employment of ground intervention during the mission.

Spacecraft relative navigation consists in processing measurements from onboard sensors to determine the relative position and attitude of two different space objects, namely an active satellite and a space target. The targets of interest are indeed various, as they inhere to both natural celestial bodies and artificial objects, i.e. other spacecraft. Thus, the relative navigation problem is of interest for several applications. For what concerns missions to moons, asteroids, comets and planetary terrains, it is fundamental for the operations of approach, descend and landing on the celestial body. Whereas, with regard to On-Orbit Servicing (OOS), Formation Flying (FF) and Active Debris Removal (ADR) missions, relative navigation allows complex operations such as approach, docking and capture. As a general consideration, for all the

above mentioned missions, the relative navigation is a mandatory step, being part of the spacecraft Guidance Navigation and Control (GNC) chain.

A key element to achieve a successful navigation is the choice and the exploitation of navigation sensors. In case of cooperative targets, the relative navigation architecture can rely on Radio Frequency (RF) and Global Positioning System (GPS) sensors. Instead, when the target is uncooperative the available options restrict to Electro-Optical (EO) sensors. EO sensors include passive sensors, as monocular cameras and stereocameras, and active sensors, as Light Detection and Ranging (LiDAR) sensors and Time-of-Flight (ToF) cameras [1], [2].

This Thesis deals with the relative navigation problem, in particular focusing on the exploitation of EO sensors for the navigation.

1.1 Context & motivation

Past and current missions have tackled only some of the many challenges related to the navigation problem. The achieved results were possible thanks to the heritage of missions of the past decades. This process towards autonomy is on-going, as major improvements still need to be performed in this direction, in order to enable new, more ambitious and complex mission objectives.

An important mission demonstrating autonomous navigation is PRISMA, that dealt with relative GNC between a chaser and a target spacecraft. The mission, launched in 2010, included the demonstration of several GNC techniques to enable future FF missions. The experiments consisted in performing different maneuvers and relative approaches around the target spacecraft, employing GPS-based, RF or vision-based techniques. Concerning the vision-based approaches, at far-range a camera was used for LoS (Line of Sight) navigation from about 100 km down to 100 m. While at short-range, another camera was exploited to carry out pose determination with the known cooperative target. A set of active markers (LEDs) was employed on the target, positioned according to a coplanar, non-collinear pattern. An output of the mission was the collection of space images and of the related navigation data, that were then used on-ground to develop algorithms for the case of an uncooperative known spacecraft target, relying on the knowledge of its model [3].

The Automated Transfer Vehicle (ATV, 2008 - 2014) missions employed a videometer, a system that operates similar to a Flash LiDAR as navigation sensor. At far distances, the sensor was employed to derive range, range-rate and LoS measurements. Whereas at close distances, the relative position and relative attitude were computed thanks to retro-reflectors on the ISS [4]. Of particular interest for the development of improved navigation algorithms was

the ATV-5 experiment in 2014, which allowed the collection of synchronized data from visible, thermal cameras and scanning LiDARs [5].

Regarding the close-range navigation with an uncooperative natural body, one of the most recent missions tackling this problem is Hayabusa 2, that in 2018 reached the asteroid 162173 Ryugu. A retroreflective target marker was deployed on the asteroid's surface and used as an artificial landmark on the natural terrain. A flash lamp installed near the on-board camera illuminated the marker, that reflected most of the light unlike other natural bright objects on the surface, allowing the retrieval of differential images for landing [6]. On the other hand, for the first part of the descent operations, the autonomy of the GNC system was still low. In fact, the vision-based navigation relied on ground-loops, where the control operators manually had to find by hand the pose of the asteroid by means of a Graphical User Interface (GUI) program, with an iterative process. Operations entailed then the ground computation of the spacecraft maneuver, propagating the estimated state for 65 min. This was the time needed to uplink the maneuver's command, downlink new images and compute the successive command. Whereas the altitude was estimated on-board relying on LiDAR measurements [7].

For the first time, during the OSIRIS-REx mission in 2020, an autonomous optical navigation system relying on natural terrain features was used to guide a spacecraft to a planetary surface. The spacecraft's position and velocity with respect to the asteroid was estimated autonomously on-board, thanks to prior on-ground generation of digital terrain models, equipped with relative albedo data [8]. The on-board system worked by rendering the landmark features from the digital terrain models and then performing a correlation in real-time with navigation images.

In 2022, the DART mission reached and impacted the Didymos system, relying on a vision-based navigation system that was autonomous for the last hours up to the impact. The mission was planned to perform blobbing, centroiding, targeting to derive LoS measurements of Didymos. The spacecraft inertial pointing was exploited to predict the stars and Didymos' centers in the images [9]. The thorough navigation results are still not publicly available. From first insights, the most relevant challenges in the application of the navigation technique were the number and frequency of images, the need to add multiple images and the distortions caused by the camera rolling shutter [10].

The high Technology Readiness Level (TRL) relative navigation techniques based on EO sensors are reported in Table 1.1, where $\rho, \dot{\rho}$ are range and range-rate, $\mathbf{r}$ and $\mathbf{v}$ are position and velocity, h is the altitude and $\mathbf{q}$ stands for the relative attitude. As clearly visible, relative navigation in proximity of a known or unknown uncooperative target is clearly still a great technological challenge.

In fact, the solutions adopted up to now largely rely on the active cooperation of the target, on the employment of markers, or on ground-loops.

Table 1.1: State-of-the-art relative navigation techniques with EO sensors.

Navigation sensor	Target	Range	Measurement	Notes on autonomy
Camera	known cooperative s/c	far/mid	LoS	on-board
Camera	known cooperative s/c	close	$\mathbf{r}$, $\mathbf{q}$	on-board, relying on markers
Camera	known asteroid	far/mid	$\mathbf{r}$, $\mathbf{v}$	on-board, ground maplets
Camera	asteroid	mid/close	LoS	on-board, relying on absolute nav output
Camera	asteroid	close	LoS, ρ	on-board, marker deployment
LiDAR	known cooperative s/c	far/mid	$\rho, \dot{\rho}$, LoS	on-board
LiDAR	known cooperative s/c	close	$\mathbf{r}$, $\mathbf{q}$	on-board, retroreflectors
LiDAR	asteroid	mid/close	h	on-board

New sensors technologies, as scanning LiDARs and uncooled microbolometer thermal cameras, have enhanced their TRL in the recent years, opening the doors to new possibilities for navigation. The LIRIS ATV 5 experiment in 2014 has been a technological demonstration for the employment of new sensors, to support the advancement of GNC non-cooperative rendezvous technologies. LIRIS experiment consisted of two thermal infrared cameras, a visible camera and a LiDAR system. The new sensors' data were recorded on-board without any on-board processing and then were used on ground to develop and test relative navigation algorithms [5].

Future missions are planned with more ambitious objectives that require the employment of such recently proposed and tested sensors technologies. ESA's

Hera mission payloads will include both an AFC (Asteroid Framing Camera) for navigation and imaging and the TIRI (Thermal InfraRed Imager) contributed by JAXA, which is proposed to be used for navigation and observation of the asteroid night side [11]. For future active debris removal missions [12], LiDARs are suggested as key technology enabling the rendezvous with an uncooperative target, in the mid/close-range operations. While final rendezvous and capture should incorporate a combination of sensors, namely cameras working in the visible and infrared spectrum and LiDAR.

Hence, an appealing opportunity for spacecraft navigation consists in the clever and concurrent exploitation of multiple EO sensors, with the aim of compensating the limitations of the respective technologies.

The choice of the relative navigation sensors and of the related algorithms is not trivial, as it is tightly connected to the mission analysis and operations design, since the first mission design phases. In fact, the sensors measurements depend on the spacecraft trajectory relative to the target, on the absolute trajectory, i.e. on the environmental conditions (such as the illumination) and on the mission operations, which determine the different sensing modes to be employed in all mission phases and contacts with ground. Thus, on one side the sensors characteristics constitute an input to the trajectory design, and on the other side the input to sensors are dependent on the nominal trajectory and environmental conditions.

The most commonly adopted sensors are visual cameras, that are passive and provide a wide operative range, generally required by the mission's operations. Indeed, spacecraft relative navigation techniques largely rely on optical measurements taken from monocular cameras working in the visible spectrum. On the other hand, space imaging strongly suffers from high contrast, low Signal to Noise Ratio (SNR) and smearing. Moreover, the target appearance is subject to large variations due to the changes of the illumination conditions in orbit.

In addition to visible cameras, missions to celestial bodies typically carry on board scientific instruments working in different spectral bands, and which could be exploited also for navigation purposes. The Rosetta mission carried VIRTIS (Visible InfraRed and Thermal Imaging Spectrometer), an hyperspectral imager performing 0.25 µm to 5.1 µm imaging spectroscopy on two separate channels [13]. The Hayabusa 2 mission carried on board the TIR experiment, that collected far to close range thermal images of the surface in the Long Wave InfraRed (LWIR) spectrum, in the range 8-12 μm [14]. In particular, it has been shown that during the descent some of the thermal images could have been exploited for enhancing the detection of markers placed on the asteroid surface, that were detectable as cold spheres [15]. Such recent outcomes have highlighted the possibility to exploit sensors working in the infrared band for navigation purposes, along side with a more classical sensor suite. Possible

useful applications include vision-based GNC algorithms for approaching a target body, operations in the night or in the dark area, but also moons and meteoroids detection, markers tracing, landing site selection and hazards detection.

The main idea behind multispectral imaging consists in collecting images in specific and distinguished targeted spectral bands and exploit the differences among images to obtain the algorithms performance improvement. Thermal sensors typically present a smaller array size compared to photon sensors and for this reason they are less prone to navigation. On the other hand, the thermal infrared band is less sensitive to illumination than the visible band [16]. Thus thermal cameras could be employed also in the dark side if the body surface temperature is inside the instrument detection range. Other advantages of thermal images compared to the visible ones are the absence of bright spots such as background bright stars, bad pixels, irradiation of solar wind particles or galactic rays. Moreover, there is no need to change the overexposure when the target is seen from a single pixel unit until a numerous pixel-sized body [15].

While visible and thermal cameras are both imaging instruments, LiDAR sensors are laser-based and thus offer a quite different alternative. In fact, they are employed with success for deriving range measurements, which are more difficult to obtain with vision sensors. Moreover, LiDAR sensors are affected neither by illumination conditions and by the presence of far-range celestial bodies in their field of view, nor by the temperature of the target. On the other hand, LiDAR are active sensors and they typically require a larger computational processing power, which is indeed very limited in space applications, due to the necessity of using rad-hard components. Some other disadvantages presented by LiDAR are the limited Field of View (FoV), motion distortion (i.e. point clouds measurements received at different times during continuous LiDAR motion) and failure of scan matching in degenerate scenes dominated by a planar surface. In ground applications, visual-LiDAR fusion strategies have shown an increased robustness in terms of strong light changes, which is a typical condition for space applications, and in terms of motion speed, which could be the case of a high tumbling target or slow processing frequency available on-board. Since images provide rich visual texture information, and LiDAR has sparse but precise distance measurement and 360-degree field of view, the combination of vision sensors and LiDAR sensors may complement each other in motion estimation tasks.

When dealing with the development of new navigation techniques, an important aspect consists in their verification and test. To be accepted for flight, GNC algorithms require running Hardware/Processor-In-the-Loop (HIL/PIL) tests campaigns in highly representative environments. PIL and HIL experiments

are mandatory to further increase the TRL of EO-based navigation algorithms. There are several reasons for which such experimental approach is fundamental for the testing and validation process. In first place, the navigation algorithms development widely relies on synthetic images and data, because of the scarce availability of complete and repeatable real imagery datasets. To validate such approach, experiments are necessary. In addition, while synthetic images allow to create large datasets, they are not always suitable for carrying out closed-loop PIL tests. In fact, the high computational cost of the rendering process may not allow real time tests, which may thus require the use of a real sensor. Finally, the whole navigation system performance can be assessed only by means of complete HIL tests, connecting the composing parts together to verify mutual influences. In addition to traditional image-based navigation techniques, the recent development of navigation algorithms based on Artificial Intelligence (AI) has posed further challenges to the verification and testing of the GNC chain. Therefore using both kind of data, synthetic and experimental, is mandatory for the development and testing of relative navigation algorithms.

1.2 The research problem

1.2.1 Objectives

The enhancement of autonomy in space missions is the core motivation of the Thesis, given the context explained in Section 1.1. More specifically, the principal research objective of the present Thesis is:

> *to study the exploitation of Electro-Optical sensors for enhancing relative navigation and proximity operations with uncooperative space targets.*

In order to develop the proposed objective, tools and techniques must be found merging and adapting knowledge from the two realms of space navigation and computer vision. The former offers consolidated algorithms with low autonomy, but light, robust and suitable for an on-board implementation; the latter provides advanced image processing and machine learning algorithms with high autonomy, and sensor fusion methods.

The Thesis always keeps into account that research is not independent of the real application. For this reason, the verification and testing of techniques with realistic data is an aspect that can not be ignored. In light of the fact that datasets from real missions are limited, or not public, or not existing for some scenarios, the work starts with the problem of realistic datasets generation. The work methodology consisted in exploiting both synthetic and experimental data for the techniques development, verification and testing.

1.2.2 Research questions

The Thesis objective is argued in the research, looking for the answer to the following questions:

1. How to generate representative synthetic visible images, thermal images and LiDAR data? How to obtain realistic experimental data and which is the ground truth pose accuracy achievable in the laboratory for a multi-sensor system?
2. Which is the actual difference between synthetic data and experimental or real data? To which degree can synthetic data be used for developing and testing of navigation algorithms for space applications?
3. Which are the limitations of visible imaging algorithms in complex mission scenarios with multiple targets of different nature? Can mapping with visible imaging be improved in the direction of an autonomous exploration and specifically how is the performance affected by factors and uncertainties not accounted for during the technique ground development?
4. Can multispectral imaging improve relative navigation and mapping? What is the contribution of thermal imaging? In which conditions a multispectral approach can lead to benefits to the navigation?
5. Can LiDAR-based methods provide robust and accurate solutions for pose estimation with an uncooperative target? How could they be used in synergy with camera for pose estimation?
6. What is the gap in pose estimation performance between synthetic and real data?

1.3 Dissertation overview

The Thesis dissertation follows the research questions listed above. The dissertation starts from researching ways for obtaining realistic synthetic and experimental sensors' data. From the comparison among them, some insights on their employment are drawn and then followed for the navigation algorithms development and testing. Then, innovative approaches for relative navigation are proposed. First, the employment of visual cameras for navigation and mapping is considered, dealing with non-traditional approaches and highlighting which are the limits that could be improved by employing other sensors. Then, the exploitation of thermal cameras and LiDAR scanners for pose estimation is studied, considering the sensors alone or in synergy with a visual monocular camera. More specifically, the Thesis is organized as follows.

Chapter 2 describes the background and state-of-the-art knowledge needed to develop the present work. The models of the different EO sensors are introduced and the state-of-the-art techniques for relative navigation with multispectral imaging and LiDAR sensors are presented.

Chapter 3 discusses the employment of vision data for development and validation of on-board algorithms. The different aspects and challenges related to the generation of synthetic data are faced, argued for visible cameras, thermal cameras and scanning LiDARs. The work then includes experimental activities in GNC facilities, tackling the issues related to calibration of new mock-ups and sensors. A comparison of artificial with experimental or real data is carried-out for the three kinds of sensors.

Chapter 4 deals with the navigation and mapping relying on visible imaging, with the aim of probing both the benefits and applicability limits of such kind of sensors, considering challenging applications. First, a scenario with multiple targets of different nature is examined, proposing a centralized relative navigation for far-range CubeSats in a binary asteroid system. Then, an AI-based policy for mapping an asteroid in the close-range is verified, testing it with uncertainties not accounted for during the learning and checking the advantages obtainable with the on-ground mapping from the collected images.

Chapter 5 investigates the possibility and advantages of employing imaging systems working in different spectral bands for the purposes of on-board navigation. In detail, relative navigation in proximity of an unknown asteroid is tackled, defining the boundaries for exploiting a multi-modal Simultaneous Localization And Mapping (SLAM) with visible and thermal images. Than, image-level multiscale transform-based fusion of visible and thermal images is performed and the SLAM with fused images is compared with the prior approach.

Chapter 6 focuses on LiDAR-based methods for pose estimation with an uncooperative spacecraft. In particular, a Point Completion Network (PCN) is trained for the global pose estimation with a known uncooperative spacecraft. The possibility of employing a visible camera along with LiDAR sensors for pose tracking is studied proposing a Visual-LiDAR Odometry (VO-LO) architecture. For both PCN and VO-LO results are discussed comparing the performance achieved with synthetic and real data.

Finally, Chapter 7 synthesizes conclusions about each aspect studied in the Thesis work. The major findings and results are highlighted, drawing a recommended road-map for future developments.

1.4 Bibliographic disclaimer

The present work is the result of three years of Philosophiae Doctor (Ph.D.), during which parts of the research were presented in conferences or published on peer-reviewed journals. The most significant publications related to this Thesis work are listed below, in chronological order.

- Piccinin, M., Zanotti, G., Silvestrini, S., Capannolo, A., Pasquale, A., and Lavagna, M. E. (2020). CubeSat Exploration Missions to Binary Asteroids: on Board Autonomy and Intelligent Imaging Towards Science Return Enhancement. In *2020 AAS/AIAA Astrodynamics Specialist Conference* (Vol. 175, pp. 1015-1028).
- Piccinin, M., Silvestrini, S., Capannolo, A., Lavagna, M. and Fernandez, J.G. (2021). Centralized Relative Navigation of Multiple Spacecraft with reduced visibility: validation campaign of the vision-based approach. In *11th International ESA Conference on Guidance, Navigation and Control Systems*.
- Silvestrini, S., Piccinin, M., Capannolo, A., Lavagna, M., and Fernandez, J. G. (2021). Centralized Autonomous Relative Navigation of Multiple Cubesats around Didymos System. *The Journal of the Astronautical Sciences*, 68(3), 750-784.
- Piccinin, M., Silvestrini, S., Zanotti, G., Brandonisio, A., Lunghi, P., and Lavagna, M. (2021). ARGOS: calibrated facility for Image based Relative Navigation technologies on ground verification and testing. In *72nd International Astronautical Congress* (IAC 2021) (pp. 1-11).
- Piccinin, M., Civardi, G. L., Quirino, M., and Lavagna, M. (2021). Multispectral Imaging Sensors for asteroids relative navigation. In *72nd International Astronautical Congress* (IAC 2021) (pp. 1-12).
- Piccinin, M., Lunghi, P., and Lavagna, M. (2022). Deep Reinforcement Learning-based policy for autonomous imaging planning of small celestial bodies mapping. *Aerospace Science and Technology*, 120, 107224.

CHAPTER 2

Background & State of Art

In this Chapter, the background and state-of-the-art knowledge needed to develop the present work are presented. First, the models of EO sensors are described. Then, techniques for relative navigation are presented, starting from LiDAR-based methods, and then describing multi-sensor methods, including visible-thermal imaging and visual-LiDAR techniques. Finally some tools for validation and testing are introduced.

2.1 Electro-Optical sensors models

In this Section, models for EO sensors are introduced. First, cameras' working principles are presented, including the response model of the two major categories of detectors for imaging in the visible and thermal spectral bands. Then, an overview of LiDAR sensors for space applications is presented.

2.1.1 Cameras

Cameras are EO sensors that collect light through an optical system onto a detector, thus capturing an image. A fundamental aspect of vision-based navigation algorithms consists in associating pixel coordinates of image points to real world coordinates.

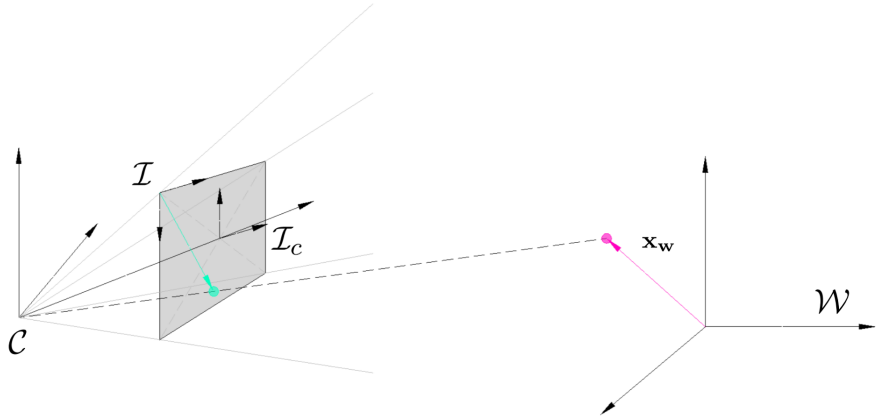


Figure 2.1: Camera model.

To define the relationship between world points and the image's pixels, the following reference frames, shown in Fig. 2.1, are considered:

- the world reference frame $\mathcal{W}$, being an inertial reference frame;
- the camera reference frame $\mathcal{C}$, that is fixed with the camera sensor. Its origin coincides with the camera center of projection, while the third axis is coincident with the principal camera axis and the other two axes are aligned with the detector edges;
- the image-centered reference frame $\mathcal{I}_c$, which is here defined as fixed with respect to $\mathcal{C}$, with the origin translated on the image plane;
- the image reference frame $\mathcal{I}$, which typically finds its origin in the center or external corner of the upper left pixel.

To obtain the camera coordinates of the world point $\mathbf{x_w} = [x_w \ y_w \ z_w]^T$, a transformation is applied to the point homogeneous world coordinates $\mathbf{x_w^H} = [x_w \ y_w \ z_w \ 1]^T$.

$$\mathbf{x_c^H} = [\mathbf{R_w^c}|\mathbf{t}]\mathbf{x_w^H} \tag{2.1}$$

where $\mathbf{R_w^c}$ is the rotation matrix from world to camera coordinates and $\mathbf{t}$ is the translation vector. The roto-translation matrix given by $[\mathbf{R_w^c}|\mathbf{t}]$ is also called *extrinsic matrix*.

Then, the point can be expressed in the $\mathcal{I}_c$ frame:

$$\mathbf{x_{I_c}} = \mathbf{K}\mathbf{x_c^H} \tag{2.2}$$

where $\mathbf{x_{I_c}} = [x_{I_c}\ y_{I_c}\ 1]^T$ and its components are expressed in pixel units. The camera model needs to be characterized by means of the intrinsic calibration, which consists in retrieving the *intrinsic camera matrix* $\mathbf{K}$ and the camera distortion coefficients. A simple pinhole camera model can be used for the intrinsic matrix:

$$\mathbf{K} = \begin{bmatrix} f_x & 0 & p_x \\ 0 & f_y & p_y \\ 0 & 0 & 1 \end{bmatrix} \tag{2.3}$$

where f_x and f_y are the camera focal lengths and (p_x, p_y) is the position of the principal point. The intrinsic camera matrix transforms the coordinates of a point from the camera frame to the image pixel coordinates $\mathbf{x_{I_c}}$ by means of a projective transformation [17].

Finally, the image coordinates of the point $\mathbf{x_I}$ can be found by means of a rigid translation in the image plane:

$$\mathbf{x_I} = \begin{bmatrix} x_I \\ y_I \end{bmatrix} = \begin{bmatrix} x_I - n_x/2 \\ n_y/2 - y_I \end{bmatrix} \tag{2.4}$$

where $[n_x, n_y]$ is the detector array size in pixel units.

Camera distortion For a real camera model, the two main distortion effects are often considered, being the radial and tangential distortion.

Regarding the radial distortion, the actual projected point pixel coordinates (x'', y'') are related to the ideal point coordinates (x', y') by a radial displacement described as:

$$\begin{bmatrix} x'' \\ y'' \end{bmatrix} = L(r') \begin{bmatrix} x' \\ y' \end{bmatrix} \tag{2.5}$$

where $r' = \sqrt{x'^2 + y'^2}$ is the radial distance from the center and $L(r') = 1 + k_1 r' + k_2 r'^2 + k_3 r'^3 + ...$ is an approximation of an arbitrary function.

Tangential distortion can be corrected applying the following transformation:

$$\begin{aligned} x' &= x'' + [p_1(r''^2 + 2x''^2) + 2p_2 x'' y''] \\ y' &= y'' + [p_2(r''^2 + 2y''^2) + 2p_1 x'' y''] \end{aligned} \tag{2.6}$$

Detector model Focal Plane Array (FPA) detectors are considered in this Thesis, being currently the dominant technology for cameras. Such detectors can be divided into two main categories: *photon detectors*, that can work from the visible (VIS) band to LWIR band and *thermal detectors*, which work in the LWIR band, as shown in Fig. 2.2. The sensors responses for the two types are hereafter described.

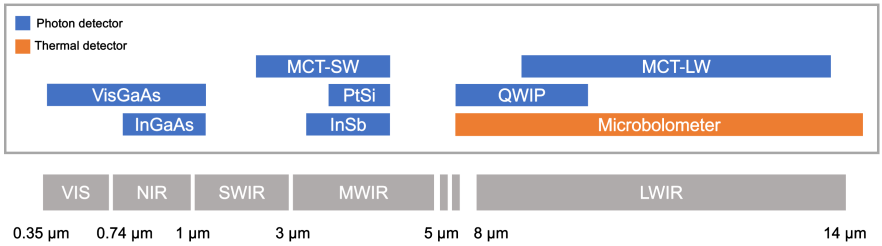


Figure 2.2: Typical detector materials [18] operational range along the spectrum.

2.1.1.1 Photon detector model

Photon detectors convert the absorbed electromagnetic radiation directly into a change of the electronic energy distribution in a semi-conductor by the change of the free charge carrier concentration. The working principle of such materials consists in the generation of a photocurrent proportional to the intensity of the incident radiation. In particular, when the material is exposed to impinging photons, if their wavelength is inside the sensibility range of the material, electrons can be moved from their valence band to the conduction band, generating a current.

Sensor response Considering a point target, in first approximation the number of photons incident on the detector during the exposure time t_{exp} can be computed as [19]:

$$P_{px} = P_{sun}\left[\pi\frac{d^2}{2}\rho g(\phi)\right]\frac{1}{r^2}\left[\pi\frac{d_a^2}{4}\right]t_{exp} \tag{2.7}$$

where P_{sun} the photon flux coming from the Sun at the target body, r is the range target-camera, ρ is the target reflectance, ϕ is the phase angle Sun-target-camera. The function $g(\phi)$ models the reflection on the target surface. The pixel response in Digital Numbers (DN) is computed as:

$$r_{px} = \eta\frac{P_{px}}{G} \tag{2.8}$$

where G is the camera gain and η is the detector efficiency, computed based on the fill factor FF, the detector quantum efficiency QE and the optical throughput η_{oe}:

$$\eta = \eta_{oe}\text{QE FF} \tag{2.9}$$

Sensor noise Background noise (expressed in $[DN]$) is computed as the Root Mean Square (RMS) of the noise sources: read-out noise, dark current and shot noise [20] [19].

$$\begin{aligned} N &= \sqrt{\sigma_{shot}^2 + \sigma_{dark}^2 + \sigma_{read}^2} \\ &= \sqrt{N_{shot} + (N_{dark}t_{exp})^2 + N_{read}^2} \end{aligned} \tag{2.10}$$

Shot noise is typically modeled as a Poisson process, hence its variance is equal to the mean: $N_{shot} = \eta P_{px}/G$.

2.1.1.2 Thermal detector model

Thermal detectors convert the absorbed electromagnetic radiation into thermal energy causing a rise in the detector temperature.

A common type of thermal detector is the uncooled microbolometer, which is made of a metal or semiconductor material. Due to recent developments of this technology, such sensors are becoming interesting for space applications. Microbolometer sensors present numerous advantages, such as the possibility to operate without cooling system; thus, having reduced dimensions, this technology is a promising option for deep-space CubeSats as well, thanks to the small dimensions of such compact devices.

The working principle of a microbolometer consists in applying a bias voltage to a film of thermo-sensitive material through the readout circuit. If the thin film is irradiated by infrared radiation, it heats up and its resistance value is changed; such resistance change is converted to a current or voltage signal that can be output by the readout circuit.

Sensor response The measurement provided by a microbolometer is proportional to the incoming scene radiation.

The pixel response r_{px} in $[DN]$ of the instrument can be modeled as:

$$r_{px} = G_{px}L + D_{px} \tag{2.11}$$

where L is the scene radiance and G_{px} and D_{px} are parameters determined experimentally during the instrument characterization. In particular, G_{px} is the camera gain and D_{px} the camera offset, also known as dark signal. The parameters can be determined experimentally by imaging a black body that occupies the whole scene, at different temperatures. This procedure allows to retrieve the response of each pixel of the detector array as function of the scene radiance. Due to the non-uniformity of the detector, each pixel has a different response and thus the parameters G_{px} and D_{px} must be determined for each pixel *(i, j)*. Typically, the offset and sensitivity variations on the array are quite relevant, thus during operations dark frames are acquired to correct the response.

Scene radiance Ideally, the emitted scene radiance L_e can be computed by means of the Stephan-Boltzmann law as:

$$L_e = \epsilon \sigma T^{\alpha} \tag{2.12}$$

where $\alpha = 4$. Whereas, the incoming radiance accounts for the view factor from the emitting body to the detector pixels:

$$L = F_{a-p} L_e \tag{2.13}$$

The view factor F_{a-p} between the target object and pixel surfaces is:

$$F_{a-p} = \frac{1}{A_p} \int_{A_a} \int_{A_p} \frac{(\hat{\mathbf{n}}_a \cdot \mathbf{s}_{ap})(\hat{\mathbf{n}}_p \cdot \mathbf{s}_{pa})}{\pi S^4} dA_p dA_a \tag{2.14}$$

where $\hat{\mathbf{n}}$ is the surface normal, $\mathbf{s}_{ij} = \mathbf{r}_j - \mathbf{r}_i$ is the position vector connecting the points on surface i with those on surface j and S is its magnitude ($S = ||\mathbf{s}_{ji}|| = ||\mathbf{s}_{ij}||$), as shown in Fig. 2.3.

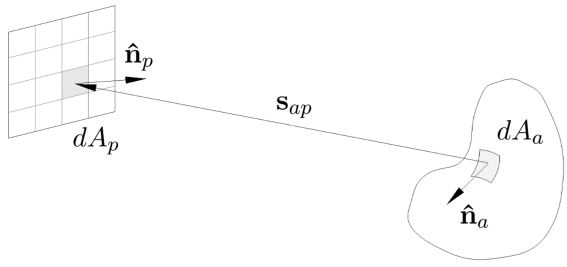


Figure 2.3: View factor between detector array and target area.

Sensor responsivity The responsivity R (or sensitivity) of a microbolometer array is defined as the ratio between the detector output, i.e. the voltage

variation, and the corresponding input, i.e. the temperature variation. The responsivity is thus expressed in $[DN/K]$ and corresponds to the slope of the response curve:

$$R(T) = \frac{dr_{px}(T)}{dT} \tag{2.15}$$

Using equations 2.11, 2.12 and 2.13 it turns to be:

$$R(T) = \alpha G_{px} F \epsilon \sigma T^{\alpha-1} \tag{2.16}$$

From experiments, the sensitivity has been proved to be quite homogeneous on the detector, thus it can be approximated as independent from the (i, j) pixel [21].

Sensor noise Microbolometers sensors are characterized by the Noise Equivalent Temperature Difference (NETD), that is the minimum ΔT that the instrument can resolve. In particular, it corresponds to the T difference that a signal equal to the temporal noise (SNR = 1) would produce. It can be expressed as:

$$NETD = \frac{N_t}{R} \tag{2.17}$$

where N_t is the temporal noise ($[DN]$) and R the camera responsivity ($[DN/K]$).

Experimentally, the NETD can be determined as the difference in temperature between two side-by-side blackbodies which, when viewed by the camera, gives rise to a difference in SNR of 1 in the electrical output of the two halves of the array. It is important to highlight that the NETD depends on the instrument f-number, the exposure time and the operating temperature of the camera.

$$NETD = NETD(f_{num}, t_{exp}, T_{cam}) \tag{2.18}$$

2.1.2 LiDAR sensors

While cameras capture the color, texture and appearance information, LiDAR sensors supply the 3D structure information of the environment. The working principle of a LiDAR sensor consists in illuminating the target with a light source, providing a ranging measurement ρ by detecting the radiation scattered by the target:

$$\rho = \frac{c\ t}{2} \tag{2.19}$$

where c is the light speed, and t the time incurred from the light emission to the detection of the scattered radiation.

The source of each laser pulse is a semi-conductor laser diode, composed by stacked n-p layers of a semi-conductor material. When current is applied across the diode, photons are emitted and routed into a focused beam.

Several types of LiDAR systems can be employed for space navigation purposes [4]:

- **Scanning LiDAR**. Such sensor is made by a row of laser diodes and detectors, that are put into rotation by a motor, scanning the environment with a tight laser. They emit and detect the light with a high frequency, scanning the scene. Such systems contain moving parts.
- **Detector arrays**. An array of lasers and detectors is kept fixed, illuminating the scene at once.

For what concerns LiDAR scanners and arrays, different approaches can be used to measure the light ToF:

- **Pulse/flash**. Discrete laser pulses are emitted, and the return pulse is awaited at the detector.
- **Continuous Wave**. A signal is modulated onto the laser and the phase shift is tracked onto the returning signal.
- **Pseudo Random Number**. A PRN is encoded onto the laser and then auto-correlation is performed with the sensed return.

Scanning LiDAR In case of a scanning LiDAR, in addition to the point range typically other information is provided. In particular, for each measurement the angular position of each laser can be derived in the LiDAR reference frame $\mathcal{L}$ (see Fig. 2.4). Hence, it is possible to measure the position of a world point in the sensor's frame, with a simple transformation from spherical to cartesian coordinates:

$$\begin{cases} x_L = \rho \cos\omega \sin\alpha \\ y_L = \rho \cos\omega \cos\alpha \\ z_L = \rho \sin\omega \end{cases} \tag{2.20}$$

where ω is the elevation angle, related to the laser detector ID, and α is the azimuth angle, given by the time of the measurement.

Therefore, the LiDAR output consists in a sparse point cloud, i.e. a set of unordered points, invariant to spatial transformations and where neighbouring points form a meaningful subset. A point cloud point can be represented only with its 3D coordinates or also adding other kind of information, such as color or surface normal. To change the format of such representation into a regular format (such as vortex grids or 3D images) is not convenient for a

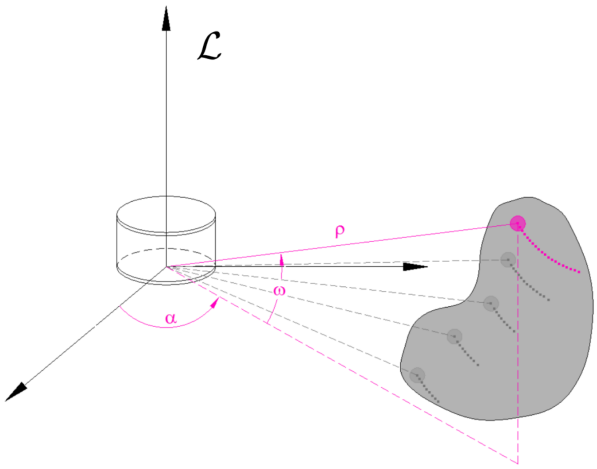


Figure 2.4: LiDAR model.

light online application, as it implies a high computational cost and storage memory. Therefore it is preferable to operate directly on the point cloud.

2.2 LiDAR-based relative navigation techniques

2.2.1 Classical methods

Classical methods for LiDAR-based pose estimation have been employed in both the ground robotics and in the space realms.

In the ground robotics field, LiDAR sensor's employment for retrieving the robot pose is quite consolidated and used in real-time applications. One of the most classical approaches is LiDAR Odometry And Mapping (**LOAM**) method proposed in [22] and depicted in Fig. 2.5.

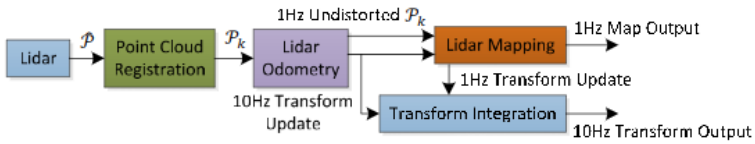


Figure 2.5: LOAM scheme [22].

Concerning the research in the space realm, in [23] a scanning LiDAR is employed for pose acquisition and tracking of a known uncooperative target. In particular, the pose acquisition is performed with a binary Template Matching

(**TM**) algorithm exploiting a voxel representation of the data. The pipeline is tested with proof-of-concept experiments.

In [24], another TM approach is proposed for navigation in close-proximity of a known uncooperative satellite. The TM operates directly on the point cloud and is in charge for the pose initialization, while the classical Iterative Closest Point (**ICP**) algorithm is employed for pose tracking. The presented approach is reliable, but has the limits of the huge amount of data to be stored on-board required by the TM and the consequent high computational cost. A highlight of the study is that the ICP performance is affected by the partial view of the target spacecraft.

In [25], a 3D LiDAR is used for providing an initial pose estimate, performing a registration of the acquired point cloud with a known model point cloud, exploiting the matching among a query point set and a match point set. The algorithm is tested in DLR EPOS facility, with real LiDAR point clouds.

2.2.2 Deep Learning methods

While several Deep Learning (DL) methods exist for estimating the 6-DoF pose of an object from visual information, such architectures can not be employed with point cloud data. In the recent years, some DL methods that handle this kind of data have been studied.

In [26], PointNet has been proposed to process directly point clouds for classification and segmentation problems. The authors introduce an architecture based on the use of a single symmetric function, max pooling, since point clouds are invariant to permutation of their members. The advantage of employing a symmetric function is in fact to aggregate information from all the points, without the need of performing a training with input permutation or without the need of finding a good sorting strategy for the input points (which is not a trivial problem). In [27], PointNet++ addresses the problems of partitioning point sets and of learning local features. PointNet++ applies PointNet recursively on a nested partitioning of the input set. In [28] a Point Completion Network (**PCN**) is presented as a new approach for shape completion, building up on PointNet and generating higher resolution outputs in an efficient manner.

In the context of pose estimation, in [29] it is proposed to employ the PCN architecture for the task of pose estimation with an uncooperative known satellite, as hereby detailed.

Point Completion Network architecture The PCN architecture is shown in Fig. 2.6. The PCN original task consists into shape completion of classes of objects. Taking in input a partial shape of an object, structured as a sparse

set of points, the PCN is capable of retrieving the whole shape of the object. It is composed by an encoder and a decoder.

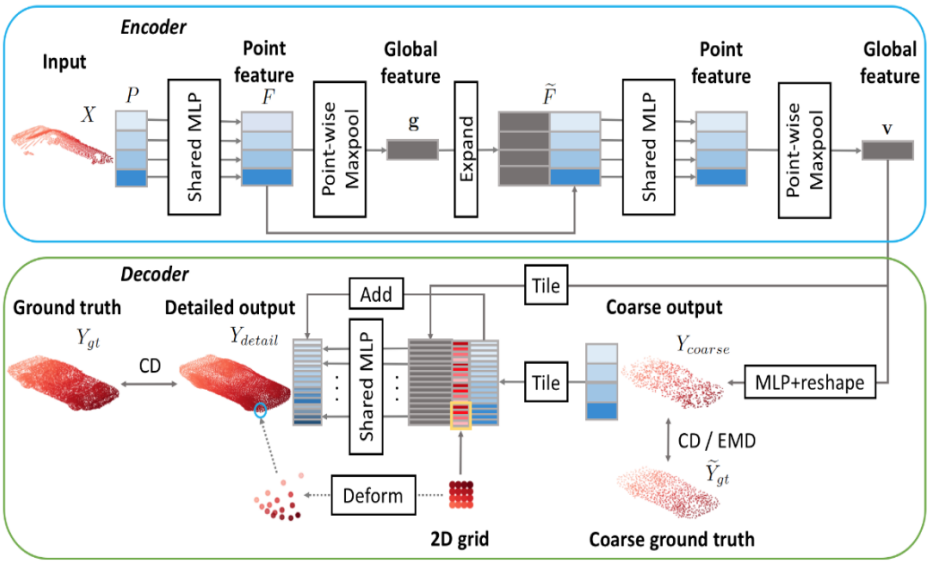


Figure 2.6: Point Completion Network architecture [28].

The encoder stacks 2 PointNet layers [30]. It takes as input a point cloud X, composed by a set of m points. In particular, each point $\mathbf{p_i} = (x, y, z)$ is described as a set of 3D coordinates and it is individually parsed into a shared Multilayer Perceptron (MLP), that outputs a feature per point. This operation repeated for each point results into a matrix F of point features $\mathbf{f_i}$. The set F undergoes the maxpool operator, from which only one global feature $\mathbf{g}$ of dimensions k is retained, where $\mathbf{g_j} = \max_{i=1,..,m}(F_{ij})$. Then, the obtained global feature is stacked with each individual feature in F and parsed into another shared MLP, obtaining other point features. The maxpool operator is then applied again, finally retaining the global feature $\mathbf{v}$.

The decoder is a multistage process composed by a fully connected decoder and a folding-based decoder. The fully connected decoder takes in input the global feature vector $\mathbf{v}$ and outputs a coarse point cloud $\mathcal{Y}_{coarse}$. The feature $\mathbf{v}$ is parsed into a MLP and reshaped, to obtain $\mathcal{Y}_{coarse}$, which has a fixed dimension dependent on the latent space dimension. Then, the folding operation is performed, creating a dense point cloud $\mathcal{Y}_{dense}$ of a smooth surface.

The strength of this algorithm is that it is independent on the dimension of the input point cloud, which varies depending on the LiDAR-target relative pose.

Loss function The loss function used for the training is a combination of the Chamfer Distance (CD) and the Earth's Mover Distance (EMD).

The CD is defined as:

$$CD(S_1, S_2) = \frac{1}{|S_1|} \sum_{x \in S_1} \min_{y \in S_2} ||x - y|| + \frac{1}{|S_1|} \sum_{y \in S_2} \min_{x \in S_1} ||y - x|| \tag{2.21}$$

where $S_1 \in \mathbb{R}^{mx3}$ and $S_2 \in \mathbb{R}^{nx3}$ are respectively the PCN output point cloud and the ground truth point cloud, while $x, y \in \mathbb{R}^3$ are their points. The CD metrics indicates the distance among ground truth and output: the output point cloud must lie close to the ground truth, and all the ground truth points shall be covered by the output.

The EMD distance is defined as:

$$EMD(S_1, S_2) = \min_{\phi: S_1 \to S_2} \frac{1}{|S_1|} \sum_{x \in S_1} ||x - \phi(x)|| \tag{2.22}$$

where ϕ consists into a bijection mapping that minimizes the average distance between the corresponding points of S_1 to S_2.

Finally the loss function is defined as:

$$\begin{aligned} L(\mathcal{Y}_{coarse}, \mathcal{Y}_{dense}, \mathcal{Y}_{gt}) = CD(\mathcal{Y}_{coarse}, \hat{\mathcal{Y}}_{gt}) + EMD(\mathcal{Y}_{coarse}, \hat{\mathcal{Y}}_{gt}) \\ + \alpha \; CD(\mathcal{Y}_{coarse}, \mathcal{Y}_{gt}) \end{aligned} \tag{2.23}$$

where α is an hyperparameter, $\mathcal{Y}_{gt}$ is the ground truth dense point cloud, $\hat{\mathcal{Y}}_{gt}$ is its sampling, while $\mathcal{Y}_{coarse}$ and $\mathcal{Y}_{dense}$ are the PCN outputs.

Pose estimation The PCN is trained with synthetic scans of a known client satellite. After training, a codebook is generated offline containing the encoder global features and a labelbook is created to store the corresponding poses. For the online pose estimation, a new scan is parsed into the PCN to encode its global feature; then, the pose is predicted finding the closest global feature in the codebook and its correspondence in the labelbook.

2.3 Multi-sensor relative navigation techniques

2.3.1 Visible-thermal imaging navigation

The use of multispectral imaging today is present in particular ground applications, such as detection and tracking of targets in the agricultural and military applications. The most common combination of cameras for multispectral

imaging is thermal and visual [16]. Concerning relative navigation techniques, thermal imaging is proposed as a method to overcome low-light conditions in visual odometry algorithms by means of non-uniformity correction [31].

Different techniques for exploiting the multispectral information exist, since the fusion can take place at different steps of the processing. An option consists in performing **visible-thermal images fusion**. Regarding pixel-level image fusion algorithms, the main categories are multi-scale transform, sparse representation, neural networks, subspace and saliency-based methods [32]. Such approaches aim at enriching the information content of the images. Visible and thermal images can be used as a **stereo-pair**. Multispectral VO for unmanned air vehicles based on stereo visible-thermal images has been presented and experimentally validated [33]. Other works consider the sensors as independent and **non-stereo**. A multispectral SLAM approach for robot navigation [34, 35], confirming experimentally [35] that the information added by the thermal camera improves the performance of the monocular SLAM approach. Another option consists in leaving the **measurements as separated** inputs for the navigation filter.

Concerning space applications, some studies on the possibility of using thermal and multispectral imaging were made in the recent years, following the ATV-5 LIRIS experiment that collected thermal images of the ISS.

In [36] a SLAM based approach is employed for the scenario of an ADR mission and tested with experimental thermal images of a heated mock-up, showing successful results. The importance of adopting more realistic space imagery data especially from real space debris to verify developed algorithms is highlighted.

In [37] a method for approximated thermal analysis of space debris along their orbits is proposed and analyses are made to evaluate the signature of space debris in the thermal infrared bands. Based on the debris materials, thermal inertia and orbit, their signatures are evaluated and guidelines for the applicability of thermal imaging for navigation are provided.

Within the ESA-funded Multispectral Sensing for Relative Navigation (MSR-NAV) project, in [38] the scenarios of non-cooperative rendezvous with a dead satellite (Envisat) and of landing on one of the components of the binary asteroid Didymos were studied. The Image Processing (IP) algorithm compares input images against a codebook with image representations (aggregate of visual features) associated to poses. To exploit the information provided by the different bands of the multispectral camera, the codebook is built for each band and the fusion of measurements is based on majority voting. The major findings are the importance of the thermal band for image segmentation when the Earth is in the background and in eclipse; while with favorable illumination

conditions configurations with a larger number of multispectral bands provide better results.

Finally, considering a far-range target, in [39] the infrared signal provided by a satellite target and received by a hyperspectral space telescope is simulated. A machine learning approach is then proposed to reconstruct the attitude motion from the time series of the CubeSat spectral light curve.

2.3.2 Visual-LiDAR navigation

In the recent years, in the robotics realm fusion approaches have been investigated, that exploit LiDAR information into monocular camera pipelines, which can be VO or SLAM approaches.

Indirect methods In [40] the **V-LOAM** method is presented. Visual odometry (DEMO) is combined with LiDAR odometry (LOAM) with two sequential stages, in order to estimate the pose and to build a metric map of the environment. Improvements are achieved with respect to previous odometry methods, in terms of accuracy but also robustness to ambient light changes and high speed of motion. The fusion is exploited in two blocks: the Depth Map Registration and in the Sweep to Sweep Refinement. The Depth Map Registration consists in a 2D-KT where the features depth is stored. The depth is computed by correspondences with the LiDAR point cloud and with triangulation of corresponding visual features in subsequent frames. The Sweep to Sweep Refinement consists in refining the LiDAR point clouds exploiting the VO knowledge. Finally, the transforms computed with motion estimation of camera and LiDAR (operating at different frequencies) are integrated.

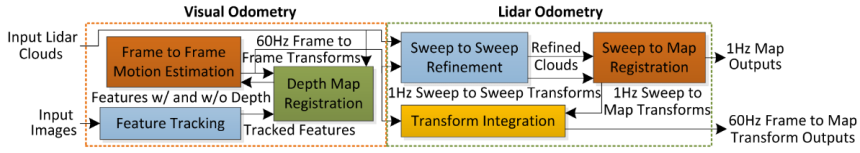


Figure 2.7: V-LOAM method scheme [40].

In [41], the **LIMO** architecture exploits the LiDAR points, projected onto the image plane, to estimate the depth of feature points in the image by fitting local planes.

In [42], a LiDAR-monocular visual odometry using point and line features is proposed. Point and line features are extracted from a monocular sequence; then, their depth is estimated by correlating the 2D points with the LiDAR 3D points; finally, a frame-to-frame odometry is performed. The sensor fusion

is also exploited in a scale correction optimization routine, which prevents the scale drift.

Direct methods The **LAPS** pipeline in [43], presents a cross modal approach where the camera localization method uses appearance of prior structure. The difference in appearance of LiDAR points in images is minimized from different view-points.

In [44], the **DLV-SLAM** method is presented, being a Direct Sparse Odometry method employed with window-based optimization. This direct method projects the LiDAR points on the current frame, forming an RGB sparse depth image. Image patches in the surrounding of the LiDAR points are considered in the reference frame. The tracking is performed with a Gauss-Newton optimization of photometric error of image patches between the current and reference frame. Similarly, a window-based optimization among keyframes is performed.

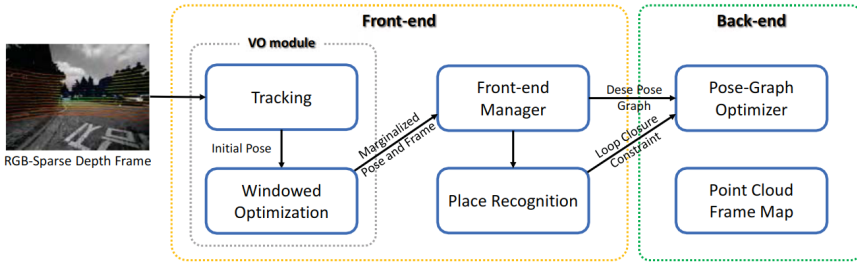


Figure 2.8: DLV-SLAM method [44].

In [45], **DV-LOAM** combines a two-stage direct tracking VO module with a LiDAR mapping module, based on a front-end for tracking and LiDAR scan-to-map optimization and a back-end for loop closure and pose graph optimization.

Regarding space applications, research in the LiDAR-camera fusion field has not yet been conducted, to the author's knowledge.

2.4 Algorithms validation and testing tools

In this Section, the tools used in the Thesis for the algorithms' validation and testing are introduced.

2.4.1 Tools for synthetic datasets generation

Real mission's datasets availability is generally scarce, but some datasets are made publicly available from space agencies. This is particularly true for scientific missions. Data typically comprise natural bodies' shape models, science or navigation instruments data and SPICE kernels. If available, such data are the best option for testing algorithms, but typically they are scarce due to the data transfers limits during missions and of course they are related only to a particular target, instrument and orbit/environmental condition. All the following datasets were employed for the development of this Thesis: ESA's COSMOS [46], NASA's Planetary Data System [47] and JAXA's DARTS for Lunar and Planetary Science [48].

In light of the fact that datasets from real missions are limited, or not public, or not existing for some scenarios, it is fundamental to produce artificial datasets for the aim of algorithms development, validation and testing. Synthetic visible image rendering is a well-known task that is achieved via ray tracing. Ray tracing is a rendering technique that relies on the concept of evaluating and simulating the path of view lines. Lines start from the observer camera and end on generic virtual objects that, together with the light rays simulated from the light sources to the virtual object, allow the computation of the color intensity of the related pixels. Several available tools allow a user-friendly environment to develop 3D scenes to generate images via ray tracing. In particular, general-purpose open source programs are available, like POV-Ray (Persistence of Vision Raytracer) by [49] and Blender, developed by [50]. While a commercial tool specific for rendering space targets is Pangu (Planet and Asteroid Natural scene Generation Utility), presented by [51]. In the context of this Thesis, all the three aforementioned rendering programs are employed.

Datasets with already generated images are also available, as the SPEED dataset, which contains images of the Tango satellite [52]. This option can be useful for development and comparison of machine learning-based techniques in the scenario of a satellite target.

Regarding thermal images simulation, less solutions are available. Commercial software can be used to synthesize thermal images in space scenarios, e.g. infrared sequences of Envisat were produced using the Astos' Camera Simulator in [53] and images have synthesized exploiting the Vega Prime software in [54]. Among commercial software, PANGU v6 will include a tool for TIR image rendering through a lookup table-based thermal image rendering model for natural scenarios, including physics-based features such as thermal lag and local variations in emissivity and absorptivity. Hence, such approach is limited to already explored natural bodies. Instead, for artificial bodies the thermal rendering is equation-based with a model that accounts for thermal energy

from solar, planetary reflectance, planetary emission, background radiation, and internal heat sources as reported by [55]. Such tools for thermal images rendering are not available for the development of this Thesis.

2.4.2 Tools for experimental datasets collection

A step further in the navigation's algorithms testing is the verification by means of real sensors data. When real datasets for the scenario of interest do not exist, the only option is then the one of experimental activities in dedicated GNC laboratories. When dealing with the employment of sensors in a GNC facility, their calibration is a fundamental aspect for allowing the utilization of the sensors and retrieval of the ground truth data. The calibration methods for cameras and LiDAR sensors are numerous. In the context of this Thesis the ones here after presented are used, being or the most classical or required/already tailored for the specific application of interest.

Camera intrinsic calibration A classical method for retrieving the intrinsic parameters of a monocular camera, is OpenCV's library chessboard-based calibration procedure [56]. The traditional procedure consists in finding the internal chessboard corners pixel coordinates and associate them to their world coordinates, knowing the chessboard dimensions and lying the corners on the same plane. Several images of the chessboard are taken, from different viewing angles, allowing to estimate the intrinsic matrix and distortion parameters [17]. First, the intrinsic parameters are initially computed, assuming no distortion. Then, the initial camera pose is estimated, solving the Perspective n Point problem (PnP) [57]. Finally, the global Levenberg-Marquardt optimization algorithm is employed to minimize the reprojection error.

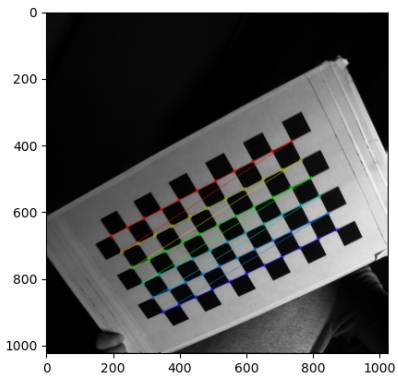


Figure 2.9: Camera calibration.

Dual-LiDAR system calibration Concerning scanning LiDAR sensors, a method for a dual-LiDAR system calibration is presented in [29]. Such method allows to find the relative pose between two LiDARs and the transformation among them and a Calibration Target (CT). Starting from an initial guess, the points belonging to the scans are transformed into the CT frame. A routine segments the points into two distinct sets:

- inner points S_{in}, whose projection on the CT plate falls inside the CT;
- outer points S_{out}, whose projection on the CT plate falls outside the CT.

An optimization is run to find the optimal transformations, i.e. the ones that minimize the distance of outer points from the CT edges and minimizes the distance of inner points from the CT plate.

In the notation here employed, $F1\_T\_F2$ stands for the transformation from reference frame $F2$ to reference frame $F1$. The reference frames of interest are:

- the calibration Target frame CT;
- the vertical LiDAR $L1$ and the horizontal LiDAR $L2$ frames.

The optimization procedure minimizes the following cost function:

$$\mathcal{L} = \frac{1}{N} \sum_{n=1}^{N} [\mathcal{L}_1 + \mathcal{L}_2] \tag{2.24}$$

where N is the total number of scans of the calibration target acquired with the LiDAR system, $\mathcal{L}_1$ is the loss computed from the points $\mathbf{p_1}$ of L1 scans, $\mathcal{L}_2$ is the loss computed from the points $\mathbf{p_2}$ of L2 scans.

$$\mathcal{L}_1 = \sum_{\mathbf{p_{1,i}} \in S_{in}^n} f^2\Big(CT\_T\_L1^n \ \mathbf{p_{1,i}}\Big) + \sum_{\mathbf{p_{1,j}} \in S_{out}^n} g^2\Big(CT\_T\_L1^n \ \mathbf{p_{1,j}}\Big) \tag{2.25}$$

$$\begin{aligned} \mathcal{L}_2 = &\sum_{\mathbf{p_{2,i}} \in S_{in}^n} f^2\Big(CT\_T\_L1^n \ L1\_T\_L2 \ \mathbf{p_{2,i}}\Big) \\ &+ \sum_{\mathbf{p_{2,j}} \in S_{out}^n} g^2\Big(CT\_T\_L1^n \ L1\_T\_L2 \ \mathbf{p_{2,j}}\Big) \end{aligned} \tag{2.26}$$

where $f(.)$ is the point to plane distance and $g(.)$ is the point to edge distance.

The optimization problem is formulated as follows:

$$(CT\_T\_L1^1, ..., CT\_T\_L1^n, L1\_T\_L2)^* = \text{argmin}(\mathcal{L}) \tag{2.27}$$

CHAPTER 3

Comparison and usage of synthetic and real vision data

In this Chapter, the gap between synthetic and real data is studied for the sensors of interest, carrying out a systematic comparison and drawing general guidelines for the data usage, considering different mission targets and various scenarios.

3.1 Synthetic data generation

3.1.1 Visible images

The realization of synthetic visible images can rely on the heritage from rendering programs, like Blender and POV-Ray, and on space-dedicated rendering tools, as Pangu. The key aspects in the data creation are here studied for different types of space targets and three different applications:

1. **Moon terrain**: landing on the South Pole.

2. **Point objects and asteroid terrain**: rendezvous with a binary asteroid system, with point CubeSats orbiting around the system.

3. **Space debris**: rendezvous with the Envisat space debris.

Moon terrain The chosen software for the lunar landing is Pangu, as it already provides nice starting features for this kind of application. An artificial Digital Elevation Model (DEM) of the Moon is exploited to generate simulated images taken by a landing navigation camera, to create a dataset useful for instance for training of AI techniques or end-to-end simulations of complete landing maneuvers. Two different approaches were considered, depending on the spacecraft altitude: one relies on a real DEM for high altitudes, the other on a completely synthetic DEM for low altitudes.

DEMs for the Moon are available from the LROC mission data [58] and depending on the area the available resolution can be between 100 m/px up to 5 m/px (for instance at latitudes from 87.5S to the South Pole). The use of the DEM is paired with an update Moon Crater Database considering all the craters with a diameter greater than 1-2 km. The real DEM is imported in Pangu and converted, with the possibility of exploiting particular projections. In the considered mission scenario, for the highest altitudes the Moon surface curvature effect is still relevant if the camera attitude is not downward. For this reason, a polar stereographic projection of the flat DEM surface is adopted in the area of the South Pole (60S to 90S), as shown in Fig. 3.1. Once the DEM has been projected, it is used as base to generate the world model. From realistic mission constraints, a resolution of 2 m is needed at an altitude of 3 km. Therefore, the DEM resolution has to be increased, depending on the spacecraft altitude upon the surface. Pangu provides the possibility of adding layers to the DEM area with increased resolution, maintaining the model light in regions where this is not necessary. Moreover, Pangu supports the addition of realistic craters and boulders with the desired dimensions, ages and frequency distribution. Craters and boulders are added to include such elements in the scene even for the smallest resolution; realistic sizes and distributions are considered. An example of synthetic image rendered from the modified real DEM is shown in Fig. 3.3.

Similarly, for low altitudes a flat DEM is created, perturbed with fractal noise and enriched with the other relevant terrain features, i.e. craters and boulders. Craters constitute the most relevant terrain feature, therefore they are often exploited in relative and absolute navigation algorithms. An example of synthetic image rendered from the synthetic flat DEM is shown in Fig. 3.2.

In order to generate a rich and representative dataset, the environmental variables in Table 3.1 are randomly varied within the reported ranges. By

doing so, the dataset can cover the wide feature-space that is expected in the operational scenario.

Table 3.1: Environmental variables and their range of variation for the dataset generation.

Variable	Range
Altitude	3 - 100 km
Attitude pitch (wrt vertical)	0° - 20°
Sun illumination angle - Elevation	0° - 90°
Sun illumination angle - Azimuth	0° - 360°
Synthetic crater frequency	1.8e6 - 3e6
Synthetic craters dimension	6 - 500 m

The craters reported in Table 3.1 refer to the synthetic generation performed in Pangu, with lunar impact crater size and distribution extracted from [59]. Considering a ballistic landing trajectory, during the coasting phase the spacecraft travels half of the transfer orbit, covering 180° in true anomaly. This implies that the illumination conditions, especially the elevation of the Sun over the terrain, that the navigation system is expected to encounter are extremely variable, from the Sun slightly above the horizon in polar regions to potentially straight illumination with 90° of Sun elevation close to the Lunar Equator. Regarding the Sun Azimuth angle, considering the South Pole region, it is related to the Moon rotation. Therefore, it can vary in the whole range between 0° to 360°. Actually, this wide range is applicable only for high latitudes close to the Poles. No particular constraints that bound the Sun inclination to a specific range (like the execution of the whole landing maneuver close to the lunar terminator) are assumed. Some examples of the generated images are reported in Fig. 3.5a, with the associated ground truth craters present in the images highlighted in Fig. 3.5b.

The main challenges found in synthetic data generation for this application are:

- the many constraints linking the availability of real DEMs, the considered Moon area, the particular projection needed depending on the spacecraft altitude, the minimum resolution needed depending on altitude and camera pitch;

- the conciliation between the need to limit the model resolution as much as possible and the rigid methods that allow the manipulation of the DEM in the software, which cause the unavoidable creation of very large models;

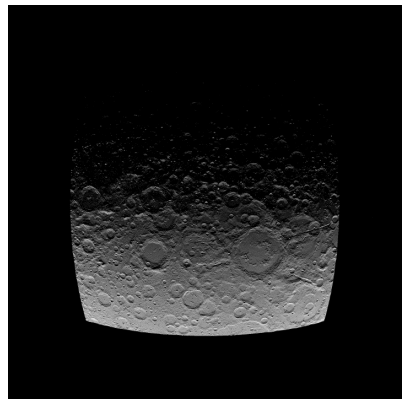


Figure 3.1: Example of polar stereographic projection of the 60S to 90S.

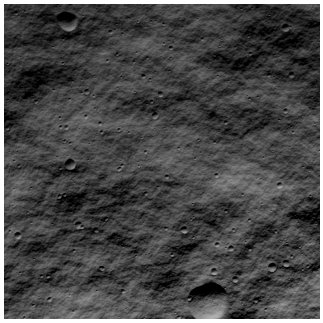


Figure 3.2: Synthetic image and DEM.

Figure 3.3: Synthetic image, real DEM.

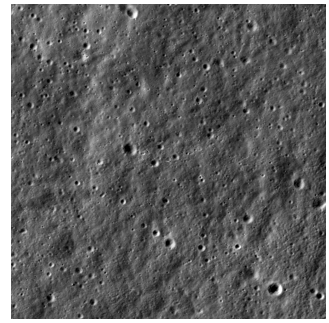
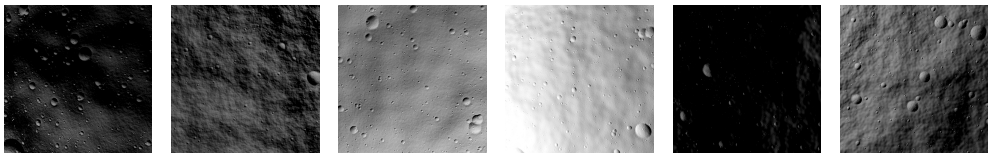
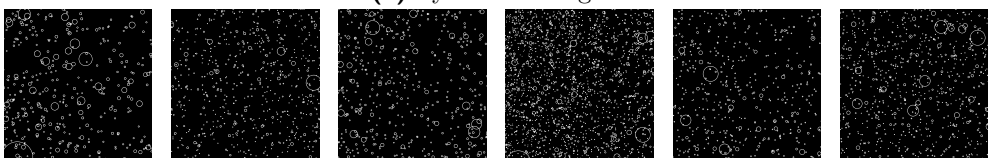


Figure 3.4: Real image from LROC.



(a) Synthetic images.



(b) Ground truth craters.

Figure 3.5: Dataset samples for different altitudes and illuminations.

- the achievement of a precise craters ground truth on a terrain that is projected or presents mountains/fractal noise.

Point objects and asteroid terrain For an application in which the main goal is to detect point CubeSats in proximity of asteroids, again Pangu is selected as preferred software. The main reason is that the key element is here found to be the EO sensor model [20], which in Pangu can be easily included considering the effects typical of the space environment. Representative images are synthetically generated using Pangu [60], as shown in Fig. 3.6.

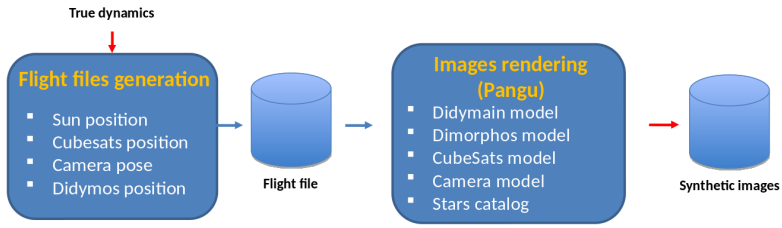


Figure 3.6: Implemented simulator block for images generation.

All the necessary elements for the images generation with Pangu are reported in the scheme in Fig. 3.6. The true dynamics of the objects is taken as input from the trajectory generation. The considered binary asteroid is Didymain. The camera is assumed to point towards the primary asteroid, while two CubeSats (CubeSat1 and CubeSat2) are assumed to be Sun pointing for their brightness analysis. The CubeSats visibility is confirmed, thanks to the low Sun phase angle.

The following modeling assumptions have been considered:

1. **Primary asteroid**. Didymain model is created starting from a sphere of radius 387.5 m and then adding Perlin noise, craters and boulders.

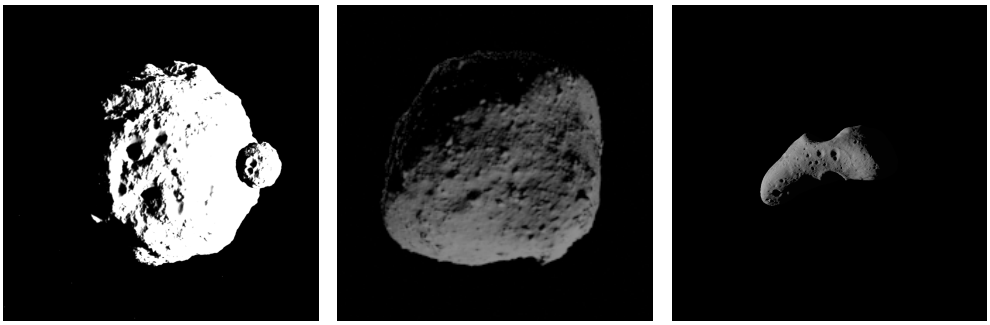
2. **Secondary asteroid**. Dimorphos model is created starting from a sphere of radius 81.5 m and then adding Perlin noise, craters and boulders.

3. **Spacecraft**. Considering their point-like object appearance, the CubeSats are modeled as cubes and rendered as imposter images. Since solar panels are not modeled, the side of the cube is assumed to be 1 m. The surface is assumed to be white and with a combination of diffuse and specular reflection.

4. **Stars**. The stars background is simply generated with the default stars catalogue and relating it with the trajectory kernels.

5. **Camera**. The camera is modeled according to the characteristics of a narrow angle camera, including distortion, Point Spread Function (PSF), shot noise and dark current [61].

An example of generated synthetic image including all the above mentioned elements is shown in Fig. 3.7a; it can be noticed that with the given camera exposure time and Sun phase angle, the asteroids saturate the image. Both point CubeSats and stars with high magnitude are visible in the image. This condition is suitable for a navigation in which the LoS of the objects has to be derived and also in the case in which point objects need to be detected.



(a) Didymos - Pangu. **(b)** Ryugu - Blender. **(c)** Eros - POV-Ray.

Figure 3.7: Examples of synthetic images for asteroids terrain.

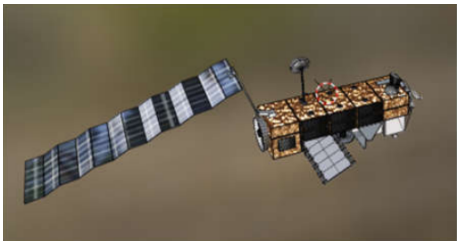
The main suggestion for the generation of data for such application is to tailor the data generation on the algorithm's need. Indeed, at least within Pangu framework, it is not possible to conciliate the need of an accurate rendering of an unknown asteroid terrain with the need of rendering far point objects. Thus the latter aspect has been privileged, at the expense of a detailed asteroid terrain with custom shape. Instead, if more detailed features of the terrain are needed, the suggestion is to start from a real asteroid mesh. This can be done with Pangu, but also with POV-Ray (see Fig. 3.7c) or Blender(see Fig. 3.7b). The latter program also allows an easy manipulation of the mesh in case custom shapes need to be created. On the other hand, the use of Pangu is suggested for the rendering of point objects and stars with an accurate sensor model.

Space debris Concerning the rendering of a spacecraft object, two different options are investigated. The first possibility is to use Pangu. Such tool is born mostly for rendering natural objects and it allows also to treat artificial objects, but with some limitations. The second option is to use a different tool, such as Blender, a photorealistic rendering software which is more flexible for the artificial object modeling.

In case Pangu is used, the artificial target needs to be first modeled in a Computer-Aided Design (CAD) program and then imported in the rendering software. In particular, the Envisat debris is here considered. Pangu allows to import models of artificial objects and to convert them in a more convenient format for the rendering. Nevertheless, Pangu does not allow to directly import the object geometry together with its materials. The final effect would be similar to the one in Fig. 3.8a. To overcome such limitation, the approach can be to break down the geometry into different parts and convert each part into a Pangu model. Each Pangu part can then be included by hand in an XML file in order to add the material properties and its appearance, specifying the BRDF (Bidirectional Reflectance Distribution Function) and color. Some additional features are supported, as the Multi Layer Insulation (MLI) textures. Anyway, this procedure can be cumbersome with a complex geometry such as the one of Envisat.

The other possibility is to render Envisat with Blender. The geometry can be imported from a CAD of the spacecraft and materials can be easily modeled by means of a GUI. Textures can be added and several options are available for specifying the materials properties, including the BRDF. An example of obtainable images is reported in Fig. 3.8b. Blender can be easily interfaced with Python for the database creation, creating an automatic sequence of camera poses to be rendered, varying the illumination as well. On the other hand, less parameters for the camera model can be specified. Therefore, for instance noise needs to be added after images have been rendered with some post-processing.

(a) Pangu framework.



(b) Blender framework.

Figure 3.8: Example of artificial target modeling and rendering tools.

3.1.2 Thermal images

While traditional and consolidated rendering techniques allow to generate visible synthetic images, analogous tools for thermal images rendering do not exist.

The architecture shown in Fig. 3.9 is proposed for the generation of realistic thermal images. Such tool will be referred to as ADE. The proposed framework focuses on a natural body, in particular on an asteroid target, because of the availability of real data from the Hayabusa 2 mission for the process validation, but it can be extended to other types of targets, such as terrains or artificial object.

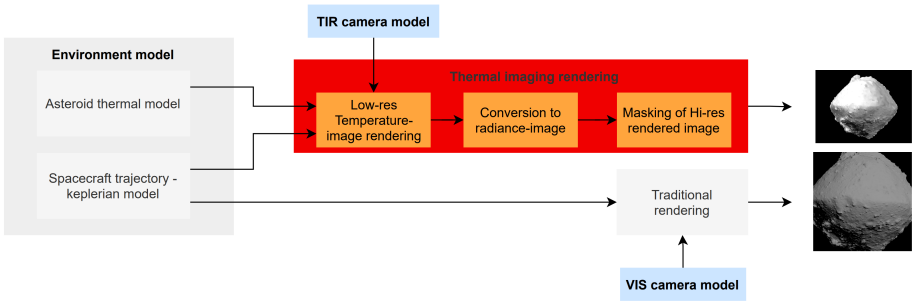


Figure 3.9: Thermal images chain model.

The proposed tool starts from the thermal simulation of the target body, which is performed on a low resolution shape model, due to computational resources availability. The thermal simulation of the body is performed starting from the asteroid and space environment inputs and deriving the temperature distribution field at the trajectory epoch. The output of the simulation consists into a low resolution shape model with the temperature field, in .vtk format.

The temperature information is re-elaborated inside a rendering software (Blender), with the aim of quickly project it from the 3D point into a 2D temperature image.

Then, the response of an uncooled microbolometer sensor is modeled, accounting for the input radiation and the related NETD. This model allows the conversion of the temperature image into a radiance image and then into a digital image.

As previously explained, the thermal simulation is run on a low resolution shape model. The consequence is that the projected shape model facets are larger than the image pixel. As first step of the tool development, the issue is solved by means of an artifact. A high-resolution image of the asteroid surface is rendered and exploited to mask the low-resolution temperature-image, adding the details while keeping the mean brightness value for the pixels corresponding to each facet. This post-processing step aims at obtaining a more representative appearance of the target asteroid. In particular, to match the expected microbolometer physical response - proportional to the radiation power absorbed by the detector pixels - and re-introduce low level details, which nevertheless do not derive from a thermal simulation.

As an extension of the tool, greater computational resources shall be allocated to work directly with larger shape models and the thermal simulation shall be performed by means of a CFD software, as OpenFoam. This step is currently under development at Politecnico di Milano, and for this reason the tool is compatible with the .vtk output provided by the software.

Asteroid thermal model The thermal model considers the contributions of the incoming Sun heat flux Q_{in}, the radiated flux Q_{out} and the heat transfer towards the ground layers Q_s:

$$Q_{in} = Q_{out} + Q_s \tag{3.1}$$

The solar flux depends on the distance between the Sun and the asteroid and on the incidence angle ϕ between the direction of the Sun and the normal to the heated surface:

$$Q_{in} = S_{\odot}(1 - A)\left(\frac{AU}{r}\right)^2 \cos(\phi) \tag{3.2}$$

where $S_{\odot}$ is the solar constant, A is the bolometric Bond albedo and r is the distance to the Sun expressed in AU. The flux emitted from the asteroid at the thermal equilibrium is commonly written through the Stephan-Boltzmann law:

$$Q_{out} = \epsilon\sigma T^4 \tag{3.3}$$

where ϵ is the emissivity of the surface of the asteroid, σ is the Stefan-Boltzman constant and T is the temperature of the asteroid. The heat flux at the surface of the asteroid is linked to the the 1D temperature gradient

$$Q_s = -k \left.\frac{\partial T}{\partial x}\right|_{x=0} \tag{3.4}$$

being k is the conductivity of the ground of the asteroid. The x coordinate is taken in the radial direction, positive downwards. The heat conduction equation is then applied, in which a constant conductivity is assumed for simplicity:

$$\rho c_p \frac{\partial T(t,x)}{\partial t} = k \frac{\partial T(t,x)}{\partial x^2} \tag{3.5}$$

Finally, boundary conditions are set to complete the model:

$$\begin{cases} T(0,x) = f(x) & \forall\ x \in [0, l_s] \\ T_x(0,t) = \frac{Q_{out} - Q_{in}}{k} & \forall\ t \geq 0 \\ T_x(l_s,t) = 0 & \forall\ t \geq 0 \end{cases} \tag{3.6}$$

Where T_x denotes the partial derivative of temperature with respect to the spatial coordinate x and the skin depth l_s is regarded as the depth of the ground where an adiabatic assumption can be acceptable. A numerical finite-difference technique is used to solve the 1D heat conduction equation and an iterative technique is used to solve the surface boundary condition.

Microbolometer response model Starting from the thermal sensor model presented in Section 2.1, the following assumptions are made. The responsivity is assumed homogeneous on the detector, therefore for a specific scene temperature $\bar{T}$:

$$R(\bar{T}) = \alpha(i,j)G(i,j)\bar{T}^{\alpha(i,j)-1} \tag{3.7}$$

A mean value of the exponent α is taken on the detector pixels (i,j) [21], leaving the spatial dependence on the gain only:

$$G(i,j) = R(\bar{T})/(\alpha\bar{T}^{\alpha-1}) \tag{3.8}$$

The non-uniformity of the detector gain and offset are accounted for, exploiting data available in the literature from experimental activities on space microbolometers sensors for asteroids thermal mapping [21]. Such effects are important to be modeled in case of the absence of non-uniformity corrections, which may be the case for on board application because of re-calibration needs or limited computational power reasons.

Thus the pixel response $r(T,i,j)$ in $[DN]$ is computed as:

$$r(T,i,j) = G(i,j)F\epsilon\sigma T^{\alpha} + D(i,j) + N_t \tag{3.9}$$

where $N_t = R(\bar{T})$NETD is the temporal noise, which depends on the detector responsivity and thus on the temperature of the observed scene.

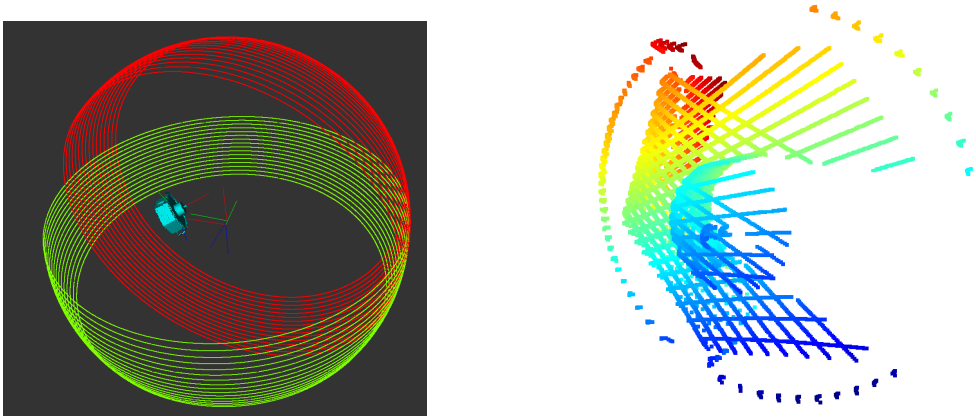
A simplified radiation view factor F can be adopted for a quasi-spherical body, as the case of emission from a sphere of radius R to small flat surface at distance H:

$$F = \left(\frac{R}{H}\right)^2 \tag{3.10}$$

3.1.3 LiDAR scans

For a scanning LiDAR, a synthetic point clouds can be generated by means of ray-tracing from the sensor to the target. A tool developed at DLR with this purpose [29] has been employed. According to the vertical and horizontal resolution achieved with the scanner, a ray is traced for each grid point from the origin of the sensor. The data point is found by intersecting the ray with

the target model mesh. The tool has been slightly modified in few aspects, to tailor it for its specific application in the OOS-SIM facility. The tool framework and an example of generated scan are shown in Fig. 3.10a and 3.10b.



(a) Tool framework. **(b)** Example of synthetic scan.

Figure 3.10: LiDARs synthetic data generation.

3.2 Laboratory data gathering

Some parts of the present doctoral research has been developed in two different vision-based GNC facilities: ARGOS (Advanced Robotics & GNC Optical-based Simulator) at Politecnico di Milano and OOS-SIM (On-Orbit Servicing SIMulator) at the DLR RMC Institute. In this Section, a brief description of the two facilities is provided and then the calibration activities carried-out are described.

3.2.1 Vision-based GNC facilities

3.2.1.1 ARGOS facility

ARGOS facility at PoliMi comprises the fundamental following elements:

- Navigation camera.
- Illumination system.
- Moon diorama.
- Artificial satellite mock-up.

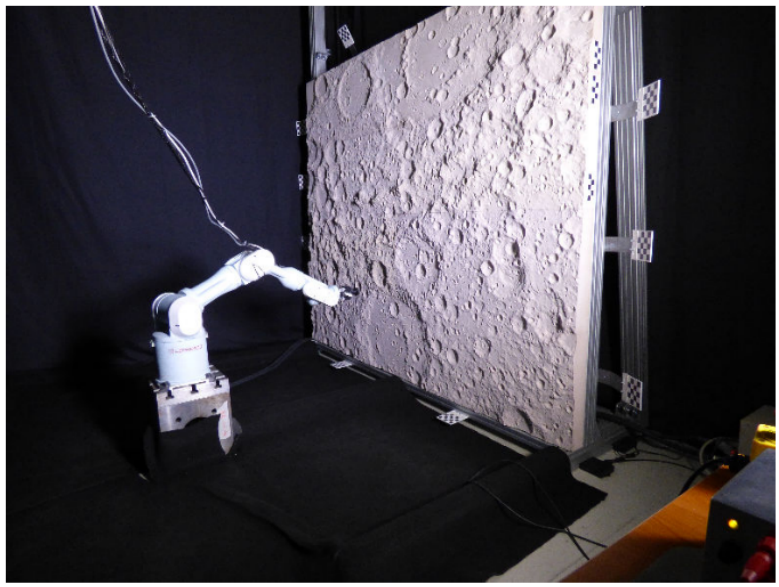


Figure 3.11: ARGOS facility at DAER-Polimi.

Navigation camera The main sensor of the reproduced autonomous GNC architecture is made up by the navigation camera, as representation of a possible flight hardware. Given the laboratory envelope, the camera has a wide FoV and fixed focus is exploited, without the need of having high resolutions. For this facility, a Chameleon 3 by FLIR is used, whose characteristics are reported in Table 3.2. Such camera can be easily tuned through a dedicated C++ API, and can be interfaced to the rest of the system by USB3.1 connection and a programmable GPIO. A grey-scale images configuration is adopted for simulating flight hardware, exploiting lower frame rate and a 1024×1024 resolution.

Table 3.2: Navigation camera specifications.

Parameter	**Value**
Resolution	1280x1024
Frame rate	149
Megapixels	1.3
Chroma	Colour/grey-scale
ADC	10 bit
Sensor format	1/2-inch-type CCD
Focal length	6 mm
FoV	63.5°

Illumination system In order to properly validate the images, it is fundamental to ensure a proper illumination and environmental conditions for obtaining realistic images. To achieve that, the light shall not be diffuse, given that the operational conditions under investigation all outside the atmosphere. For such reason, the facility has been realized in a dedicated dark room, preventing light reflection with black curtains and floor covers. To simulate the Sun illumination, a dedicated high-CRI LED array with 60° beam angle and 5700 K light temperature is employed, characterized by the features in Table 3.3.

Table 3.3: Illumination system specifications

Parameter	Value
Light temperature	5700 K
Beam angle	60°
Led array dimension	1024x1024

Moon diorama A portion of the Far-Side Lunar surface has been realized fully at the PoliMi-Daer laboratories, to be exploited as a mock-up for the landing simulations. To guarantee correct optical properties, the material used for the diorama is Urethane foam due to its surface finish. The diorama is made up by 8 separated tiles, measuring 1200×500 mm, leading to an overall size of 2400×2000 mm and a scale factor of 2000:1. In order to have mixed terrain features, i.e. from plains to rough slopes, a DEM from the GLD-100 NASA LROC dataset was employed. The facility was designed to test navigation algorithms for pinpoint landing with desired touch-down accuracy in the order of 10 m, which in the scaled mock-up corresponds to 5 mm. In order to keep a terrain resolution of at least one order of magnitude better than the required accuracy, the machine-working accuracy of at least 0.5 mm was required.

The facility was also equipped with a robotic arm for simulating trajectories and carrying the camera on the end effector. Unfortunately, the robotic arm was not available during the period of the present research.

3.2.1.2 OOS-SIM facility

The OOS-SIM is a simulator for on-orbit servicing of the Robotik und Mechatronik Zentrum part of the DLR [62].

As visible in Fig. 3.12, the facility is equipped with two satellites mock-ups, a servicer and a chaser. The servicer carries a sensitive lightweight robot arm with gripper to perform tasks as capture of the client satellite. The entire servicing system is called free-flying robot in orbital robotics. Two large

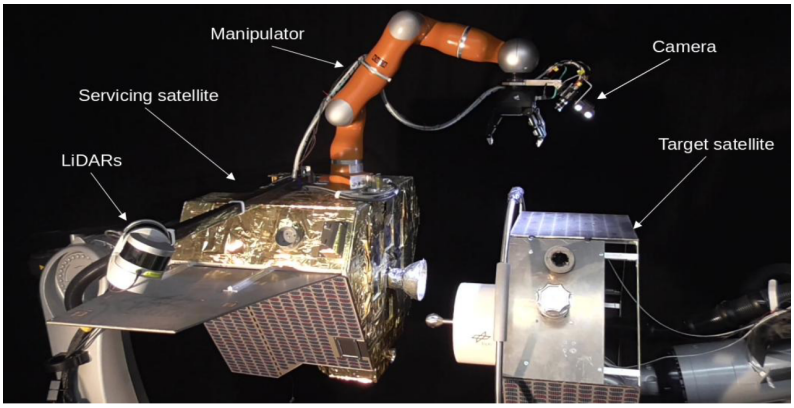


Figure 3.12: OOS-SIM: on-orbit servicing simulator [29]

industrial robots hold the satellites mock-ups, simulate their weightlessness and relative motion [63]. In order to perform the close approach with the client, the servicer is equipped with a stereo camera attached to gripper of robot and a LiDAR system composed of two scanning LiDARs, attached to a boom.

The facility thus allows to study complex tasks such as assembly, maintenance and repair work on satellites. It also investigates the system's applicability for the removal of non-functioning orbiting target satellites [64]. It provides a suitable framework for development and validation of the orbital robot control with the support of computer vision methods through data acquired by the servicer sensors.

Navigation LiDAR In the OOS-SIM, two VLP-16 by Velodyne are available. The two lidars are mounted on the servicer boom, tilted of 90° with respect to each other. The VLP-16 is a scanning LiDAR emitting laser beams with 16 channels working in the Near InfraRed (NIR). A laser channel is a single 903 nm laser emitter and detector pair. Each laser channel is fixed at a particular elevation angle relative to the horizontal plane of the sensor. The characteristics of a VLP-16 are reported in Table 3.4.

An output data point of the device consists in the calibrated reflectivity of the target (which can be classified in diffused or reflective material) and in the range measurement.

In order to define the 3D coordinates of a data point, in addition to the distance, two angular measurements in the LiDAR-fixed reference frame are necessary. Both angles are given in the data output, along with the time stamp (necessary for the multi-sensors algorithms).

Table 3.4: VLP-16 characteristics.

Parameter	Value	Unit
Wavelength	903	nm
Maximum range	100	m
Channels	16	-
HFoV	360	deg
VFoV	30	deg
Rpm	300-1200	rpm
Horizontal resolution	0.1-0.4	deg
Vertical resolution	2	deg

3.2.2 Experimental activities in ARGOS

The activities carried-out in ARGOS comprised the creation of a satellite mock-up of Envisat and the gathering of images of the created mock-up and of the Moon diorama, with the corresponding pose ground truth.

Artificial satellite mock-up The Envisat mock-up is 1:50 scale of the real satellite, entailing all the most critical geometrical and appearance features. The mock-up is manufactured using 3D Stratasys Fortus450 3D printer available at Politecnico di Milano premises. The model dimensions are maximized to reduce the scaling of the experiment, compliant with the printer capabilities. In this way, the accuracy to validate the algorithms can be reached. A spray acrylic white paint, usually used for modeling, was applied twice on the surface to reproduce the surface of the spacecraft. The details were adjusted by a thin brush. Silver aluminum foils were used to reproduce the classical thermal protection of spacecraft, made of MLI. The white paint and silver aluminum foil have been applied to maximize the reflectivity of the materials and enhance the contrast with respect to the laboratory background.

Camera pose retrieval The present work is based on images acquisition without the employment of the robotic arm, that was not available due to contingencies external to this research work. In this case, it is possible to derive the camera pose using visual markers applied on the target or in the target proximity. If the acquired images contain the visual marker, standard libraries can be used to compute the camera pose with respect to the marker (as OpenCV [56]), retrieving the translation from camera to marker $\mathbf{t_{c,m}}$ and the rotation matrix $\mathbf{R_c^m}$.

Thus, if the position and rotation of the marker with respect to the world frame are known ($\mathbf{t_{m,w}}$, $\mathbf{R_m^w}$), the camera pose in the world frame is simply computed as:

$$\mathbf{x_{c,w}} = \mathbf{t_{m,w}} + \mathbf{x_{c,m}} \tag{3.11}$$

$$\mathbf{R_c^w} = \mathbf{R_c^m R_m^w} \tag{3.12}$$

Markers calibration The marker calibration procedure consists in the marker pose measurement with respect to the world reference frame. Two strategies are tried:

1. **Direct measurement in the facility**. The marker center position is measured with respect to a known point on the target, while its orientation is fixed in order to easily derive a known rotation with respect to the target frame. Multiple position measurements were taken from different points and the mean was retained.
2. **Dense reconstruction of the target including markers**. If an accurate mesh can be reconstructed including both target (Moon or satellite mock-up) and applied markers, then the transformation can be numerically computed, finding the corners' vectors of the marker in the reconstructed mesh.

A dense matching method has been selected to perform the shape reconstruction, based on the Colmap software [65]. Several photos from different angles are taken; then, structure from motion algorithms are used to obtain the camera pose for each image. Finally, dense cloud point models are obtained by triangulation of optical features between different frames. For the Lunar terrain mock-up, reconstruction is performed for the testing region. The dense reconstruction is composed of three main steps that were performed by means of the Colmap software [65]: feature detection and extraction; feature matching and geometric verification; structure and motion reconstruction; multi-view stereo and dense reconstruction.

Results Both the adopted procedures successfully allowed to collect the images with the related poses, an example is shown in Fig. 3.13. Aruco markers were employed, exploiting a predefined dictionary of 50 7-pixels markers. Different illumination conditions were used and measured by means a sundial, built for this purpose.

Fig. 3.14 shows a reconstructed point cloud of a limited testing region in the Moon diorama, refined through Screened Poisson surface reconstruction. As shown in Fig. 3.13,

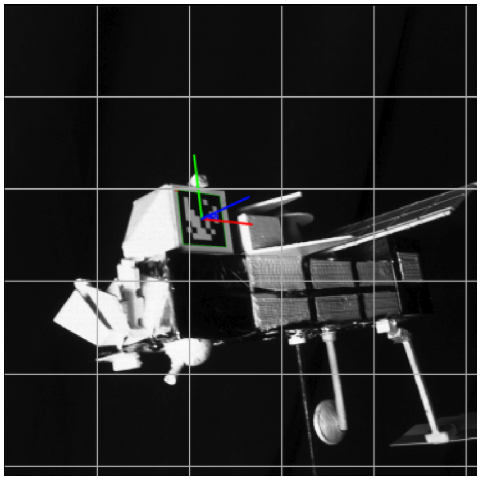


Figure 3.13: Pose retrieval with visual marker.

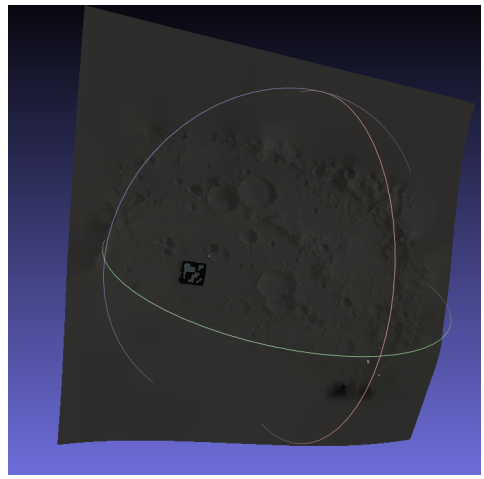


Figure 3.14: Dense point cloud reconstruction of testing region.

3.2.3 Multi-sensor calibration in OOS-SIM

Concerning the activities carried out at the OOS-SIM, the main goal is to calibrate the dual-LiDAR system in the facility. The calibration of the two LiDARs in the OOS-SIM is necessary to obtain the ground truth of their pose in the laboratory. In particular, the aim is to find their pose with respect to the servicer satellite on which they are mounted. As a consequence, after the calibration is performed, it will be possible to align the measurements of the stereo-camera and the LiDARs and thus develop and test multi-modal computer vision algorithms in the laboratory.

The approach here proposed consists in performing two different calibration paths, and comparing the obtained results, providing a quantification of the error. The method is schematized in Fig. 3.15. Two routines for calibrating the stereo-camera and the dual-LiDAR system are exploited. They are presented hereafter.

LiDAR-LiDAR extrinsic calibration The methodology proposed and developed in [29] has been adopted for calibrating the LiDAR system with the use of a calibration target. For the method formulation please see Section 2.4. The method has been tested with synthetic LiDAR data in [29]. From such numerical simulations, the method has shown to be capable to deal with synthetically introduced noise. For the usage with real data, an extension of the method is desirable. When gathering a scan of the CT, the whole scan from the LiDARs will be present, including the surrounding environment. The

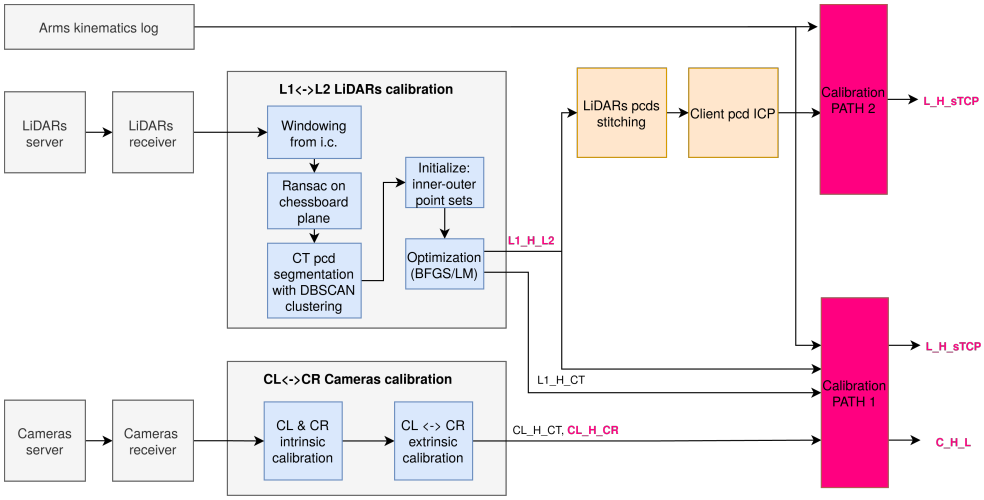


Figure 3.15: Multi-sensor calibration method.

problem of segmentation of the CT point cloud from the overall point cloud is not trivial. The following prior steps are introduced:

1. Extraction of interest bounding box from ambient point cloud.
2. RANSAC routine for finding the CT plane in the point cloud and outliers removal.
3. Clustering algorithms for removal of inliers not belonging to the CT.

The LiDAR output is organized in UDP packets, providing the measured distance and two angles. The box extraction is based on a configuration file, related to the used set-up, in which the bounding limits are defined in terms of spherical coordinates, as they are directly related to the lasers identification and to the LiDAR pointing direction. The RANSAC algorithm is then used to find the plane model and remove outliers, i.e. points lying outside the calibration plate [66]. Then, the obtained point cloud is segmented in each of its composing laser scan. Each scan line is processed with the DBSCAN algorithm to identify the cluster of points belonging to the chessboard and the noise points. Finally, the optimization routine is run and the transformation $\mathbf{H}_{\mathbf{L2}}^{\mathbf{L1}}$ from the L2 reference frame to the L1 frame is found. Also the transformation with the calibration target $\mathbf{H}_{\mathbf{L1}}^{\mathbf{CT_c}}$ is retrieved, where the frame CT_c is placed on one of the two centers of the stereo-calibration target.

Stereo-camera calibration algorithm The stereo-camera calibration relies on DLR toolbox CalLab [67]. As first step, the intrinsic calibration of each camera is performed. This consists in retrieving the camera intrinsic parameters of the left camera (CL) and of the right camera (CR) and thus

properly distort and un-distort image coordinates. At least four shots of the chessboard are necessary, inclining the board up, down, right and left. As a second step, a static image of the chessboard is taken with the two cameras at the same time and the relative pose among them $\mathbf{H}_{\mathbf{CR}}^{\mathbf{CL}}$ can be computed, along with the transformation with respect to the chessboard $\mathbf{H}_{\mathbf{CT}}^{\mathbf{CL}}$.

3.2.3.1 Calibration PATH 1

Calibration PATH 1 is schematized in Fig. 3.16 consists in exploiting a calibration target (a rigid chessboard) and gathering data of the CT with the LiDAR system and the stereo-camera, keeping the system static and requiring a single pose to perform the calibration.

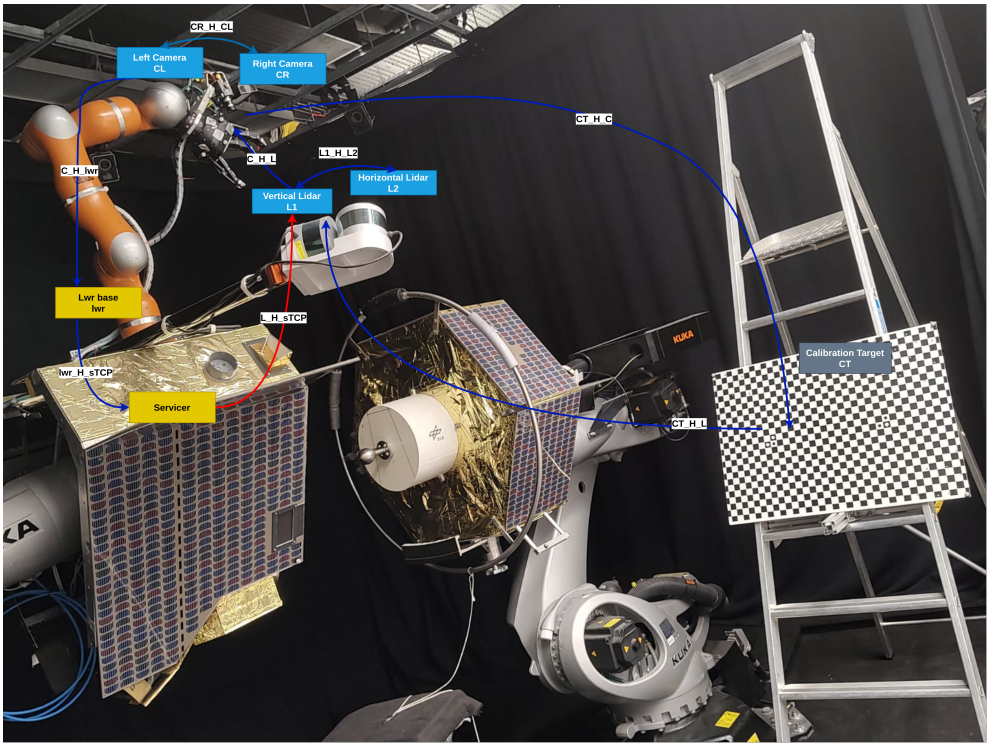


Figure 3.16: Calibration PATH 1.

The images and scans are then processed with the two calibration algorithms, the stereo-camera system calibration and the dual-LiDAR system calibration. Their output is the pose of all the sensors with respect to the CT. This allows to find the relative pose between LiDAR system and stereo-camera.

The stereo-camera and LiDARs are used to image and scan the calibration target in a static set-up. From the stere-camera and LiDAR system calibrations algorithm, the following transformations are found respectively $\mathbf{H_{CL}^{CTc}}$, $\mathbf{H_{L1}^{CT}}$.

The transformation $\mathbf{H_{CTc}^{CT}}$ is known from the calibration chessboard manufacturing and measurement.

$$\mathbf{H_{CL}^{L1}} = \mathbf{H_{CT}^{L1}} \cdot \mathbf{H_{CTc}^{CT}} \cdot \mathbf{H_{CL}^{CTc}} \tag{3.13}$$

This provides a direct calibration among the two sensor systems.

From the industrial robots kinematics, it is possible to log the transformation $\mathbf{H_{s}^{s0}}$ of the servicer center point with respect to its start-up position.

From the manipulator kinematics, it is possible to log the transformation $\mathbf{H_{lwr}^{s0}}$ of the manipulator target center point with respect to the servicer start-up position.

The camera system pose with respect to the light weight robot ($\mathbf{H_{CL}^{lwr}}$) is known from prior extrinsic calibration activities performed at the OOS-SIM.

The extrinsic calibration of the LiDAR system in the servicer body-fixed frame, thus results to be:

$$\mathbf{H_{s}^{L1}}\Big|_{PATH1} = \mathbf{H_{CL}^{L1}} \cdot \mathbf{H_{lwr}^{CL}} \cdot \mathbf{H_{s0}^{lwr}} \cdot \mathbf{H_{s}^{s0}} \tag{3.14}$$

The possible error sources for this path are expected to be:

- not exact calibration plate segmentation in the LiDARs scans;
- inaccuracies in the kinematics of the light weight robot;
- exploitation of past results from the stereo-camera extrinsic calibration with respect to the light weight robot.

3.2.3.2 Calibration PATH 2

Calibration PATH 2 is schematized in Fig. 3.17. The PATH 2 calibration exploits the fact that the client is an object with known position in the lab, contrary to the chessboard. In fact from the industrial robots kinematics, it is possible to log the transformation $\mathbf{H_{s0}^{c}}$ of the client center point with respect to the servicer start-up position.

This calibration path is based only on the LiDAR system, and thus still requires the two LiDAR to be calibrated with a CT, in order to obtain a stitched point cloud of the client.

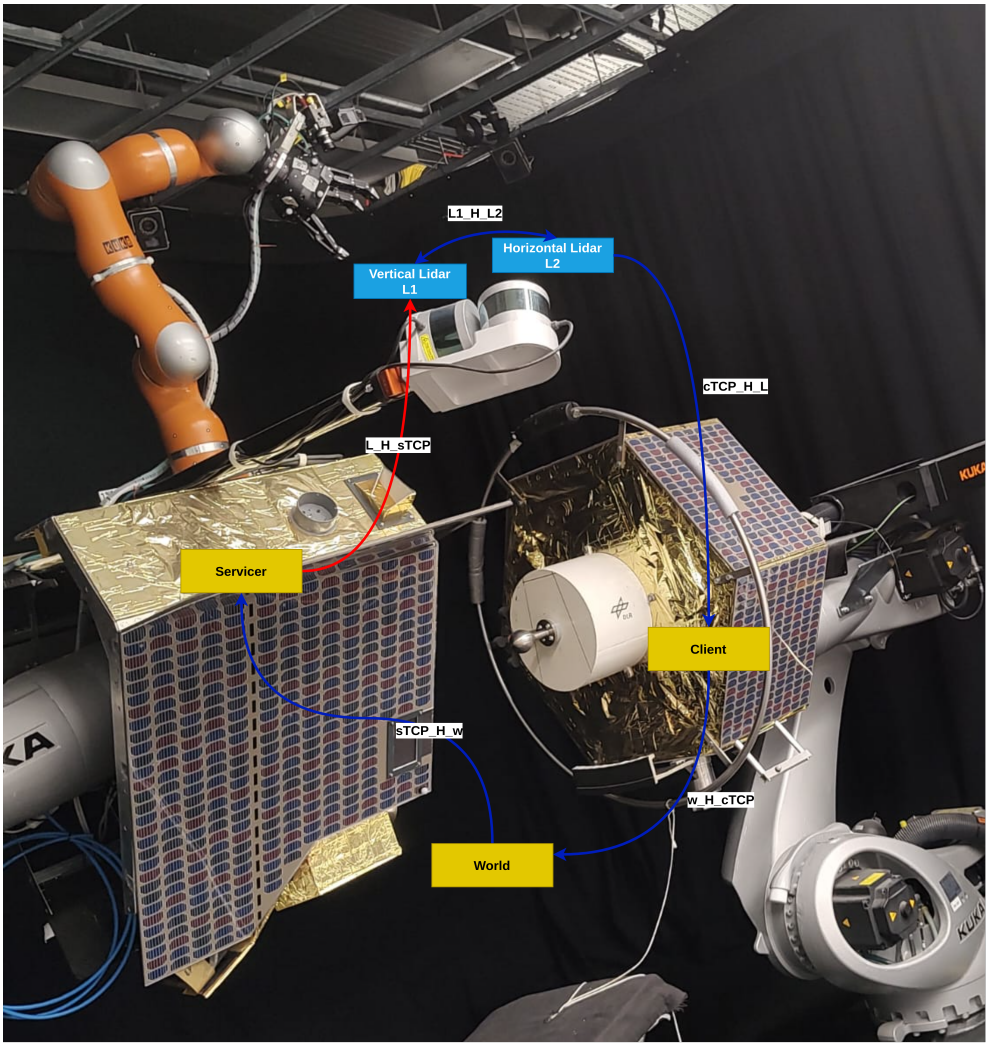


Figure 3.17: Calibration PATH 2.

Given an initial guess, the ICP algorithm can be exploited to find the transformation between LiDAR system and client $\mathbf{H_c^{L1}}$. This is possible because a model of the client is available, even if some error may arise because of differences between the client model and the manufactured client. In order to converge, the ICP needs a good initial guess. This procedure has been automatized, requiring in input the position of 3 known points of the client in the LiDARs scans. Such input needs to be manually provided.

To summarize, the second calibration path can be computed as:

$$\mathbf{H_s^{L1}}\Big|_{PATH2} = \mathbf{H_c^{L1}} \cdot \mathbf{H_{s0}^{c}} \cdot \mathbf{H_s^{s0}} \tag{3.15}$$

In this case, the major error source is expected to be the ICP.

Accuracy computation The translation error e_t and rotation error e_r among the two transformations obtained for $\mathbf{H_s^{L1}}$ are computed as:

$$e_t = ||\mathbf{t_2} - \mathbf{t_1}||^2 \tag{3.16}$$

$$e_r = \text{acos}\left(1 - \frac{\text{trace}(I - \mathbf{R_2}\mathbf{R_1}^{-1})}{2}\right) \tag{3.17}$$

Where $\mathbf{t_i}$ is the translation vector and $\mathbf{R_i}$ the rotation matrix, composing the transformation matrix $\mathbf{H}$.

3.2.3.3 Results

Experiment set-up The calibration target is a metallic chessboard with external dimensions of 88.0 cm width and 63.4 cm height. In order to take static scans, the chessboard was put on a stair and different set-ups have been tested, either tilting the calibration target or the dual-LiDAR system.

LiDAR system calibration results The LiDAR calibration routine successfully finds the transformation among the two sensors. Different set-ups and variations of the algorithm are tested, as reported in Table 3.5.

Table 3.5: LiDAR system calibration results.

Set-up	Outliers removal	Loss function	Optimizer	Final loss [mm]
CT tilted	Y	L1 + L2	LM	0.177465
CT tilted	Y	L1 + L2	BFGS	0.177463
L1-L2 tilted	Y	L1 + L2	LM	0.182658
L1-L2 tilted	Y	L1 + L2	BFGS	0.182651

where LM is the Levenberg-Marquardt optimization routine and BFGS is the Broyden-Fletcher-Goldfarb-Shanno algorithm. In all the successful cases the loss function was including the point clouds of both LiDARs and the outlier removal steps were applied. With respect to the results obtained with synthetic scans in [29], the final loss is one order of magnitude larger. The loss function trend during the optimization is shown in Fig. 3.18 and 3.19, along with the points belonging to the inner and outer set.

It has been observed that the optimization is highly sensitive to the presence of outliers or in-liers not belonging to the chessboard. An example of run

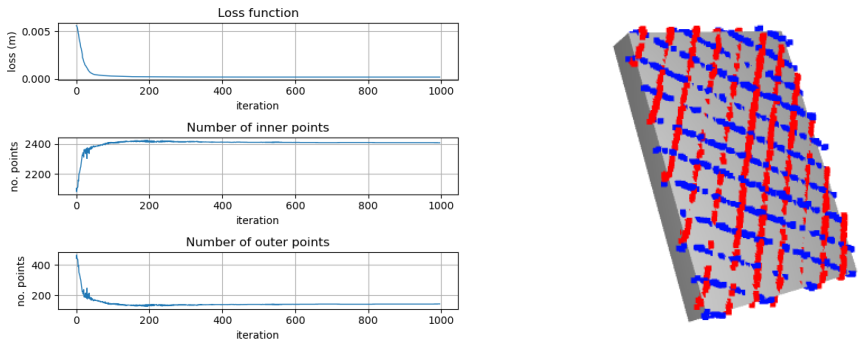


Figure 3.18: Outliers removal - LM optimizer.

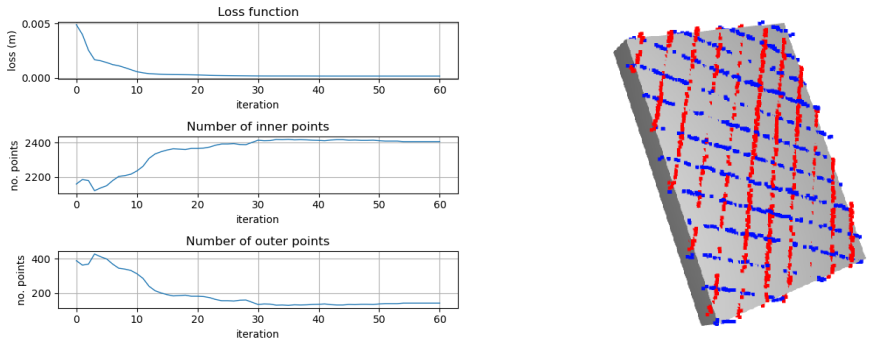


Figure 3.19: Outliers removal - BFGS optimizer.

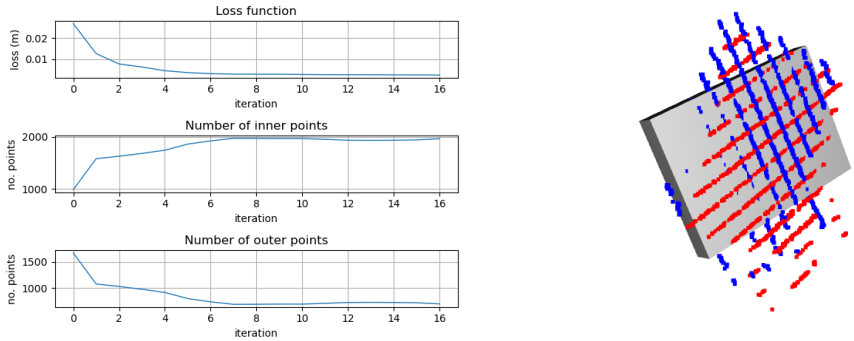
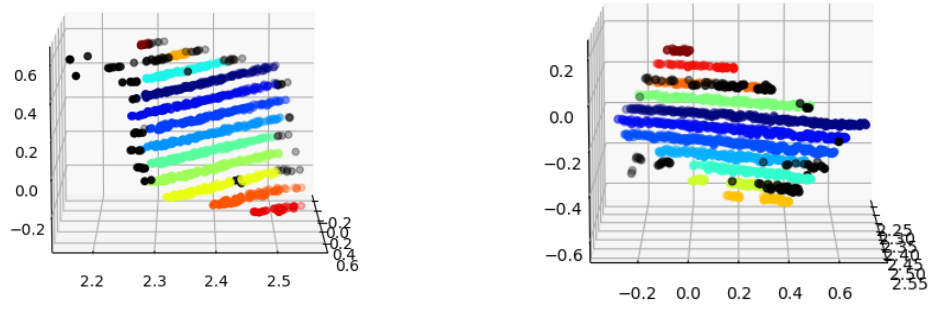


Figure 3.20: Optimization wrong convergence without outliers removal.

performed without the outliers removal is shown in Fig. 3.20. In this case, the optimizer converges to the wrong solution.

The edges detection is fundamental for an accurate result, as the loss function computation is also based on the identification of points sets that are inside and outside the board edges. Therefore, as best practice, it is suggested to take care in the experimental set-up preparation to the positioning of the chessboard. In fact, if the chessboard plane lies on the exact same plane of another object it can be hard for the clustering step to correctly perform the segmentation on edge points. As it can be seen in Fig. 3.21, also for a human eye in this case it is impossible to properly distinguish the chessboard points. It has been found that, as best practice, the calibration board shall not be lying on a plane coincident with the one of the background stair, otherwise its segmentation from the ambient point cloud will be harsh.



(a) Correct clustering.

(b) Incorrect clustering.

Figure 3.21: Chessboard segmentation sensitivity to the set-up preparation.

LiDARs-OOS-SIM calibration results The error among $\mathbf{H_S^{L1}}\Big|_{PATH1}$ and $\mathbf{H_S^{L1}}\Big|_{PATH2}$ is reported in Table 3.6. The achieved accuracy is lower than the pose estimation requirement, thus the result can be considered successful for generating a ground truth in the lab. It is worth mentioning that the achieved accuracy is comparable to the accuracy obtained with the real LiDAR data on the ATV-5 experiment, which is 30 cm at 20 m and 5 cm at docking [68].

Table 3.6: Calibration accuracy.

	e_t [cm]	e_r [deg]
Calibration accuracy	2.98	1.44
Pose estimation req	5	2

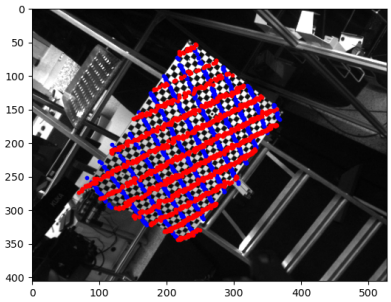


Figure 3.22: Left image.

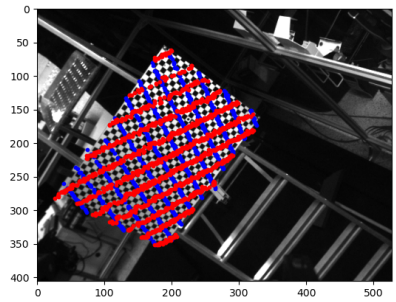


Figure 3.23: Right image.

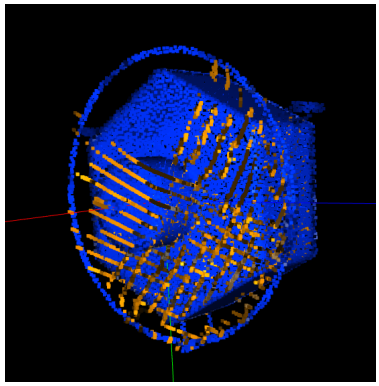


Figure 3.24: Calibration path 1.

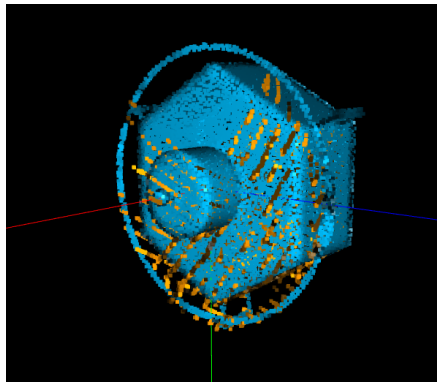


Figure 3.25: Calibration path 2.

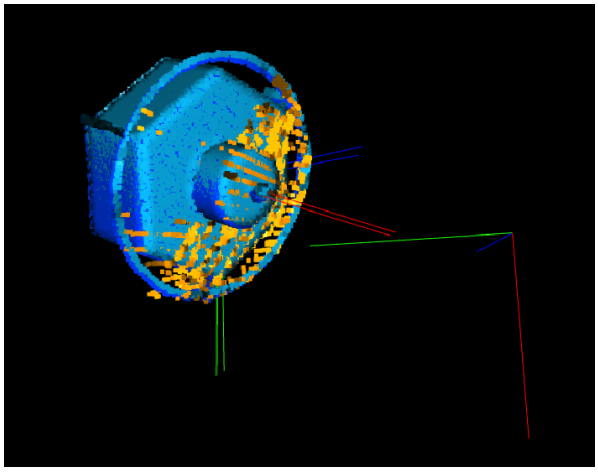


Figure 3.26: Error between the two calibration paths.

3.3 Data validation and comparison

In this Section, the data (images and scans) generated synthetically in Section 3.1 are compared with the ones acquired in the GNC facilities as explained in Section 3.2 or with real missions data. The comparison is presented in order for the three EO sensors: visible cameras, thermal cameras and scanning LiDAR.

3.3.1 Visible images results

The comparison and cross-validation of visible images has been carried out at PoliMi's ARGOS. After the ARGOS facility calibration, an images acquisition campaign has been conducted for the two following scenarios.

1. **Lunar landing**. The scenario is based on a descent trajectory targeting the Lunar South Pole. In order to correctly simulate it, some requirements must be taken into account to set-up the facility to acquire a realistic sequence of camera images. Therefore, the facility must be able to reproduce the conditions listed in Table 3.7.

Table 3.7: List of the facility requirements for the Moon scenario.

Variable	Range
Altitude	3-100 km
Attitude pitch	0°-20°
Sun Elevation angle	0°-90°
Sun Azimuth angle	0°-360°

2. **Debris inspection**. The second scenario studied concerns a relative vision-based navigation system for debris or space object inspection or capture. In particular, the objective is to acquire the image dataset of a trajectory around Envisat. In Table 3.8 the facility requirements for the Envisat inspection are listed.

Table 3.8: List of the facility requirements for the Envisat scenario.

Variable	Range
Chaser relative position	0.5-50 m
Camera attitude pitch/yaw angle	±90°
Sunlight direction azimuth angle	0°-360°
Target attitude angle	0°-360°

Images cross-validation and comparison metrics To verify the accuracy of the calibrated facility, images taken in ARGOS are compared with the synthetic images generated by a rendering software with the same pose. The validation aim is to cross-verify the reliability of laboratory and synthetic images by means of quantitative indices, comparing them for the two scenarios. In particular, the synthetic images are generated exploiting the ESA's software Pangu for the Moon landing scenario, whereas the open source Blender suite is used for the debris inspection scenario.

For each scenario, the validation set is made of 10 images spanning the dataset ranges in terms of illumination and poses.

Four indications are examined to evaluate the facility images goodness:

1. **Image histogram**. The histograms information is a low level information which gives a good representation of the image content. Such method has been already used to evaluate images quality for testing of space navigation algorithms [69].

2. **Shadow index**. The synthetic and laboratory images are thresholded to identify shadows. The value of the threshold is identified automatically using the Otsu algorithm. The Otsu method is a deterministic and automatic way to discriminate shadowy and illuminated target parts. Then, the two resulting binary images are subtracted to obtain a shadow disparity map. The accuracy of the shadow representation, which can be considered as representative of the accuracy of the general shape of the sample, is evaluated by a scalar shadow index (J_s), defined as:

$$J_s = 1 - \frac{D_s}{S_{real}} \tag{3.18}$$

   Where D_s is the sum of the disparity map and S_{real} is the sum of the pixels classified as shadow in the real image. J_s expresses the fraction of pixels in shadows correctly reproduced in the synthetic model.

3. **Contrast index**. A second index is then identified. For both images, the real and the synthetic, an illumination ratio RI is identified as:

$$R_I = \frac{I_L}{I_s} \tag{3.19}$$

   Where I_L is the mean intensity of the pixel classified as in light, and I_S is the mean intensity of the pixel classified as in shadow. Then, the contrast index J_c is defined as:

$$J_c = \frac{R_{Ireal}}{R_{Irend}} \tag{3.20}$$

4. **Features quality index**. Typical navigation algorithms rely on feature extraction steps, thus a comparison among real and synthetic images is considered a good indication of the similarity of behaviour among the two. The Feature Quality Index (FQI) indicates the similarity of features extracted in two corresponding frames (a real and a synthetic one) and it is defined as:

$$FQI = 1 - \frac{\mu(H_d)}{H_{d,max}} \tag{3.21}$$

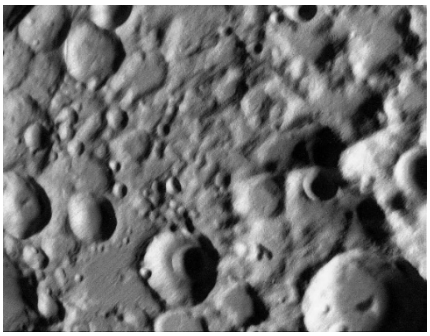
where H_d is the Hamming distance between two corresponding features descriptors and $H_{d,max}$ are the maximum possible hamming distance. The mean value $\mu(H_d)$ is computed on 10 corresponding ORB features [70].

The higher the indexes, the best the image is represented. Requirements to satisfy the validation are based on the scalar shadow index J_s and on the contrast index J_c. In particular, it is required $J_c > 0.90$, $J_s > 0.75$ and $FQI > 0.80$.

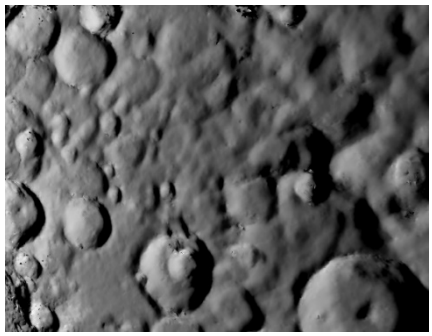
Results of obtained images Hereafter, the images validation results are presented, starting from the Moon landing scenario and then examining the debris inspection case. For the Moon application, the synthetic images have been rendered first with the dense reconstruction of the diorama, and then, after some re-calibration activities, it was possible to use directly the DEM for the diorama manufacturing.

Moon scenario - dense reconstruction results In Fig. 3.27a an image of the Moon diorama taken with the FLIR Chameleon-3 camera is shown; in Fig. 3.27b it is shown the corresponding image generated in Pangu from the reconstructed diorama point cloud. The illumination data are recorded exploiting multiple sundial's measurements. Results show that the validation tool is capable to correctly reproduce the pose and the illumination conditions.

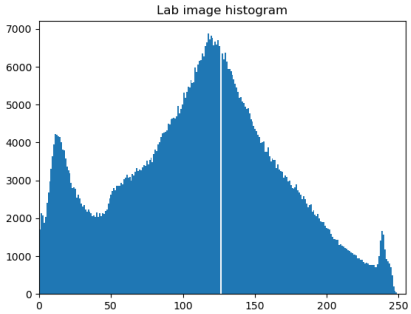
In Fig. 3.27e the disparity between shadow areas is shown. The corresponding index of merit is $J_s = 1 - D_s/S_{real} = 0.82$. As it can be seen, there are still some shadow areas not matched and the requirement is not yet satisfied. The error causes are multiple: the uncertainty in the pose, errors in the determination of the camera intrinsic parameters, the irradiance settings on the point cloud. The black peaks in the synthetic images are mainly due to the presence of holes in the dense point cloud of the facility. A recalibration of the Moon diorama facility has been performed, which solved the problems yielding a successful validation. On the other hand, the contrast index requirement is met, being $J_c = 0.80$, showing a good capability of brightness representation.



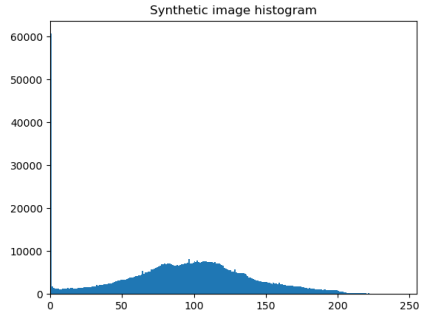
(a) Laboratory image.



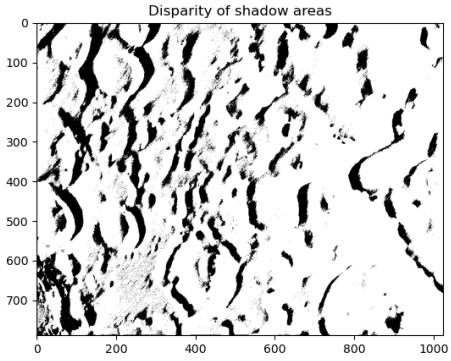
(b) Rendered image.



(c) Laboratory image histogram.



(d) Rendered image histogram.



(e) Disparity map of shadow areas.

Figure 3.27: Comparison and cross-validation of laboratory image and synthetic image, before recalibration activities.

As a further insight on the two images, their histograms are shown in Fig. 3.27c and in Fig. 3.27d. It can be observed that the histogram peak is similar for the two and that in the synthetic image there is the presence of completely black pixels, as expected.

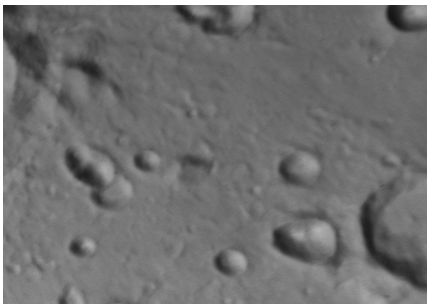
The histograms values are comparable, in particular the central peak. Nevertheless, in the synthetic images a black peak is found, which scales the plots differently. The white peak present in the lab images was due to reflections in the lab, which are aimed to be further minimized. As mentioned, the black peak found in the synthetic image is due to the presence of holes in some region of the diorama mesh used to generate Pangu world.

Moon scenario - manufacturing DEM results The validation campaign aimed at using the robotic facility to acquire real images, to check the adequateness of synthetically generated images. The validation images cover different region of the diorama and are taken with different set-up of the facility illumination. The results of the validation campaign (restricted to 10 images) are summarized in Table 3.9. An example of validated images and corresponding histograms, together with the resulting disparity map is shown in Fig. 3.28.

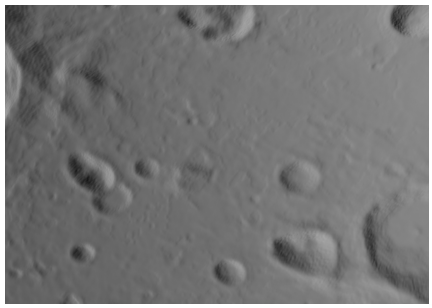
Table 3.9: Summary of validated images and corresponding indices - Moon scenario.

Frame	$\mathbf{J_s > 0.9}$	$\mathbf{J_c > 0.75}$	**Sun [Az, El]**
0	0.92	0.88	[262°, 56°]
1	0.96	0.79	[84°, 60°]
2	0.91	0.82	[84°, 60°]
3	0.91	0.78	[84°, 60°]
4	0.96	0.78	[84°, 60°]
5	0.92	0.82	[86°, 35°]
6	0.91	1.00	[262°, 41°]
7	0.92	0.97	[86°, 26°]
8	0.91	0.80	[86°, 26°]
9	0.91	0.91	[86°, 26°]
10	0.95	0.91	[86°, 26°]

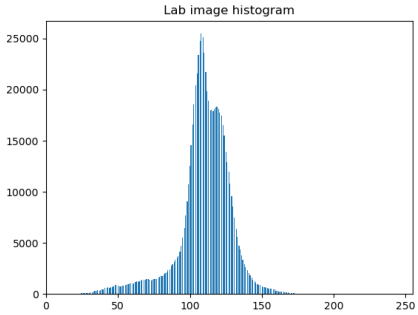
Debris scenario results The images obtained for the debris scenario are here analyzed. Detailed results are shown for a sample pose and illumination. The laboratory image and the render are compared with the extracted ORB features in Fig. 3.31. It can be seen that the pose is correctly retrieved, even if



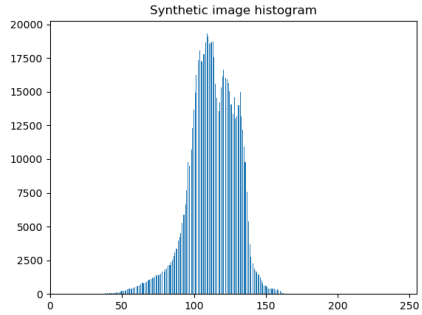
(a) Laboratory image.



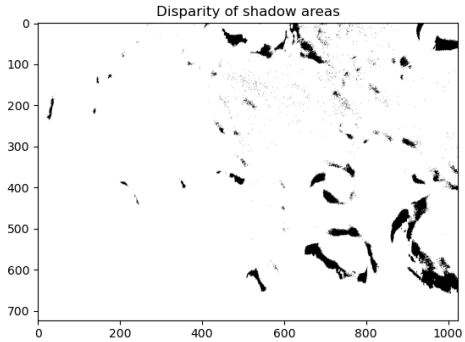
(b) Rendered image.



(c) Laboratory image histogram.



(d) Rendered image histogram.



(e) Disparity map of shadow areas.

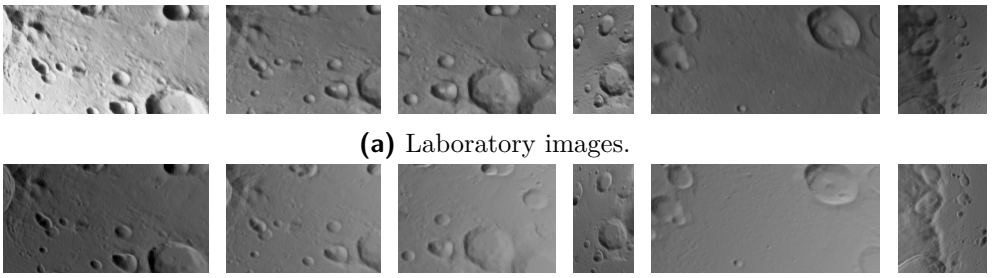
Figure 3.28: Comparison and cross-validation of laboratory image and synthetic image, after recalibration activities.

the rigidity of the mock-up's solar panel has to be improved to match the ideal positioning visible from the synthetic image. The extracted features are very close both in number and location; the features quality index is $FQI = 0.82$. Please note that in Fig. 3.31 the marker in the laboratory image has not been cropped, but that the FQI is computed based on corresponding features only. An high FQI indicates that the ORB descriptors are similar in the two images and thus that the illumination and mock-up materials are realistically reproduced. It has been noticed that, the two images histograms are not exactly matching. The major difference is present for low intensity pixels, and in particular it can be attributed to the diffuse light inevitably present in the facility. The shadow index is $J_s = 0.98$, indicating a shadow distribution very close to the one obtained with the rendered ground truth.

The indices relevant for an artificial target are reported in Table 3.8 for 10 validation images. The images are shown in Fig. 3.30 and correspond to different poses and illumination conditions. The validation campaign can be considered successfully concluded, even if some differences among the two kinds of images could not be completely eliminated.

Table 3.10: Summary of validated images and corresponding indices - Debris scenario.

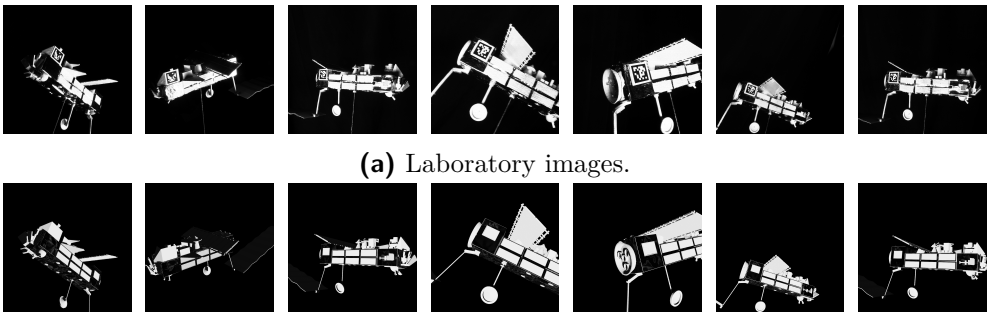
Frame	$\mathbf{J_s > 0.9}$	**FQI > 0.8**	**Sun [Az, El]**
0	0.97	0.82	[318°, -28°]
1	0.95	0.83	[120°, -63°]
2	0.91	0.80	[120°, -63°]
3	0.94	0.82	[120°, -63°]
4	0.96	0.80	[120°, -63°]
5	0.94	0.82	[120°, -63°]
6	0.96	0.85	[120°, -63°]
7	0.92	0.82	[120°, -63°]
8	0.94	0.84	[334°, -17°]
9	0.98	0.81	[334°, -17°]
10	0.97	0.82	[334°, -17°]



(a) Laboratory images.

(b) Synthetic images.

Figure 3.29: Images from Moon validation set (frames 1 to 6).



(a) Laboratory images.

(b) Synthetic images.

Figure 3.30: Images from Debris validation set (frames 1 to 7).

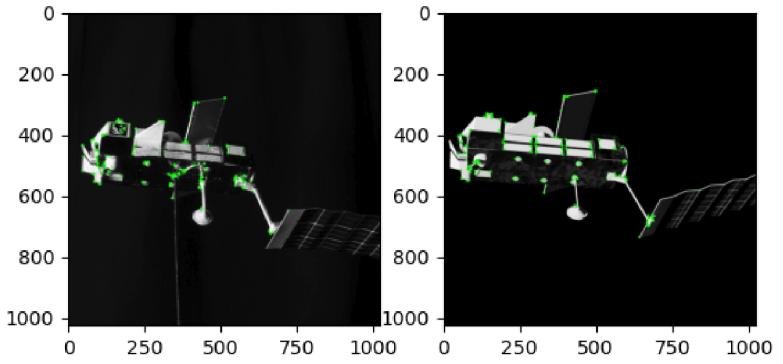


Figure 3.31: Example of features extraction comparison.

3.3.2 Thermal images results

Validation method Concerning thermal images, the validation process is carried benchmarking synthetic images rendered with the ADE tool (see Section 3.1), with a sequence of real images from the Hayabusa 2 Science Data Archive [71]. In particular, Hayabusa 2 TIR 20180720-071632-l1 to 20180720-150217-l1 47 images are selected, since the distance from target is adequate for the application case considered in this work. Spice kernels are used to retrieve the Sun position, the TIR camera position and orientation in the Ryugu-fixed reference frame.

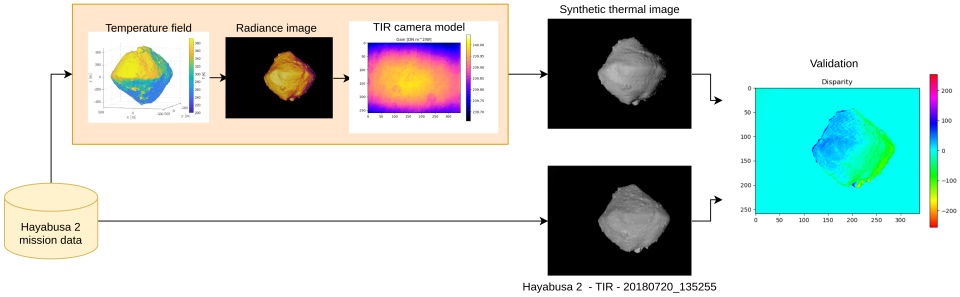


Figure 3.32: Synthetic thermal images validation scheme.

In Fig. 3.32 the validation process is schematized. For each image, the thermal simulation is run to derive the corresponding temperature field, under the assumption of a stationary thermal behaviour, thus considering to reach the thermal equilibrium after the transient, for each time instant. The considered asteroid 162173 Ryugu's thermophysical properties are reported in Table 3.11 [72]. The asteroid shape model used is available from Hayabusa 2 mission data [71].

Table 3.11: Ryugu's thermophysical properties [72].

Parameter	Symbol	Value	Unit
Emissivity	ϵ	0.9	-
Bond albedo	a	0.0146	-
Thermal inertia	Γ	225	$Js^{1/2}K^{-1}m^{-2}$
Density	ρ	1192.8	kgm^{-3}
Heat capacity	c_P	600	JK^{-1}

Then, from the temperature field, a radiance image is obtained. The TIR camera model is based on Hayabusa 2 Thermal Infrared imager, whose characteristics are reported in Table 3.12.

Table 3.12: Cameras characteristics, from [14].

TIR	
Array size	344 x 260
FoV	16.7° x 12.7°
NETD	0.4 K
Bit depth	12
Detectable temperature	150 - 460 K
Spectral band	8 - 12 μm

Overall, the disparity error is quantified by means of a disparity index, which has been defined as the absolute difference between pixel intensities in the two images, only considering the Region Of Interest (ROI), i.e. the asteroid.

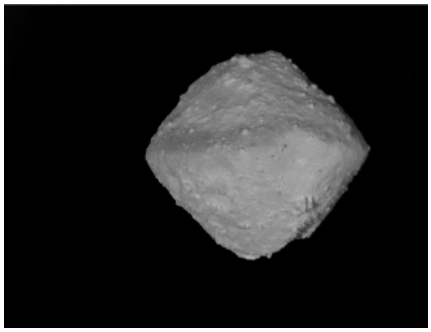
$$J_D = |I_{real} - I_{synth}|_{ROI} \tag{3.22}$$

Such index is a low level measure of the difference among the images.

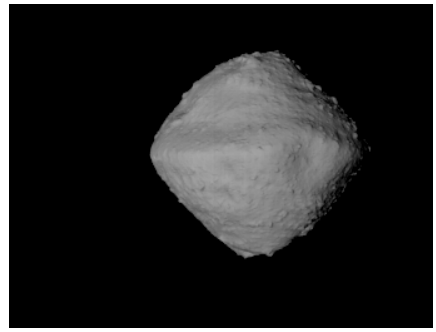
Validation results The first real image of the sequence can be seen in Fig. 3.33a and its correspondent synthetic image in fig. 3.33b. The disparity image $|I_{real} - I_{synth}|$ among the two is shown in Fig. 3.33e. The images histograms are shown in Fig, 3.33c and 3.33d.

The disparity image is useful to qualitatively understand the error sources. On a macroscopic basis, the disparity is tending to zero, showing that the ADE tool can successfully reproduce the mean brightness of the body and represent the major image features. Nevertheless, low level details are not always well represented. For instance, some rocks on the surface appear to be brighter in the real images. The main reason is that a uniform emissivity has been assumed for the whole body, but Hayabusa 2 studies [72] highlighted that the surface of the asteroid presents different materials, with different emissivity values. Such error is considered acceptable here, as its entity is contained and the ADE tool purpose is to be a general tool. A fine hand tailoring of the emissivity value would be necessary to reproduce exactly the same materials that are present on Ryugu's surface. As second option, a random emissivity map can be applied on the shape model. Another error visible in the disparity image, is instead related to a gradient in the longitudinal direction. The origin of this error is instead associated to the hypotheses performed in the thermal simulation. The ADE tool is compatible with OpenFoam, and a future extension will include more complex thermal analyses.

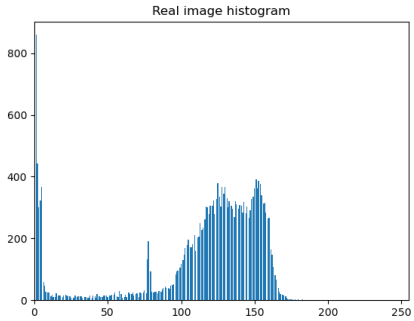
The mean JD_μ and standard deviation JD_{std} in the ROI have been computed for J_D. Results for the whole sequence are shown in Fig. 3.33f: JD_μ is in average the 2.83% of the gray levels while JD_{std} is the 7.05%.



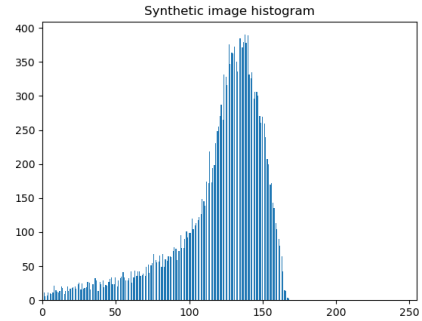
(a) Real image.



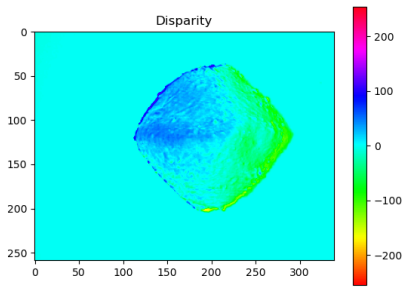
(b) Rendered image.



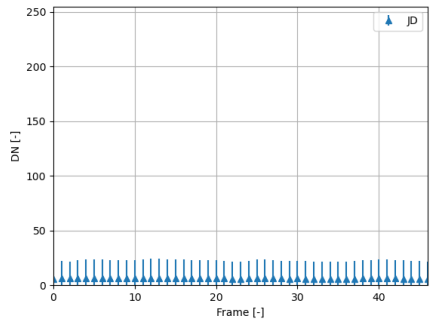
(c) Real image histogram.



(d) Rendered image histogram.



(e) Disparity image.



(f) JD index for the whole sequence.

Figure 3.33: Comparison of synthetic and real thermal images belonging to the 20180720-071632-l1 to 20180720-150217-l1 image sequence.

3.3.3 LiDAR data results

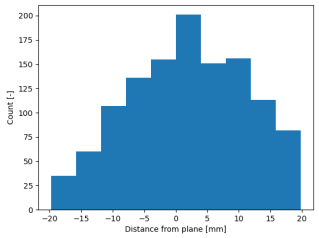
A first characterization of the scanning LiDAR point cloud is performed, thanks to the data gathered in the OOS-SIM.

Observing the appearance of the LiDAR's point cloud of a flat plate, it has been noticed that the major noise component is in the laser firing direction. This is easily explainable considering the working principle of the sensor.

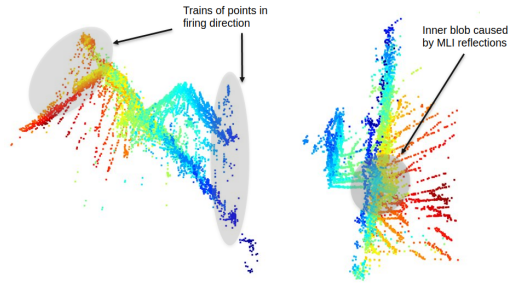
The noise of LiDAR point cloud has been quantified using the plate point cloud and considering the distance of the points from the plane model computed with the RANSAC algorithm. The point-plane distances distribution is shown in Fig. 3.34, being a Gaussian with mean of 0.2863 mm and standard deviation of 8.693 mm. This characterization can be exploited to generate noisy synthetic scans.

On the other hand, observing the real point cloud of the client satellite, other kinds of noisy features were observed. Such features can not be reproduced with a simulator in a straightforward way, being:

- massive blobs of points inside the client, caused by MLI reflections in proximity of the borders of the client central cylinder;
- train of points when the impinging light is tangent to the surface, observed especially on the external ring which is non-planar and thin with respect to the vertical grid of laser firings;
- noise components in directions other than the firing one, due to random reflections on the surface, which sometimes cause a distortion of the grid.



(a) Point-plane distance distribution on the plate.



(b) Noisy features in the client satellite point cloud.

Figure 3.34: LiDAR noise characterization.

Such considerations are a first starting point for filling the gap between real and synthetic point clouds.

CHAPTER 4

Visible imaging navigation and mapping systems

4.1 Far range relative navigation

4.1.1 Centralized relative navigation of multiple far-range CubeSats

The problem of performing a centralized autonomous navigation algorithm able to reconstruct the trajectories of a fleet of CubeSats relative to an asteroid binary system is here studied.

The algorithm has to be executed on-board the main spacecraft, which takes relative measurements using a narrow optical camera. The image processing algorithm goal is to detect and track the CubeSats, deriving Line of Sight (LoS) measurements. Such measurements can be exploited into a navigation filter to obtain the CubeSats position relative to the asteroid.

The optical navigation capability of the mothercraft is exploited to detect and track the CubeSats, deriving LoS measurements. In particular, the following challenges are tackled:

- to exploit only low exposure images that are already taken for the mothercraft's own navigation;
- to detect and distinguish far-range objects in presence of starry background and large natural celestial bodies in the FoV.

Depending on the target distance, characteristic dimension and camera characteristics, different IP techniques can be adopted. In Fig. 4.1, the image portion occupied by a CubeSat target with characteristic dimension of 60 cm is displayed at different distances for images taken with a narrow angle camera. When the CubeSat is close to the mothercraft, algorithms for pose estimation can be employed, from which the LoS measurement can be derived. Considering that the pixels occupied by the object should be at least of the order of 50-100 px to estimate the pose, the employment of such strategy is limited only to very close range distances. Such close-range distances are of the order of 50 m, thus likely coinciding with a very reduced mission phase, e.g. the CubeSats release. For the mid and far range distances, it is possible only to employ blob detection algorithms to safely compute the LoS.

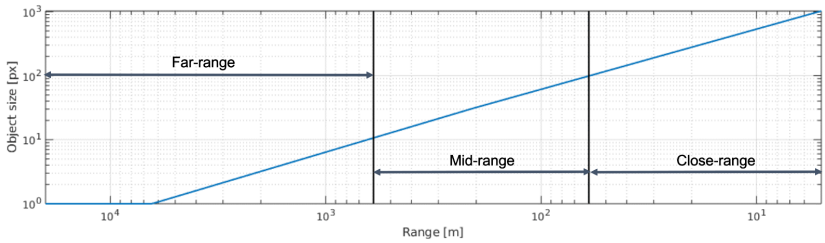


Figure 4.1: CubeSats pixel size at possible mission ranges.

Additional challenges related to the CubeSats detection and identification are the presence of other point objects in the image (i.e. stars or other CubeSats).

4.1.1.1 Mission scenario

Scenario description For the evaluation of the far-range techniques, without loss of generality, the Hera mission scenario is considered [11]. The main spacecraft (Hera) will be executing a set of hyperbolic passages around the binary system 65803 Didymos. A 6U CubeSat (CubeSat1) will be orbiting closer to the system, exploiting the three-body problem equilibrium points of the binary gravity field. Another 6U CubeSat (CubeSat2), will be orbiting

around the whole binary system, in a Self-Stabilizing Terminator Orbit (SSTO) whose period is such that an integer number of orbits are completed within one week [73].

For the subsequent analyses here presented, the specific orbits derived in [74] are taken (see Fig. 4.2):

1. Hera's trajectory has been selected from the planned hyperbolic passages sequence and has been propagated with a Keplerian + solar radiation pressure (SRP) dynamical model.
2. CubeSat1 is located to a naturally stable orbit, centered on the Lagrangian point L4 and leveraging the binary system dynamics, known as Short Period Orbit (SPO). The resulting orbit has a bounded, non-periodic motion with a synodic period of 12 hours and 20 minutes approximately.
3. CubeSat2 is placed in a SSTO, a trajectory naturally stabilized by the effect of the SRP, with a radius of 3.3 km and a period of 56 hours. Due to the geometry of the system, the orbit is nearly perpendicular to the binary system's orbital plane, thus providing good coverage of the whole system.

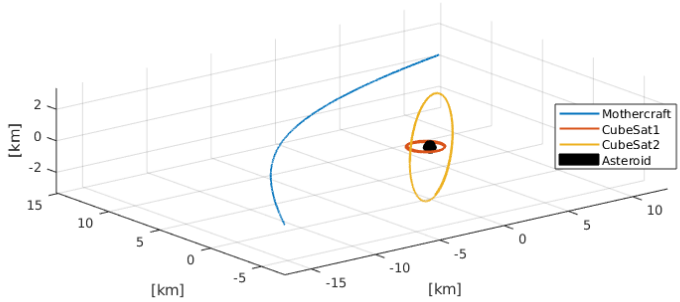


Figure 4.2: Mission scenario under study.

Electro-optical sensor characteristics Hera visual sensor and its parameters are essential for the study of a centralized vision-based navigation. The Asteroid Framing Camera (AFC) characteristics of interest are reported in Tab. 4.1, as from [61]. Such camera model is used to generate realistic synthetic images in Pangu. In operating conditions the AFC is affected by a dark noise with rate $N_{dark} = 0.03$ DN/s and a read-out noise with standard deviation $N_{read} = 2$ DN. This paper wants to investigate the possibility of exploiting images already taken by Hera for its own navigation. Therefore, images are

acquired with a low exposure time $t_{exp} = 0.1$ s, necessary to avoid smear and saturation, since Didymos is the main target [11].

Table 4.1: AFC characteristics.

Parameter	Value
Field Of View	FOV = 5.5°
Aperture diameter	$d_a = 20$ mm
Focal length	$f = 150$ mm
Pixels	1024x1024
Bit Depth	$BD = 14$
Analog chain gain	$G = 18$ electrons/DN
Quantum Efficiency (max)	$QE_{max} = 0.2$
Point Spread Function (PSF) diameter	$d_{PSF} = 1.7$ px

4.1.1.2 IP alternatives definition for trade-off

In this section, the possible architectures for the IP algorithm are defined.

Prior information availability Two different general approaches can be adopted for the IP algorithm:

1. the IP assumes *a-priori* information from a navigation filter;
2. the IP does not rely on *a-priori* information from the navigation filter.

The first possibility is to assume to have the filter state feedback. In this case, no lost-in-space detection is performed: the IP takes as input the CubeSats image coordinates estimated by the filter and their covariance. This allows to crop a region of interest in the image where the CubeSats detection and identification can be performed. The IP outputs are the measured image coordinates of the targets, that are fed into the filter.

The second possibility is not to rely on filter feedback. The objective of the IP algorithm here is to find the CubeSats in a lost-in-space scenario, where no information comes from the filter. It takes in input two subsequent images and its output is the CubeSats position in image coordinates or a not-found message. Within the image pre-processing block, asteroids are masked and the centroids corresponding to the other celestial and artificial objects is computed; then such points are characterized as features and matched from one image to the other. Finally, disparity is computed and the candidate points are screened, identifying the CubeSats.

Centroid computation For the blob centroid computation different techniques are available, e.g. developed for star trackers [20]. The *Moment* computation is the simplest and computationally efficient technique. This basic technique is used to find the target object center (x_c, y_c) in the image I and can be used to provide a LoS measurement. The simplest method is the computation of the momentum:

$$x_c = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m} I(i,j)i}{\sum_{i=1}^{n}\sum_{j=1}^{m} I(i,j)} \tag{4.1}$$

$$y_c = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m} I(i,j)j}{\sum_{i=1}^{n}\sum_{j=1}^{m} I(i,j)} \tag{4.2}$$

The momentum method is very simple as robust, thus suitable for the implementation of a light on-board algorithm. On the other hand, *Gaussian Fit* can achieve a larger accuracy at the price of a larger computational effort [75]. This method can be employed when the target object is smaller than the detector resolution. The object appearance is mainly related to the optics and stray light effects that determine the Point Spread Function (PSF) of the camera. The method consists in fitting a 2D gaussian distribution with a linear least square based algorithm:

$$I(x,y) = \frac{t_{exp}}{2\pi\sigma_x\sigma_y} e^{-\frac{m0-m}{2.5}} e^{-\frac{(x-x_c)^2}{2\sigma_x^2}} e^{-\frac{(y-y_c)^2}{2\sigma_y^2}} \tag{4.3}$$

where t_{exp} is the exposure time, $m_0 = -1.5$ is the visual magnitude of Sirius and σ_x, σ_y is the standard deviation of the camera PSF. Its main applications of interest are star trackers, as it exploits the known magnitude of the object and allows centroiding with sub-pixel accuracy.

The *Moment* algorithm is selected in virtue of its simplicity and robustness, considering also that the main challenge of the IP is related to the CubeSat identification among the detected blobs rather than to the centroid accuracy.

Features Extraction (FE) Each candidate needs to be matched with the corresponding one in the previous image. In this process it is essential to have points in the center of highly variable area, which helps in the matching process. Well defined points are found in the presented scenario (i.e. stars, CubeSats), but they are all placed over a background poor of features. Typical features used for stars matching are based on geometric patterns or brightness information. Therefore they are the ones considered here.

1. **FE1: Point coordinates** With the scenario image acquisition frequency, the disparity between the images is small. Therefore an option is simply to consider as a feature the points coordinates in the two images.

2. **FE2: Triangular Feature** For each point the two closest ones are considered, forming the three of them a triangle, as shown in Fig 4.3. Parameters used by Liebe for stars matching are the angles from the star to the closest two neighbors and the inner angle of the triangle [76].

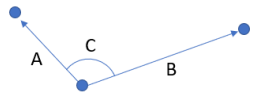


Figure 4.3: Triangular Feature A,B,C: A,B = angular distance C = inner angle.

3. **FE3: Triangular feature + brightness** Geometry may be different because the CubSats are moving with respect to the stars. Therefore brightness information should increase their detectability. Brightness is therefore considered as additional parameter. As brightness parameter the brightest pixel of the blob can be taken, or the mean of surrounding pixels.

Outliers Identification (OI) After the disparity computation the candidate points corresponding to CubeSat1 and CubeSat2 need to be selected. While stars movement is due to the movement of Hera, the CubeSats disparity in the image is the result of their relative motion with respect to the mothercraft. If a sufficient number of stars is detected, CubeSats can be recognized as outlier points, having a different disparity.

1. **OI1: Median** The point is considered an outlier if its deviation from the median is larger than three standard deviations.
2. **OI2: GESD** The Generalized Extreme Studentized Test is performed to verify if a point is an outlier or not. Such test is useful when the number of outliers is not known a priori, therefore it is suited to the present scenario. [77].
3. **OI3: Mean** The point is considered an outlier when the distance of its disparity from the mean is more than a certain threshold. Such threshold is a number of pixels chosen depending on the image acquisition frequency, given the orbital dynamics scenario. When an outlier is found, it is removed and other outliers are searched for with an iterative procedure.

4.1.1.3 CubeSats' Visibility analysis

Given the orbits in Fig. 4.2, the CubeSats dimension is always less than the pixel resolution.

An analysis is performed to assess when the CubeSats are visible. Four geometric constraints that affect the CubeSats detection are taken into account:

1. the target is in the camera Field Of View (FOV);
2. the target is not overlapped to Didymain or behind the body;
3. the target is not overlapped to Dimorphos or behind the body;
4. the target is not in Didymain or Dimorphos shadow.

Please note that the overlap with a celestial body makes the target impossible to detect because of its small dimension in pixels.

Results are reported in Tab. 4.2 for both CubeSat2 and CubeSat1. In Fig. 4.4 and 4.5 Hera orbit traits are highlighted when the targets are visible. It can be observed that CubeSat1 is not visible mainly because it lies on Didymos orbital plane as Hera, therefore it is often covered by Didymain. It is visible in a discontinuous manner. While CubeSat2 is almost always outside the FOV and the trait in which it is visible is unique.

Table 4.2: CubeSats Visibility.

Cubesat	**Contraint**	**Visibility**
CubeSat1	1	35.08 %
CubeSat1	2	78.39 %
CubeSat1	3	100 %
CubeSat1	4	85.58 %
CubeSat1	all	9.17 %
CubeSat2	1	4.58 %
CubeSat2	2	100 %
CubeSat2	3	100 %
CubeSat2	4	100 %
CubeSat2	all	4.58 %

From the geometric constraints analysis, it is clear that the overall algorithm performance will be seriously affected by the targets' low visibility and occasional measurements provided by the IP. Such an outcome is highly dependent on the reference hyperbolic arc of Hera. The reference arc is selected based on the trajectory analyzed in [11]. An out-of-plane arc is expected to be beneficial with respect to geometrical constraints mentioned here. Nevertheless, the impact of such constraints on the navigation needs to be evaluated through extensive tests.

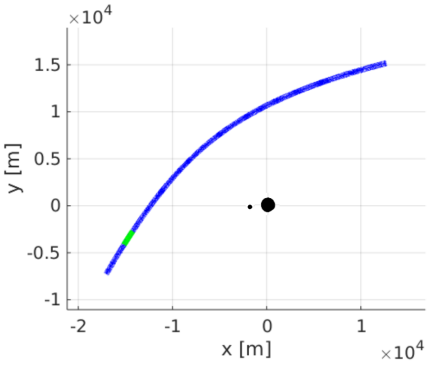


Figure 4.4: CubeSat2 visibility during Hera hyperbolic trajectory (green = visible, blue = not visible)

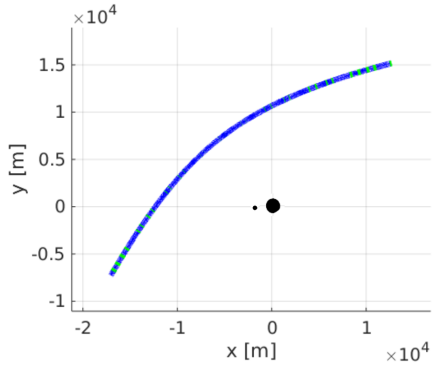


Figure 4.5: CubeSat1 visibility during Hera hyperbolic trajectory (green = visible, blue = not visible)

4.1.1.4 IP alternatives trade-off results

For the algorithm alternatives trade-off, a simulation is carried-out over the first 1000 measurements taken along Hera hyperbola. The measurements are acquired every 60 s. The images in which the CubeSats are visible are 239. Pangu generated images are used for the simulation, considering only CubeSats, stars, sensor model and noise.

The IP architecture trade-off is reported in Tab. 4.3. The accuracy of the two strategies is similar and the number of false positive (FP) and false negative (FN) are comparable. The first alternative is computationally lighter than the second: simulating one Hera hyperbolic arc, the mean CPU time dedicated to one step of the IP with filter feedback is 0.024 s, while without feedback it is 0.130 s. The simulation has been run on an Intel(R) Core(TM) i7-10510U CPU @ 1.80GHz. The preliminary analyses in the sections above have pointed out that the CubeSats visibility is low, so the IP algorithm measurements provided to the filter are scarce. Hence, in case of assuming preliminary info from the filter, if the filter diverges the IP will not be able to provide any measurement, causing a failure of the navigation architecture. Therefore, in virtue of its robustness, the IP without filter feedback is selected as baseline.

Table 4.3: IP main architecture trade-off

Alternative	Accuracy	Robustness	Cost
IP with filter feedback	Medium	Low	Low
IP without filter feedback	Medium	High	Medium

For a lost-in-space scenario the algorithm robustness is critical. In particular, two criteria are selected to evaluate the other alternatives:

1. **False Positives (FP)**. False positives need to be avoided, as providing a wrong measurement to the filter will cause its divergence, compromising the navigation success. Therefore the percentage of false positives in sample cases needs to be minimized.

2. **False Negatives (FN)** False negatives would miss the opportunity of detecting the CubeSat when it is in view. Therefore the percentage of false negatives in sample cases needs to be minimized.

Table 4.4 shows the results for the features extraction trade-off. For the success of the algorithm it is essential to have enough stars correctly detected and matched, in order to have a term of comparison for CubeSats disparities. The best performance is obtained with FE2 and FE3 for both FP and FN. Between the two FE2 is selected as a baseline, since geometrical information is sufficient and brightness adds no relevant information. Some FP are due to the fact that the satellite center of mass is not inside the fov, but the PSF is actually visible.

Table 4.4: Features extraction trade-off

Alternative	FN	FP
FE1 - image coordinates	50	145
FE2 - triangular features	27	37
FE3 - triangular features and brightness	27	37

Table 4.5: Outlier identification trade-off

Alternative	FN	FP
OI1 - median	26	1049
OI2 - gesd	20	466
OI3 - mean	27	37

Table 4.5 shows the results for the outlier identification trade-off. OI2 and OI3 largely exceed the FP number of OI3. Therefore the mean method is selected as baseline.

4.1.1.5 Baseline description

The detailed IP baseline design resulting from the trade-off is shown in Fig. 4.6.

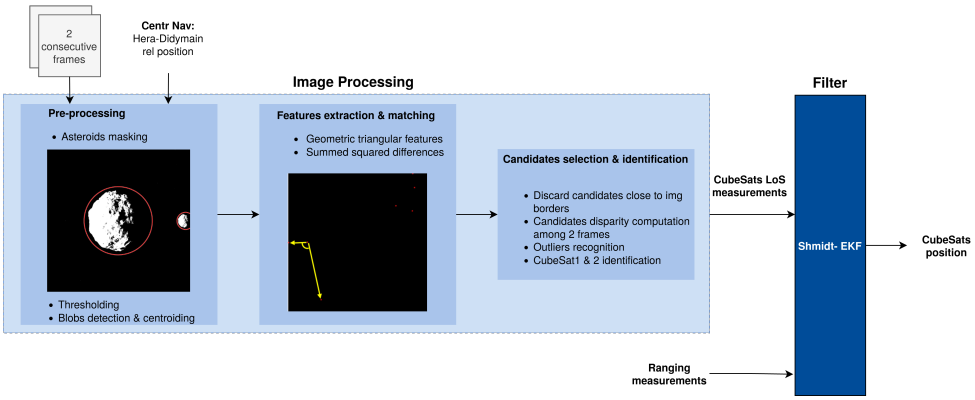


Figure 4.6: IP baseline algorithm

Masking The asteroids masking is performed as first step in the image pre-processing. In particular, Didymain position is assumed known from the mothercraft central navigation, while Dimorphos position is not considered known. Dimorphos is assumed on a circular orbit around Didymain, allowing a coarse knowledge of its position. The error in pixels between the two cases is shown in Fig. 4.7 and can reach about 80 pixels. According to this analysis, to establish the margin, the uncertainty on Dimorphos position is assumed to be equal to the body radius.

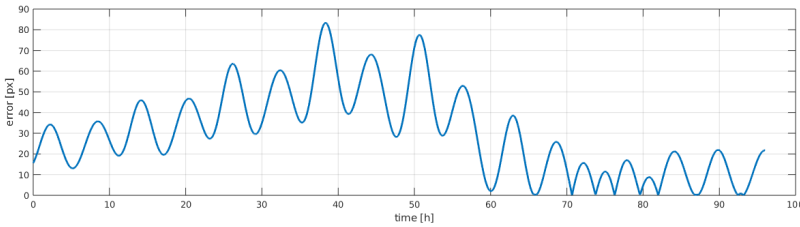


Figure 4.7: Error in Dimorphos assumed position, between circular and elliptical orbit.

Threshold and centroids Afterwards, a threshold is applied to the image to remove background noise. The threshold is tuned according to the present scenario. The result is a binary image where only stars and CubeSats bright enough are visible. Their centroids are computed as center of the detected blobs. Such points constitute the pool of candidates.

Features extraction and matching Each candidate needs to be matched with the corresponding one in the previous image. Triangular features are extracted to associate the stars and CubeSats geometric pattern in the two

frames. For each point the two closest ones are considered, forming the three of them a triangle, as shown in Fig 4.3. Parameters used by Liebe for stars matching are the angles from the star to the closest two neighbours and the inner angle of the triangle [76]. Features are matched with a Sum Squared Differences metric, in order to associate candidate points of the two images and understand how they moved from one frame to the other.

Candidate points on the image border Feature matching is necessary for associating candidate points of the two images and to understand how they moved from one frame to the other. It is possible that a candidate in one image is not visible in the subsequent one; in this case the algorithm shall recognize that the candidate is no longer present in the image. Triangular features are identified by searching for the two closest points for each point. When a point appears or disappears in one image, being near to the image border, the pattern can not be correctly matched in the two frames, as the closest points will be different. As an additional check in the candidate selection process, centroids within a few pixels of the image border are removed from the candidate pool, because they are likely not found in subsequent images.

Disparity and outliers identification The disparity of the matched candidates is computed. The stars shall have a similar disparity, as the relative motion between camera and fixed stars is due only to Hera trajectory and pointing. A candidate is selected if its disparity differs from the mean more than a threshold. Such threshold is a number of pixels chosen depending on the image acquisition frequency, given the orbital dynamics scenario. When an outlier is found, it is removed and other outliers are searched for with an iterative procedure.

Distinguish CubeSat1 and CubeSat2 Once candidates have been screened and outliers found, it is necessary to distinguish between CubeSat1 and CubeSat2. The knowledge of CubeSat disparity can be exploited to distinguish among them. The criteria used are the following:

1. a candidate is recognized as CubeSat1 if its disparity, which has a direction similar to the CubeSat velocity, is mainly horizontal and if the CubeSat is close to the asteroid plane;

2. a candidate is recognized as CubeSat2 if it is not CubeSat1 and its velocity is mainly vertical, as it should be during visibility periods from the scenario preliminary analysis.

4.1.2 Results of centralized relative navigation

4.1.2.1 Centralized navigation performance

The IP algorithm is tested on the reference scenario using Pangu synthetic images. The performance is evaluated in terms of FP, FN and pixel error for the reference nominal scenario.

The vision-based navigation algorithm has been evaluated in the reference scenario. Representative images of the AFC have been synthetically generated using Pangu, including the binary asteroid system, the CubeSats, the Sun and the stars (Hypparcos catalog), coherently with the reference dynamics of the built trajectories. An example of generated image is shown in Fig. 4.8. Since the phase angle among camera asteroid and Sun is low, the asteroids saturate the image. This condition is desirable for the mothercraft's own navigation and allows have a low phase angle also for the CubeSats. Nevertheless, it can be observed that even if CubeSats and stars are present in the image, they are hardly visible, and a longer exposure time would improve their brightness.

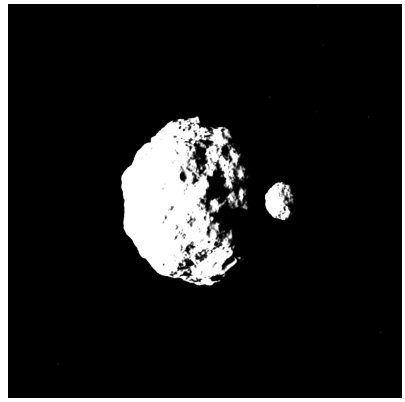


Figure 4.8: Example of synthetic image of Didymos system, stars and CubeSats, generated in Pangu considering the AFC model.

Test 1: images without asteroids. At first, the algorithm capabilities are tested on images without the asteroids rendering. This test aims at addressing the IP performance without the masking block. Since the CubeSats visibility is not affected by the asteroids presence, such test involves a larger number of images where the spacecraft are visible and so it involves a wider set of situations in terms of illumination and trajectory. The test results are shown in Table 4.6.

Table 4.6: IP validation - test 1.

	Visible	Correct Matches	FP	FN	Error μ $[px]$	Error σ $[px]$
CubeSat1	1676	1628	2	48	3.03	2.15
CubeSat2	276	242	0	34	7.72	2.33

The algorithm under test can be considered validated, as the number of FP is kept very low and the FN are contained.

The main causes of the presence of FN are here analyzed:

- **Candidate near to the image border.** In the algorithm implementation there is a 5 pixel border near to the margin in which candidates can not be detected. Such margin is necessary to deal with candidate points that appear or disappear in one of the two involved frames and have the same number of candidate points in the two frames.
- **Appearance of the CubeSat.** When a CubeSat enters the FoV (overcoming the margin), it is not possible to detect it, simply because it is not present in the former frame.
- **Feature description.** It has been discovered that the 67% of CubeSat2 FN are due to a degenerate case in the feature matching. In fact, the three following conditions happen simultaneously:
  1. the two closest candidates form with CubeSat2 an isosceles triangle (see Fig. 4.9);
  2. those two stars are involved in the same triangle when forming the triangular feature (in other words, there are not other candidates closer to them), making CubeSat2 feature description identical to another one;
  3. the two identical features are mismatched.

Test 2: Dimorphos position assumed known. For a proper validation of the masking block as well, images including the asteroids are now considered. In this case, the CubeSats visibility is affected by the presence of the two bodies. In this test, Dimorphos position is assumed to be exactly known from Hera central navigation. Results of test 2 are reported in Table 4.7.

As expected, with respect to test 1 CubeSat1 visibility decreases, whereas it does not change for CubeSat2. In fact, in CubeSat2 case the visibility is affected mainly by the FoV and not by the asteroids presence. It can be noticed

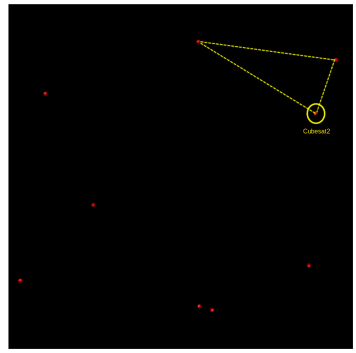


Figure 4.9: FN due to degenerate case in feature description.

Table 4.7: IP validation - test 2.

	Visible	**Correct Matches**	**FP**	**FN**	**Error μ** $[px]$	**Error σ** $[px]$
CubeSat1	535	423	8	112	4.85	1.69
CubeSat2	269	211	12	58	7.90	2.30

that the inclusion of the asteroids in the images leads to a significant increase of FN.

The new FN causes are here analyzed (additional to the ones already presented):

- **Insufficient number of candidates.** This is due to the fact that a large portion of the sky is not visible anymore and so the number of reference stars decreases, making it harder or impossible to distinguish the CubeSats as outliers. For instance, the 55% of CubeSat1 FN and the 9% of CubeSat2' ones are happening when the number of candidate points is 3 or 2. Such a low number of candidates is due to the fact that a large potion of the image is being masked. For instance, at the perigee almost the whole image needs to be masked (see Fig. 4.10).
- **Appearance and disappearance of candidates.** The problems arising when a candidate appears and disappears are now much more frequent. In fact, such phenomena happen not only when a candidate is in the nearby of image borders, but also when it is in proximity of the asteroids. Moreover, the way in which geometric features of candidates close to the appearing/disappearing point are extracted is also affected. There is a routine in the algorithm in order to identify which is the point appearing or disappearing. In Fig. 4.11 and 4.12 an example is shown.
- **Asteroids margin.** Finally, the asteroids masking is done by masking a circular area centered in them. Such area does not correspond to

the bodies exact contour and in the case of Dimorphos it includes a significant margin to deal with uncertainty in his position knowledge. As a result, points that are actually visible in the images are lost during the masking process (see Fig. 4.12). For instance, in CubeSat2 this happens for the 45% of FN. A possible solution could be a more complex image segmentation, that could limit the margin impact. On the other hand, if such solution is followed for an on-board implementation, the computational cost should be carefully evaluated.

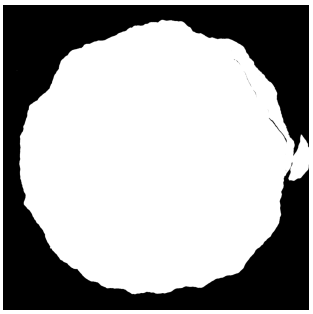


Figure 4.10: Frame at perigee - largest masking needed.

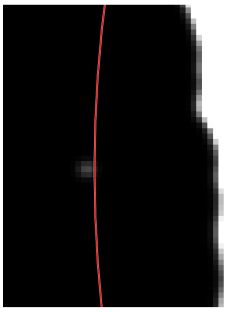


Figure 4.11: CubeSat1 before being masked (frame k).

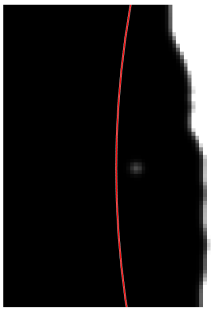


Figure 4.12: CubeSat1 after being masked (frame k+1).

Test 3: Dimorphos position unknown As last test for the IP validation, the most challenging situation is considered, i.e. Dimorphos position is considered not known, but the body is assumed (on-board) to be on a circular orbit around the primary. Please note that the masking margin is equal to Test 2, but in this case the centre of the masked area changes.

Results of measurements coming from the algorithm are resumed in Table 4.8. In particular, a match is considered correct if the error between nominal and measured position is < 10 pixels for CubeSat1 and < 15 pixels for CubeSat2. Otherwise, the point is considered to be a FP. The number of FN is deduced from the visibility analysis. Please note that the high number of FN is due to asteroids masking and especially to Dimorphos, for which the masking margin is 1.5.

Table 4.8: IP validation - test 3.

	Visible	**Correct Matches**	**FP**	**FN**	**Error μ** $[px]$	**Error σ** $[px]$
CubeSat1	535	425	14	110	4.85	1.68
CubeSat2	269	209	12	60	7.93	2.29

Please note that test 3 results are almost unchanged with respect to test 2. This means that the margin for masking has been correctly tuned: the current margin leads to similar results with a less restrictive assumption on Dimorphos position knowledge. In other words, asteroids are correctly and fully masked. The IP performance is not affected by the fact that on-board knowledge on Dimorphos position is limited, but by the fact that less stars in background are visible (with respect to test 1).

With this test the IP is considered validated: in the reference scenario the algorithm is capable of detecting the CubeSats and to identify them. The performance can be considered satisfactory in terms of FP: the FP number is $< 2.6\%$ (with respect to the total number of visible time instants) for CubeSat1 and $< 4.4\%$ for CubeSat2. A low number of FP is fundamental for the robustness of the navigation architecture. The FN number is still relevant, being the 20.6% for CubeSat1 and the 22.3% for CubeSat2. The number of correct matches corresponds to the 79.4% for CubeSat1 and the 77.7% for CubeSat2. The mean error excluding false positives is about 5 pixels for CubeSat1 and 8 pixels for CubeSat2. All performance criteria are met, so the validation is successful.

IP measurement model The IP algorithm is run for the whole trajectory, using images from Pangu. Results of measurements coming from the algorithm are presented in Tab. 4.9. In particular, a match is considered correct if the error between nominal and measured position is < 10 pixels for CubeSat1 and < 15 pixels for CubeSat2. Otherwise, the point is considered to be a FP. The number of FN is deduced from the visibility analysis. The IP is able to detect the CubeSats and correctly distinguish them from stars, background noise and among them, notwithstanding their limited visibility and the presence of other objects in the scene (Didymain and Dimorphos). The number of FP is kept low, avoiding to provide incorrect measurements to the filter. Please note that the high number of FN is due to asteroids masking and especially to Dimorphos, for which the masking margin is 1.5.

Table 4.9: IP performance.

	Visible	**Correct Matches**	**FP**	**FN**	**Error μ** $[px]$	**Error σ** $[px]$
CubeSat1	535	425	14	110	4.85	1.68
CubeSat2	269	209	12	60	7.93	2.29

The IP measurement error model is shown in Fig. 4.13 for CubeSat1 measurements and in Fig. 4.14 for CubeSat2. The error is the difference between the measured pixel coordinates and the pixel coordinates that would be obtained

with a perfect imaging system (i.e. no PSF, no distortion). It can be seen that

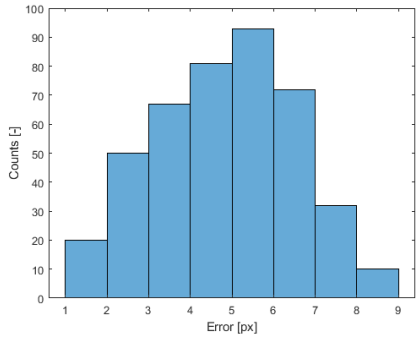


Figure 4.13: CubeSat1 measurements error

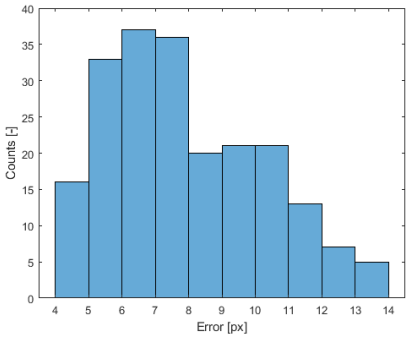


Figure 4.14: CubeSat2 measurements error

the error differs for the two CubeSats. This may be due to different reasons, here listed:

- CubeSat2 when visible is close to the image border, where the camera distortion is stronger and therefore the error in the measured position is larger;
- the different distance and therefore brightness of the CubeSats can cause a different PSF intensity distribution. The threshold therefore affects differently the PSF areas and if the kept area is larger, the error may be larger as well.

4.1.2.2 CubeSats detection analysis

The detectability of the CubeSats in the visible band is studied in this section.

With the considered trajectories and exposure time, no smear occurs. Smear becomes relevant only with exposure times larger than 10 s, as shown in Fig. 4.16 and Fig. 4.15. Therefore, the CubeSats' irradiance can be considered to arrive at one pixel only for the SNR computation.

For the considered trajectories, the resulting range and camera-target-Sun phase angle are shown respectively in Fig. 4.17 and in Fig. 4.18. Considering the surface area of the object to be $0.36\,\mathrm{m}^2$, and the materials reflectivity of 0.8, the expected SNR along the trajectory is computed. As visible in Fig. 4.19, the SNR of the target objects is above the typical acceptability limit of 10 dB. Hence the CubeSats are detectable along the whole trajectory.

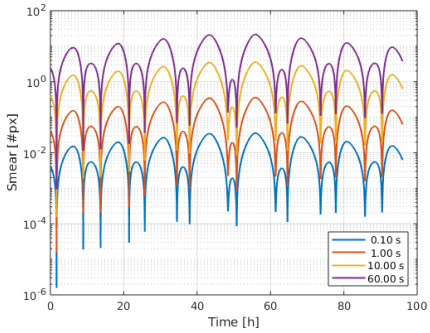


Figure 4.15: CubeSat1 smear.

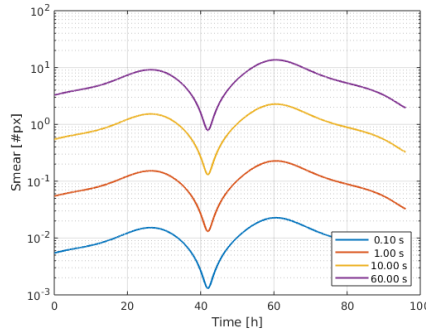


Figure 4.16: CubeSat2 smear.

Nevertheless, assuming a worst case scenario in which the CubeSat is badly oriented, exposing a surface area of $0.01\,\mathrm{m}^2$ with a low reflectivity of 0.2, the CubeSat would not be detectable anymore (see Fig. 4.20).

Considering the irradiance from the target to the sensor, two different contributions can be distinguished: the light reflected from the Sun and the target's own thermal emission. In Fig. 4.21, the two contributions are reported for an object with reflectance $\rho = 0.8$ and emittance $\epsilon = 0.8$, at different temperatures. The selected temperature range is wide: from $-100\,^{\circ}\mathrm{C}$ to $150\,^{\circ}\mathrm{C}$, being a possible range for a non-operating or debris spacecraft. For an operating spacecraft the range can be restricted to $-40\,^{\circ}\mathrm{C}$ to $80\,^{\circ}\mathrm{C}$. It can be noticed that among the different possible temperatures, the emitted component plays a dominant role with respect to the reflected one in the LWIR (Long Wave Infrared) range, i.e. [8 to 15 μm]. The IR band interesting for this mission scenario is therefore the LWIR, since in this band the reflected radiation component is less than the thermal emitted component, allowing robustness to illumination and target visibility also in eclipse (lowest temperatures). The choice of such band can be considered robust with respect to orbit and target materials, since the appropriate band would be always the LWIR.

In conclusion, a possible enhancement of the far-range object detection could be performed with multi-spectral imaging and in particular with a TIR camera.

4.2 Close range imaging for mapping

In this section the problem of using an AI architecture inserted in the GNC chain during the close range relative navigation with an unknown target is tackled.

The quality of the mapping process of a small celestial body can be evaluated based on the spacecraft viewing angle (i.e. the incidence angle i) and the

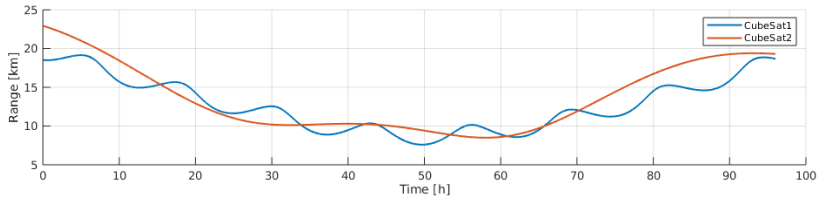


Figure 4.17: Target - camera range.

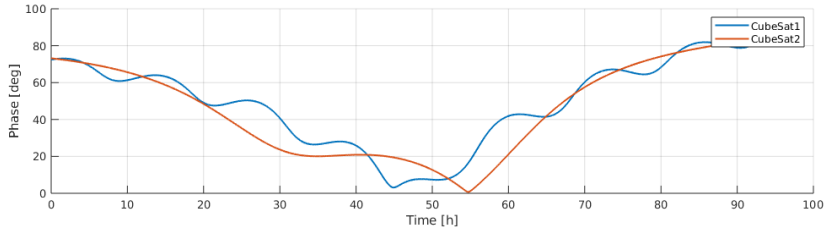


Figure 4.18: Camera-target-Sun phase angle.

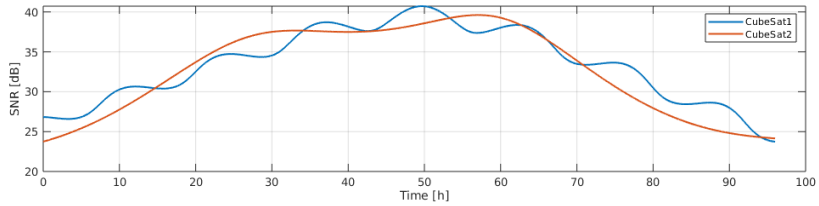


Figure 4.19: Target SNR - best case.

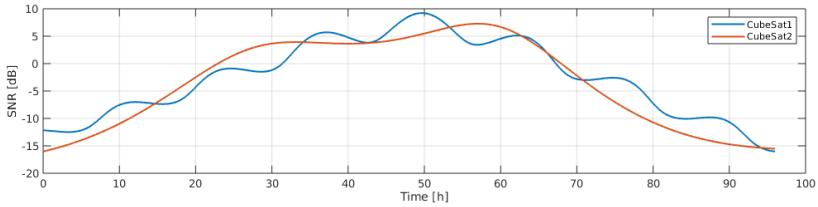


Figure 4.20: Target SNR - worst case.

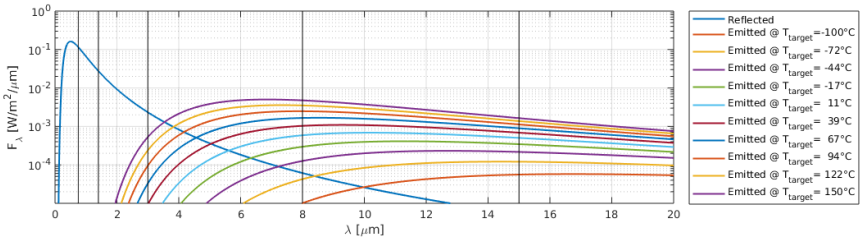


Figure 4.21: Target irradiance: contribution of reflected and emitted radiation.

illumination of the surface (i.e. the emission angle e) [78]. Considering a polyhedron shape model of the body, a score S^i can be associated to each facet:

$$S^i = w_1 S_i^i + w_2 S_e^i + w_3 S_{\Delta e}^i + w_4 S_{\Delta \alpha}^i + w_5 S_{\Delta \beta}^i \tag{4.4}$$

where S_i^i is the inclination score, S_e^i the emission one, $S_{\Delta e}^i$ the emission variation score and $S_{\Delta \alpha}^i$ and $S_{\Delta \beta}^i$ the solar and spacecraft azimuth angle scores. According to the images history, the facet mapping index m^i is defined for the i-th facet:

$$m^i = S^i \min\left(1, \frac{n}{N}\right) \tag{4.5}$$

where n and N are respectively the number of taken images and the number of ideally necessary images.

As a general formulation, small bodies mapping can be described as a continuous states and actions POMDP (Partially Observable Markov Decision Process) [78]:

$$\pi^\star = \text{argmax}_\pi \mathbb{E}_\pi \left[\sum_{k=0}^{T} \gamma^k r_k(\boldsymbol{a_k}, \boldsymbol{b_k}) \right] \tag{4.6}$$

where γ is the discount parameter, $\boldsymbol{b_k}$ is the belief of spacecraft state and map state $\boldsymbol{s_k} = (\boldsymbol{x_k}, \boldsymbol{m_k})$, $\boldsymbol{a_k}$ the actions performed by the agent following policy π and r_k the reward that models the planner objectives.

The spacecraft can act according to an AI-policy for improving the mapping performance. The goal of the policy is to better the mapping operations of a partially unknown small body, enhancing the mission science return and optimizing the amount of collected data. A Deep Reinforcement Learning (DRL) algorithm is employed for an on-ground learning, allowing to obtain a flexible, light and performing policy to be used on-board. Such policy betters the body coverage ensuring good illumination conditions of the acquired images, outperforming a uniformly scheduled image acquisition. In particular, two different techniques are compared: Neural Fitted Q (NFQ) [79] and Deep Q Network (DQN) [80], where an Artificial Neural Network (ANN) is used to approximate the Q-value, i.e. the expected return over time. Therefore the optimal policy is the one that maximizes the Q-value:

$$\pi^\star = \text{argmax}_\pi Q_\pi(\boldsymbol{a_k}, \boldsymbol{s_k} | \theta_\pi) \tag{4.7}$$

where θ_π are the ANN weights and biases. The action that can be performed by the agent is the acquisition of an image at the current time step. States have been designed to synthesize only the information necessary and useful for decision making and comprise the *memory state*, *map state* and current viewing and illumination *angles state*. The AI-based algorithm has been selected also in virtue of its low computational burden and flexibility with regard to the mission scenario, which would facilitate an on-board implementation.

The implementation of the AI-policy relies on the work done in [81].

4.2.1 Verification of AI-based policy for mapping operations improvement

AI-policy verification with inputs state belief The images acquisition policy takes in input the estimated pose from the GNC system (see Fig. 4.22) and outputs the command of acquiring an image, if it is worth, keeping into account the body mapping and constraints from the available on-board memory.

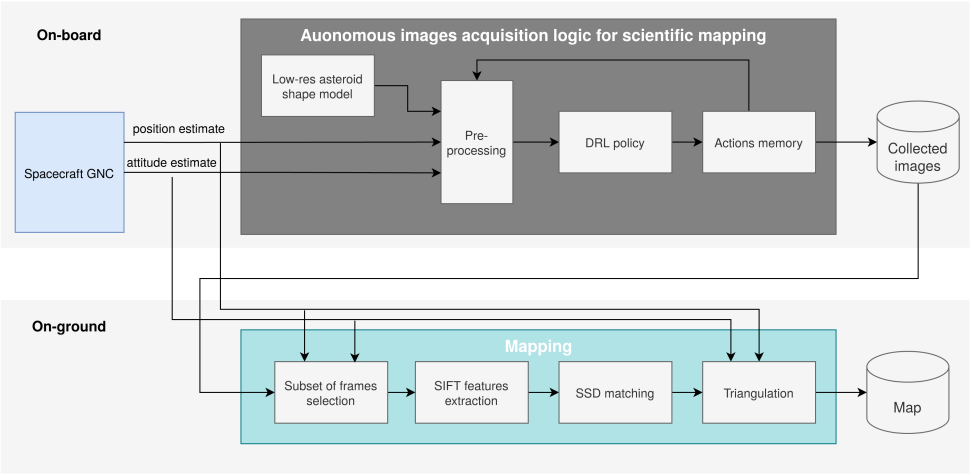


Figure 4.22: Reference trajectory in the inertial frame.

In the present paper, the AI-policy for high-value images collection is included in the system as shown in Fig. 4.22. In particular, the acquisition logic takes in input the estimated relative pose with respect to the asteroid system and processes such values to compute the input state for the AI-policy. The on-board algorithm is supposed to have a coarse polyhedral shape model of the body available.

AI-policy verification with on-ground point cloud generation Because of the need of verifying the mapping efficiency, a simple IP algorithm is implemented. The policy is meant to optimize images collection for SPC, nevertheless SPC is a very complex and iterative process, usually performed with human experts supervision and its implementation is beyond the purpose of this analysis.

The equivalent of an on-ground mapping algorithm is implemented for generating a point cloud of the body.

A simulation of the on-board system is run first, then the on-ground equivalent algorithm takes as input the spacecraft position and orientation as from the on-board navigation and the acquired frames from the AI-policy. According to

the actions performed by the agent, the corresponding synthetic images are generated.

A reference image is considered and images that overlap are selected, based on the navigation data. SIFT features are extracted and correspondences in the images are searched for: when two features match, based on a Sum Squared Differences (SSD) metric criterion, the point is triangulated [82]. The process is repeated for each image. In such a way a point cloud of the body is created.

The obtained shape is then compared with the available high-resolution shape model, used for generating the images and taken as reference ground truth. Such triangulation-based IP wants to be a verification method, with the aim to assess the effectiveness of the images collection policy, rather then a shape reconstruction algorithm to be used during mission operations. This kind of verification has two different purposes:

- to assess the effectiveness of the AI-policy;
- to compare the performance of the AI-policy and benchmark policies, using a criterion that overcomes the limitation of having hand-based rewards.

4.2.2 AI-based policy verification results

4.2.2.1 AI-policy verification with inputs state belief

The reference trajectory is a circular orbit at 4 km from the Didymos system, with an inclination of 11° (see Fig. 4.23). The perturbations introduced by the secondary body of the system are deemed negligible at the distances in play and for the purpose of this analysis, reason for which the dynamical model used to generate the reference ground-truth trajectories is the perturbed 2-body problem, with the whole mass of the system concentrated in the primary asteroid. A camera with a 10° FoV is considered as science imaging payload.

AI-policy ideal behavior at Didymos The AI-policy is tested on the reference scenario, with the mapping of Didymain only as objective. The presence of Dimorphos is not known by the net. The known input to the strategy comprises a 1000 facets shape model of Didymain and a known memory constraint of maximum 50 images to acquire during one orbital period, which lasts about 74 h. The AI-policy is employed with a lower frequency with respect to the navigation one, and in particular a frequency of 150 time steps per orbit is chosen.

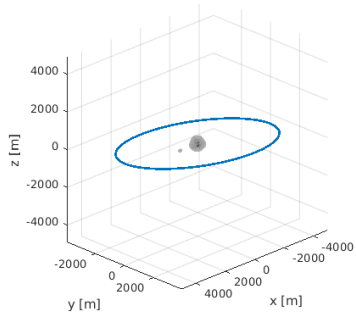


Figure 4.23: Reference trajectory in the inertial frame.

The performance of the AI-policy is compared to the mapping that would be obtained taking all the images, at each of the 150 steps. In Table 4.10, the *all-frames acquisition policy* (ALL) is compared with two AI-based strategies (DQN and NFQ) and also with the UNI policy, that considers a fixed frequency for acquiring the frames, to completely exploit the available memory. The

Table 4.10: Comparison between UNI, DQN and NFQ and the ideal ALL policy.

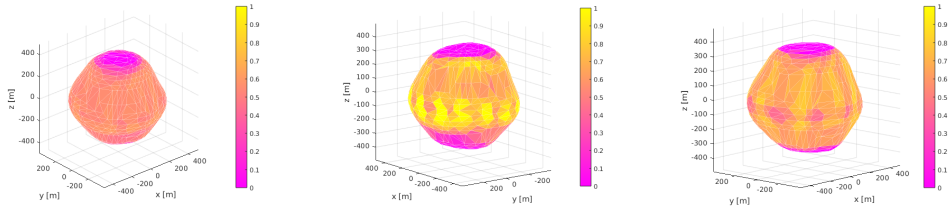
Policy	**Mapping index** [%]	**Frames acquired** [-]
ALL	47.8	150
UNI	41.9	50
DQN	46.8	37
NFQ	44.3	18

best mapping is reached with the DQN strategy, which gets close to the ideal maximum quality achievable (ALL), outperforming both UNI and NFQ. Both the AI-based policies respect the constraint on maximum memory. The NFQ strategy is less prone to collect new data and reaches an inferior mapping quality, even if still better than UNI. As additional note, the UNI takes 6 images of Didymain when it is completely shadowed. Please note that for such test, a perfect knowledge of the spacecraft state is available as input to the AI-policy.

To have a better insight on the obtained mapping, the results achievable with the *all-frames acquisition policy* are shown in Fig. 4.24. In particular, in Fig. 4.24a it can be seen that the achievable mapping is quite uniform with exception of the poles, that can not be covered staying in the considered orbit. Regarding the emission score (in Fig. 4.24b), it is dictated by the quite regular

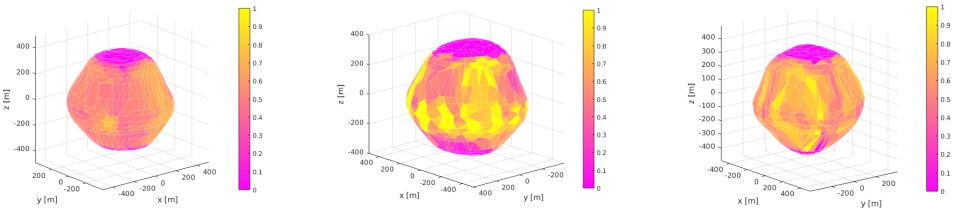
shape of the body and the Sun direction, that is approximately in Didymain equatorial plane. For this reason, the collected images have almost always emission angles in the acceptable range of $10 - 50$ ° in the equatorial area and, with the poles exception, a good score is obtainable. The incidence score (in Fig. 4.24c), has a more uniform distribution, as expected from the spacecraft position relative to the body and the pointing towards its center of mass direction.

The results obtained with the AI-policy (DQN) are shown in Fig. 4.25. Looking at Fig. 4.25b and Fig. 4.25c, it can be noticed that the score is less homogeneous among neighboring facets, compared to the ALL case. Nevertheless, the AI-policy performance is close to the *all-frames acquisition policy*, granting a good overall coverage of the body.



(a) Mapping index. **(b)** Emission score. **(c)** Incidence score.

Figure 4.24: All-frames acquisition mapping performance (ALL).



(a) Mapping index. **(b)** Emission score. **(c)** Incidence score.

Figure 4.25: AI-policy mapping performance (DQN).

AI-policy behavior with input state belief Another test of the AI-policy is performed introducing some errors in the knowledge of the relative pose of the spacecraft. Such tests are meant to verify the DQN robustness to uncertain inputs, which were not considered at all during the training, but may be relevant in case of application of the strategy to a small platform in a challenging environment such as the one of a binary asteroid system.

In particular, 7 tests are performed considering an increasing uncertainty separately for position and pointing, which are perturbed by a white noise.

Finally, test 8 considers both effects, with the uncertainty expected for the study case, i.e. 100 m for the relative position and 3° for the pointing. Each test has been run with 50 simulations; mean value and standard deviation of the mapping index and acquired frames are reported in Table 4.11. Please note that the acquired frames are reported as percentage of the maximum capability of the memory.

Table 4.11: AI-policy performance with uncertainties on relative state.

Test	Position σ [m]	Pointing σ [°]	Mapping index [%] (σ)	Frames acquired [%] (σ)
0	0	0	46.8 (-)	74
1	100	0	47.4 (0.8)	75 (4)
2	250	0	44.0 (1.5)	71 (4)
3	500	0	30.4 (4.2)	70 (5)
4	0	0.05	46.8 (0.2)	73 (2)
5	0	3	45.6 (0.7)	88 (3)
6	0	5	46.3 (0.4)	94 (7)
7	0	7.5	46.0 (0.7)	79 (12)
8	100	3	47.2 (0.6)	84 (7)

The results in Table 4.11 show that the AI-policy is quite robust to both kinds uncertainties. It can be noticed that the final mapping obtained is close to the ideal value. This is due to the fact that, to have a fair term of comparison, in each simulation and test case the actual pointing and position of the spacecraft is the same and only the state belief is different.

The first 3 tests show that the number of frames acquired is comparable to the test-0, but the actual mapping performance decreases with a higher position uncertainty. Since the AI-policy assumes to point to Didymain CoM from the belief of its relative position, the information regarding to the illumination of the surface in view is affected, leading to a lower quality mapping. It can be noticed that in test-1 the mapping index outperforms the one of test-0, meaning the the AI-policy is actually not optimal for the present test case. Therefore, with a slightly different belief of the state, it acts better than in the nominal situation. Please note that the AI-policy has been trained in a completely different scenario [81], concerning both body shape and orbit, and has been designed in order to be easily employed into a wide variety of different mission scenarios. Therefore, an optimal behavior is actually not expected to be reached, but only an enhancement in scientific mapping and data collected compared to simpler acquisition strategies [81].

Test 4 considers a pointing error in line with the result presented in Fig. **??**. Such error does not significantly affect the policy behavior, since about the same performance is reached. On the contrary, tests 5-7 highlight a different behavior of the net: a larger number of images is collected, maintaining a good mapping quality thanks to the larger amount of data. With a perfect pointing, at 4 km Didymain would be almost inscribed in the image, thus maximizing the resolution, keeping the whole body in view. In test 6 almost all the images are collected: the belief of the current mapping is worse than the actual one because part of the body is believed to exit the FoV; thus the policy continues collecting data to complete the body coverage even if they are not necessary. Increasing again the pointing determination uncertainty, a large amount of data is still collected, but decreasing with the uncertainty. Such effect can be related to the fact that when a significant portion of the asteroid is believed not to be in view, it is not worth collecting the data.

In the combined test (test 8) the policy is robust to uncertainty in the state, leading to a good mapping of the object, but with a rise of data acquired with respect to absence of uncertainties. The considered uncertainties can be considered in line with a realistic scenario and the AI-policy is verified to outperform a classical UNI scheduling both in terms of amount of data and images quality.

As an additional consideration, it is observed that the coverage of the body's surface can not be always completed because some areas persist to be in shadow. Thus, as a general consideration, the mapping could benefit from the employment of multispectral cameras.

4.2.2.2 AI-policy verification with on-ground point cloud generation

As a target for testing the images collection DRL-based policy, comet 67P has been chose in virtue of its irregular shape. The orbital scenario is a keplerian hyperbolic arc with pericenter at 70 km from 67P, shown in Fig. 4.26. A narrow angle camera with a 5° field of view and 1024x1024 pixels is considered as optical sensor. Therefore the body characteristic dimension is about 800 pixels and the resolution with which the surface is imaged is 7 m per pixel. The phase angle Sun-body-spacecraft ranges from 10° to about 90° along the orbit, granting an appropriate illumination for the body imaging and at the same time allowing sufficiently variable illumination conditions for testing the algorithm in a challenging scenario.

Since the body is still far, the aimed number of images is assumed to be 30 and the preliminary knowledge of the shape model is assumed to be quite low, as represented in Fig. 4.27, corresponding to a 100 facets shape model.

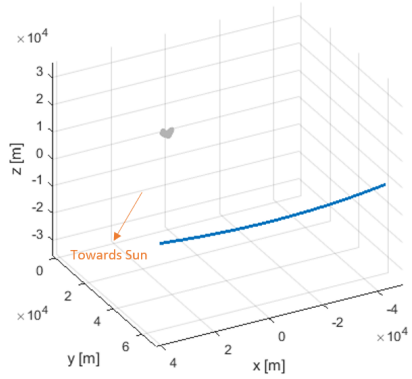


Figure 4.26: Orbital scenario - hyperbolic arc at 67P.

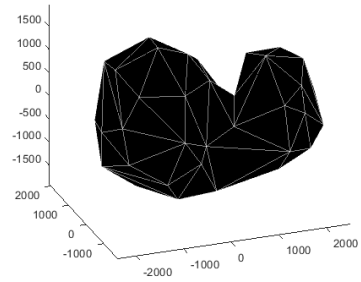


Figure 4.27: Low resolution shape model available on-board.

Verification of effectiveness for on-ground IP The collected images are synthetically generated with the ray-tracing software POV-Ray and the IP algorithm is applied to reconstruct the shape. SIFT features are extracted from images of the same body areas and matched (see Fig. 4.28).

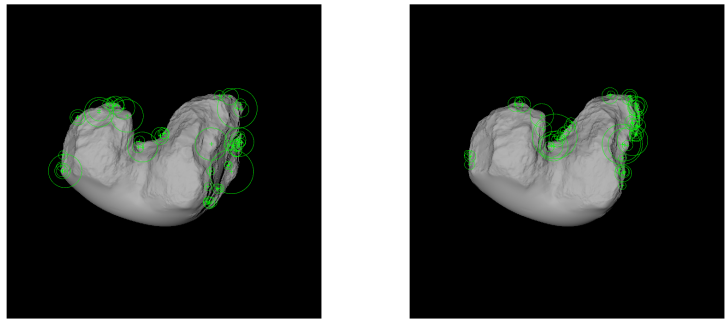


Figure 4.28: SURF features extracted from POV-Ray generated images.

The DRL-based policy is capable of dealing with memory constraints, collecting a total number of 33 images (slightly larger that the aimed number of 30 images). The point clouds of the UNI and DRL policies are shown respectively in Fig. 4.29 and 4.30. Following the DQN policy the mapping quality betters and excluding outliers 938 points are triangulated, whereas with the UNI policy only 162 points are correctly matched triangulated.

This proves that an image collection based on the DRL policy results in a better capability of reconstructing the body shape and being compliant with the storage memory requirements. Nevertheless, the body shape is still far

from being completely reconstructed, because of the scarcity of salient features in the generated images and because of simplified IP adopted.

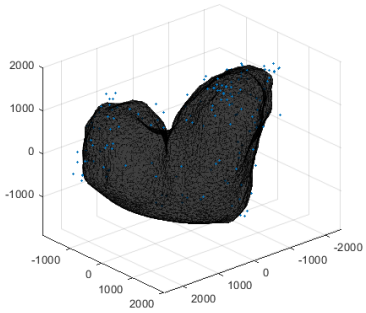


Figure 4.29: UNI policy point cloud.

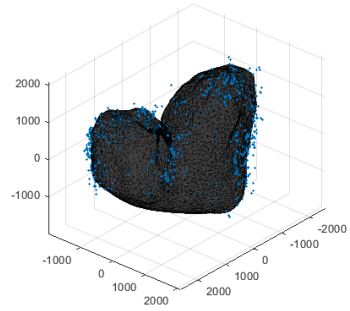


Figure 4.30: DQN policy point cloud.

CHAPTER 5

Visible-thermal relative navigation and mapping systems

5.1 Multimodal visible-thermal relative navigation and mapping

In this Section, the possibility of using visible and thermal images for navigation in proximity of an asteroid is studied. The aim of this study is to understand if thermal images can be employed for navigation purposes, and what are the conditions in which their exploitation can lead to benefits for the navigation.

Multi-modal VIS-TIR SLAM architecture It is here proposed to use a multi-modal approach to employ images from cameras working in the visible and thermal spectrum for navigating in proximity of an asteroid.

For this purpose, the validated visible and thermal images are employed as input to a vision-based navigation algorithm, as shown in Fig. 5.1.

The selected vision-based navigation algorithm builds on concepts of Visual Odometry (VO) and Visual-SLAM (V-SLAM) and relies on the implementation from [83]. Features are extracted and tracked from the incoming image stream,

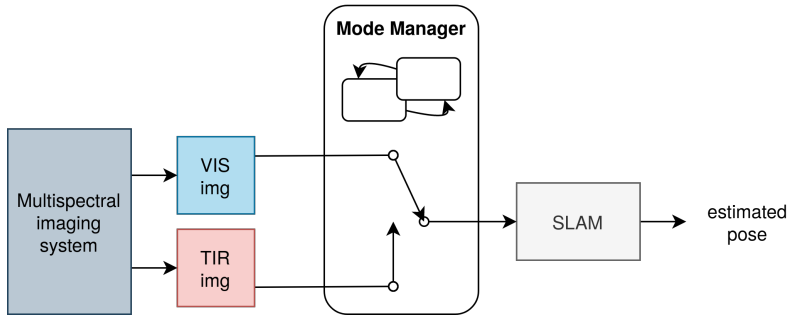


Figure 5.1: Multi-modal visible-thermal relative navigation and mapping.

while at the same time a sparse 3D map of the asteroid is reconstructed and used for navigation. A detailed description of the algorithm can be found in [83].

5.1.1 Phase angle and thermal inertia analysis

Mission scenario definition The scenario under study is the one of asteroid 162173 Ryugu, for which real mission data are available. The navigation architecture is tested considering a keplerian circular orbit at 5 km distance from Ryugu, with an inclination of 11°, shown in Fig. 5.3. Please note that Ryugu's mean diameter is 870 m.

Ryugu's thermophysical properties used for the thermal analysis are reported in Table 5.1. The asteroid shape model used is available from Hayabusa 2 mission data [84].

Table 5.1: Ryugu's thermophysical properties [85].

Parameter	Value	Unit
Emissivity	0.9	-
Bond albedo	0.0146	-
Thermal inertia Γ	225	$Js^{1/2}K^{-1}m^{-2}$
Density	1192.8	kgm^{-3}
Heat capacity	600	JK^{-1}

Sensors for multispectral imaging selection Given the orbital scenario, the grey-body emitted radiation (Planck's law) and the radiation reflected by the target at a 5 km distance is shown in Fig. 5.2. Reflectance and emissivity coefficients for Ryugu are $\rho = 0.02$ and $\epsilon = 0.9$ from the Hayabusa 2 mission data analyses [15] [86].

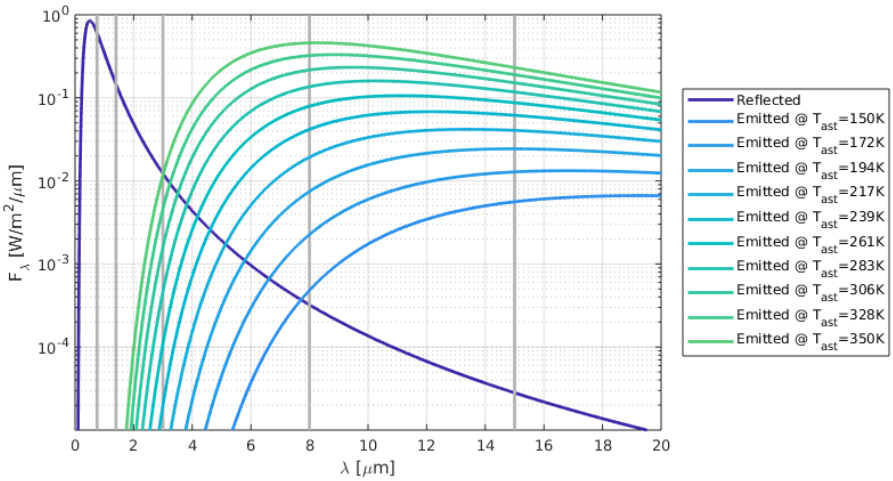


Figure 5.2: Radiation emitted and reflected from Ryugu at 5 km.

The selected bands of interest for the multispectral imaging are consequently the VIS and LWIR band, being not overlapping and corresponding respectively to the peaks of reflection and emission. The main purpose of this study is in fact to analyze the benefits of exploiting imaging coming from well separated bands and thus carrying different information.

The TIR and VIS cameras characteristics are reported in Table 5.2. In particular, for the TIR the real HW characteristics of Hayabusa 2 TIR payload have been considered, while for the VIS sensor common characteristics have been chosen, in order to have as expected a wider array size and a similar but smaller FoV (Field of View).

Analysis parameters Four case studies are analyzed:

1. **Low phase angle.** In this case the s/c is on the light side of the asteroid.
2. **Intermediate phase angle.** The s/c is looking at the terminator.
3. **High phase angle.** The s/c is on the eclipse side of the asteroid.
4. **High phase angle + high thermal inertia.** The s/c is on the eclipse side of the asteroid and a high thermal inertia is assumed for Ryugu.

The considered cases differ only regarding the illumination conditions (phase angle ϕ) and asteroids thermal inertia Γ. Such parameters are in fact potential weaknesses of VIS-only and TIR-only navigation and thus a potential area of improvement with a multispectral approach. The actual Γ of Ryugu is employed in cases 1-3 and a different one is chosen for case 4.

Table 5.2: Cameras characteristics.

TIR	
array size	344 x 260
FoV	16.7° x 12.7°
NETD	0.4 K
bit depth	12
detectable temperature	150 - 460 K
spectral band	8 - 12 μm
VIS	
array size	1024 x 1024
FoV	10° x 10°
bit depth	8
spectral band	380-750 nm

Each arc has a duration of about 3 h and the considered sampling rate is 1 frame per minute, due to the slow relative dynamics.

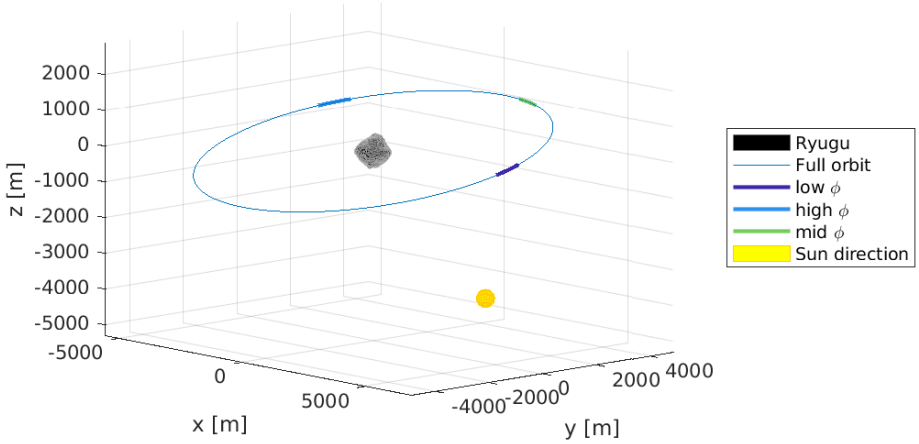


Figure 5.3: Spacecraft orbit in inertial frame with different illumination study cases.

5.1.1.1 Images generation results

Thermal simulation results For each case study, the minimum and maximum temperatures on the asteroid surface are approximately constant along the whole orbit, being in range 146 K to 397 K. Such results are in accordance with the actual Ryugu surface temperature, which is estimated to typically range from 250 K to 400 K during daytime and down to 150 K during nighttime [14].

From the thermal simulations in cases 1-3, it has been observed that the low thermal inertia causes fast temperature changes on the surface as the asteroid rotates on its spin axis. This means that the phase angle is tightly related with the temperature field on the surface.

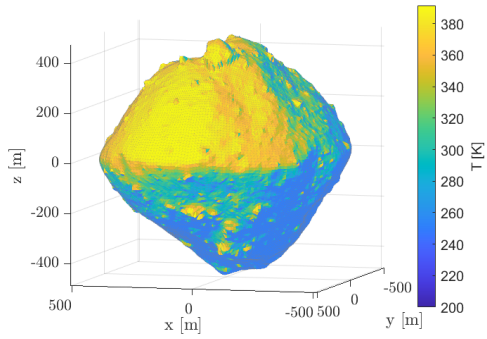
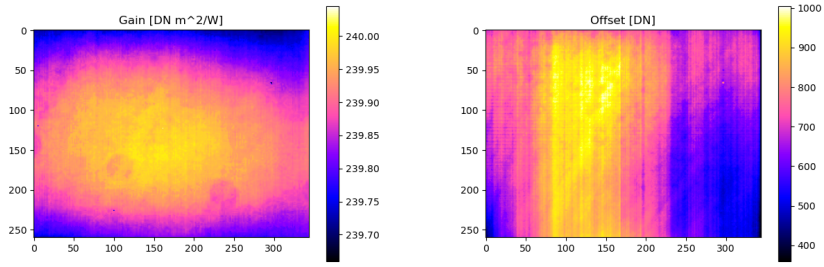


Figure 5.4: Temperature field, high Sun phase angle.

Microbolometer response result Representative data for the sensor model have been taken from calibration experiments on microbolometer for asteroids observation [21]. In particular, the available non uniformity textures of responsivity and offset have been tailored on the present study case. The textures are resized with TIR array dimensions; while the offset and gain values are scaled according to the TIR bit depth and operative detection temperature range. The resulting detector response and offset values at the mean expected temperature are shown respectively in Fig. 5.5a and 5.5b.

An example of pixels' responses with large differences in offset and sensitivity is shown in Fig. 5.6 for the whole operative detection temperature range. As it can be noticed, the non-uniformity effects can be quite relevant. Thus, their modeling shall be included for realistic thermal images generation. In fact, for a real-time on board application, as the relative navigation treated here, corrections might not applied to images or they might degrade in time, becoming potentially dangerous to the navigation.



(a) Modeled detector gain. **(b)** Modeled detector offset.

Figure 5.5: Modeled detector for blackbody scene at $T_{mean} = 305$ K. Base texture from literature real sensor data[21].

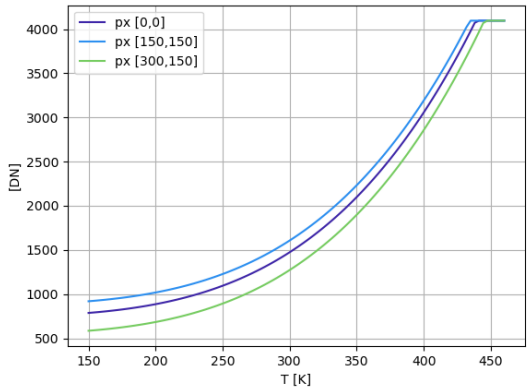


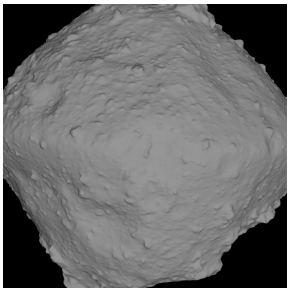
Figure 5.6: Pixel response examples.

Generated images VIS images are generated by means of classical rendering techniques, while TIR images are obtained from the the asteroid surface thermal behaviour modeling. Examples of the VIS images, the generated temperature field and the TIR images are shown for each case-study respectively in Fig. 5.7, Fig. 5.8, Fig. 5.9 and in Fig. 5.10. What can be noticed is the large appearance difference between the VIS and TIR images, not only in relation to form factor and resolution, but also in the brightness distribution and presence of features on different surface areas. Possible improvements to the generation process are the inclusion of different emissivity values of the asteroids terrain and rocks and the computation of the asteroid-pixel view factor F_{a-p} from a shape model, instead of the spherical approximation.

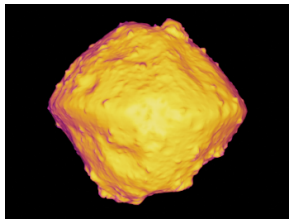
In cases 1-3, due to the very low thermal inertia Γ of asteroid Ryugu, the phase angle ϕ has a relevant influence on the thermal images as well. As a

consequence, in the low ϕ case, the TIR observes mainly the hottest regions of the surface; while in the high ϕ the coldest ones. Simulations have highlighted that due to Ryugu low thermal inertia, the thermal images in eclipse are quite dark and the brightness level is comparable to the noise. Being large areas of the body neither visible in the optical band, neither from the thermal camera, both bands result to be not useful for optical navigation in the high ϕ scenario. In fact, considering that the simulated temperatures are in range 146 K to 397 K and that Hayabusa TIR covers temperatures in range 150 K to 460 K, but can work properly only in the range 230 K to 420 K [14], such result is in line with the mission data.

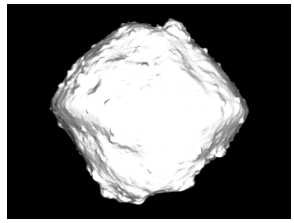
For this reason, the 4th study case is introduced, which is identical to case 3, but considers a fictitious, even though realistic, physical property, i.e. a higher thermal inertia, which leads to hotter temperatures in shadow. In this case, VIS images have a bad quality in shadow, but the asteroid is observable with the TIR camera.



(a) VIS image

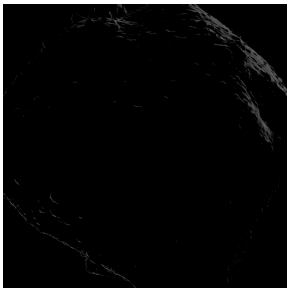


(b) Temperature field.

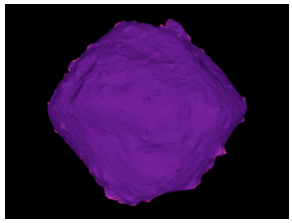


(c) TIR image.

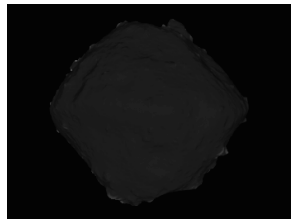
Figure 5.7: Case 1: Low ϕ (light).



(a) VIS image

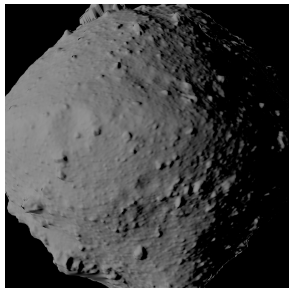


(b) Temperature field.

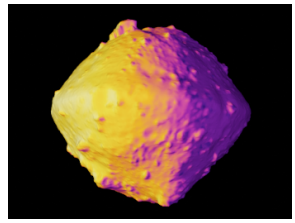


(c) TIR image.

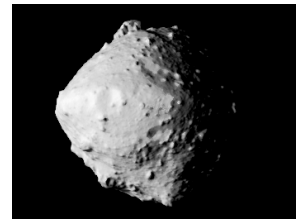
Figure 5.8: Case 2: High ϕ (shadow).



(a) VIS image

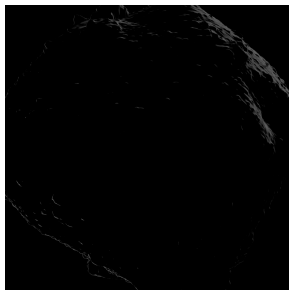


(b) Temperature field.

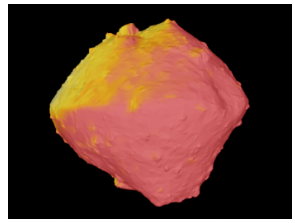


(c) TIR image.

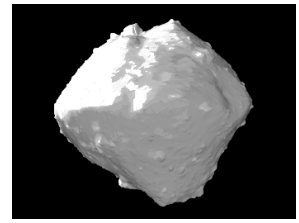
Figure 5.9: Case 3: Mid ϕ (terminator).



(a) VIS image



(b) Temperature field.



(c) TIR image.

Figure 5.10: Case 4: High ϕ (shadow), high Γ.

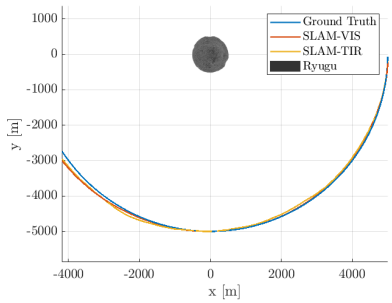
5.1.1.2 Navigation and mapping system results

The SLAM algorithm is then tested on the generated synthetic images.

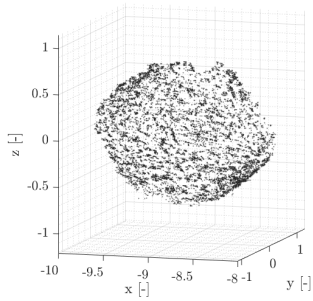
Fig. 5.11a shows the reconstructed trajectory for the two different sensing modalities, considering the low phase angle case study. The trajectory is correctly retrieved by both the VIS and TIR mode. As expected, VIS images still provide the best results in terms of navigation accuracy, having a larger resolution. The localization error always tends to increase towards the end of the sequence, that is a typical feature of any SLAM algorithm, since the pose error accumulates despite bundle adjustment.

To quantify the navigation algorithm performances, the overall position error is computed as:

$$e_\rho = \sqrt{(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 + (z_i - \hat{z}_i)^2} \tag{5.1}$$



(a) Reconstructed trajectory in Ryugu body-fixed frame.



(b) 3D Sparse map

Figure 5.11: SLAM results for case n. 1.

where $\hat{x}$,$\hat{y}$,$\hat{z}$ are the position components estimates. The attitude error is instead computed following [87]:

$$e_R = arccos\left(1 - \frac{tr(\mathbf{I} - \mathbf{A}^T\hat{\mathbf{A}})}{2}\right) \tag{5.2}$$

with $\hat{\mathbf{A}}$ being the estimated rotation matrix. The Root Mean Square Error (RMSE) is then reported in Table 5.3 for each test case scenario.

It has been noticed that the SLAM performance can vary depending on the map initialization step, which in certain simulations leads to an immediate failure of the algorithm. It is here clarified that the results in Table 5.3 report the performance in cases in which the algorithm does not fail, as the focus of the analysis is on the navigation accuracy that can be achieved with different input data and not on the SLAM initialization robustness.

As previously mentioned, since both VIS and TIR images in eclipse tend to be quite dark, the vision-based navigation algorithm cannot provide a meaningful solution. When dealing with favorable illumination conditions, i.e. low phase angle, VIS images still retain a clear advantage with respect to TIR ones, due to their wider array size. However, considering a high phase angle, both VIS and TIR images lead can not be employed for navigation purposes. In the fourth case, a higher thermal inertia for Ryugu's thermal model has been assumed, which consequently led to a temperature field compatible with Hayabusa TIR capabilities even on the eclipse side. In this last scenario, VIS images are still too dark to be used for navigation purposes, while a TIR-based localization solution becomes feasible.

Concerning now the mapping performances of the implemented SLAM algorithm, it can be noticed that the best 3D sparse map is obtained using VIS images with a high phase angle, despite the lower navigation accuracy. The

Table 5.3: Localization error.

	VIS		TIR	
	e_ρ $[m]$	e_R $[deg]$	e_ρ $[m]$	e_R $[deg]$
Case n.1	103.08	2.45	173.81	3.49
Case n.2	236.15	11.79	280.25	12.43
Case n.3	–	–	–	–
Case n.4	–	–	328.77	13.86

reason is that this illumination condition makes it possible to extract features that are more evenly spread on the asteroid surface, which in turn translates into a spatially uniform map. Fig. 5.11b shows the output 3D sparse map. Please notice that due to the selected camera FoV it was not possible to reconstruct the whole shape.

5.1.2 Conclusions

This analysis studies the possibility to exploit multispectral imaging sensors for asteroids relative navigation.

The current technologies of photon and thermal detectors are examined and accurately modeled. Based on the sensors' characteristics and on the asteroids spectral radiation emission and reflection profiles, the visible and thermal LWIR bands are selected as the most promising.

A vision-based SLAM algorithm is selected as relative navigation architecture and tested with visible and thermal synthetic images. A physics-based thermal imaging generation process is employed, starting from the asteroid thermal simulation up to the sensor response. The major outcomes of such analysis are that:

- TIR sensors can be employed for navigating, offering an interesting opportunity for relative navigation in close proximity of asteroids.
- While both VIS and TIR sensors are exploitable, they contribute in different ways to navigation: VIS imaging is superior in terms of navigation accuracy and map quality, due to the higher resolution of the detectors available for this technology; TIR imaging enables the navigation in new scenarios, i.e. in the case of high phase angle with a high thermal inertia asteroid.
- Both sensors can not be employed when images are too dark, being the target comparable to the noise level of the image.

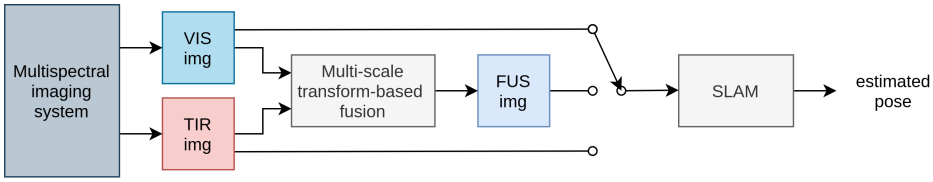


Figure 5.12: Images fusion FUS architecture.

5.2 Visible-thermal images fusion

Images fusion In this section, it is proposed to enrich the multi-modal VIS-TIR architecture for navigation and mapping in proximity of asteroids, with a third possibility: the fusion of the two images, as shown in Fig. 5.12. Visible (VIS) and thermal infrared (TIR) images are acquired from the on-board imaging system. A fusion at image-level is performed, producing fused (FUS) images via a multi-scale transform-based approach. Then, the images sequence is processed by means of a Simultaneous Localization and Mapping (SLAM) algorithm, whose main output is the estimated pose. The SLAM block can be used to process VIS, TIR or FUS images: according to the operating conditions the most suitable option shall be selected. Thus, three different operative modes are available on board: VIS or TIR stand-alone navigation and mapping or fused images FUS navigation and mapping.

Multiscale transform-based fusion Several methods exist for direct image fusion, such as multi-scale transforms, sparse representation and neural networks. In this architecture a multi-scale transform approach is adopted, being efficient, robust and widely used in other application fields [32]. Please note that for the selected application, other successful methods as saliency-based methods are not considered: their application in this context is not meaningful, as the image of an asteroid does not contain any relevant geometric feature, except from its own contour.

The selected image fusion pipeline is reported in Fig. 5.13. First, each source image is decomposed into a series of multi-scale representations. Then, the multi-scale representations of the source image are fused according to the selected fusion rule. Finally, the fused image is acquired using corresponding inverse multi-scale transforms on the fused representations.

5.2.1 Results

Images fusion results Two major methods for the transform have been compared: the Laplacian transform and the wavelet transform, adopting the

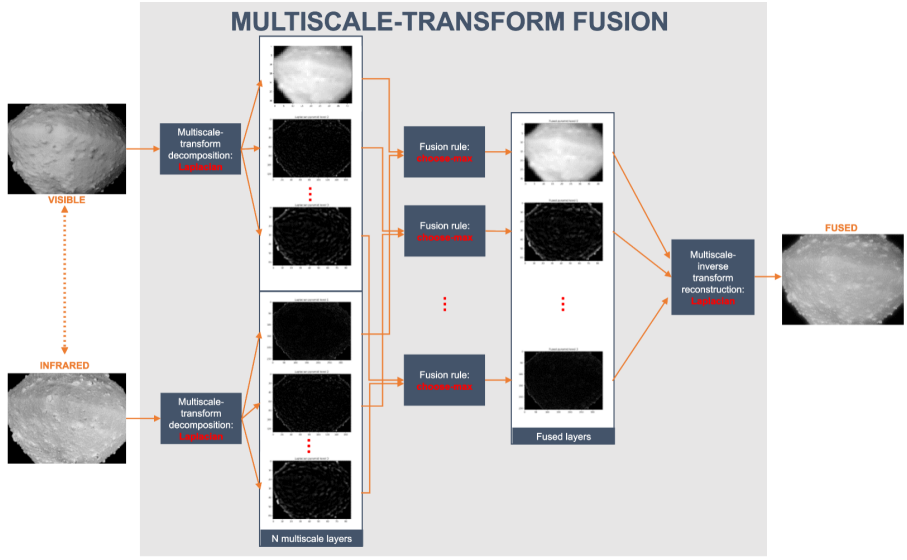


Figure 5.13: Images fusion method scheme.

choose-max as fusion rule. The methods are compared in Table 5.4 for the fusion of a couple of Ryugu VIS-TIR synthetic images. The image entropy H is exploited as fusion metric. The fused image shall contain the largest amount of information and thus have the highest entropy [32], which is defined as:

$$H = -\sum_{l=0}^{L-1} h_l log(h_l) \tag{5.3}$$

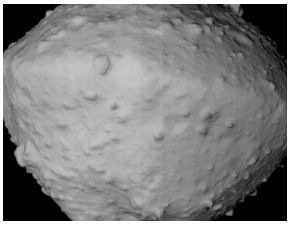
Where L is the number of gray levels and h_l is the normalized histogram of the image.

Results are reported in Table 5.4. Both methods are effective to increase the

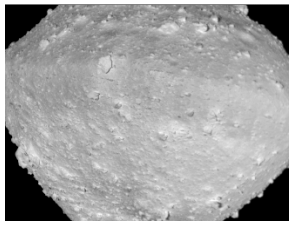
Table 5.4: Images entropy for transform selection.

Transform	H_{VIS}	H_{TIR}	H_{FUS}
Laplacian	1.507	3.490	4.959
Wavelet	1.507	3.490	3.494

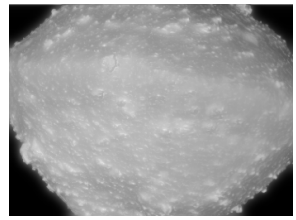
image entropy, being the Laplacian the best performing. Moreover, it has been observed the presence of some aliasing using the Wavelet transform for some images, around the asteroid contour, this the Laplacian transform is selected. An example of fused image is shown in Fig. 5.14.



(a) Synthetic VIS image.

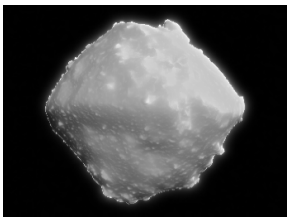


(b) Real TIR image.

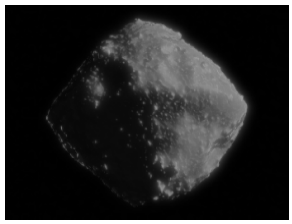


(c) FUS image.

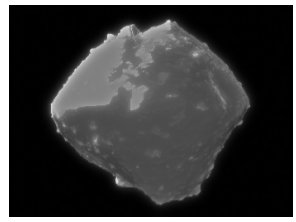
Figure 5.14: Example of fusion application with a real TIR image and its corresponding VIS synthetic image.



(a) Case 1: Low phase angle ϕ (light).



(b) Case 2: Mid ϕ (terminator).



(c) Case 4: High ϕ (shadow), high Γ.

Figure 5.15: Example of fusion application with corresponding VIS and TIR synthetic images.

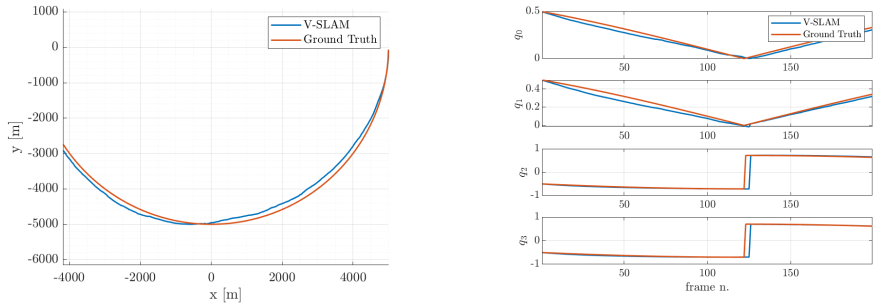
Relative navigation performance results In the present analysis it is assumed to have a VIS and TIR sensors with the same FoV and array size. Since the technology with the most relevant limitations is the TIR one, TIR characteristics presented in Table 5.2 are considered.

The same scenario and case studies as in Section 5.1 are considered. Results for the navigation are reported in Table 5.5. Please note that with respect to Section 5.1, the VIS sensors is now different.

Table 5.5: Localization error.

	VIS		**TIR**		**FUS**	
	e_ρ $[m]$	e_R $[deg]$	e_ρ $[m]$	e_R $[deg]$	e_ρ $[m]$	e_R $[deg]$
Case n.1	106.36	2.91	173.81	3.49	258.89	5.37
Case n.2	317.92	7.24	280.25	12.43	225.53	7.74
Case n.3	–	–	–	–	–	–
Case n.4-a	–	–	328.77	13.86	–	–
Case n.4-b	–	–	204.21	7.97	165.44	4.47

In case 1, the low ϕ case, it can be seen that FUS images allow navigation, but their performance is similar, even if slightly lower than the VIS and TIR one. This may be due to the fact that FUS images carry a larger noise and working with row data results in a better performance. The reconstructed pose with FUS images is shown in Fig. 5.16.



(a) Estimated trajectory.

(b) Estimated attitude.

Figure 5.16: SLAM results with FUS images for case 1: Low phase angle ϕ (light).

In case 2, FUS images show have the best performance for the translation part, while the rotation error is similar to the VIS images. In case 3, when both VIS and TIR images can not be exploited, also the FUS images do not allow the navigation, as the SLAM does not provide any meaningful solution. Case 4 has been divided in two sub-case: case 4-a represents the whole trajectory arc, while case 4-b includes only the first half of the arc. It has been noticed that navigation with FUS images is less robust than with TIR images only (case 4-b), as SLAM diverges sooner. Nevertheless, when the trajectory is shorter, FUS images do provide a best performance. Hence, FUS images can allow to better the TIR performance, at the price of a lower robustness.

As a general conclusion of this analysis, it can be said that the FUS images can lead to improvements to the SLAM performance. This analysis is the first step to define how to exploit the 3 types of data depending on some relevant mission parameters. The aim of this study was to provide an insight on the relationship between some major variables (i.e. the illumination and the thermal behavior of the body) and the navigation performance, as this constitutes a first guideline for the design of vision-based GNC systems based on a concurrent exploitation of multispectral imaging.

CHAPTER 6

Visual-LiDAR relative navigation systems

6.1 Visual-LiDAR pose estimation

An architecture to fuse the information from LiDAR and visual sensors to perform pose estimation with respect to an uncooperative target is proposed in Fig. 6.1. The point clouds of a dual-LiDAR scanning system are processed into a LiDAR Odometry (LO) pipeline, while the consecutive frames coming from a monocular camera are processed with a Visual Odometry (VO) pipeline. During the close approach phases with an uncooperative spacecraft, a large robustness needs to be granted. This requires not only a local pose estimation algorithm but also a global pose estimation block to be used for initialization and for lost-in-space cases. Hence, the architecture is provided with an initialization or re-initialization module, based on a Point Completion Network (PCN) for global pose estimation.

The rational behind such choices is to separate the tasks of local and global pose estimation, in order to split the need for robustness, granted by the pose initialization module, from the one of accuracy, provided by the VO and LO pipelines, keeping them computationally light.

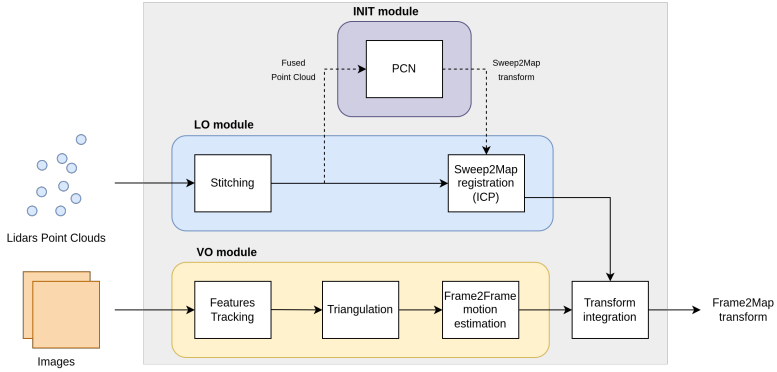


Figure 6.1: Visual-LiDAR pose estimation architecture.

Please note that the hereby considered target is a known object, but this framework can be extended to an unknown target, by adding blocks for the object map initialization and refinement and by extending the PCN to the usage with different objects classes.

In this Chapter, the blocks composing the LiDAR-based global pose estimation and Visual-LiDAR local pose estimation will be described, presenting results obtained with numerical and laboratory experiments.

6.2 LiDAR-based global pose estimation

6.2.1 Point Completion Network for pose initialization

The PCN architecture presented in Chapter 2 is employed for the global pose estimation. In particular, the pose estimation is accomplished by writing a *codebook* and a *labelbook*. After training the PCN with the target satellite point clouds at different poses, instances of the PCN are run with the training data, saving the global features v into the *codebook* and the associated ground truth pose into the *labelbook*.

Two different algorithms are exploited for the pose estimation task. The first one is based on the employment of the encoder part only. When a new test pose is run into the PCN, the closest global feature in the *codebook* is found with a cosine similarity. Then, the corresponding pose in the *labelbook* is read, retrieving the relative translation $\mathbf{t}$ and rotation $\mathbf{R}$.

Synthetic training dataset The synthetic dataset is built with the following constraints:

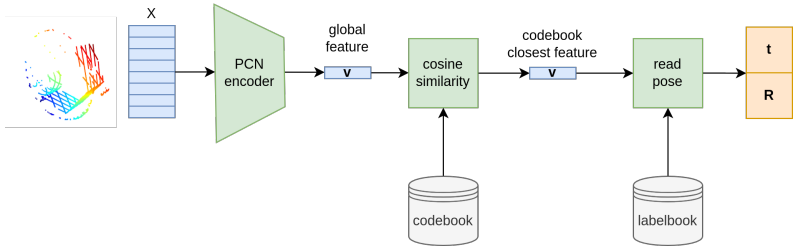


Figure 6.2: Pose estimation with PCN encoder.

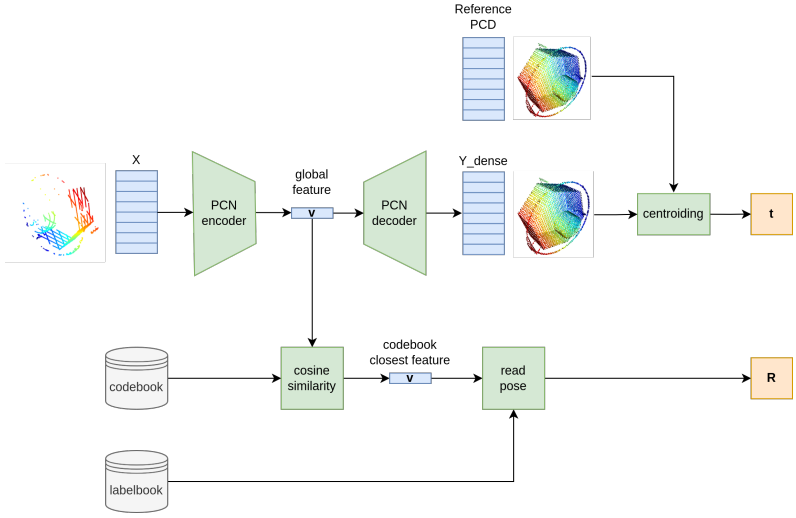


Figure 6.3: Pose estimation with PCN encoder-decoder.

- position of the LiDARs in front of the target;
- distance in range 1 m to 2.5 m;
- at least 50 % of the vertical LiDAR VFoV includes the target.

Except from the VFoV constraint, the attitude is randomly generated. The dataset is split in the classical manner into 80 % for training and validation and 20 % for testing. The synthetic dataset contains 16800 data points, and it is shown in Fig. 6.4.

(a) Relative translation.

(b) Relative rotations angles.

Figure 6.4: Synthetic dataset.

6.2.2 Point Completion Network results

Training results The training has been performed with the following parameters.

- Batch size: 32.
- Learning rate: 1e-4.
- Decay rate: 1e-6.
- Epochs: 300.

The used GPU is a NVIDIA® Quadro® GV100, and the training time 27 h. The learning curve is reported in Fig. 6.5.

Performance on synthetic dataset Four different algorithms are compared for the pose estimation purposes:

1. The architecture based on PCN encoder.
2. The architecture based on both PCN encoder and decoder.

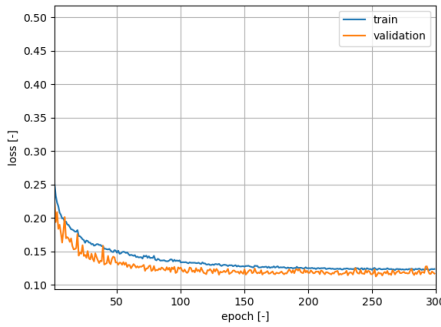


Figure 6.5: Learning curve.

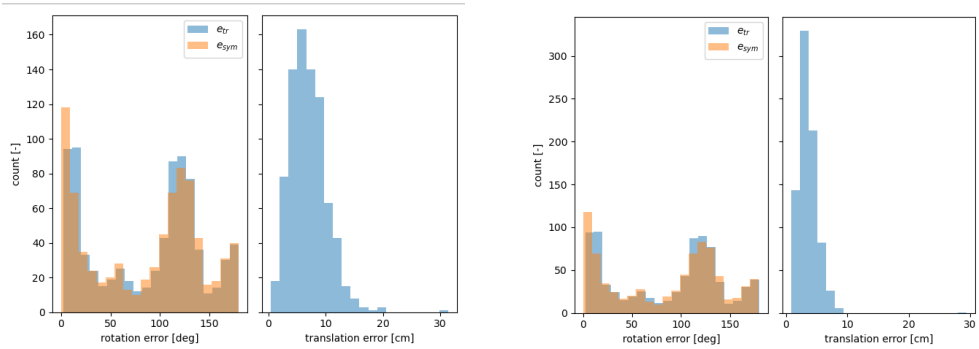
3. The architecture based on PCN encoder, followed by a nested ICP to refine the pose.

4. The architecture based on both PCN encoder and decoder, followed by a nested ICP to refine the pose.

As it can be seen in Fig. 6.6, the PCN is capable to correctly predict the translation, with a satisfying accuracy. The rotation estimation performance is not accurate and some peaks can be observed at 60°, 120° and 180°. This fact is related to the symmetry of the target, as the error on the symmetry axis e_{sym} almost overlaps with the overall rotation error e_r. This is confirmed by looking at the rotation error parametrizing the pose with Euler angles, as shown in Fig. 6.9.

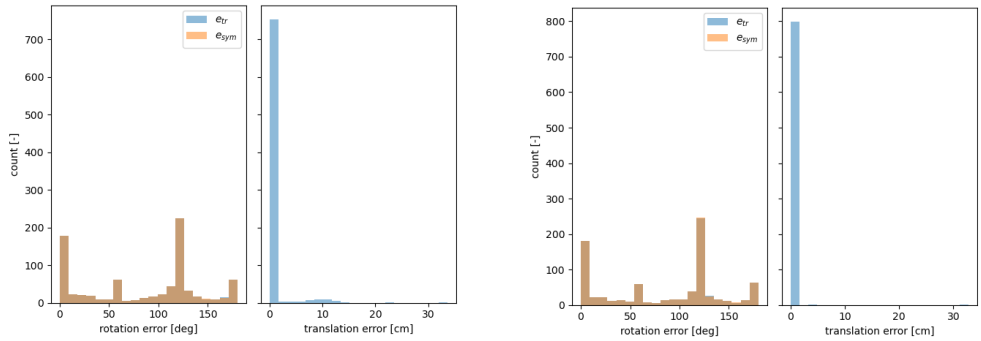
As it can be seen in Fig. 6.7 the pose estimation error largely benefits from the ICP refinement. So it can be concluded that even if the PCN performance is poor, it is nevertheless sufficient to allow the ICP convergence, thus providing finally a proper and accurate initialization of the pose. The peaks related to the symmetry axis are much more evident after the ICP, which converges almost always to the closest peak, providing a better symmetry breaking.

Geometric ambiguities Concerning the geometric ambiguities related to the symmetry, the PCN is capable of solving them up to 60°. This is thanks to the edges of the target spacecraft main body and to the 3 handles attached to it. The spacecraft has also a grasping point, which could allow to break the symmetry completely. It is observed that the grasping point in the point clouds is not easily visible, either because it is not included in the grid of points, or because it is not sufficiently thick to distinguish it from the external ring. In conclusion, the symmetry breaking task could benefit as well from the employment of both sensors, since the PCD is always capable to provide a



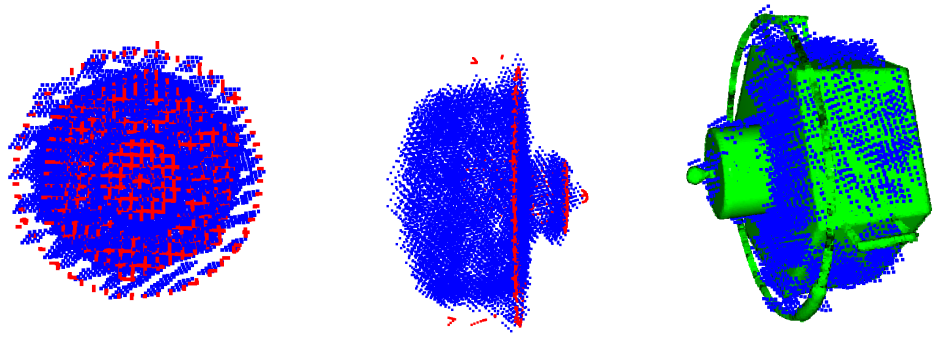
(a) Encoder. **(b)** Encoder + Decoder.

Figure 6.6: PCN pose estimation error.



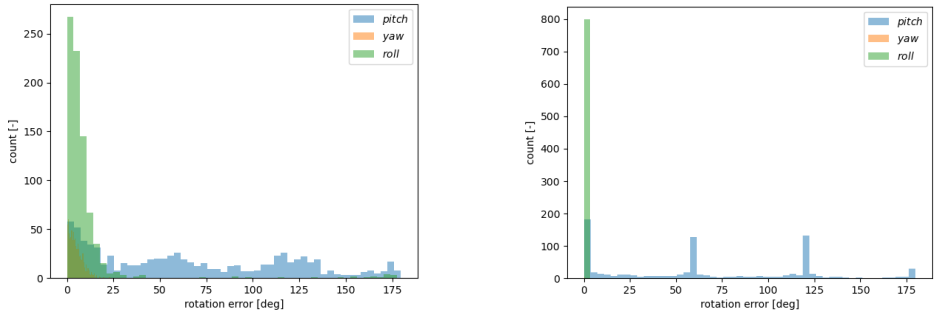
(a) Encoder. **(b)** Encoder + Decoder.

Figure 6.7: PCN + ICP pose estimation error.



(a) PCN dense point cloud and input scan. **(b)** PCN dense point cloud and input scan. **(c)** PCN dense point cloud and reference model.

Figure 6.8: Comparison among reconstructed dense point cloud (blue), input scan (red), and projected reference point cloud (green).



(a) PCN. **(b)** PCN + ICP.

Figure 6.9: PCN rotation error, looking at Euler angles.

partial breaking of the symmetry, and the images could more easily provide the information of semantic segmentation when the grasping point is in their FoV.

Experimental dataset acquisition The PCN module is tested in the OOS-SIM with experimental data.

The data are acquired in the facility and saved without running the pipeline online. The online run is still not ready to be performed for two main set-up-related reasons:

1. The output of the LiDAR system in the OOS-SIM consists into a 360° scan. Given the sensors ground truth trajectory, the laboratory point cloud needs to be processed to prune it from data points that would not exists in space (i.e. the ones of the room and of the target satellite base), as shown in Fig. 6.10.

2. Another problem is the time synchronization. This issue is solved by means of an artifact: since the sensors are not synchronized with the industrial arms logs, the two time clocks are aligned by recognizing when the arm movement starts, from the log data, and when changes are seen in the point clouds. This is done with an automatic procedure after the data acquisition.

Six different trajectories have been acquired, as reported in Table 6.1. The trajectories selection has followed the criteria of exploring different possible use-cases, given the spatial constraints available in the facility. For the second set of trajectories, it was possible to acquire both images and LiDAR scans. Please note that the axes referred to in Table 6.1 are not the ones of the sensors or satellites frames, but they are the ones of the industrial arms in the laboratory.

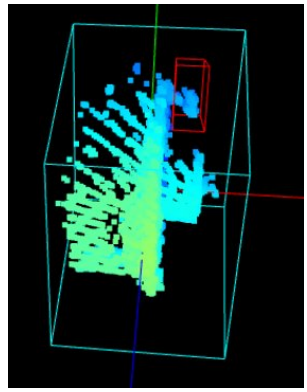


Figure 6.10: Cropping of the target satellite and its base.

Table 6.1: Real data sequences.

Sequence	Description	Points	Data
SQ-01	1-axis rotation	34	L1-L2
SQ-02	2-axes translation	29	L1-L2
SQ-03	1-axis translation	39	L1-L2
SQ-04	3-axes roto-translation	27 + 270	L1-L2 + CL-CR
SQ-05	1-axis rotation	21 + 210	L1-L2 + CL-CR
SQ-06	1-axis translation	40 + 400	L1-L2 + CL-CR

Performance on real dataset A dataset composed by real data sequences is created acquiring 6 sequences of data points, as reported in Table 6.1. The acquired sequences correspond to possible trajectories in proximity of the target. The global pose estimation algorithm shall be capable of initializing the sequence not only for an initial case, but more generally along the whole trajectory, in case of failure of the local pose estimation.

The estimation of the translation is still good with real data with the exception of SQ-04. The PCN is challenged by the real data appearance and the rotation estimation performance is degraded consistently. In particular, relevant errors are introduced also in the roll angle. Concerning this analysis, it must be highlighted that some of the sequences present data points that are outside the position box exploited for the training dataset. It can be said that the PCN is not robust for usage outside the training box, but a new real dataset shall be built for a more proper testing.

The comparison of the obtained errors for the two datasets is reported in Table 6.2.

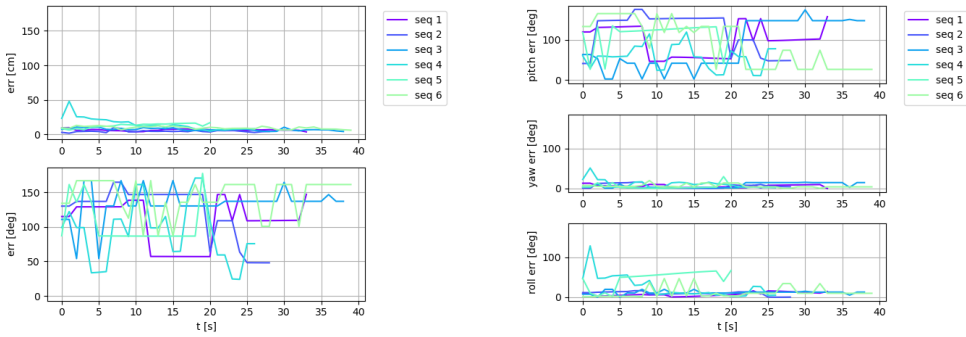
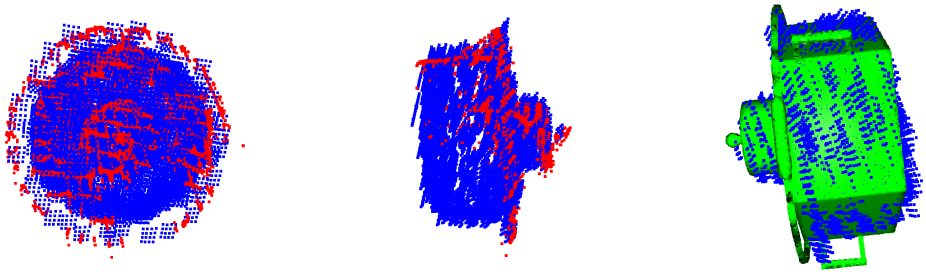


Figure 6.11: PCN position and attitude errors.



(a) PCN dense point cloud and input scan.

(b) PCN dense point cloud and input scan.

(c) PCN dense point cloud and reference model.

Figure 6.12: Comparison among reconstructed dense point cloud (blue), input scan (red), and projected reference point cloud (green).

Mean error	Synthetic dataset	Real dataset
e_t	0.287 cm	7.37 cm
e_{yaw}	0.068°	2.96°
e_{roll}	0.280°	10.22°

Table 6.2: Mean error for synthetic and real data.

Computational time The mean inference time of the PCN on a 8 GB NVIDIA-GeForce GTX 1080 is 0.867 s.

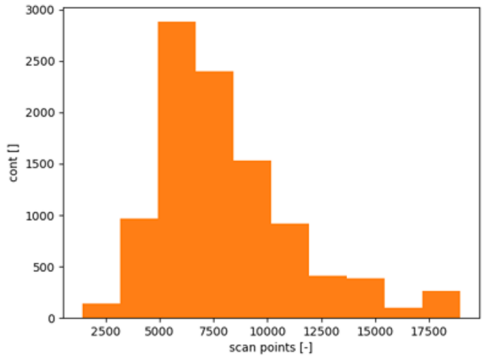


Figure 6.13: Number of points in the point clouds composing the synthetic dataset.

Some hints for reducing the computational time for an on-board implementation are the following:

- Optimizing the PCN architecture, reducing the number of parameters.
- Resampling the point cloud to reduce the number of points, or set a different resolution on the VLP-16 LiDAR, that can be lowered without adding a resampling step in the algorithm.
- Retaining only the encoder part in the on-board algorithm.

Conclusions The final conclusions are that:

- The global pose estimation has an accurate performance for translation on both synthetic and real data.
- The rotation estimation has a satisfying performance with synthetic data, but the performance is severely degradated with real data.

- The global pose estimation is capable to break the symmetry up to 60°, thanks to the combination of edges and handles. It could still benefit by fusion with a visual information related to the identification of the grasping point.

6.3 Visual-LiDAR local pose estimation

6.3.1 Visual-LiDAR Odometry

LiDAR Odometry The LiDAR Odometry takes in input a client scan $\mathcal{P}_1$ from the vertical LiDAR and another scan from the horizontal LiDAR $\mathcal{P}_2$. The two point clouds are stitched, composing the stitched point cloud $\mathcal{P}_{12} = \mathcal{P}_1 + \mathbf{H}_{\mathbf{L2}}^{\mathbf{L1}}(\mathcal{P}_2)$. Where $\mathbf{H}_{\mathbf{L2}}^{\mathbf{L1}}$ is the transformation from $L2$ to $L1$ frame, known from the system calibration.

Given a sufficiently close initial condition $\hat{\mathbf{H}}_{\mathbf{T}}^{\mathbf{L1}}$, the reference model point cloud $\mathcal{P}_{ref}$ is transformed from the target (or client) frame T_{TCP} into the L1 frame:

$$\mathcal{P}_{ref,t0} = \hat{\mathbf{H}}_{\mathbf{T}}^{\mathbf{L1}} \mathcal{P}_{ref} \tag{6.1}$$

A nested ICP algorithm is employed to compute the sweep refinement $\mathbf{H_{ICP}}|_{\mathbf{t0}}$ among $\mathcal{P}_{12,t0}$ and $\mathcal{P}_{ref,t0}$.

$$\mathbf{H}_{\mathbf{T}}^{\mathbf{L1}}|_{\mathbf{t0}} = \mathbf{H_{ICP}}|_{\mathbf{t0}} \hat{\mathbf{H}}_{\mathbf{T}}^{\mathbf{L1}} \tag{6.2}$$

The pose at the previous step is used as initial condition for the point cloud registration to the known model.

$$\mathcal{P}_{ref,t_k} = \mathbf{H_{ICP}}|_{\mathbf{t_{k-1}}} \dots \ \mathbf{H_{ICP}}|_{\mathbf{t0}} \mathbf{H}_{\mathbf{T}}^{\mathbf{L1}}|_{\mathbf{t0}} \ \mathcal{P}_{ref} \tag{6.3}$$

Hence the computed pose is always registered with respect to a known map, but it depends on the initial guess from the previous step:

$$\mathbf{H}_{\mathbf{T}}^{\mathbf{L1}}|_{\mathbf{t_k}} = \mathbf{H_{ICP}}|_{\mathbf{t_k}} \dots \ \mathbf{H_{ICP}}|_{\mathbf{t0}} \ \hat{\mathbf{H}}_{\mathbf{T}}^{\mathbf{L1}} \tag{6.4}$$

Visual Odometry The VO is initialized by extracting ORB features from the first available frame. The features are then tracked among frames computing the optical flow with the Lukas-Kanade Piramydal method.

The keypoints in the image are undistorted with the camera intrinsic parameters, known from the camera intrinsic calibration.

Then, the motion is estimated with 2D to 2D correspondences. The essential matrix is computed by means of the 5-point algorithm, exploiting the RANSAC method for removing outliers. The essential matrix is then decomposed with Singular Value Decomposition and then the possible roto-translations are recovered. The correct transformation $(\mathbf{R}, \mathbf{t})$ is selected by means of a Cheirality check.

Once the motion is computed, the keypoints are triangulated to create the current map. This step is necessary to retrieve the translation scale, which is computed from the ratio between corresponding point pairs among the old and new map.

Applying the scale to the translation, the transformation among subsequent frames $\mathbf{H}^{\mathbf{C_k}}_{\mathbf{C_{k-1}}}$ is thus computed. Then the transformations are concatenated to retrieve the pose with respect to the initial frame:

$$\mathbf{H}^{\mathbf{C_k}}_{\mathbf{C_0}} = \mathbf{H}^{\mathbf{C_k}}_{\mathbf{C_{k-1}}} \dots \mathbf{H}^{\mathbf{C_1}}_{\mathbf{C_0}} \tag{6.5}$$

Being the transformation of interest the one among servicer and target satellite, only one reference frame is retained for the whole sensors system. In particular, the *L1* frame is selected. Therefore, the camera pose is reported in the dual-LiDAR system reference frame and the initial pose with respect to the target is exploited to finally get the transformation of interest:

$$\mathbf{H}^{\mathbf{L1}}_{\mathbf{T}}|_{\mathbf{t_k}} = \mathbf{H}^{\mathbf{L_1}}_{\mathbf{C}} \ \mathbf{H}^{\mathbf{C_k}}_{\mathbf{C_0}} \ \mathbf{H}^{\mathbf{C}}_{\mathbf{L1}} \ \hat{\mathbf{H}}^{\mathbf{L1}}_{\mathbf{T}} \tag{6.6}$$

where $\mathbf{H}^{\mathbf{C}}_{\mathbf{L1}}$ and its inverse are known from the system calibration.

Transforms integration Ideally the two pipelines can run at different frequencies, for instance with the VO at a higher frequency.

The transformations can be integrated by resetting the initial condition for the camera motion at each time when a new LO measurement is available:

$$\mathbf{H}^{\mathbf{C_k}}_{\mathbf{C_0}} = \mathbf{H}^{\mathbf{L1}}_{\mathbf{T}}|_{\mathbf{t_k}} \ \mathbf{H}^{\mathbf{T}}_{\mathbf{L1}}|_{\mathbf{t_0}} \tag{6.7}$$

Of course this condition applies when all the sensors are fixed with the servicer satellite.

6.3.2 Visual-LiDAR Odometry results

The VO and LO modules are tested in the OOS-SIM using the experimental dataset already collected for the PCN.

6.3.2.1 LiDAR Odometry

The results of the LiDAR Odometry are reported in Fig. 6.14 for the 6 trajectories. The position is expressed in the L1 frame $\mathbf{x}_{c-L1}^{L1} = (x, y, z)^T$ and the rotation is parameterized in Euler angles $\mathbf{R}_{c-L1}^{L1} = \mathbf{R}_y(\beta)\mathbf{R}_z(\gamma)\mathbf{R}_x(\alpha)$, where α is the roll angle, β the pitch angle and γ the yaw angle. The presented results assume a perfect initial condition.

The LiDAR Odometry is successfully capable of computing the correct pose for all the sequences. The algorithm is quite robust, never leading to a divergence.

In the sequences where the position or rotation variation is less evident, it is possible to observe that the estimated pose presents an offset. The possible origins of such an error are the following.

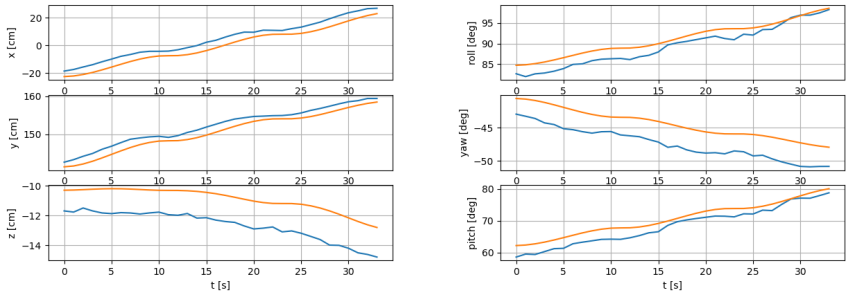
- **Time synchronization artifact**. The correspondence among the LiDAR scans and the ground truth time stamps is not exactly known. In fact, it can be observed in Fig. 6.14 that the time axis of the ground truth is not always coherent with the real scans time stamp. For instance, such shift is visible in the SQ-01 x and y coordinates and corresponds in this case to about 1.5 s. This error can vary from one trajectory to another.

- **Calibration accuracy**. The visible offset is often of the same order of magnitude as the calibration accuracy, i.e. about 3 cm in translation and 1.5° in rotation.

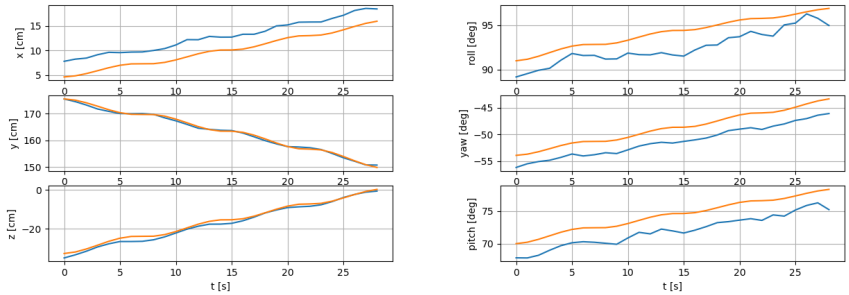
- **Boom flexibility**. The LiDARs are mounted on a flexible boom. Even if the trajectories have been acquired in conditions with no apparent vibration, the boom configuration could have been different than the calibration condition or some minor vibrations or deformations not visible by the human eye could be present along the trajectories.

- **Data noise**. As already shown, the client satellite point clouds present a consistent noise, due to the MLI reflections. Such data noise influences the ICP accuracy.

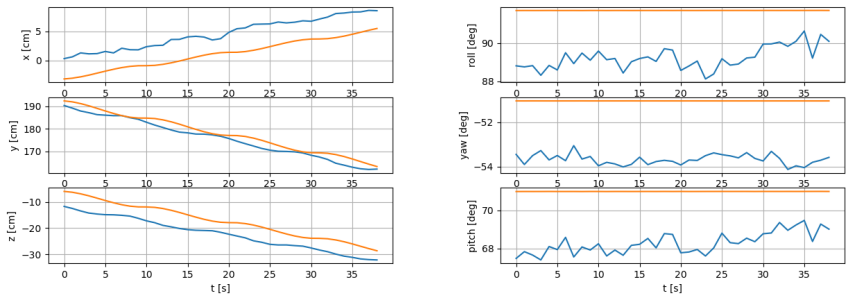
The error for the six sequences is shown in Fig. 6.16, with the statistics reported in Table 6.3. The general observation is that both the translation and rotation error standard deviations are low, compared to the mean error. Such observation is related to the offset noticed in Fig. 6.14. Another note is that the translation error is computed on vectors expressed in the L1 frame and not in an inertial frame.



(a) SQ-01.



(b) SQ-02.



(c) SQ-03.

Figure 6.14: LiDAR Odometry trajectories (blue: reconstructed, orange: ground-truth).

Table 6.3: LiDAR Odometry error statistics - real data.

Sequence	$\mu(e_t)$ [cm]	$\sigma(e_t)$ [cm]	$\mu(e_r)$ [deg]	$\sigma(e_r)$ [deg]
SQ-01	5.29	1.00	3.54	0.20
SQ-02	3.24	0.34	2.86	0.39
SQ-03	6.59	1.21	3.18	0.22
SQ-04	7.37	3.68	2.66	0.37
SQ-05	4.33	0.22	1.91	0.34
SQ-06	6.70	1.11	2.02	0.37

Robustness: sensitivity to initial condition A Montecarlo simulation is run to see the effect of a wrong initial condition on the LiDAR Odometry. For each run, the random initial condition has a rotation error of 5° and 10 cm.

As it can be seen in Fig. 6.15, the algorithm is robust enough to deal with such initial conditions. The result is reported for the SQ-04, which has the highest mean translation error, but the robustness is verified for all the sequences.

Comparison with synthetic data The LO is also tested with synthetic data. The same trajectories acquired in the laboratory are replicated with the LiDARs simulator presented in Section 3.1. It can be observed from Fig. 6.16 that all the sequences present a lower error with respect to real data, as expected. In particular, the error is one order of magnitude lower for both translation and rotation, with the exception on SQ-03, as reported in Table 6.4. The SQ-03 is a one-axis translation motion towards the target, so the subsequent point clouds are very similar with the exception of the component in the target symmetry axis direction. Therefore, the possible reason for this trajectory to be harder to be solved by the ICP could be related to the particular geometry of this sequence.

Table 6.4: LiDAR Odometry error statistics - synthetic data.

Sequence	$\mu(e_t)$ [cm]	$\sigma(e_t)$ [cm]	$\mu(e_r)$ [deg]	$\sigma(e_r)$ [deg]
SQ-01	0.06	0.02	0.13	0.09
SQ-02	0.07	0.02	0.11	0.07
SQ-03	2.74	2.71	0.29	0.69
SQ-04	0.08	0.02	0.09	0.07
SQ-05	0.07	0.02	0.16	0.11
SQ-06	0.07	0.02	0.09	0.05

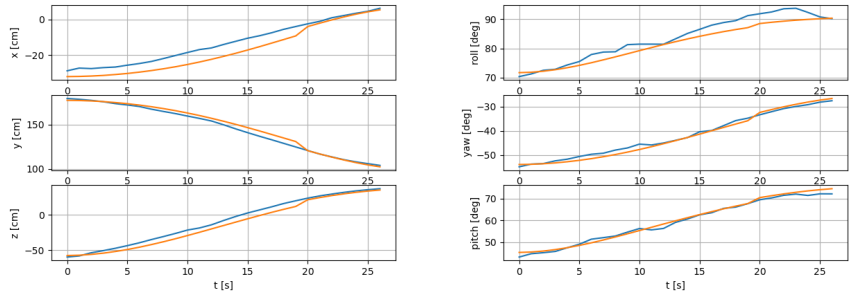
6.3.2.2 Visual Odometry

For what concerns the VO module, ORB features are extracted and tracked correctly, as shown in Fig. 6.17, notwithstanding the MLI reflections, visible in the images.

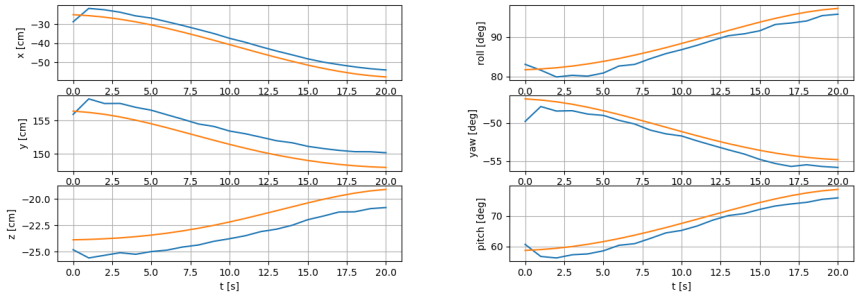
The performance results for VO are shown in Fig. 6.18. Three different frequencies have been tested: 0.5 Hz, 1 Hz, 10 Hz. As it can be seen, for the 1 Hz SQ-04 the VO begins to diverge after about 10 s, whereas at 0.5 Hz it is much more stable. It is evident that at 10 Hz the error accumulates fast. Therefore it is noticed that the best performance is obtained with the lowest frequency, while the error accumulates faster for higher frequencies. This might be related to two reasons:

1. the 2D to 2D motion estimation is particularly hard at high frequencies because the motion is very slow, and thus close to degenerate cases;
2. the error accumulation is related to the number of processing steps that are performed.

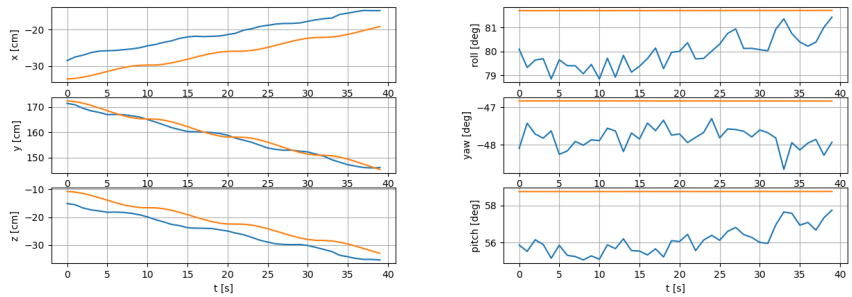
Conclusion It is concluded that the LO is more robust and accurate than the VO and that the integration of VO-LO can be advantageous only when the frequency of the VO is higher than the LO one. Another observation is that the performed tests were limited and did not include the case of a fast relative dynamics with a high speed tumbling target. In that case, the performance of VO could be the most robust and accurate one.



(d) SQ-04.



(e) SQ-05.



(f) SQ-06.

Figure 6.14: LiDAR Odometry trajectories (blue: reconstructed, orange: ground-truth).

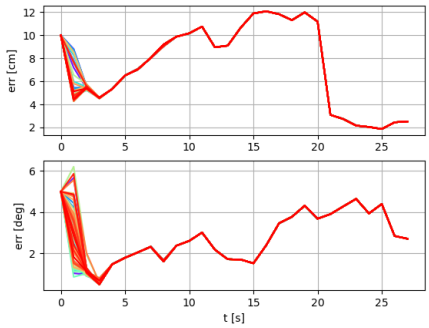
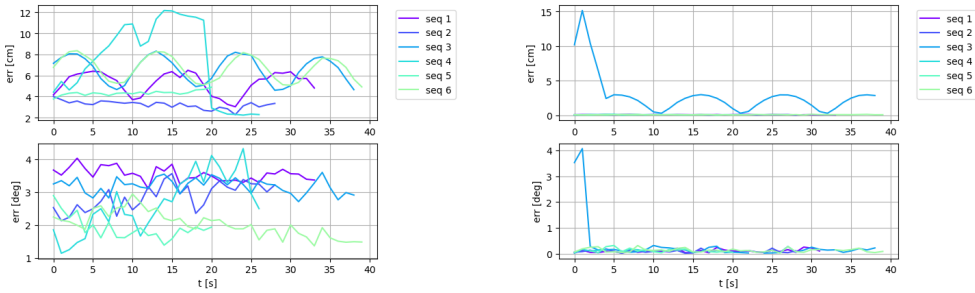
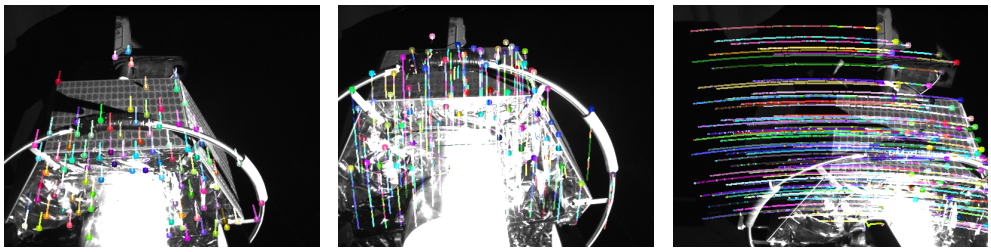


Figure 6.15: Example of LiDAR Odometry Montecarlo simulation: SQ-04.



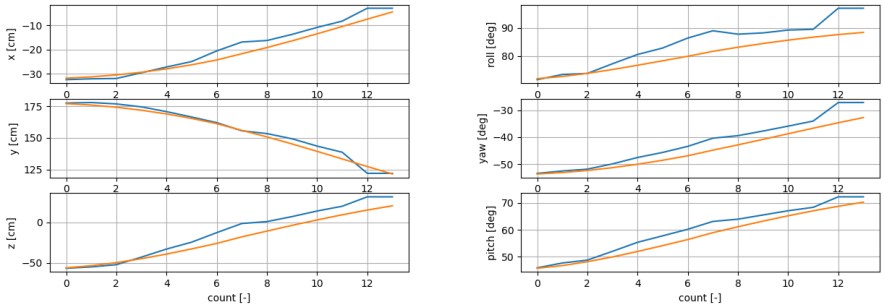
(a) Real data. **(b)** Synthetic data.

Figure 6.16: Translation and rotation error for sequences from SQ-01 to SQ-06.

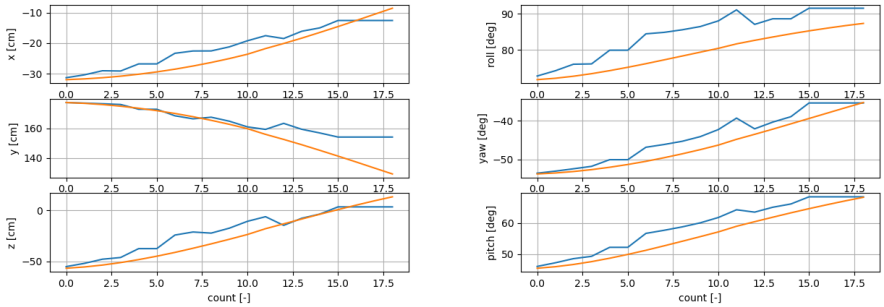


(a) SQ-04. **(b)** SQ-05. **(c)** SQ-06.

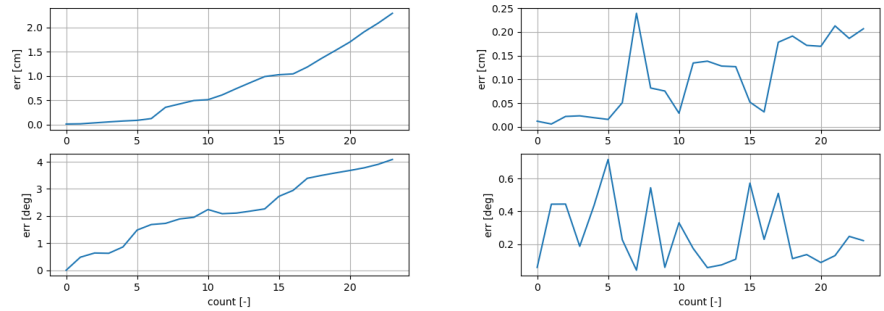
Figure 6.17: Examples of tracked features for sequences from SQ-04 to SQ-06.



(a) SQ-04 at 0.5 Hz (orange ground truth, blue VO).



(b) SQ-04 at 1 Hz (orange ground truth, blue VO).



(c) Error of SQ-04 at 10 Hz.

Figure 6.18: Translation and rotation error for sequence SQ-04.

CHAPTER 7

Conclusions

This dissertation dealt with relative navigation in proximity of uncooperative space targets, with the clear objective of improving or enabling operations by means of the clever exploitation of multiple sensors, namely visible cameras, thermal cameras and scanning LiDARs. Guidelines for the navigation sensors exploitation have been derived, highlighting the respective advantages and limitations in some applications of interest. Given the vastness of this topic, it has not been completely exhausted with this Ph.D. Thesis, which focused on some specific techniques among all the possible methodologies and on particular applications and scenarios. Nevertheless, the Thesis transversely covered various aspects of the development, verification and testing of relative navigation techniques with multiple sensors.

The starting topic tackled in the Thesis is the realistic sensor's data generation as preparatory part, necessary to the further studies and analyses. Visible images were generated employing three different existing rendering programs. Similarly, LiDARs scans are simulated employing an existing tool. Instead, due to the lack of an available free software, an innovative approach is proposed for the rendering of thermal images and specifically applied to small celestial

bodies. The method starts from a simplified thermal simulation of the asteroid, based on a coarse shape model. The output is then manipulated to obtain the radiance data of the object model in matrix form, that is then converted to a thermal infrared image by means of the sensor's model. Experimental activities are carried-out to gather visible images and LiDAR's scans, in two different space GNC labs. At PoliMi's ARGOS, images of a Moon diorama and of Envisat's mock-up are taken at different illumination conditions and poses for comparison with the corresponding synthetic images. At DLR's OOS-SIM, a multi-sensor calibration method is completed and performed to calibrate a dual-LiDAR system in the lab.

A data-level comparison is performed among visible synthetic and experimental images. A successful validation of the data is achieved; by means of comparison some differences among synthetic and real data which could not be removed are highlighted. The generated synthetic thermal images are compared with real mission data from Hayabusa 2; the capability of the tool of representing some major image features is shown, even if the low dimensional details are not always well represented, and future improvements on the thermal model are desirable. The multi-sensor calibration is successful and its accuracy is derived, finding the major error sources in the process. The LiDAR's scan noise is characterized and a qualitative comparison with synthetic point clouds is shown.

This concludes some preliminary activities that aim at critically contextualize the validity of the data used for developing and testing the navigation algorithms, subsequently investigated and proposed.

Dealing with relative navigation, the first part of the work focuses the employment of visible imaging. An Image Processing (IP) algorithm for the centralized relative navigation of far-range CubeSats orbiting around a binary asteroid system is designed, implemented and tested. Some major challenges are posed by the utilization of the mothercraft navigation camera: the detection and identification of multiple point targets over a starry background, the presence of two large celestial bodies with a low Sun phase angle, the lack of control over the mothercraft images, meaning the availability of low exposure images only and related to an attitude pointing to the primary asteroid. The proposed solution relies on the asteroids masking and then blobbing and centroiding to find all the point objects; geometric features are exploited to find patterns in the images, to be matched among two frames; the disparity of points is then evaluated and the CubeSats are identified by distinguishing their motion from the one of the stars. The IP has been numerically tested with synthetic images and the results show a low number of false positives, with a good accuracy of the CubeSats centroid identification. Among the many challenges, the most relevant one has been found to be the visibility of the CubeSats in

the images. A visibility analysis is performed to breakdown the causes of the lack of observations, considering geometric occlusions, shadowing and also the object brightness. It is observed that an improvement factor could be the exploitation of a camera working in a different spectral band.

Always dealing with visible imaging, the problem of mapping the surface of an asteroid with a camera is analyzed. In detail, an AI-policy for optimizing the images collection process is verified and tested in a twofold manner. First, an analysis is made verifying the AI-policy with input state beliefs, and then the benefits to the mapping process are cross-checked implementing a simplified ground-IP for the shape reconstruction. It is verified that the AI-policy leads to some improvements with respect to a uniform imaging scheduling and it is observed that the coverage of the body's surface can not be always completed because some areas persist to be in shadow. Again, some advantages could be provided by concurrently using a multispectral camera.

In the second part of the work, the possible enhancement of relative navigation with multispectral imaging is studied, considering the scenario of an unknown uncooperative target, in particular an asteroid. Starting from the examination and modeling of the current technologies of photon and thermal detectors, some considerations about the asteroid spectral radiation emission and reflection profiles are made and the visible and thermal LWIR bands are selected as the most promising for navigation.

A multi-modal Simultaneous Localization And Mapping (SLAM) architecture is selected and analyzed with the aim of defining the logic for switching from one sensor to the other. The vision-based SLAM algorithm is tested with visible (VIS) and thermal (TIR) synthetic images which correspond to realistic orbital and environmental conditions. The analysis considers the variation of relevant parameters which affect the appearance of the target in the visible and thermal infrared band: the Sun phase angle and the thermal inertia of the asteroid. The major outcomes of such analysis are the following. It is found that TIR sensors can actually be employed for navigating, offering an interesting opportunity for relative navigation in close proximity of asteroids. While both VIS and TIR sensors are exploitable, they contribute in different ways to navigation: VIS imaging is superior in terms of navigation accuracy and map quality, due to the higher resolution of the detectors available for this technology; TIR imaging enables the navigation in new scenarios, i.e. in the case of high phase angle with a high thermal inertia asteroid. Both sensors can not be employed when images are too dark, being the target comparable to the noise level of the image.

Then, a third alternative is introduced, proposing to fuse VIS and TIR images (FUS) and use them as a input to the SLAM algorithm. A multiscale transform method is implemented for image-level fusion, quantitatively comparing

the images obtained with a Laplacian and Wavelet transforms. The former transform is selected, incrementing the more the image entropy. Finally, a comparative analysis is made considering two sensors with the same FoV and images with the same resolution, to focus on the spectral bands aspect only. The major finding is that the FUS images can lead to improvements to the navigation error, but cause more easily a divergence of the SLAM.

Finally, the employment of LiDAR sensors is studied. The problem of pose estimation with a known uncooperative symmetric satellite is tackled by training a Point Completion Network (PCN). A dataset of synthetic scans is generated and used to train the PCN. After training, the global feature output by the PCN encoder part is stored in a *codebook* and the corresponding ground truth pose is saved in a *labelbook*. Two methods for pose estimation are compared: one relying on the global feature encoding only and the other also on the dense point cloud reconstructed by the PCN, deriving the relative translation vector by centroiding the dense point cloud with a reference point cloud. In both cases the pose is then refined with an Iterative Closest Point (ICP) algorithm. The two methods are tested with a synthetic testing dataset and compared. Both methods prove to be suitable for providing an initialization of the pose, being the second method more accurate for the translation estimation. The PCN accuracy enables the convergence of ICP, that allows to reach an accurate solution. The global pose estimation is capable to break the symmetry only partially, thanks to the combination of the target satellite's edges and handles. It could still benefit by fusion with a visual information related to the identification of a grasping point present on the target, too thin to be appreciable in the point clouds.

Considering the necessary computational time, the suggested employment of the PCN architecture is the pose initialization, while another lighter method can be used for the pose tracking. For this purpose, a Visual-Lidar Odometry (VO-LO) is proposed and implemented. The LO has been tested on different trajectories, starting from ideal and perturbed initial conditions; it is observed that the LO successfully tracks the pose and is highly robust. A first implementation of the VO module is tested; as first result it is observed that the error growth is dependent on processed frames selected frequency. Hence it is deduced that a key factor for transform integration with the LO module is the LiDAR versus camera working frequency. In particular, the VO-LO integration can be useful when the VO frequency is larger than the LO frequency.

A comparison of the performance with input synthetic and real point clouds is carried-out. Regarding the pose initialization module, the global pose estimation has an accurate performance for translation on both synthetic and real data. The rotation estimation has a satisfying performance with synthetic data, but the performance is severely degraded with real data. Concerning the

pose tracking, the accuracy is quite good in both case, event if it is found the LO error with real scans is about 2 orders of magnitude larger with respect to synthetic sequences. Even if synthetic data allow to cover major aspects in algorithms development and testing, even without a perfect sim2real framework, it has been shown that real data do still affect in a relevant way the navigation performance.

As a general conclusion, all the different approaches presented in this Thesis provide possible answers to the main challenges of relative navigation with uncooperative objects. It has been shown that proximity operations can benefit from the usage of multiple sensors, deriving some guidelines and insights on this vast topic. Based on the findings of the work, a road-map with indications for future developments is here outlined:

- regarding the synthetic data generation, future work can try to further reduce the gap with respect to experimental data, refining the modeling of objects and their materials' properties. For the thermal images generation, analyses can be carried-out with a dedicated thermal modeling software, to deal with higher resolution shape models and with artificial targets. The extension of the tool developed in this Thesis is already on-going at Politecnico di Milano [88], as the tool has been designed since its first conception to be interfaced with the mentioned thermal analyses programs.
- it can be interesting to investigate also other methods for fusing the sensor's information, comparing the approaches. For instance, methods, that here were only in part investigated, can perform the fusion at different stages (at data-level, within a tightly fused processing, after the data processing, or within the navigation filter).
- the presented approaches can be consolidated with more thorough test campaigns, expanding the test cases and considering not only data from real sensors but proceeding also to Processor-In-the-Loop and complete Hardware-In-the-Loop tests.

Bibliography

[1] R. Opromolla, G. Fasano, G. Rufino, and M. Grassi, "A review of cooperative and uncooperative spacecraft pose determination techniques for close-proximity operations", *Progress in Aerospace Sciences*, vol. 93, pp. 53–72, 2017.

[2] L. P. Cassinis, R. Fonod, and E. Gill, "Review of the robustness and applicability of monocular pose estimation systems for relative navigation with an uncooperative spacecraft", *Progress in Aerospace Sciences*, vol. 110, p. 100 548, 2019.

[3] S. D'Amico, M. Benn, and J. L. Jørgensen, "Pose estimation of an uncooperative spacecraft from actual space imagery", *International Journal of Space Science and Engineering 5*, vol. 2, no. 2, pp. 171–189, 2014.

[4] J. A. Christian and S. Cryan, "A survey of LIDAR technology and its use in spacecraft relative navigation", in *AIAA Guidance, Navigation, and Control (GNC) Conference*, 2013, p. 4641.

[5] A. Masson, C. Haskamp, I. Ahrns, *et al.*, "Airbus DS Vision Based Navigation solutions tested on LIRIS experiment data", in *ESA 7th Space Debris Conference*, ESA Space Debris Office Darmstadt, Germany, vol. 2017, 2017.

[6] N. Ogawa, F. Terui, Y. Mimasu, *et al.*, "Image-based autonomous navigation of Hayabusa2 using artificial landmarks: The design and brief in-flight results of the first landing on asteroid Ryugu", *Astrodynamics*, vol. 4, no. 2, pp. 89–103, 2020.

[7] G. Ono, F. Terui, N. Ogawa, *et al.*, "Design and flight results of GNC systems in Hayabusa2 descent operations", *Astrodynamics*, vol. 4, no. 2, pp. 105–117, 2020.

[8] C. Norman, C. Miller, R. Olds, *et al.*, "Autonomous Navigation Performance Using Natural Feature Tracking during the OSIRIS-REx Touch-and-Go Sample Collection Event", *The Planetary Science Journal*, vol. 3, no. 5, p. 101, 2022.

[9] D. Bekker, R. Smith, and M. Q. Tran, "Guiding DART to Impact - the FPGA SoC Design of the DRACO Image Processing Pipeline", in *2021 IEEE Space Computing Conference (SCC)*, IEEE, 2021, pp. 122–133.

[10] B. P. Rush, D. M. Mages, A. T. Vaughan, J. Bellerose, and S. Bhaskaran, "Optical Navigation for the DART Mission", in *3rd Space Imaging Workshop*, 2022.

[11] ESA-TECSAG-TN-011315, *HERA: Proximity Operations*. ESA, 2018.

[12] A. Hall, P. Steele, J. Moulin, and E. Ferreira, "Airbus Active Debris Removal Service", *Advances in Astronautics Science and Technology*, vol. 4, no. 1, pp. 1–10, 2021.

[13] O. Groussin, N. Attree, Y. Brouet, *et al.*, "The thermal, mechanical, structural, and dielectric properties of cometary nuclei after Rosetta", *Space Science Reviews*, vol. 215, no. 4, pp. 1–51, 2019.

[14] T. Okada, T. Fukuhara, S. Tanaka, *et al.*, "Thermal infrared imaging experiments of C-type asteroid 162173 Ryugu on Hayabusa2", *Space Science Reviews*, vol. 208, no. 1, pp. 255–286, 2017.

[15] T. Okada, "Thermography of Asteroid and Future Applications in Space Missions", *Applied Sciences*, vol. 10, no. 6, p. 2158, 2020.

[16] R. Gade and T. B. Moeslund, "Thermal cameras and applications: a survey", *Machine vision and applications*, vol. 25, no. 1, pp. 245–262, 2014.

[17] A. Heyden and M. Pollefeys, "Multiple view geometry", *Emerging topics in computer vision*, vol. 3, pp. 45–108, 2005.

[18] Teledyne FLIR. (2023). The ultimate infrared handbook for R&D professionals, [Online]. Available: `https://www.flirmedia.com/MMC/THG/Brochures/T559243/T559243\_EN.pdf` (visited on 07/01/2021).

[19] A. M. Didion, A. K. Nicholas, J. E. Riedel, R. J. Haw, and R. C. Woolley, "Methods for Passive Optical Detection and Relative Navigation for Rendezvous with a Non-Cooperative Object at Mars", in *AAS/AIAA Astrodynamics Specialist Conference*, Snowbird, UT, 2018.

[20] A. B. Katake, "Modeling, image processing and attitude estimation of high speed star sensors", Ph.D. Dissertation, Texas A&M University, 2006.

[21] E. Brageot, O. Groussin, P. Lamy, and J.-L. Reynaud, "Experimental study of an uncooled microbolometer array for thermal mapping and spectroscopy of asteroids", *Experimental Astronomy*, vol. 38, no. 3, pp. 381–400, 2014.

[22] J. Zhang and S. Singh, "LOAM: Lidar Odometry and Mapping in Real-time.", in *Robotics: Science and Systems*, Berkeley, CA, vol. 2, 2014, pp. 1–9.

[23] P. Jasiobedzki, S. Se, T. Pan, M. Umasuthan, and M. Greenspan, "Autonomous satellite rendezvous and docking using lidar and model based vision", vol. 5798, SPIE, May 2005, p. 54. DOI: 10.1117/12.604011.

[24] R. Opromolla, G. Fasano, G. Rufino, and M. Grassi, "Uncooperative pose estimation with a LIDAR-based system", *Acta Astronautica*, vol. 110, pp. 287–297, May 2015. DOI: 10.1016/j.actaastro.2014.11.003.

[25] F. Rems, J. A. Moreno González, T. Boge, S. Tuttas, and U. Stilla, "Fast initial pose estimation of spacecraft from LIDAR point cloud data", 2015.

[26] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation", Dec. 2016.

[27] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space", Jun. 2017.

[28] W. Yuan, T. Khot, D. Held, C. Mertz, and M. Hebert, "Pcn: Point completion network", in *2018 International Conference on 3D Vision (3DV)*, IEEE, 2018, pp. 728–737.

[29] K. Nguyen Ba, "Lidar-based 6D Pose Estimation of a Satellite in an On-Orbit-Servicing Simulator (OOS-SIM) Scenario", Master's thesis, Saarland University, 2021.

[30] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 652–660.

[31] P. V. K. Borges and S. Vidas, "Practical infrared visual odometry", *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 8, pp. 2205–2213, 2016.

[32] J. Ma, Y. Ma, and C. Li, "Infrared and visible image fusion methods and applications: A survey", *Information Fusion*, vol. 45, pp. 153–178, 2019.

[33] A. Beauvisage, N. Aouf, and H. Courtois, "Multi-spectral visual odometry for unmanned air vehicles", in *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, IEEE, 2016, pp. 001 994–001 999.

[34] L. Chen, L. Sun, T. Yang, L. Fan, K. Huang, and Z. Xuanyuan, "Rgb-t slam: A flexible slam framework by combining appearance and thermal information", in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2017, pp. 5682–5687.

[35] M. Magnabosco and T. P. Breckon, "Cross-spectral visual simultaneous localization and mapping (SLAM) with sensor handover", *Robotics and Autonomous Systems*, vol. 61, no. 2, pp. 195–208, 2013.

[36] O. Yilmaz, N. Aouf, L. Majewski, M. Sanchez-Gestido, and G. Ortega, "Using infrared based relative navigation for active debris removal", in *ESA*

GNC 2017: 10th International ESA Conference on Guidance, Navigation & Control systems, 2017.

[37] Ö. Yılmaz, N. Aouf, E. Checa, L. Majewski, and M. Sanchez-Gestido, "Thermal analysis of space debris for infrared-based active debris removal", *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 233, no. 3, pp. 811–822, 2019.

[38] A. Fiengo, J. Bravo, P. Rosa, N. Guercio, J. Christy, and V. Dubanchet, "Multispectral Sensing for Relative Navigation", in *EUCASS, 8th european Conference for Aeronautics and Space Sciences*, 2019.

[39] M. Vasile, L. Walker, R. D. Dunphy, *et al.*, "Intelligent characterisation of space objects with hyperspectral imaging", *Acta Astronautica*, 2022.

[40] J. Zhang and S. Singh, "Visual-lidar odometry and mapping: Low-drift, robust, and fast", vol. 2015-June, Institute of Electrical and Electronics Engineers Inc., Jun. 2015, pp. 2174–2181. DOI: 10.1109/ICRA.2015.7139486.

[41] J. Graeter, A. Wilczynski, and M. Lauer, "Limo: Lidar-monocular visual odometry", in *2018 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, IEEE, 2018, pp. 7872–7879.

[42] S.-S. Huang, Z.-Y. Ma, T.-J. Mu, H. Fu, and S.-M. Hu, "Lidar-monocular visual odometry using point and line features", in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2020, pp. 1091–1097.

[43] A. D. Stewart and P. Newman, "Laps-localisation using appearance of prior structure: 6-dof monocular camera localisation using prior pointclouds", in *2012 IEEE International Conference on Robotics and Automation*, IEEE, 2012, pp. 2625–2632.

[44] Y.-S. Shin, Y. S. Park, and A. Kim, "Dvl-slam: sparse depth enhanced direct visual-lidar slam", *Autonomous Robots*, vol. 44, no. 2, pp. 115–130, 2020.

[45] W. Wang, J. Liu, C. Wang, B. Luo, and C. Zhang, "DV-LOAM: Direct visual lidar odometry and mapping", *Remote Sensing*, vol. 13, no. 16, p. 3340, 2021.

[46] European Space Agency. (2023). Cosmos: the portal for users of ESA's science directorate's missions, [Online]. Available: https://www.cosmos.esa.int/ (visited on 02/01/2023).

[47] National Aeronautics and Space Administration. (2023). NASA's Planetary Data System, [Online]. Available: https://pds.nasa.gov/ (visited on 02/01/2023).

[48] Japan Aerospace Exploration Agency. (2023). JAXA's DARTS: Lunar and Planetary Science, [Online]. Available: https://www.darts.isas.jaxa.jp/planet/ (visited on 02/01/2023).

[49] T. Plachetka, "POV Ray: persistence of vision parallel raytracer", in *Proc. of Spring Conf. on Computer Graphics, Budmerice, Slovakia*, vol. 123, 1998, p. 129.

[50] Blender Online Community, *Blender - a 3D modelling and rendering package*, Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018.

[51] S. Parkes, I. Martin, M. Dunstan, and D Matthews, "Planet surface simulation with PANGU", in *Space OPS 2004 conference*, 2004, p. 389. DOI: 10.2514/6.2004-592-389.

[52] T. H. Park, M. Märtens, G. Lecuyer, D. Izzo, and S. D'Amico, "SPEED+: Next-generation dataset for spacecraft pose estimation across domain gap", in *2022 IEEE Aerospace Conference (AERO)*, IEEE, 2022, pp. 1–15.

[53] M. Avilés, D. Mora, M. Canetri, and P. Colmenarejo, "A complete IP-based Navigation Solution for the Approach and Capture of Active Debris", in *67th Int. Astronautical Congress (IAC), Guadalajara, Mexico*, 2016.

[54] J. Tao, Y. Cao, M. Ding, and Z. Zhang, "Visible and Infrared Image Fusion-Based Image Quality Enhancement with Applications to Space Debris On-Orbit Surveillance", *International Journal of Aerospace Engineering*, vol. 2022, 2022.

[55] I. Martin, M. Dunstan, D. Vural, and M. Gestido, "Simulating Thermal Infrared Images of Planets, Asteroids and Spacecraft", English, in *8th ICATT (International Conference on Astrodynamics Tools and Techniques) 2021 conference proceedings*, vol. ICATT, Netherlands: European Space Agency, Aug. 2021.

[56] OpenCV team. (2023). OpenCV library., [Online]. Available: https://opencv.org (visited on 02/01/2023).

[57] F. Fraundorfer and D. Scaramuzza, "Visual odometry: Part ii: Matching, robustness, optimization, and applications", *IEEE Robotics & Automation Magazine*, vol. 19, no. 2, pp. 78–90, 2012.

[58] Applied Coherent Technology. (2023). LROC Master database, [Online]. Available: http://target.lroc.asu.edu/output/lroc/lroc_page.html (visited on 02/01/2023).

[59] G. Neukum, B. König, and J. Arkani-Hamed, "A study of lunar impact crater size-distributions", *The moon*, vol. 12, no. 2, pp. 201–229, 1975.

[60] N. Rowell, S. Parkes, M. Dunstan, and O. Dubois-Matra, "PANGU: Virtual spacecraft image generation", in *5th Int. Conf. on Astrodynamics Tools and Techniques, ICATT*, 2012.

[61] S. Holger, H. Keller, R. Jaumann, H Michalik, B. T, B. F, *et al.*, "The Dawn framing camera", *Space science reviews*, vol. 163, no. 1-4, pp. 263–327, 2011.

[62] J. Artigas, M. De Stefano, W. Rackl, *et al.*, "The OOS-SIM: An on-ground simulation facility for on-orbit servicing robotic operations", in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2015, pp. 2854–2860.

[63] H. Mishra, A. M. Giordano, M. De Stefano, R. Lampariello, and C. Ott, "Inertia-Decoupled Equations for Hardware-in-the-Loop Simulation of an Orbital Robot with External Forces", in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2020, pp. 1879–1886.

[64] R. Lampariello, H. Mishra, N. Oumer, P. Schmidt, M. De Stefano, and A. Albu-Schäffer, "Tracking control for the grasping of a tumbling satellite with a free-floating robot", *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3638–3645, 2018.

[65] J. L. Schönberger and J.-M. Frahm, "Structure-from-Motion Revisited", in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[66] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[67] K. H. Strobl, W. Sepp, S. Fuchs, C. Paredes, M. Smisek, and K. Arbter. (). DLR CalDe and DLR CalLab, Institute of Robotics and Mechatronics, German Aerospace Center (DLR), [Online]. Available: `http://www.robotic.dlr.de/callab/`.

[68] B. Cavrois, A. Vergnol, A. Donnard, P. Casiez, and O. Mongrard, "LIRIS demonstrator on ATV5: a step beyond for European non cooperative navigation system", in *AIAA guidance, navigation, and control conference*, 2015, p. 0336.

[69] M. Kisantal, S. Sharma, T. H. Park, D. Izzo, M. Märtens, and S. D'Amico, "Satellite pose estimation challenge: Dataset, competition design, and results", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 56, no. 5, pp. 4083–4098, 2020.

[70] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF", in *2011 International conference on computer vision*, Ieee, 2011, pp. 2564–2571.

[71] JAXA. (2023). Hayabusa 2 Science Data Archives, [Online]. Available: `https://www.darts.isas.jaxa.jp/planet/project/hayabusa2/` (visited on 10/01/2022).

[72] Y. Shimaki, H. Senshu, N. Sakatani, *et al.*, "Thermophysical properties of the surface of asteroid 162173 Ryugu: infrared observations and thermal inertia mapping", *Icarus*, vol. 348, p. 113 835, 2020.

[73] H. R. Goldberg, Ö. Karatekin, B. Ritter, *et al.*, "The Juventas cubesat in support of ESA's Hera mission to the asteroid Didymos", 2019.

[74] S. Silvestrini, A. Capannolo, M. Piccinin, M. Lavagna, and J. Gil-Fernandez, "Centralized Autonomous Relative Navigation of Multiple

Spacecraft Around Small Bodies", in *AIAA Scitech 2020 Forum*, 2020, pp. 1–20. DOI: 10.2514/6.2020-1204.

[75] R. C. Stone, "A comparison of digital centering algorithms", *The Astronomical Journal*, vol. 97, pp. 1227–1237, 1989.

[76] C. C. Liebe, "Pattern recognition of star constellations for spacecraft applications", *IEEE Aerospace and Electronic Systems Magazine*, vol. 7, no. 6, pp. 34–41, 1992.

[77] B. Rosner, "Percentage points for a generalized ESD many-outlier procedure", *Technometrics*, vol. 25, no. 2, pp. 165–172, 1983.

[78] V. Pesce, A.-a. Agha-mohammadi, and M. Lavagna, "Autonomous navigation & mapping of small bodies", in *2018 IEEE Aerospace Conference*, IEEE, 2018, pp. 1–10. DOI: 10.1109/AERO.2018.8396797.

[79] M. Riedmiller, "10 steps and some tricks to set up neural reinforcement controllers", in *Neural networks: tricks of the trade*, Springer, 2012, pp. 735–757. DOI: 10.1007/978-3-642-35289-8_39.

[80] V. Mnih, K. Kavukcuoglu, D. Silver, *et al.*, "Human-level control through deep reinforcement learning", *Nature*, vol. 518, no. 7540, p. 529, 2015. DOI: 10.1038/nature14236.

[81] M. Piccinin, V. Pesce, S. Silvestrini, and M. Lavagna, "Smart Autonomous Imaging Plan For Small Bodies Efficient Mapping", in *Advanced Space Technologies in Robotics and Automation ASTRA*, 2019.

[82] B. Giese, J. Oberst, R. Kirk, and W. Zeitler, "The topography of asteroid Ida: A comparison between photogrammetric and two-dimensional photoclinometric image analysis", *International Archives of Photogrammetry and Remote Sensing*, vol. 31, B3, 1996.

[83] M. Piccinin, G. L. Civardi, M. Quirino, and M. Lavagna, "Multispectral Imaging Sensors for asteroids relative navigation", in *72nd International Astronautical Congress (IAC 2021)*, 2021, pp. 1–12.

[84] Watanabe et al. (2019). (2019). Shape models, their derivatives, and relating data and tools of the asteroid Ryugu, [Online]. Available: https://data.darts.isas.jaxa.jp/pub/hayabusa2/paper/Watanabe_2019/README.html (visited on 07/01/2022).

[85] Y. Shimaki, H. Senshu, N. Sakatani, *et al.*, "Thermophysical properties of the surface of asteroid 162173 Ryugu: Infrared observations and thermal inertia mapping", *Icarus*, vol. 348, p. 113 835, 2020. DOI: https://doi.org/10.1016/j.icarus.2020.113835.

[86] K. Kitazato, R. Milliken, T. Iwata, *et al.*, "The surface composition of asteroid 162173 Ryugu from Hayabusa2 near-infrared spectroscopy", *Science*, vol. 364, no. 6437, pp. 272–275, 2019.

[87] L. Markley and D. Mortari, "Quaternion Attitude Estimation Using Vector Observations", *Journal of the Astronautical Sciences*, vol. 48, Apr. 2000. DOI: 10.1007/BF03546284.

[88] G. L. Civardi, M. Bechini, A. Colombo, M. Quirino, M. Piccinin, M. Lavagna, *et al.*, "Vis-tir imaging for uncooperative objects proximity navigation: a tool for development and testing", in *11th International Workshop on Satellite Constellations Formation Flying (IWSCFF 2022), International Astronautical Federation, IAF, Milan, Italy*, vol. 6, 2022.

Colophon

This thesis was typeset with LaTeX and BibTeX, using a typographical look-and-feel created by Margherita Piccinin. The style was inspired by A. Capannolo, S. Silvestrini, A. Colagrossi, D.A. Dei Tos `PhD_Dis` and by J. Stevens, L. Fossati `phdthesis` styles.