CRANFIELD UNIVERSITY

MARCO ZACCARIA DI FRAIA

A FRAMEWORK FOR OPTICAL FEATURES SELECTION AND MANAGEMENT FOR CAMERA-ONLY AUTONOMOUS NAVIGATION IN THE PROXIMITY TO SMALL CELESTIAL OBJECTS

Cranfield Defence and Security

PhD Thesis
Academic Year: 2020 - 2021

Supervisor: Lounis Chermak
Associate Supervisor: Joan Pau Sanchez Cuartielles
August 2021

CRANFIELD UNIVERSITY

Cranfield Defence and Security

PhD

Academic Year 2020 - 2021

MARCO ZACCARIA DI FRAIA

A FRAMEWORK FOR OPTICAL FEATURES SELECTION AND MANAGEMENT FOR CAMERA-ONLY AUTONOMOUS NAVIGATION IN THE PROXIMITY TO SMALL CELESTIAL OBJECTS

Supervisor: Lounis Chermak
Associate Supervisor: Joan Pau Sanchez Cuartielles
August 2021

To all those who feel lost.

“It is sometimes said that scientists are unromantic, that their passion to figure out robs the world of beauty and mystery. But is it not stirring to understand how the world actually works — that white light is made of colors, that color is the way we perceive the wavelengths of light, that transparent air reflects light, that in so doing it discriminates among the waves, and that the sky is blue for the same reason that the sunset is red? It does no harm to the romance of the sunset to know a little bit about it.”

(C. Sagan)

-

“There ain't no such thing as a free lunch”

(Popular adage, but often attributed to R. Heinlein)

ABSTRACT

Small celestial bodies such as asteroids and comets are abundantly present in the Solar System, yet their surfaces remain largely unexplored. Achieving regular access to these surfaces would have a major impact on capabilities such as planetary defence and in situ resource utilisation and lead to significant scientific insights.

However, missions close to small celestial objects remain challenging in at least two aspects: technically, due to weak gravity fields, complex operational environments and latency from long communication times, and commercially, with the applications still being few and cost-ineffective.

A potential solution to reducing development and operational costs and obtaining robust, scalable operations, could be using small, camera-only spacecraft with an elevated degree of autonomy. Enabling a camera-based autonomy requires building appropriate computer vision pipelines. All computer vision pipelines start with the detection of features - salient patterns within the scene. This thesis presents multiple methods and tools enabling the appropriate selection and management of different features for autonomous navigation in proximity to asteroids.

To that end, relevant contributions developed during this work consist of:

- The development of a software toolbox for prototyping and testing optical navigation technologies through a parametrisable synthetic 3D visual environment;
- An analysis of the response of feature detectors to internal factors (e.g., feature model) and external factors (e.g., illumination). This response, once known, can be used for designing the system or to obtain situational awareness
- An assessment of the response of template matching methods when the template (model) does not perfectly match the observed target (asteroid, with illumination).

Through the above contributions, it was shown that considering environmental cues and the perception model helps in achieving robust camera-only navigation processes.

This capability could lead to small satellites autonomously exploring hundreds or thousands of small celestial objects or be employed on more powerful spacecraft for redundancy.

Keywords:

Computer Vision, Optical Navigation, Feature Detection

ACKNOWLEDGEMENTS

The PhD is nothing, but a mere background,

A scene with fake trees to which one feels bound

Till one's not, and the PhD is fed on being around

Exploring the world, embarking on journeys

That strip away formalised worries

A school will give you an award or a title

But a waste of time is aiming only to both

For what one gets from a journey is growth.

(And amazing memories!)

This PhD has been closer to an Odyssey. And yet, at the end of it I have nothing but gratitude, for the only way to reach home, and myself, was to keep travelling through stormy seas. Weirdly fitting, for a PhD that addresses navigation.

Truth to be told, this work would not be here had it not been for some amazing people. Also, truth to be told, I don't believe I've met anything but amazing people during this big adventure. Those people have been the stars that made me find a way when I felt lost in a dark ocean. There is a Sigur Rós song the title of which translates to lost at sea: Sæglópur. Give it a listen if you can.

If I look at all these stars surrounding me, I have one certainty: the future is bright.

This PhD was sponsored by Thales Alenia Space. I remain immensely grateful for having had the privilege to receive this scholarship that has allowed me to find my calling in research: giving technological access to those who haven't it.

Yet, I believe that my first thanks must go to Lounis. It's a deep thanks, and word would hardly do any justice to it. He has been fighting with me, and for me when I could not. He has always believed in me, and from that trust, I've grown trust on my own. He's just amazing – but he'll probably never confess that.

Thanks to Leonard and Pau, for joining my supervision team when I didn't have much to offer. For their trust, their patience, and for all that they've taught me. Note to you, the reader: they're just amazing too.

Thanks also to Antonio, for welcoming me to the UK on a stormy night, and for being there for a "short" year and a half making life in Swindon exciting and funny. Guess what? Amazing, indeed…and he'll probably confess that. But he can afford to.

Thanks to Xavier, for his mentorship, his trust and his care.

Thanks to Roberto and Giancarmine, who were the first to teach me the art of research. Their incredible example has helped me in navigating the field of research on more than one occasion.

The biggest non-academic thanks go to Claire. She has shown me the meaning of many things and has brought happiness and excitement into my life during the last days of thesis writing, ensuring that this work could be finished.

Thanks to my family, for their unfaltering love and support. Thanks to my parents Enzo and Titti for their pure, immense love, and thanks to my sister Roberta, who is the best sibling one could wish for. Except for when one has to share snacks.

Also, thanks to all the Cuginetti for having always been there. I realised only later in life how special what we have is, and thanks to all my aunt and uncles for all they've given me. In particular, thanks to my uncle Lello, for showing me so many things, and for the beers before I left for the UK. Zio, you're definitely one of the reasons why I'm here.

A huge thank you to Eddie, for always pushing me to not give up, for always pushing me to be better, and for always pushing me to have fun. Duck feeders forever.

Thanks to Ray, for the deep conversations, the music, and the banter.

Thanks to Marleen, who is wonderful at making things special.

Thanks to the Unni, for having made 10 years of university bearable and fun. If you find me crying with laughter there's a 33% chance that it's because of them.

Thanks to Akhil, Amelie, Samuel, Alix, Brandon, Jacopo, Federico, Giovanna, Fabiola, Emily, Ruben, Monica, Mengjia, Leon, Encina, Federica, Özgün, Luc, Axel, Alejandro, Joe, William, James, Duarte, Shahmi, Shabarish, Tatjana, Hanna, Sarah, Ben, Safiah, Andrea, Nasyitah, Carole, Hugo, Damien, Mathieu, Nicholas, David, Alexandra, Michal and all those that I've forgotten (because I'm terrible with names) for never allowing loneliness to be a thing in a campus in the middle of nowhere. Also, thanks to the CFI for all the sweets (So long, and thanks for all the ~~fish~~ cake), except for when they try to poison me with strawberries. I guess that could still be a chance to be valuable to forensics people and pay them back…

Thanks to Bea, Mandy and Annie, for having been so consistently awesome.

Thanks to Salvatore. You have a bright mind and an even brighter soul. Gosecha is going to happen soon. Can't wait to go to Pupetta with you again…

Thanks to Niko, for being just who he is and not realising how much that means.

Thanks to my group of friends from home. I wish I could hug and see you so much more. Yet, thanks for always making me feel like I've never been away.

Thanks to Fabrizio, for being one the most incredible humans around and for *having been around* for the last 23 years. Oh, the patience of this guy…

Thanks to Simone, who is exceptional beyond words. I always wish I was better at keeping in touch with you.

Thanks to Aspasia, who I haven't hugged in too long. You're special☺

Thanks to Katrine, who has helped me in finding my voice with her blue leaves.

Thanks to Sabrina, Sara, Rosaria and Alessia, for always being there believing in me and encouraging me to write and scheme.

Thanks to Dario and Davide, for appearing exactly when I need them (but not when I want them, because I always want them around), and for always bringing amazing things into my life.

Thanks to Veselin, Luca and the whole Planet Patrol group for making the long wintery lockdown bearable, for being the high point of Fridays, and for a lot of other amazing things.

Thanks to Pablo and all the people in SGAC for keeping space a friendly, happy, cool place.

Thanks to Paolo Sorrentino, for (possibly) giving me the chance to be on Netflix. We'll see once the movie comes out!

Thanks to Mike and Shefali, for giving me the best opportunity I could wish for.

Lastly, thanks to Bob Balaram, for flying a drone on Mars, and for telling me in front of the Old Faithful, without any person in sight: "Next time we see this, we will be on Enceladus". Quite a few times I've held on to those words to keep going during the heaviest days of Covid.

To landing on Enceladus!

Thanks to all those people who in the last four years have gifted their presence to me. You all have a home in my heart.

And thanks to all my cats, dogs and my tiny cat-son Misu for being always there to bring smiles on my face. Ciao, Nero!

A haiku $\begin{cases} 6n+1 \\ 6m+5 \end{cases}$:

Primes the uppers

And so, the lowers too but

Their product is not

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

SYMBOLS, UNITS AND NAMES

Symbol	Unit	Name
$\boldsymbol{\varphi}$	Degrees/Radians	Sun phase angle
$\boldsymbol{\epsilon}$	Dimensionless	Coefficient of restitution
$\boldsymbol{f}$	$\boldsymbol{mm}$	Focal length
$\boldsymbol{I_k}$	-	k-th image
$\boldsymbol{s^L}$	$\boldsymbol{m}$	Landing position of the probe, 3x1 array
$\boldsymbol{T_S}$	-	Threshold for Pivot-Independent features, >>1
$\boldsymbol{T_H}$	-	Threshold for Pivot-Led features, $\ll \mathbf{1}$
$\boldsymbol{Q_{0k}}$	-	Metric score of the Pivot (OAs)
$\boldsymbol{V_e}$	$\boldsymbol{ms^{-1}}$	Escape velocity
$t_0 \;\; t_{end}$	$\boldsymbol{h}$	Initial and final time for acquisitions
$\Delta\boldsymbol{t}$	$\boldsymbol{h}$	Time interval between acquisitions
$\dot{\boldsymbol{x}}$	-	State vector
$\boldsymbol{A}$	-	Jacobian matrix of $\dot{\boldsymbol{x}}$
$\boldsymbol{p_{jk}}$	-	Quality of the j-th feature in the k-th image
R_a	$\boldsymbol{m}$	Average radius of Bennu. $\boldsymbol{R_a = 248\ m}$
R_{max}	$\boldsymbol{m}$	Maximum radius of Bennu. $\boldsymbol{R_{max} = 282.5\ m}$

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
ANN	Artificial Neural Network
ASP	Action, Scenario, Perception
AU	Astronomical Unit
BA	Bundle Adjustment
CCD	Charged Coupled Device
CDF	Cumulative Distribution Function
CLOAKs	Countermeasures Leveraging Optical Attractor Kits
CMOS	Complementary Metal-Oxide-Semiconductor
COTS	Commercial Off the Shelf
CRF	Camera Reference Frame
DART	Double Asteroid Redirection Test
DEEVE	Dynamically Equivalent Equal Volume Ellipsoid
DSN	Deep Space Network
DNN	Deep Neural Network
DoF	Degree of Freedom
DSI	Deep Space I
EDL	Entry, Descent and Landing
EM	Electromagnetic
EO	Electrooptical
ETC	Estimation Test Cases
ESA	European Space Agency
FAST	Features from Accelerated Segment Test
GN&C	Guidance, Navigation & Control
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GPU	Graphics Processing Unit
GSD	Ground Sample Distance
ISRU	In-Situ Resource Utilisation
JAXA	Japan Aerospace Exploration Agency
NASA	National Aeronautics and Space Administration

NAV-L	Noncooperative Artificial Visual Landmarks
NEA	Near Earth Asteroid
NEO	Near Earth Object
NFT	Natural Feature Tracking
NLS	Nonlinear Least Squares
OA	Optical Attractor
OAKs	Optical Attractor Kits
OBIRON	Onboard Image Reconstruction for Optical Navigation
OpNav	Optical Navigation
ORB	Oriented FAST and Rotated BRIEF
PI	Pivot Independent
PL	Pivot Led
RANSAC	Random Sample Consensus
RGB	Red, Green, Blue
SIFT	Scale Invariant Feature Transform
SURF	Speeded-Up Robust Features
SWaP	Size, Weight and Power
TC	Test Case
TCTF	Target-Centered, Target-Fixed frame
THALASSA	Technologies for Hazard Avoidance and Landing for Autonomous Spacecraft with Situational Awareness
TM	Target Marker
TRN	Terrain Relative Navigation
UAV	Unmanned Aircraft Vehicle
UE4	Unreal Engine 4
VO	Visual Odometry
VPU	Visual Processing Unit
VRF	Visual Reference Frame

1 INTRODUCTION

1.1 Scope of the Thesis

This thesis presents multiple approaches developed to enable appropriate selection and management of observables for visual autonomous navigation workflows in proximity to small celestial bodies.

In particular, the research question addressed here is whether it is possible to predictively identify regions within a celestial target where adequate computer vision feature detection techniques will perform well and regions where they will inherently fail to find observables. As the detection of features represents the starting point of almost every computer vision workflow, the capability to predict the features' behaviour would have significant implications in developing robust algorithms and procedures. For example, predictive identification of suitable and non-suitable observables could then be used to optimise the possible navigational trajectory for approaching a celestial target or the navigation software.

The vision of this approach is to ultimately use a single monocular camera for navigation.

The doctoral degree was kindly sponsored by Thales Alenia Space. Thales Alenia Space's interest was to investigate new navigation approaches for landing. These approaches were required to be:

- Extendable, meaning that they could be redeployed to different targets and/on different spacecraft without having to fundamentally redesign the mission and the payload.
- Robust, meaning that their behaviour would be only weakly susceptible to external factors and perturbations, and possess safeguards against failures or anomalous behaviours.

This context shaped the scope of the study and the methodology presented. Within this set of requirements, it was jointly agreed to focus on small celestial bodies as a study case, to develop foundation work for the core package.

This is because landings on small celestial bodies both present some unique challenges (e.g., weak gravity, rapid illumination change) providing interesting research and engineering problems and represent a target class destined to become of increasing commercial importance in the next decades. Particularly significant within small celestial objects are near Earth objects (NEOs), low gravity celestial objects devoid of atmosphere with orbits that bring them close to Earth.

This proximity makes them more accessible than equivalent bodies located further into space (e.g., Oort cloud objects): NEOs offer the possibility to achieve frequent deployments at a low Δv (propellant) expense [1] and can provide new potential opportunities for numerous technical and scientific developments.

The current trends within the space sector imply that most of these future missions will be operated by private entities, focused primarily on generating a profit. The business-oriented paradigm of the last decades of space is referred to by a multitude of different labels. Common labels are “Space 4.0”, “NewSpace”, or “Commercial space” [2,3]. Driving down the cost of exploration will induce higher returns on close-up operations. At one point, ideally, this process will be cost-effective and widely accessible to smaller entities and startups, similarly to what has been observed for LEO, thus driving an exponential increase in capability. The ability to investigate in situ resources and explore asteroids is a function of the cost of the space probe, its subsystems and the cost to operate them.

In this thesis, three characteristics are deemed necessary to address the need for cost-effective commercial asteroid exploring probes.

- The first characteristic is that navigation methods should be able to operate on platforms with limited onboard resources, such as little available processing power or constrained sensing capabilities. These platforms can be small probes, restricted by their limited form factor, or large probes which can operate in a resource-constrained mode.

- The second characteristic is that navigation methods should be self-contained and able to operate autonomously, with no need for operators.
- The third and last characteristic is that navigation methods should possibly be standardised.

These characteristics enable deployments in time-critical settings at distances with a large communication delay and the operation of large fleets at low development and operational costs.

Finally, two enabling technologies are seen as requirements to develop spacecraft with those three characteristics and address technological and budgetary constraints.

- The first requirement is represented by camera-only navigation. Camera-only navigation aims to enable navigation through a single sensing element, a monocular visible camera. Cameras are passive electro-optical sensors, that are low-cost and with a minimal impact on SWaP (Size, Weight and Power) budgets even in redundant configurations.
- The second requirement is represented by an approach to selecting optical features that can enable camera-only navigation. This approach is here labelled "invariant-oriented perception". This relies on the postulate that processes based on the identification of phenomena or criteria that have a high probability to be consistently present will favour the optimal prediction of related features within different use-cases scenarios. Invariant-oriented perception aims to generate robust, standardised navigation processes by carefully selecting what is observed and tracked in visual methods. Visual methods should focus on entities and processes that do not depend on the targeted NEO’s characteristics. Techniques targeting invariants would ultimately open the way to generalised navigation for approach and landing operations, using standardised and scalable methods. This paradigm aims to offer a response to the limits of hyper-specialisation. In this way, novel missions would require at most fine-tuning of well-known methods rather than the development of new navigation packages.

The latter is, in particular, a necessity around asteroids, which show a large variety of shapes, and surface and illumination conditions. While camera-only navigation by itself could exist without the second requirement, focusing its workflows on invariants would mitigate the risks concerning the absence of features or features with poor quality.

The overall scope of the thesis is, ultimately, to lay the groundwork for scalable camera-only navigation in proximity to asteroids enhanced by the perception of invariants. The following sub-sections will introduce the key elements on which this thesis is based and that are important to understand the context surrounding the methodology of this work. Namely, they will describe the reason for which this thesis focuses on NEOs, the navigation particularity that relates to this type of celestial target, the sensing requirement, and why we propose a camera-only navigation approach.

1.2 Near Earth Objects

There has been a growing international interest in space missions targeting small celestial bodies. Asteroids, comets, and meteoroids constitute the majority of the celestial objects in the Solar System. This large cardinality is also associated with a large variance in the physical characteristics within this category of objects, which greatly vary in sizes, shapes, spin states, etc [4]. Therefore, commonly these objects are split into subpopulations using their orbital characteristics.

For example, important orbital regions are represented by the Asteroid Belt (objects between Mars and Jupiter), the Kuiper Belt (objects from the orbit of Neptune at 30 AU to 50 AU from the Sun) or Near-Earth Objects (NEOs, objects orbiting in proximity to the Earth).

This thesis focuses on NEOs, which, more formally, are defined as objects having a perihelion $q < 1.3\ AU$. For reference, Mars' orbit is roughly $1.5\ AU$ from the Sun. There is a further condition on the orbital period T for Near-Earth Comets (NECs): $T < 200y$, to only include short-period comets. However, the majority of NEOs are asteroids. Near-Earth Asteroids can be further classified into *NEAs* into groups (Amor, Apollo, Aten, Atira) using their q, their semi-major axes a, and their aphelion distance Q [5]. There are more than 20000 NEOs asteroids, representing roughly 3.33% of all the asteroids in our Solar System [6].

NEOs have sizes ranging from a few meters to the size of 1036 Ganymed, considered the largest NEO with a diameter estimate of roughly 38 km [7]. Trilling et al. [8] report an estimate of $10^{6.6}$NEOs with a size larger than 10 m, and Lupishko et al. [7] report that the number of objects with a size $l > L$ is distributed according to a power law of the type

$$N(l > L) = aL^{-b} \tag{1-1}$$

With the exponent b in the range of 1.65-2.00.

The density values of NEOs are available only for a restricted number of objects, making it difficult to infer a distribution: for example, Dotson et al. [9] attempt to determine it by using meteorites.

The density highly depends on the internal structure, which ranges between the two extrema cases of rubble pile (clusters of multiple smaller rocks, low density) or monolithic (single block of material, high density).

Most NEOs originated in the Asteroid belt, and one out of 5-6 NEOs (15-20%) is believed to be part of a binary system [10].

NEOs are a particularly significant of small celestial objects due to their proximity to Earth, which implies that they can be accessed and studied more easily than other objects further away, and that, conversely, they can more easily represent a threat to Earth. Due to the former, NEOs are a prime candidate for science and commercial missions. In particular, for scientific scopes these celestial bodies contain numerous valuable chemical and mineral resources that have not been altered by any geological or Aeolian process and which could contain critical information about the history of the Solar System [11]. Moreover, asteroids can be extremely rich in commercially valuable resources, and these can be exploited both locally by opportunistic probes or to bring back rare materials to Earth. For example, there are estimates that asteroid 16 Psyche (from the Main Belt) could contain gold for a value of 700 quintillion dollars [12].

Yet, as the latter, NEOs need constant monitoring as they can pose an existential threat to humanity due to their proximity. Therefore, in-depth knowledge about them is fundamental to mounting a rapid planetary defence response. Indeed, there is a subset of NEOs labelled potentially hazardous objects (PHOs), identified by a condition on the minimum orbit intersection distance (< 0.05 AU) and by one on the absolute magnitude (≥ 22). The condition on the magnitude translates to a condition on size, with an absolute magnitude of 22 corresponding roughly to a body with a radius of 70m. According to Rumpf et al. [13] it is sufficient for an asteroid to have a size of 18 m to start causing fatalities.

In general, to classify the hazard posed by an asteroid there are two scales: the Torino scale, used for communications with the public, and the Palermo scale, used within the scientific community. For example, the Torino scale assigns to each object a value from 0 (not dangerous) to 10 (certain, destructive impact); this value describes the chance of impact and the energy associated to it [14].

1.3 Navigation and Sensing

Achieving the scientific or technical goals for some missions targeting NEOs, such as predicted space mining missions or some approaches to planetary defence, depends on physical interactions with the celestial body at a certain stage of the mission. In other words, eventually, the spacecraft has to touch the target's surfaces. Examples of these goals include sample acquisitions [15][16], impact missions for planetary defence [17], and resource mining for in situ resource utilisation (ISRU) [18][19] or subsequent utilisation of the materials on Earth [20].

To guarantee mission success in such a delicate mission phase, where the probe or its payload can be destroyed or lost, the development of Guidance, Navigation & Control (GN&C) frameworks for NEO-targeting missions must be robust to multiple complex external factors. Navigation is necessary to timely and precisely detect a difference between operational conditions and guidance-defined nominal conditions and trigger a control action to correct the trajectory. The output of Navigation, in particular, is an estimation of a set of parameters characterising the pose of the platform. For clarity, the pose is defined as the integration of attitude and position information. These characterising parameters can be linear or angular positions, velocities, accelerations, and more.

The generally weak and complex gravitational fields of asteroids lead to unstable orbits and fast divergence of errors. Being these fields hardly predictable from the ground, and being them generally associated to nontrivial distributions of masses, the probe can find itself operating in a largely unknown environment. As navigation ensures the prompt reaction of control mechanisms to conditions and perturbations bringing the spacecraft out of its nominal course, it plays a critical role in the proximity of NEOs, as it safeguards missions against the escape orbits and impact conditions.

A fundamental part of a navigation architecture is represented by the sensors, bridging the external world to the computational capabilities available onboard the probe. A core design choice in satisfying the three identified drivers of autonomy, scalability and limited usage of resources implies choosing an appropriate navigation sensor.

Navigation on Earth has a millennia-long history, encompassing methods and technologies as diverse as eyes looking for constellations, compasses, and futuristic quantum methods. In recent years, large-scale navigation capabilities have often been actively provided through signals actively generated by distributed systems. A typical example of this is represented by the Global Navigation Satellite System (GNSS). The widely used GPS (Global Positioning System) is an instance of GNSS. In GPS, carefully timed signals from a constellation of satellites are used to triangulate the receiver’s position. GNSS has become so pervasive that the lack of it defines a navigation class by itself, in GNSS-denied environments [21,22]. Deep space missions represent a typical example of operations in a GNSS-denied environment.

Duckworth and Baranoski [23] have presented a high-level classification of approaches to geolocation and navigation in GNSS-denied environments, based on the underlying sensing modality. These are: direct measurements (e.g., GPS), inertial or dead-reckoning, and environmental correlation.

Duckworth and Baranoski consider the third approach, environmental correlation, only as a form of map or template matching. By doing so, their list omits at least another instance of environmental correlation, where the correlation is operated incrementally, as is the case with a process that will be introduced later in the thesis, geometric visual odometry. In visual odometry, the matching is operated frame-by-frame between features detected within the scene.

Among the potential sensing approaches, monocular cameras appear to be the ideal payload to provide navigation capabilities under the three system constraints listed in section 1.4. This is because of the very nature and the characteristics of these sensors, as well as the nature of the output data they provide. Indeed, cameras are versatile and optimal for most budgets (as mass, energy, and cost); images provide a data structure that enables the extraction of a large variety of information, from simpler to higher levels of interpretation. Moreover, the same data used to obtain navigation information can also be used to obtain situational awareness [24] and/or perform hazard detection (such as the identification of dangerous boulders [25] or exceedingly steep slopes[26]), unlike inertial or dead-reckoning methods.

Typical operations on images comprise scene segmentation, object detection, identification or recognition, feature detection, and noise removal and can be based on intuitive heuristics to more advanced algorithms. Some of these concepts and techniques will be elaborated further in Chapters 2, 4 and 5.

A camera-based navigation solution is here labelled as optical navigation. Optical navigation used to be a human-driven process during its early development. As an example, the Apollo missions carried on board a telescope and a sextant to allow astronauts to manually perform optical navigation tasks [27].

Nowadays, this is no longer a completely human-centric process thanks to advances in computing and computer vision. Yet, there are still some very complex tasks that require the expertise of human operators.

However, these analyses are largely computer-assisted and leverage the advancements in computer vision algorithms.

Indeed, it is now possible to program software autonomously exploiting camera images to perform a multitude of different tasks. These range from reconstructing information about the motion of the probe to analysing the surface of the target, to obtaining a relative pose with respect to other elements involved in the process. Artificial intelligence (AI), especially in its machine and deep learning forms, has already proven in specific contexts and configurations to be more reliable and accurate than a human operator. An important choice when deploying the AI for a sensor is the engineering of the features to be processed and tracked.

AI identifies a technical field only loosely defined, and no commonly agreed-upon definition of AI exists. The forms of AI, exploited within the scope of this thesis relate to perception and computer vision, specialised in visible cameras. Currently, the state-of-the-art appears to be represented by learning-based methods. The methods investigated within this work, however, will not exploit learning-oriented methods such as Deep Neural Networks [28].

This is because of two reasons. The first is that these methods are still difficult to certify and deploy in space settings. The second is that these tools could greatly improve the results of the paradigm presented within this work, but only at a later stage, when processes, data and metrics have clearly been defined and outlined.

1.4 Camera-only Navigation and Invariants

It has been noted that in the proximity to an asteroid it is important to seek visual features which are reliable and consistent across a broad spectrum of different scenarios. Additionally, it was also noted that the performance and the efficacy of observables are a function of the available sensing capability.

Indeed, monocular cameras and their navigation models possess intrinsic limitations. These sensors provide information up to a scale and suffer from instabilities induced by illumination and the propagation of small inaccuracies within detections. These issues are generally addressed through methods having a heavy impact on SWaP (Size, Weight, and Power) budgets, such as additional sensors, or computational heavy methods. An example of this is the ALHAT landing package [22], developed by NASA’s Johnson Space Center (JSC), Jet Propulsion Laboratory and Langley, who envisioned three active laser sensors, hence requiring a lot more processing and energy resources to operate than a single camera. For reliability, complexity and this would be difficult to be affordable for small players.

The hypothesis addressed in the thesis is that adequately taking advantage of opportune visible features in the proximity of an asteroid could enable camera-only autonomous navigation workflows. These workflows would lay the groundwork for cost- and energy-effective proximity missions, eventually enabling widespread access to landing capabilities. This is because computationally light, camera-based, non-specific algorithms could then be redeployed as customisable plug-and-play building blocks of a modular toolset into low-cost platforms, which could then be mass-deployed around asteroids. This could be seen as an attempt to scale up HERA and DART missions, part of the Asteroid Impact & Deflection Assessment (AIDA) mission [29], which carry smaller CubeSats to assist them in their missions – either to perform post-impact flybys (for DART) or to carry similar activities and eventually land (HERA).

Ideally, the set of priors required for these operations would be constrained to a limited set of parameters, available without extensive scientific campaigns. This condition on priors ensures that the constraint on resources needed is maintained throughout the whole framework, that the resource usage is not simply moved upstream from the satellite to ground systems and that the costs are consistently low, due to the minimal hardware/sensing package proposed and the reduction in operational costs.

Various invariant features are explored in this work (see **Error! Reference source not found.** for an artistic representation of two of them) as follows:

a. The first are deployable physical structures, with a well-defined shape and known size. In this work these are represented as extendable cubes, but they could be envisioned as landers or rovers themselves. The standardised prior information about their shape can be exploited in multiple ways. At far range, these deployable infrastructures would provide strong features in environments with potentially little visible features. At closer range, this design standardisation would enable accurate pose estimation and scale reconstruction.

b. The second is the response of algorithmic processes to variations in illumination on the surface of the observed object, here an asteroid. In this work, this response is analysed by recording the quantity of available optical features as a function of the Sun phase angle for a circular Equatorial motion.

c. The third is the dynamically equivalent equal volume ellipsoid (DEEVE) of the celestial body. The DEEVE enables the reduction of the real mass distribution of the body to an ellipsoid, a uniform mathematical structure that can be easily treated if certain conditions are met.

d. The fourth is the heuristic intrinsic to feature detection models, which enables a consistently efficient selection of features to track. Detectors of visual features identify a pixel patch as a feature according to an internal algorithm. Knowledge about the shape that most strongly excites the detector's internal processes assists in determining whether such shapes would be largely available in the operational environment or not.

Lastly, during the development of this project, a fifth additional method was suggested. in line with the invariant-observation principle. However, this method is not camera-operated, and thus will not be addressed within the thesis as it relies on the observation of dust transportation cycles on a NEO. This is tracking the motion of dust over airless bodies to perform hazard detection. Dust winnowed by direct solar illumination settles in permanently shadowed zones or topographic depressions. Both surface features might be critical for landing. Since the behaviour of dust requires the body to be airless, this would represent an invariant physical process.

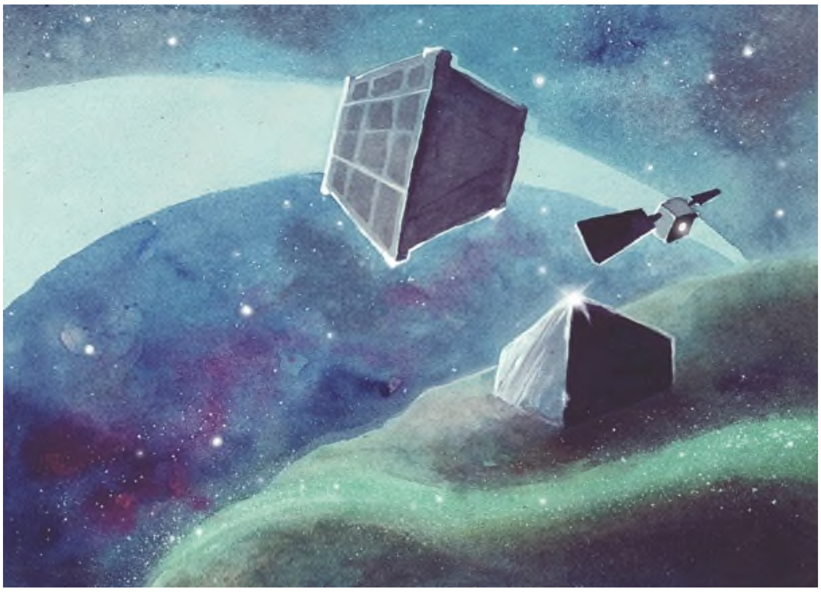


Figure 1-1 - An artistic representation of a mothership deploying a CubeSat and an artificial landmark, and dust moving over an asteroid (Illustration by Katrine Lyck)

From a purely technical perspective, the design of both the probes operating around NEOs and the probes' autonomous navigation system must address the constraint induced by the (complex) environmental factors.

Beyond purely technical requirements, however, there is currently a strong push to achieve commercial-oriented, cost-effective systems (e.g., a classic example being SpaceX[30]). In this paradigm, labelled "NewSpace", systems must be adequately de-risked; however, de-risking cannot be implemented through over-engineering, which would lead costs only to be transferred somewhere else in the design, development and deployment process.

Important factors enabling the design and possible deployment of commercial missions are: reductions in launching costs, the availability of Commercial Off The Shelf (COTS) components, the miniaturisation of hardware, and the development of complex software enabled by novel powerful small-scale processors.

The work carried out during the thesis led to individuating three key drivers for the development of a commercially oriented navigation system. This tries to take into account bottlenecks, availability of material, cost, and reliability, These system constraints are:

- Autonomy, which leads to reduced operation-related costs and enables the large-scale deployment of fleets in formation flying. It also allows probes to operate at large distances where communications times would have a much larger order of magnitude than operational times.
- Deployments on resource-constrained platforms, which allows setting an operational baseline which can be mass-deployed at a low cost while still being able to pursue its mission goal on a broad variety of platforms. These technologies may also act as a contingency or secondary navigation method for non-resource constrained probes.
- Standardisation of software and hardware (possibly with open standards), which would enable modularity, cross-platform reusing, ease of stocking and maintenance, and easy introduction of COTS.

1.5 Organisation of the Thesis and Contributions

The thesis is composed of six chapters which contextualise the scope of the research and the gaps aimed to be closed, present the tools developed to support the methodology of the proposed approaches, and ultimately introduce results and discussions about the obtained findings.

Chapter One: introduces the research questions addressed in the thesis work and makes a case for it. In particular, the focus of this chapter is on explaining the need for camera-only navigation, and invariant-oriented perception.

Chapter Two: provides further background information on deep space landings and visual-based navigation for small celestial objects. This helps in contextualising the scope of this research from a historical and engineering perspective.

Chapter Three: discusses the materials and methods developed and used throughout the thesis. It introduces, the ASP (Action, Scenario, Perception) triangle methodology, the formalisation of a design approach which underlies a large part of the results presented in the thesis and of the built software tools. ASP triangles are a design paradigm that emerged while trying to address the multifaceted navigation problem in a systemic way. To support and analyse in a data-driven manner the innovation developed by applying the ASP thinking process to relevant use-cases, the lack of accessible data and the need to control most parameters of the experiments induced the compelling necessity develop appropriate software simulation tools. These are integrated software tools for trajectory generation, synthetic image generation, estimation, and filtering, collected in a package labelled THALASSA.

Chapter Four: provides the first set of technical results, with a focus on managing the limb-fitting process in limb-based navigation. Within the scope of this work, this technique is employed in the spectrum of altitudes for which the target is clearly bigger than background stars but does not completely occupy the field of view of the camera. The limb method applied within this work approximates non-trivial spin-top NEO shapes to ellipsoids. Through these experiments, it was possible to see that errors in this estimation approach are clustered within well-defined regions around the target.

Chapter Five: is concerned with navigation based on the detection of computer vision features. These methods exploit opportunistic patterns detected within visual data and are techniques meant to support mid-range and close-range navigation. Feature-based navigation can operate at any altitude provided that some highly recognisable structures on the target are available to be observed. Specifically, this chapter will present results concerning three invariants that can affect or be propagated within feature-based processes. These invariants are concerned with increasing the robustness of the target process.

Chapter Six: provides ideas and directions to further expand on the ideas presented in the thesis, and the envisioned future work. This chapter has a twofold scope. The first part of the chapter relates to research, presenting open methodologic considerations and highlighting scientifically interesting points that could not be addressed due to time constraints. The second part relates to engineering and development, providing possible directions and requirements for implementation and deployment, and highlighting the to move the results to a higher TRL.

Scientific contributions achieved during the development of the thesis and presented in the different chapters are listed below:

- Optimization of feature-based navigation through non-computational optimizers acting at a feature’s level:
    - Environmental augmentations [31] (*“NAV-Landmarks: Deployable 3D Infrastructures to Enable CubeSats Navigation Near Asteroids”*, Conference Paper, Peer Reviewed) also connected to the principles of AI standardisation and warehousing.
    - Methods to understand the heuristic of optical feature detectors using genetic algorithms [32] (*“Countermeasure Leveraging Optical Attractor Kits (CLOAK): interpretational disruption of a visual-based workflow”*, Conference Paper, Peer Reviewed) enabling efficient selection and interpretation of the underlying perception model.

    - Heuristic filters and tools focusing on the response to illumination [33] (*“Perception fields: analysing distributions of optical features as a proximity navigation tool for autonomous probes around asteroids.”,* Conference Paper, Peer Reviewed*).*

- An iterative coevolutionary approach to design the macro-components of an AI [34] (*“ASP Triangles: Sketching the Artificial Intelligence of a Mobile Platform”*, Conference Poster).
- Development of a MATLAB/Blender hybrid plug-and-play prototyping and testing environment for developing a fully visual-based workflow (open source in the future by moving from MATLAB to Python). Modules for trajectory generation, estimation, and filtering with the goal of trajectory reconstruction (https://github.com/MarsZDF/THALASSA).
- Analyses of illumination-aware far-range optical navigation [35] (*“Autonomous Visual Trajectory Reconstruction in the Proximity of Small Celestial Objects”,* Conference Poster)*.*
- Unconventional computing: observing the dust around airless bodies, such as asteroids, to use these target objects as self-referential Hazard Detection and Avoidance computational units [36] *(“Monitoring Dust Motion Around Airless Celestial Bodies: Characterizing Suitable Landing Zones”*, Conference Paper).

2 **CONTEXT**

This chapter presents state-of-the-art and background concepts for navigation in proximity to asteroids, with a particular focus on close proximity for landing and optical navigation.

2.1 **Landing in Space**

Touching down with a spacecraft on the surface of extraterrestrial bodies is within the scope of various space mission types, ranging from local exploration to the collection and analysis of local material to impacts attempting to change the orbit of the target. Whereas most landings in space have happened on large gravity bodies, the majority of the more than one million bodies within the Solar System is represented by asteroids [37].

For mission requiring a touchdown, the landing mission stage is extremely critical, as the high speed involved can result in a complete loss of the spacecraft due to structural loads from impacts, and there is often very little margin to maneuver and react.

Moreover, in the case of low-gravity targets even non structurally destructive speeds might lead to dangerous circumstances. This is due to the low escape velocities of these celestial bodies. If the speed of the probe immediately after the impact is comparable to escape conditions, and if the craft is not anchored properly to the ground, the rebound can lead the probe to either bounce towards an unfavorable spot or into space.

To land in a safe way, means to ensure that the accelerations involved in the speed reduction process and in the touch down do not damage structures and payloads, and to ensure that the landing area is free from dangers (such as steep slopes and large boulders).

Navigation provides knowledge of the spacecraft’s speed and position with respect to the ground, which assists in maintaining the speed profile always, while scene/situation awareness provides information about what is around the probe.

When the operational distances are large (the local dynamic is much faster than the time taken by a signal to reach the probe from the ground) or the operational environment is extremely complex it is necessary to introduce an element of autonomy In order to provide the probe the capability to act or react within the short timeframe and safeguard the goal of the mission and which otherwise can hardly be accomplished as fast through remote operation.

2.1.1 Landing Approaches

In general, it is possible to distinguish between two landing approaches, depending on the speed with which the landing craft impacts the ground: soft and hard landings. High speed, often destructive, landings are labeled hard landings. When the chosen landing approach is a hard landing, the landing probe is defined as an impactor. Hard landings are rarely intentional on Earth but take place more frequently in space settings. An intentional impact landing might be selected for multiple reasons. Impactors might be chosen for their simple design, as opportunistic mission extensions, to estimate some local property of the target body, or to transfer an appropriate amount of momentum. The latter has been the case for Deep Impact mission, where an impact was used to excavate and study the nucleus of the comet Tempel 1 [38].

Performing a soft landing implies gradually reducing the speed of the craft and making it as close as possible to zero only when the craft is close to the touchdown point. A prematurely induced null velocity leads to the ballistic fall of the craft; and again, non-null velocities at touchdown generate loads proportional to the residual speed. Within the scope of this thesis, the focus will be on probes operating a soft landing, due to the stronger requirements for robust navigation.

To be successfully and reliably achieved, soft landings require highly complex architectures. The crafts landed in this way require sensing to understand when to begin the deceleration and how to perform it, and control mechanisms and intelligence to act on the sensors’ outputs. For example, typical triggers for the payload operating the slowdown are range or velocity [39]. Both triggers have different implications on the final landing condition. It is here worth mentioning that an important visual descriptor of the landing conditions is represented by the so-called "*landing ellipse*" [40]. The landing ellipse represent the bidimensional region projected on the ground where the simulations predict with a certain degree of accuracy that the lander will touch down. Generally, the chosen level of confidence is 3σ, or 99.7%. Multiple technical parameters and atmospheric (when an atmosphere is present) and gravitational perturbations influence the shape of this ellipse, centered on the expected landing spot. In addition to technologies enabling speed reduction while in the air, at times surface mechanisms damping and absorb the strength of the impact (e.g., airbags), or providing anchoring and stabilization might be necessary. These systems operating on the ground or in its proximity weakly affect the landing ellipse.

The first probe landing on a non-terrestrial surface performed a hard landing. This was the Soviet *"Luna 2"* which impacted the Moon on the 13th of September 1959. The first lunar soft landing was once again performed by a Soviet craft, the "*Luna 9*" [41]. To do so, the probe used thrusters and inflatable airbags. The trigger of the airbag inflation; a second trigger was provided at 5 meters above the ground level from a sensor registering contact with

Over the years soft landings have been enabled by multiple technologies related to perception, navigation, and speed reduction. The characteristics of these technologies are extremely different depending on whether the landing target possesses a sensible atmosphere - the envelope of gaseous layers surrounding a celestial body – or not. The presence of an atmosphere is associated to high-gravity bodies, but not every high-gravity celestial object can develop or maintain an atmosphere. A soft-landing approach on a celestial body surrounded by an atmosphere is necessarily different from one on an airless celestial object. This is because atmospheres induce a force opposing motion known as "*drag*" over the falling body, which aids in reducing speed. However, the velocities at play from an orbital entry, might lead to speeds much larger than the local speed of sound, which is well-defined in an atmosphere. These conditions are labeled "hypersonic condition". A craft moving in hypersonic conditions is subject to multiple complex thermo-fluid dynamics phenomena - in particular, extreme thermal flows. In a hypersonic regime thermal control is operated through aerodynamic design, heat shields and appropriate descent profiles. However, navigation capabilities are severely impaired during this period.

Moreover, a gravity well capable of holding an atmosphere, is necessarily strong, inducing larger forces and shorter operational timescales. The latter point is critical when designing the on-board processing unit and the number of operations that must be handled during entry, descent, and landing. However, this strong gravity removes the concern about the probe bouncing away from the first landing spot.

The presence of an atmosphere also has important implications on the operational environment close to the surface and during the descent towards the ground and on the observable elements for computer vision pipelines.

As examples, atmospheres influence conditions the type of existing geological features [42], allow the existence of vertical thermal profiles, filter the size of impactors reaching the surface and slow them down [43], determine the surface transport and deposition phenomena [44], constrain the portion of the electromagnetic spectrum capable of reaching the surface, enable a pressure build-up and the permanence of chemical species in liquid and gaseous forms, and many other things.

2.1.2 Landing on Airless Bodies

Whereas landings on bodies with atmospheres can employ solutions highly specialised to the presence of a fluid, the vast majority of celestial bodies viable for landing do not possess an atmosphere. When this type of celestial objects is chosen to be visited within a mission, there are major implications on the design of the landing strategy. Moreover, it is also possible to expect important engineering difference once the probe has landed on the surface. However, addressing surface operations is out of the scope of the thesis. Compared to landings on bodies with atmospheres, characteristics of landings on airless bodies are:

- The absence of an atmosphere implies a lack of friction to aid in the slowdown. This implies that it is not possible to slow down the probe through parachutes, and no drag forces are present
- No thermal shielding is required to survive the descent
- Whereas no thermal shielding is required, the temperature on the surface of airless bodies might be too elevated for surface operations when these celestial objects are too close to the Sun [45]
- Aeolian perturbations do not affect the trajectory of the probe
- Features observable and available on the surface
- Differences in available lighting

Again, the lack of an atmosphere does not provide any insight on the gravity of the body. However, no NEO possesses an atmosphere, and therefore all the landing conditions considered in this thesis will address airless bodies. The low gravity of NEOs, moreover, makes the pressure generated by solar radiation (SRP) a potentially significant perturbation when compared to local gravity, whereas SRP is the force generated by the Sunlight photons exchanging momentum with a surface of the spacecraft.

2.1.3 Past Landings on Small Bodies

The first decades of landing operations in astronautics have been completely dominated by the selection of large gravity landing targets. Then after the lunar Apollo missions, landing itself, for a while, appeared to become a mission type of minor relevance. It was only around the beginning of this millennium that this mission type started to become extremely significant again. The first time a synthetic device touched down on the surface of a body significant within the scope of this work, concerned with low gravity objects, was in 2001, when the probe NEAR Shoemaker [46] landed on 433 Eros, a Near Earth Asteroid (NEA). This was rapidly followed by several others missions: the Hayabusa mission to the asteroid Itokawa in 2005 [47], and the Deep Impact impactor to the comet 9P/Tempel 1 in the same year [48]; the Philae lander of the Rosetta mission to the comet 67P/Churyumov–Gerasimenko in 2014 [49], the Hayabusa2 mission to the asteroid Ryugu in 2018 [50], and more recently the OSIRIS-REx probe to the asteroid Bennu in 2020 [51]. Moreover, many more missions to similar small targets are expected to take place in the near future [52], [53]. The targets of these missions alone highlight a broad variability in the geological and morphological properties of these targets, and they represent only a handful of the roughly 25000 Near Earth Asteroids [54] that are currently known. However, not only NEAs are becoming accessible: the whole access and interaction paradigm is completely changing, largely due to the commercialization of space.

2.1.3.1 Excessive Speed at Landing on Small Bodies: The Issue of Rebounding

To further explain why landing on small bodies might be critical beyond the destruction of the payload, it is possible to use the example of the lander deployed during the Rosetta mission. Rosetta was a European Space Agency (ESA) targeting mission the comet 67P/Churyumov–Gerasimenko (Figure 2-1).

The spacecraft of the mission includes an orbiter, Rosetta, and a lander, named Philae (Figure 2-2). Rosetta and Philae were launched on the 2nd of March 2004 and reached the comet on the 6th of August 2014 - roughly 10 years later. Philae was deployed on the 12th of November, at a distance from 22.5 km from the barycentre of the comet [55] – a distance of more or less five times the extension of the largest lobe of 67P/Churyumov-Gerasimenko.

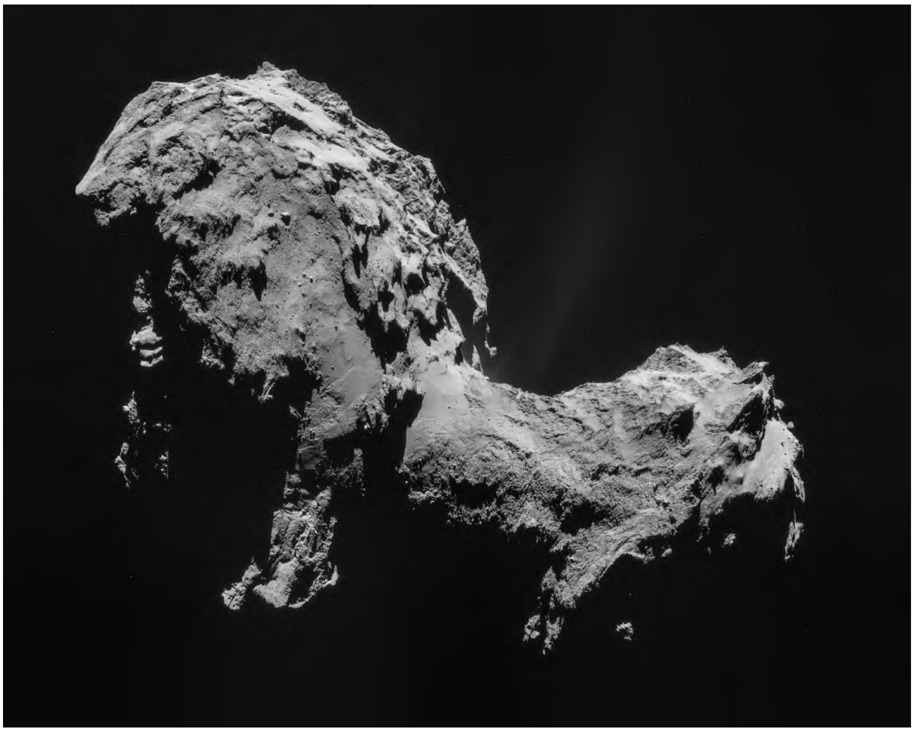


Figure 2-1 - The comet 67P/Churyumov-Gerasimenko. Image by ESA/Rosetta/NAVCAM.

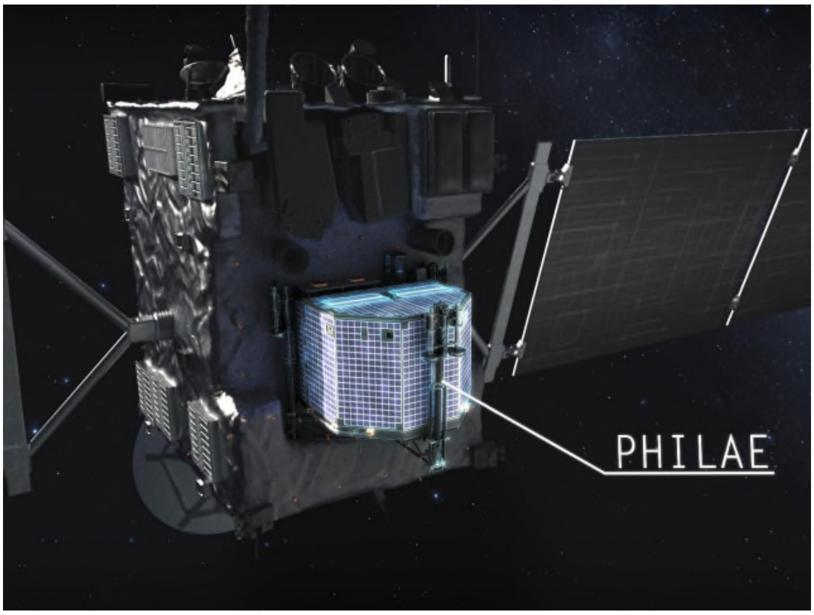


Figure 2-2 - A still from the movie "Chasing A Comet – The Rosetta Mission", showing Philae within Rosetta before deployment. Image by DLR.

Philae then successfully reached the surface of the comet and impacted it after a seven-hours long ballistic descent. The design of the lander included two mechanisms providing anchoring capabilities to Philae, ensuring that it would securely be held in place upon touch down. However, none of the devices in this anchoring package - two anchoring harpoons, ice screws, and a cold-gas thruster – worked as planned [56]. Because of these failures the probe started bouncing away from the designated landing spot and touched down on the ground three times. The speed of the probe after the first rebound was 0.38 cm/s [57]. This speed after the impact is critically comparable to the escape speed of 67P/Churyumov-Gerasimenko, where the escape speed is defined as the minimal speed that a body must possess to leave the gravity well generated by an object. For the target comet the escape speed has an order of magnitude of 0.5 m/s [58], comparable to the rebound speed of 0.38 cm/s and leading to a large displacement after the first leap. These undesired hops led the lander to settle in adverse surface conditions, with little solar illumination [59], insufficient to power the probe or recharge its batteries. These zones would be exceptionally critical for camera-only operations, as cameras could extract little to no information about them. Moreover, areas with weak or absent Sun illumination would prevent any solar powered probe from being able to operate, as has been the case for Philae. Without atmospheric scattering or reflection, airless body are more prone to presenting surface zones of permanent shadow [60–62]. More in general, airless body present a different surface illumination type than bodies with atmosphere, with stronger environmental contrasts, extremely hard shadows and, potentially, highly reflective dusty material – all critical elements for cameras.

Ultimately, it took almost two years to locate Philae using Rosetta’s camera [63]. The necessity to maintain robust navigation capabilities is again and further highlighted through the circumstances of Philae landing. Monitoring the speed of the probe enables predicting if a lander will bounce; monitoring the pose allows reconstructing, potentially in close-to-real time, the location of the final touchdown.

2.1.4 Future Landings on Small Bodies

At the time of writing, more missions to small celestial bodies have either been proposed, planned or launched without yet reaching their targets, capitalising on the successes of previous missions and the heritage technologies developed through them.

Beyond scientific purposes, these missions are also driven by the urgent need to develop readily available planetary defence systems and by the search for more sustainable ways to extract resources for Earth application and deep space ISRU.

Missions which have not yet reached their target objects and with a landing component, thus relevant within the scope of the thesis, are:

- DART: DART [29], which stands for Double Asteroid Redirection Test is a mission testing planetary defence capabilities against NEOs. DART scope will be pursued through an impact on its target, Dimorphos, the satellite of the small binary asteroid Didymos. The expected outcome of the impact is for Dimorphos to reduce its orbital distance from Didymos.

  The DART mission has two spacecraft components: the DART impactor, and LICIACube, a 6U CubeSat operating post-impact flybys at a minimum distance of 55.4 km [64]. DART was launched on the 24th of November 2021 and will reach its target in late September/early October 2022.

- Hera: DART is part of a broader space collaboration, labelled AIDA (Asteroid Impact and Deflection Assessment). Within the scope of AIDA there is a second mission named Hera, developed by the European Space Agency (ESA) [65] which is expected to be launched in October 2024. The goal of Hera is to monitor the efficacy of the DART impactor by analysing the changes on Didymos. Hera, like DART, has multiple spacecraft components: the main spacecraft and the cubesats Juventas [66] and Milani [67]. Juventas and Milani could attempt a landing on the surface of the asteroid at the end of their life.

- ZhengHe (planned to be launched in 2024): ZhengHe is an asteroid sample return mission that is being developed by CNSA (China National Space Administration) [68]. ZhengHe is set to target a NEA (469219 Kamo‘oalewa) [69] and a comet (311P/PANSTARRS). The spacecraft will collect regolith from Kamo’oalewa using a process labelled ‘anchor-and-attach’, differing from the ‘touch-and-go’ used on board OSIRIS-Rex and Hayabusa2.

In addition to these planned missions, there also are multiple planned flybys to small celestial objects; such as Lucy[70], Psyche[71], DESTINY+ [72], NEA Scout [73], Camilla [74] – also potentially including an impactor.

2.2 **Navigation for Landing**

This section provides insights about navigation solutions enabling landings on low-gravity airless bodies. The section is broken down into four subsections. The first subsection provides an overview of the six missions that in the previous years have touched down, both with soft and hard landings, on the surface of small celestial bodies. This information is presented through a table containing general information about the missions. The second subsection contains details about future missions to small celestial bodies. The third subsection presents information concerning payloads and methods for proximity optical navigation. Lastly, the fourth subsection builds upon the first three and supplies a second table exploring the landing-oriented navigation methods of the various missions discussed of interest.

2.2.1 Past Landing Missions

This subsection reports and discusses the approaches that aided in achieving proximity navigation for the six missions listed below in chronological order:

- NEAR Shoemaker (launched in 1996, landed in 2001)
- Hayabusa (launched in 2003, landed in 2005)
- Deep Impact (launched in 2005, impacted in 2005)
- Rosetta (launched in 2004, landing in 2004)
- Hayabusa2 (launched in 2014, two touchdowns in 2019)
- OSIRIS-REx (launched in 2016, touched down in 2020)

Three of these missions were deployed by NASA (NEAR Shoemaker, Deep Impact and OSIRIS-REx), two by JAXA (Hayabusa and Hayabusa2) and one by ESA (Rosetta). For some of these missions the lander is represented by the main craft itself, either performing a touch-and-go manoeuvre on the ground or effectively landing. Some other missions carry one, or multiple, deployable landers. The information concerning these missions is organised in Table 2-1 which consists of four columns presenting the main characteristics of these missions. These four columns are: *Mission Name*, *Mission Architecture*, *Landing Target,* and *Launch Date*. Mission Name reports the name of the mission, uniquely identifying it. Mission Architecture reports information about the spacecraft involved in the mission, its role, and its operational approach. Landing Target specifies the target upon which the landing took place. The name of the celestial body is followed by an (A) if the target is an asteroid, or by a (C) if it is a comet. It is worth noting that some missions might perform a tour-like mission, visiting multiple celestial bodies before reaching the one where surface operations are performed. Lastly, Launch Date reports the date in a DD/MM/YYYY format. It was preferred to report the launch date over the landing date to identify the technological landscape in which the development of the probe took place.

Table 2-1 - An overview of mission performing a touchdown on small celestial bodies

Mission Name	Mission Architecture	Landing Target	Launch Date
NEAR Shoemaker	• Orbiter, repurposed into a lander	433 Eros (A)	17/02/1996
Hayabusa	• Touch-and-go sample acquisition probe • Lander (MINERVA)	25143 Itokawa (A)	09/05/2003
Rosetta	• Orbiter • Lander (Philae)	67P/Churyumov–Gerasimenko (C)	02/03/2004
Deep Impact	• Flyby spacecraft (Orbiter) • Impactor	Tempel 1 (C)	12/01/2005
Hayabusa2	• Touch-and-go sample acquisition probe; two touch-downs • Landers/Rovers (MINERVA-II-1: Rover-1A (HIBOU), Rover-1B (OWL); MINERVA-II-2: Rover-2; MASCOT)	162173 Ryugu (A)	03/12/2014

OSIRIS-REx	• Touch-and-go sample acquisition probe	101955 Bennu (A)	08/09/2016

2.2.2 Future Landing Missions

Future landing missions appear to be moving more and more towards distributed architectures where motherships collaborate with smaller CubeSats providing ancillary capabilities.

Table 2-2 - An overview of future mission performing a touchdown on small celestial bodies

Mission Name	Mission Architecture	Landing Target	Launch Date
DART	• Impactor • Post-impact flyby cubesat	Dimorphos (A)	24/11/2021
Hera	• Flyby spacecraft • Two flyby cubesats with the potential to land at their EOL	Dimorphos (A)	10/2024 (planned)
ZhengHe	• Anchor-and-attach sample acquisition probe • Nanolander • Nanoorbiter	469219 Kamo'oalewa (A) 311P/PANSTARRS (C)	2024 (planned)

2.2.3 Optical Navigation Methods

This subsection provides technical insights into optical navigation (OpNav).

The broad range of altitudes swept while descending towards the target makes it necessary for the on-board intelligence to be able to operate in different navigation modes, dependently on what optical observable is instantaneously available. Therefore, employing optical navigation to perform a complete landing requires the integration of multiple technologies - making an even stronger case for modular developments.

Over the years many algorithmic models have been developed to exploit for navigation opportunistic (not known a priori) visual information available in space and in the proximity to the ground. This section presents the two most common architectures for the enabling camera hardware, and some OpNav methods considered important within the scope of this thesis.

2.2.3.1 Camera types: CCD and CMOS

In the modern world, digital cameras represent a ubiquitous piece of technology, integrated in a multiplicity of portable electronic devices. Nevertheless, digital cameras represent a fairly recent invention, with the first commercial releases being less than half a century old. The sensor enabling the first digital optical acquisitions is called "CCD" (Charged Coupled Device). CCD was invented in 1969, and its development has proven so important that it has granted a Nobel prize to its inventors, Boyle and Smith, 40 years later, in 2009 [75].

During the 1990s another image sensing technology emerged as an alternative to CCD: Complementary Metal-Oxide-semiconductor, or CMOS. CMOS were developed by Eric Fossum [76] at NASA's Jet Propulsion Lab, and therefore represent a technology created with space applications in mind.

CCD and CMOS appear to currently be the most widespread image sensing technologies. Both sensors create a representation of the external world by capturing and storing electromagnetic (EM) radiation. Within the scope of this thesis is of interest radiation within the visible portion of the EM spectrum, which is carried by photons with wavelengths roughly between 400 and 700 nm [77]. Cameras perform these acquisitions passively – i.e., they do not introduce any EM radiation within the scene. This is one of the limitation of cameras: they require a light source to operate - or external devices such as flashlights. However, being passive sensors makes them excellent from a power-expenditure point of view, maintaining their energetic consumption low.

The environmental radiation is acquired and stored using discrete grids of light-responsive elements labelled pixels (standing for Picture Elements). Specifically, when photons impinge on these elements, they are activated and generate an electric signal proportional to the intensity of the incoming light. This intensity is then translated to an intensity within the digital representation. The difference between CMOS and CCD sensors lies in the site where the electric signal is read – the pixel itself for CMOS, and the edge of the detector for CCD [78].

The behaviour of the two sensors is slightly different. CCD sensors perform better in conditions of scarce light; however, CMOS sensors behave better in the presence of very bright objects. CCD sensors are generally more expensive and consume more power than comparable CMOS sensors. However, CCD sensors are generally more reliable because they are better understood, having a higher degree of technological maturity, and often produce higher resolution outputs. Ultimately, the decision on depends on the application and on the constraints and requirements of the mission.

2.2.3.2 Computer Vision Features

Most of the developments discussed within the thesis target the perception of computational constructs known as features. Many computer vision processes rely on features or use them as starting point of workflows. Examples are image alignment and stitching, structure from motion, object recognition [79], and of course motion estimation. Indeed, a multitude of optical navigation designs relying on prominent external structures and patterns have been proposed for proximity operations [80–83]. The digital representation of these observables defines a feature, and the algorithms searching and selecting features in the images are deemed feature detectors. Features, then, generally require to be identified and matched across subsequent frames to return localisation information. An example of this process specialised to visual odometry method is shown in Figure 2-4. The matching stage operates through unique descriptors associated to features by algorithms labeled feature descriptors. Analysis of description and matching is out of the scope of this work.

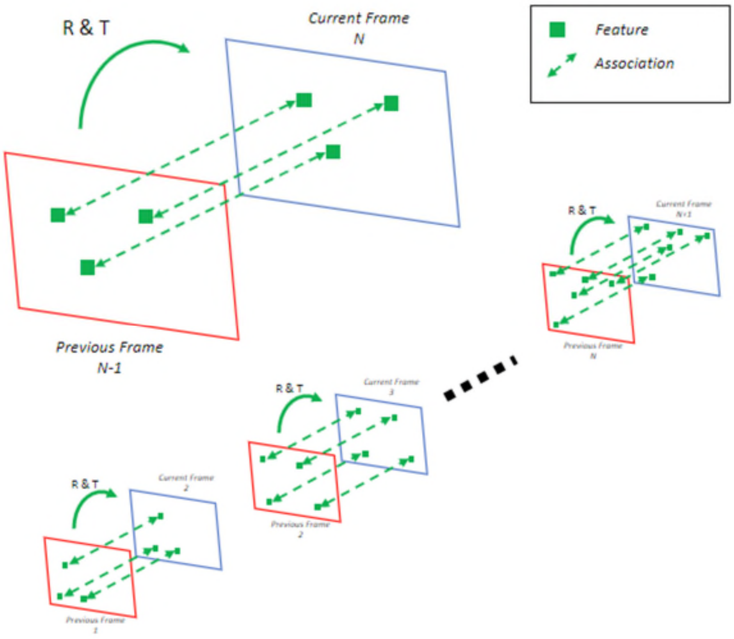


Figure 2-3 - The detection/matching process across different views. Image by Lounis Chermak.

As noted beforehand, the complex environment around asteroids can be rather disruptive for a feature detection and matching process. Features might not be stable enough under the complex illumination conditions, and their appearance could rapidly change because of the possibly fast, complex rotational dynamics. Moreover, stable large-scale features like craters might be completely absent from asteroids [80]. Generally, it is impossible to retrieve from Earth accurate knowledge about the identity and distribution of features within operational scenarios involving NEOs. Hence information about locally available features becomes accessible only when the terrain becomes resolvable to the sensors of the mission: in other words, generally, when proximity operations are already taking place. This means that it is difficult to calibrate the navigation algorithms and payloads before launch to specific local conditions or structures, as for example large boulders.

However, the presence of unique surface elements and patterns can generally be expected. The scope of this chapter is to identify commonalities between the detections of local opportunistic patterns and exploit them as invariants to enable robust camera-only operations. This is achieved both through adjustments on the internal perception model and by acting on the operational environment.

Concerning the feature perception model, two standard approaches to feature detection are significantly relevant throughout the thesis. These are Harris-Stephens corners and SURF (Speeded-Up Robust Features). SURF is a proprietary algorithm, therefore in line with an open science philosophy also an open source alternative, ORB (Oriented FAST and Rotated BRIEF), will be discussed.

The Harris-Stephens detector is the oldest of the three feature detectors largely used within the thesis. This feature detector is specialised to corners as seen in the image gradient. The algorithm was introduced to the scientific community in 1988 by Chris Harris and Mike Stephens at the Alvey Vision Conference [84] as an improvement of edge tracking methods. Figure 2-4 shows Harris-Stephens corners detected over a Bennu's picture. Heuristically, Harris-Stephen's detector classifies interest points within the image using a sliding patch. This classification is operated by analysing the average intensity variations returned when the patch is slightly shifted around in various pre-defined direction. This operation assigns the pixel patch at the centre of the sliding process to one of three possible classes. If the change is negligible regardless of the shift, the region is "flat", i.e., it presents weak changes in intensity. If the change is highly directional, the patch is centered on an edge, with translations across the edge causing large changes. Lastly, if the change is large in any direction, the region is a corner, i.e., the connection of edges.

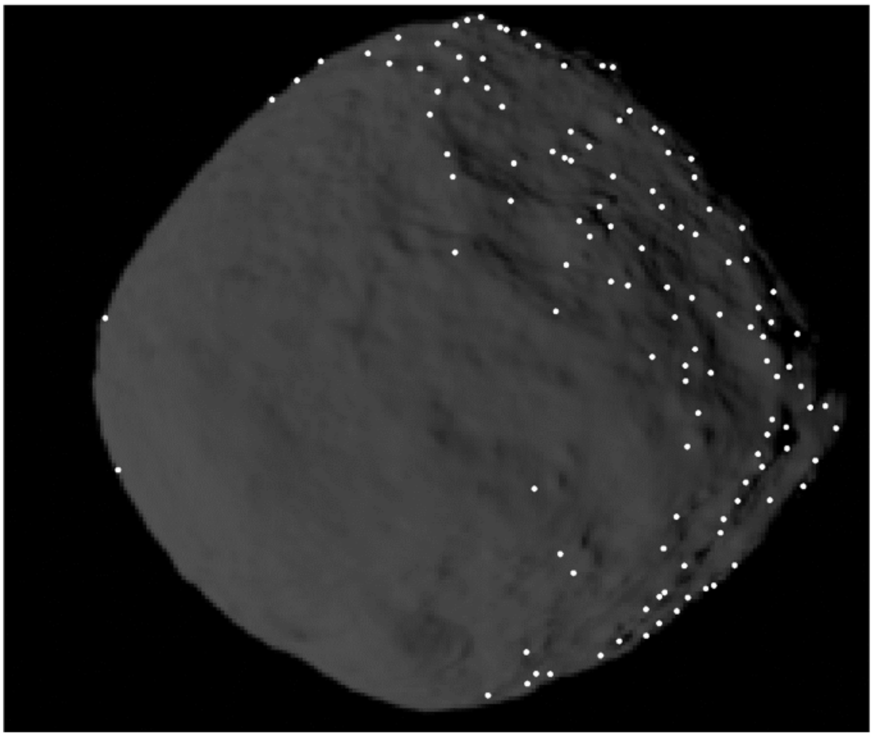


Figure 2-4 - Harris-Stephens corners detected over a Bennu simulation.

SURF was presented in 2006 - eighteen years after the Harris-Stephens detector - by Herbert Bay, Tinne Tuytelaars, and Luc Van Gool at the ECCV (European Conference on Computer Vision) [85]. As noted before, the name SURF represents an acronym, standing for “Speeded-Up Robust Features”. The *“Speeded-Up”* part within the name of this feature detector name refers to the optimality of its computational time when compared to other detectors that had been developed before SURF. SURF is often mentioned paired with SIFT of which it often seen as a faster version. Both SIFT and SURF are defined as ‘blob’ detectors. SIFT (Scale Invariant Feature Transform) was presented by David G. Lowe at the ICCV International Conference on Computer Vision in 1999 [86]. The main advantage of SIFT over corners features is that SIFT can remain stable and detectable also under scaling transformations. This property is inherited by SURF, with both algorithms operating a search on multiple scales. While SIFT and SURF have not been the first scale-invariant models (the concept itself was introduced by Lindeberg in 1994 [87]) they can certainly be considered among the most successful contributions in that respect.

Corners, indeed, are weakly suited to provide robust information in scale-changing contexts. Due to a constant patch size at different distances from the target, a corner might get ‘merged’ within an edge, resulting in the loss of finer scale features, or expanded and broken down into edge components. It is important to take this into account since the continuous altitude change in proximity operations for planets and primitive bodies often leads to hardly negligible scale changes in the observables. Thus, when possible, this limitation should be taken into consideration.

Lastly, ORB is a feature detector and descriptor, constructed from the integration of multiple prior technologies, collated and augmented. It was presented at the 2011 International Conference on Computer Vision (ICCV) by Ethan Rublee, Vincent Rabaud, Kurt Konolige and Gary Bradski [88]. The authors of ORB have presented it as an alternative to SIFT and SURF, achieving equal performance with respect to the former and improved performance at a lower computational cost with respect to the latter. Moreover, another advantage of ORB is that it can be freely used, unlike both SIFT and SURF which are patented algorithms in some countries.

The detection stage of ORB is based on a FAST (Features from Accelerated Segment Test) detector [89], which employs a rasterized circle of points with a radius of three pixels, centered on the one of interest [90]. The metric employed by the ORB process is the one defined for the Harris-Stephens detection process. ORB natively employs a BRIEF descriptor; however, addressing descriptors is out of the scope of this work. FAST, like Harris-Stephens, represents a corner detector. As the backronym in the name suggest, one of its strong points relies in its speed, a property which is preserved in ORB. This computational efficiency makes it optimal for real-time operations.

2.2.3.3 Optical Navigation

In line with the camera-only requirement, only three elements are assumed exploitable for navigation within this work:

- The monocular camera itself
- Prior information
- Geometric and textural features observed within the camera.

Therefore, navigation technologies can be classified according to either what is observed, or what knowledge is required to return estimations from the observables. Focusing on the features, within the scope of the thesis is of particular interest the set of technologies returning a navigation estimate by observing elements on the ground, commonly labelled Terrain Relative Navigation (TRN). This is because during landing the feature exploited in TRN are well resolvable within the camera. Johnson and Ivanov [91] present TRN methods as: *“These approaches sense the terrain during descent and augment the inertial navigation by providing, in real-time, position or bearing estimates relative to known surface landmarks”.* The sensors enabling TRN need to be able to create a representation of the external world within the on-board intelligence. This capability is commonly provided by electrooptical (EO) sensors such as lidars or cameras [92].

Albeit focusing on Lunar pinpoint landing, Johnson and Montgomery [92] identify three approaches to TRN:

- Global position estimation
- Local position estimation
- Velocity estimation

Most of the techniques discussed in the thesis are tested against reference values for position: therefore, a discussion on the techniques oriented towards velocity will not be provided. The difference between global and local position estimation lies in the reference frame that the position is described in. In global positioning, the position is retrieved in a global coordinate system; in local positioning the coordinate system is specific to the landing site. Regardless of this distinction, in all the four camera-based approaches for positioning described by Johnson and Montgomery some prior data about the landing site are required. These four techniques are:

1. Matching the pattern of craters to an off-line database. Generally, authors try to identify craters rather than boulders or similar element due to the larger size of craters (availability over a longer range) and their stability to perturbations
2. Matching SIFT (Scale Invariant Feature Transform[86]) features to an off-line database
3. Onboard Image Reconstruction for Optical Navigation (OBIRON)
4. Image to map correlation

Tian and Yu [93] suggest using craters for TRN on asteroids; however, again, as noted by Cocaud and Kubota generalising what had been observed on Itokawa, larger scale features such as craters could either be completely absent from small NEOs or too scarcely available to be reliably used as the starting point of visual workflows. Moreover, possessing a database of surface information cannot be taken for granted. This is because a phase constructing data assets concerning features on the ground may not be feasible for some mission architectures. This is also what makes critical the second approach as described by Johnson and Montgomery: the need for a database. SIFT represents a classical feature detector, and the approach described within the reference has two stages. Within the first stage features are detected, described, and their descriptors are stored in a database. Within the second stage the landing probe acquires a picture, generate a set of descriptors for the patterns detected as features within the image and performs localisation by comparing the features in the landing image to the one in the offline map.

Examples of solutions to the limitation induced by the need for offline data might be found in in visual odometry [94], or in an approach to TRN discussed by Christian et al. [95]. In these methods features are matched online between the outputs of sequential acquisitions rather than to a database.

Visual odometry (VO) represents a very successful approach to exploit features to reconstruct a platform’s egomotion through subsequent visual acquisitions. At the core of a VO process, optical features are identified and matched across consecutive images to return localisation information. Its functioning was envisioned by Hans Moravec in his PhD work [96], but it received its name in a paper by Nistér, Naroditsky and Bergen [97], and has found applications in many fields, albeit rarely in a monocular camera configuration. These uses range from navigation of the Mars Exploration Rovers [98] to navigation of aerial vehicles [99] to self-driving cars [100]. Main requirements for the applicability of VO approaches are that features are available, stable (i.e., don't change their appearance over time) and that the motion and acquisition are such that a population of features can be observed across subsequent frames. A plausible implementation of the VO process could be the one shown in Figure 2-5.

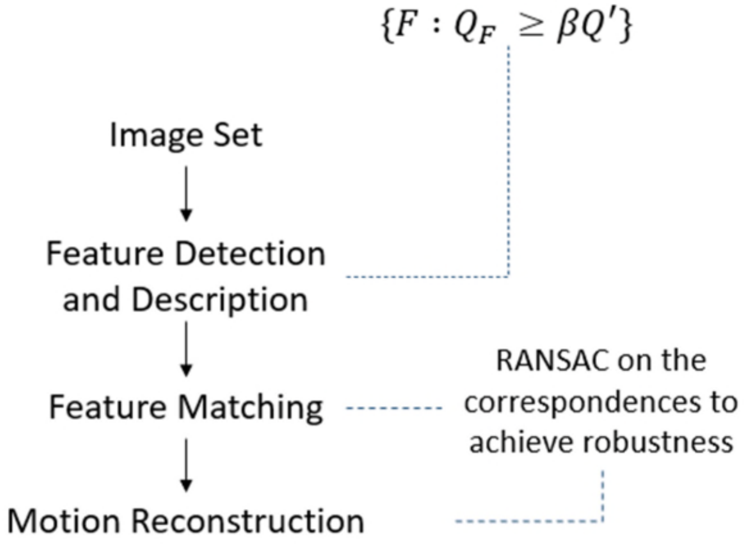

$\{F : Q_F \geq \beta Q'\}$
Image Set
Feature Detection and Description
Feature Matching
RANSAC on the correspondences to achieve robustness
Motion Reconstruction

Figure 2-5 - A possible VO process

The process operates on sequences of images and can used both offline and in real-time applications. Within each image all the optical features with a quality metric Q_F above a certain threshold, let it be $\beta Q'$, are detected, described, and stored. These features are then matched across different views, and a RANdom SAmple Consensus (RANSAC) search [101] can be used to prune outlying matches. From the matched points it is possible to estimate the motion of the platform by reconstructing matrices describing the geometric structure underlying two – or more – images. This reconstruction is operated using methods such as the five [102] point algorithm or the eight point algorithm [103]. The number of points in the name of these algorithms refers to the minimum number of points needed to extract a solution.

However, in some conditions the requirement for a minimum number of points, can lead to a scarcity of detected features, making it impossible to perform motion reconstruction. This is extremely relevant within section 5.2.2 which addresses extrema points in computer vision. These points can capture all the attention for some feature detectors, and if their number is sufficiently low, they can effectively starve the detection process and bring navigation to a halt.

In many implementations of VO, the last step of the process uses computational optimisers such as Bundle Adjustment (BA) to increase the robustness and the accuracy of the estimation [104]. One of the scopes of this thesis is to offload the heavy computational requirements of these methods to the environment and to design choices where the intelligence is coevolved with the operational scenario.

The third method for TRN discussed by Johnson and Montgomery is OBIRON. OBIRON, stands for OnBoard Image Registration for Optical Navigation, and was developed to provide close-range navigation capabilities for the Hayabusa mission [105]. The workflow of OBIRON begins by reconstructing a model of a target object using a technique known as stereophotoclinometry [106]. In stereophotoclinometry, the digital elevation model of an object is built using images of the object, by combining stereophotogrammetry (shape-from-multiple-views) and photoclinometry (shape-from-shading). This workflow requires at least three pictures per region, acquired from different perspectives, which are then used along with a set of priors about the observer's position and attitude to reconstruct the shape and the albedo of the target and the spacecraft pose.

Therefore, OBIRON requires to capture a set of specific information about a target to be deployable. To do so, the probe is required to spend some time orbiting its target to acquire a sufficient number of pictures and allow for the ground processing of them.

This approach appears in antithesis with the paradigm of the thesis, aimed at developing generalised algorithms requiring minimal prior information.

OBIRON is often mentioned in pairings with the AutoNav package. AutoNav is a foundational autonomous optical navigation in deep space. It was developed by JPL to fly on board the Deep Space I (DSI) mission [107], and then employed on multiple other spacecraft visiting small bodies [108,109]. Among the technological trajectories suggested by Riedel et al. [48] to extend the functionalities of AutoNav capitalising on the initial development, there are two which are particularly significant. The first is evolve it to target different types of celestial object, as natural satellites, or planets; the second involves including additional Guidance and Control capabilities, to obtain a full AutoGNC package. The development of a complete, nonspecific, autonomous GNC would indeed represent a huge technological milestone for missions with opportunistic landers.

Lastly, image to map correlation [92], mentioned here for completeness: *"This correlation approach compares a descent image directly to an orbital image of the landing site. First the descent image is rectified to the same scale and orientation as the map and a patch of the image is correlated with the map.".*

Current trends for navigation exploiting ground features involve using AI frameworks such as artificial neural networks (ANNs) and deep neural networks (DNN). Indeed, solutions have been studied and presented to the scientific community, particularly for feature selection [110–112]. However, ANN and DNN are mainly trained using simulated images, and thus possibly unable to capture and be robust towards natural effects. Moreover, ANN and DNN are difficult to certify for space applications, and often require significant computational capabilities to be deployed. Because of all this, neural networks have been flown on very few missions, and never in deep space.

2.2.4 Navigation in Landing Missions

Many navigation solutions have been studied or employed to operate in proximity to the surface of extra-terrestrial bodies (e.g. X has been used on the Moon/Mars). Generally, these methods employ a mixture of sensing capabilities, fused or non-fused, ranging from inertial, to Electrooptical sensors (of which cameras, and thus optical navigation, are a subset) to human controllers.

The table below presents the approaches used in the past six missions who have already touched down on a small celestial body and that will be used in future missions.

Table 2-3 Navigation approaches for missions landing on small celestial bodies

Mission Name	Navigation Strategy to Touchdown
NEAR Shoemaker	Radiometric data through the DSN; camera-based matching of visual landmarks (craters) stored off-line; laser altimeter ranging [113]. Landing was operated through four reductions in Δv [114].
Hayabusa	The pinpoint landing for a touch-and-go for the Hayabusa mission relies on two sensors - a laser rangefinder and a wide-field optical camera - and on the deployment of a fiducial marker [45]. The rangefinder is used for controlling the vertical motion. The horizontal motion is controlled through visual data, by constraining the velocity to maintain the marker centered within the camera output. It appears that MINERVA did not require any landing-oriented navigation, but only navigation for surface mobility [115].
Deep Impact	The navigation payload carried by the impactor consisted of an IMU, a high-precision star tracker, and a CCD camera labelled impactor targeting sensor (ITS) [116]. Navigation along the impact trajectory was achieved employing visual data within AutoNav algorithms [117].
Rosetta	Philae did not possess any landing GN&C capability [49].

Hayabusa2	Hayabusa2 employed two sets information during the low-altitude proximity phase the output of an altimeter and of the target marker (TM)-tracking process [45]. For the last phase of the landing sequence, where the mission requires an elevated degree of accuracy on the landing position the mission improves the strategy employed by Hayabusa. A target marker is used as an artificial topographical feature, and used to reconstruct the spacecraft position rather than only control its velocity.
OSIRIS-REx	OSIRIS-REx carries a wealth of navigation sensors. During proximity operations OSIRIS-REx navigated through star-based centroiding, and natural feature tracking (NFT). In this case, NFT extracts landmarks from digital terrain maps generated through stereophotoclinometry (or “shape-from-shading”) [118].
DART	To impact on Didymos DART will employ proportional navigation [119]. Proportional navigation is based on properties of the line-of-sight and is used to maintain a target and an impactor on a collision course.
Hera	Milani: In nominal conditions, the Milani CubeSat is set to navigate to the ground using a combination of OpNav, LiDAR and radio tracking capabilities [120]. Juventas: In nominal conditions, the Juventas CubeSat is set to navigate to the ground primarily through an inter-satellite link with Hera, assisted by camera and laser rangefinder [66]
ZhengHe	No document appears available on the topic at the time of writing.

3 SOFTWARE AND METHODOLOGY

3.1 Introduction

This chapter presents the software, the tools and the methods that have enabled the definition, construction, and analysis of the proposed camera-only processes throughout the thesis.

These are the developed software tools, organised in a modular package labelled THALASSA, and the ASP (Action, Scenario, Perception) design principle. THALASSA is a complete prototyping framework based on state-of-the-art models and algorithms that was developed to carry out t and can be considered among the core contributions of the thesis. The ASP design approach is a formalisation of an iterative coupled design approach used to develop perception approaches which include considerations about the environment and the platform's behaviour.

The name THALASSA is an acronym, standing for *"Technologies for Hazard Avoidance and Landing for Autonomous Spacecraft with Situational Awareness".* THALASSA is a modular prototyping environment developed during the doctorate with the general scope of developing, testing, and eventually deploying autonomous navigation technologies. The package has been developed to be able to control as many aspects as possible of the simulations. This approach is in line with the need to consider the interaction and propagation of every design choice highlighted within the introduction. In this work THALASSA is used to define the dynamical behaviour of all the elements appearing in the environment of the simulation (Sun, target, landing probe); to simulate an image generation process; and, ultimately, to reconstruct the trajectory the probe is moving on through visual estimation.

Within this thesis, the object used to simulate a target in THALASSA has been in almost every case the asteroid 101955 Bennu, the target of the OSIRIS-REx mission. This is due to three factors. The first is that Bennu can be considered representative of a whole class of asteroid shapes, often labelled 'spin-tops'. For example the target of the Hayabusa2 mission 162173 Ryugu [121], is visually rather similar to Bennu (see Figure 3-1).

The second reason leading to the choice of Bennu is that albeit it possesses a nontrivial geometric shape (as opposed to e.g., a sphere) it remains nearly ellipsoidal (effectively close to a superellipse) – see Figure 3-2. This shape characteristics enables an initial translation of optical navigation methods developed for high-gravity objects.

Lastly, the third factor leading to the choice of Bennu as a principal target is that the OSIRIS-REx mission has generated as a science product a shape model with a surface resolution of 75 cm [122]. This 3D CAD model is available in both an STL and an OBJ format, meaning that it can easily be deployed in most graphic engines. The only section where another asteroid appears as a target is in chapter 5, within the discussion concerning perception fields. In that section another asteroid is introduced in the visual simulation with the goal of having a control reference. This additional celestial body is the contact binary NEA (8567) 1996 HW1 [123].

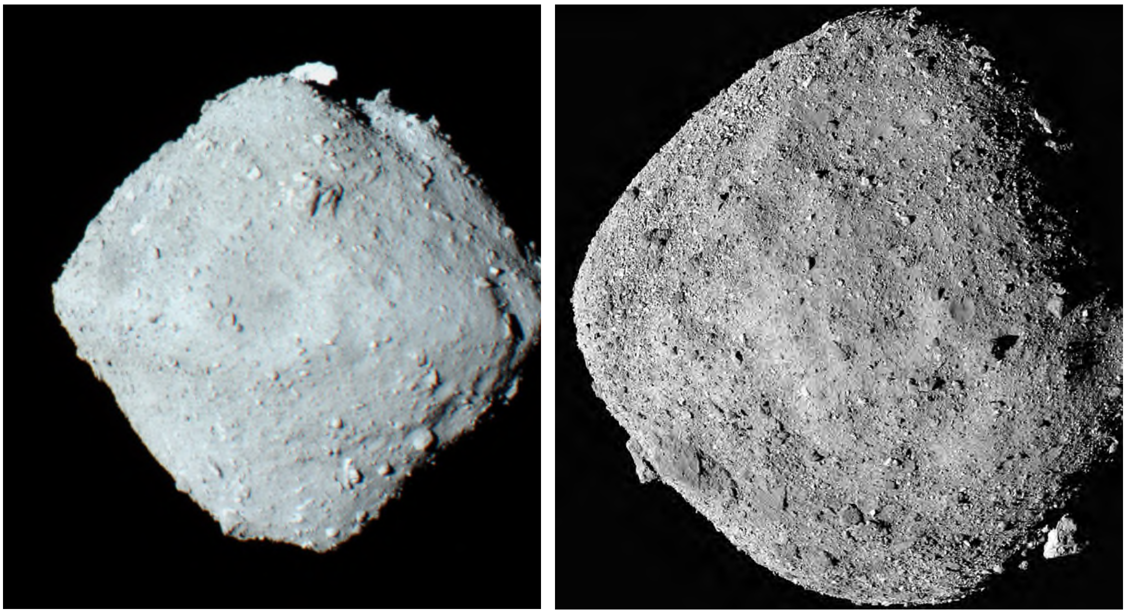


Figure 3-1 - On the left, Ryugu (Image by ISAS/JAXA, Hayabusa 2). On the right, Bennu (Image by NASA/Goddard/University of Arizona).

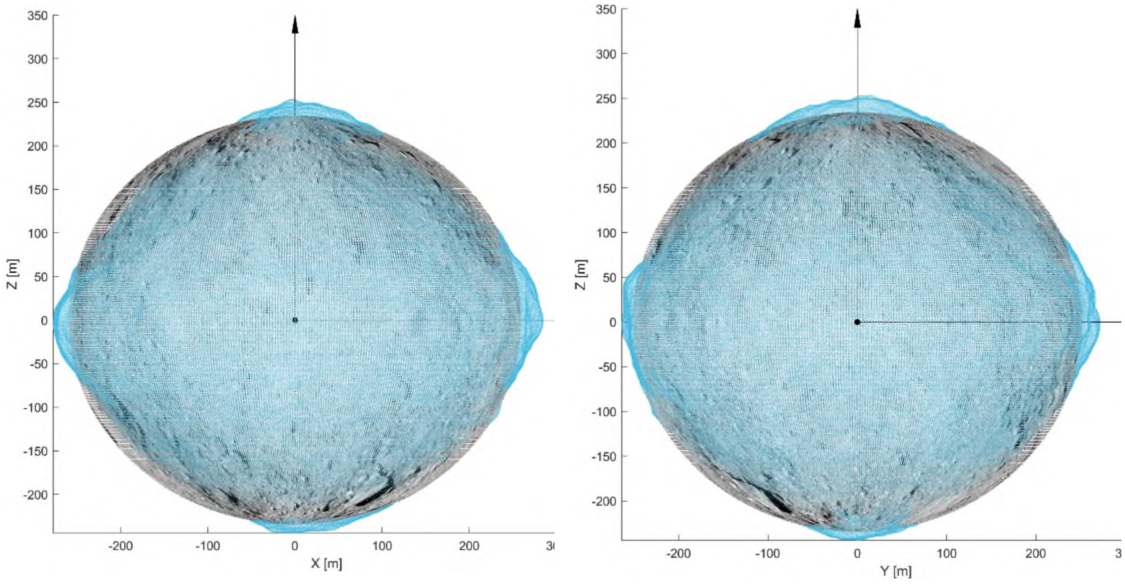


Figure 3-2 - The discrepancy between a three-axis ellipsoid and Bennu's actual shape

There are two versions of THALASSA which have been employed through the thesis, a beta version, and a THALASSA 1.0. The beta version is mostly manually operated and employs Unreal Engine 4 (UE4) as a core graphics engine. THALASSA 1.0 employs Blender as a graphics engine, and automatically generates images starting from a CSV file containing information about the camera pose and the Sun position. In both cases, computations, analyses of the data and visualisation tasks are operated within MATLAB. Future work will be concerned with using Python rather than MATLAB to make the tool completely reliant on open-source software.

While still highly experimental, THALASSA 1.0 can be found at @GitHub.

3.2 Action Scenario Perception (ASP) Triangle

This section presents ASP triangles, a design tool that was used to support and enable the innovation processes reported in this thesis. They explicitly formalise and facilitate a design paradigm that we labelled ‘coupled design’, focused on developing the on-board intelligence of a mobile autonomous platform. ASP triangle are not a software per se but enabled the consistent and systemic development of all the software tools used within the thesis.

The thesis focuses on optical features in computer vision pipelines; therefore, within this work ASP triangles capture the idea that the observable elements within a scene are a function of many interdependent factors, such as what is present within the operational environment, the chosen sensors and the model used to interpret and represent the sensor’s data, the mobility of the probe - enabling the observability of different locations and perspectives, and the timescales of all the processes. To operationalise this heuristic, these criteria were summarised in a three-elements model as motion, scenario, and perception. The intelligent behaviour is assumed to emerge from the interaction of these three coupled macro elements characterising the probe and the mission context. These are Action, Scenario and Perception, encapsulated in the acronym ASP (see Figure 3-3):

- ❖ **Action** denotes any physical change induced on the platform, its sensors, and its actuators by internal or external factors
- ❖ **Scenario** describes the environment within which the platform operates, considered up to a certain degree a designable entity
- ❖ **Perception** concerns the acquisition and initial manipulation of information enabling AI processes.

To define an optimal computer vision approach, these three elements cannot be considered as stand-alone entities, but need to be treated systematically. This is because if the computational elements are designed and deployed without considering the actual operational conditions, there might be significant discrepancies between the internal models and the world, leading to large errors or critical failures. These differences could in a first instance be addressed through the introduction of algorithmical overheads, as additional sensing capabilities or computational optimisers. For example, attempting to exploit an environmental feature that cannot be properly perceived using the on-board sensors or that cannot be consistently observed due to orbital constraints would result in suboptimal, if operational at all, navigation processes.

Moreover, the design approach of the complete navigation package cannot neglect the interactions and emergent properties arising during operations.

To obtain optimal visual acquisition methods, through ASP Triangles we suggest a development and testing process structured as a continuous coevolution of environment, perception and motion elements, until a satisfactory configuration is reached. The couplings and interactions are analysed two-by-two (e.g., environment and perception or motion and perception), and the optimisation is structured as an iterative process. However, in line with the need for standardisation, important for reducing costs and increasing reliability, one or more of these elements could be artificially made independent of the operational cases and maintained constant between missions. A potential example of this, which has been studied in this thesis, is represented by artificial navigation markers used as target features to standardise the visual operational environment. In this case, the environmental element to be observed would be a constant, and the development process could refer to a class of missions, rather than a single one. This would lead to a leaner design process, and enable fast deployments, with important consequences for readiness and the management of launch windows.

Each vertex of the triangle represents an aggregate of subcomponents that can be manipulated in some form. Moreover, each of these vertices can be seen as the composition of two layers: representation and structure. Structure describes the physical properties of the vertex, representation its necessarily approximated model within the AI.

The main inspiration for this paradigm comes from the field of aeroelasticity. Aeroelasticity is an engineering discipline, succinctly described by Wright and Cooper [124] as: *“Aeroelasticity is the subject that describes the interaction of* *aerodynamic, inertia and elastic forces for a flexible structure and the phenomena that can result.*". In 1946 Collar introduced a triangle model to illustrate the coupling of these forces, with each vertex of the triangle representing one force [125]. The systematic study of aeroelasticity has allowed researchers to identify and prevent or control a multitude of catastrophic emerging behaviours, as flutter. Coradeschi and Saffiotti [126] introduced a triangle-like model with similar vertices to the ASP triangle in 2006 to explain the operation of a symbiotic robotics system. The symbiotic robotic triangle, however, appears associated to explaining a behaviour, rather than designing it, and includes human as part of the autonomous entity's operations.

The value of ASP triangles lies in helping designers consider the couplings and interactions between the three ASP components during the design process and include these effects in the developed intelligence. Specifically, they are a graphical representation of the decisional structure underlying a plausible iterative process that could lead to the final, operational architecture. The goal of this approach is to achieve a continuous iterative coevolution process, harmonising and optimising the interplay between co-operating factors.

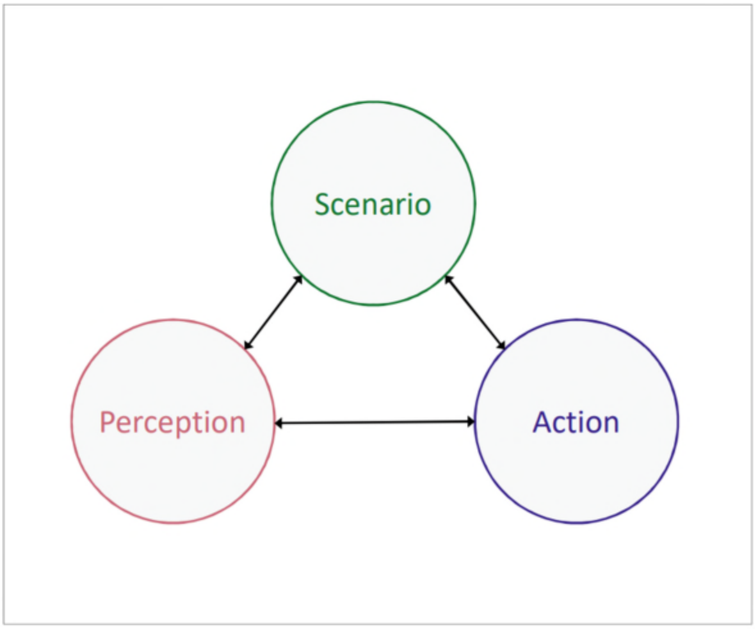


Figure 3-3 An ASP Triangle

To use an ASP triangle, the designer selects a starting vertex and designates it as level 0. All the decisions and adjustments falling under the sphere of influence of that element are taken and made, and their underlying rationales registered. The designer then selects the remaining two vertices, promotes them to level 1, observes the effect of the level 0 decisions on them, and evolves them consistently with the augmented set of constraints. A new starting vertex is then selected from any of the two level 1 ones, and the process of adaption/selection is iteratively repeated until a satisfying architecture configuration has been reached. Figure 3-4 graphically shows this methodological flow.

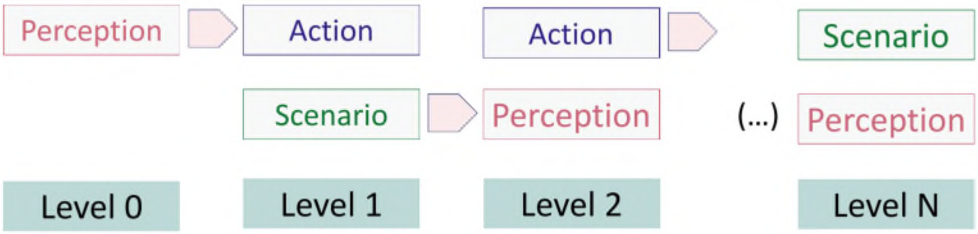


Figure 3-4 Methodological Flow for design based on ASP Triangles

From this, it is possible to interpret all the contributions of the thesis using this paradigm. In this optics, the contribution of section 4 describes how for a fixed Scenario, the Action and the Perception strategies must be adapted to obtain optimal results. Artificial visual landmarks are a way of modifying the Scenario for a given Perception process. It was however observed that this augmented Scenario is critical under some Perception models. This brought the landmark choice to a level 1. In level 1, Perception was selected as the core element, allowing to study the propagation of these fixed Scenario conditions. This analysis led to investigate extreme values in perception.

3.3 THALASSA

For fully autonomous spacecraft, even in simple models, the interdependency of parameters can generate unforeseen emergent behaviours or lead to anomalies or uncertainties. For example, the same object under different lighting conditions and in a different context can induce extremely different responses in a computer vision pipeline (in an ASP paradigm, this could be Perception as a function of the Scenario).

Hence, in designing the GN&C system, it is important to consider the complete operational context and processes. Therefore, the analyses of perception and navigation operated in this work have the scope of testing the behaviour of a complete frameworks or algorithmic process through simulated relevant conditions. These include the spacecraft motion near irregular bodies or changing illumination conditions. These necessities led to the development of a software package named THALASSA.

THALASSA was developed to have five distinct modules, labelled blocks. These are Path, Sensor, Estimation, Filtering and Decision. The last block refers to control actions and processes, and will not be addressed in this work, which is oriented towards testing the behaviour of estimation and perception algorithms. The scope of each block is to transform a set of values to return one or more collections of structured data in output. These outputs can for example be images, matrices of state information, point clouds. When available for the chosen mathematical model, the instance-specific covariance matrix is returned as a product of the block; when a closed formulation is absent the covariance can be estimated through an adequate number of Monte Carlo runs. To the five previously mentioned specialized blocks it is important to add adapter blocks, performing data transformations enabling interoperability between different blocks and software.

To facilitate a rapid design and experimental process, the blocks of THALASSA are developed to adhere to a plug and play paradigm. This is reflected in the definition of the variables and data structures, which remain fully parametric and consistently defined throughout the whole framework. The implementation of THALASSA 1.0 within Chapter 4 is shown in Figure 3-5.

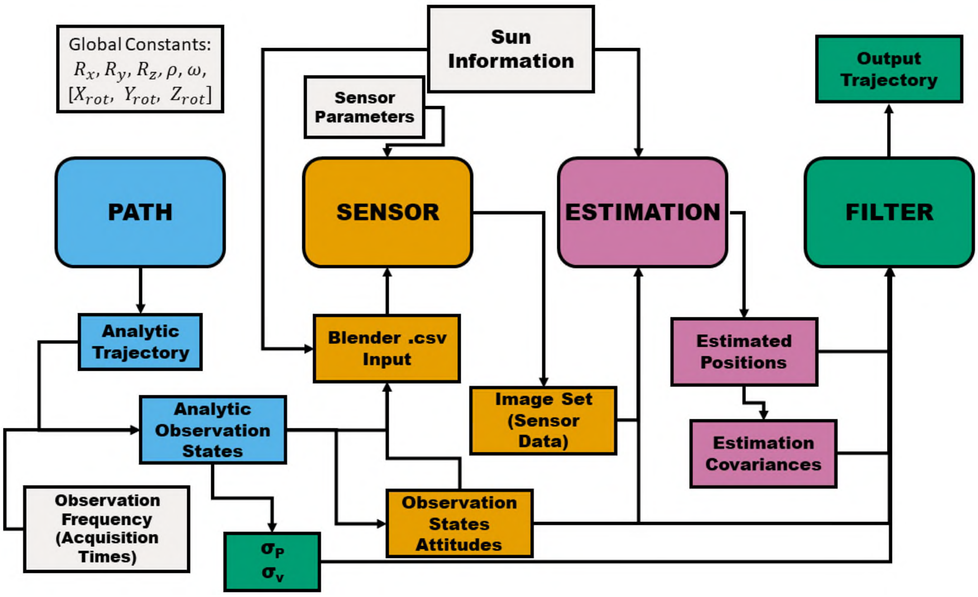


Figure 3-5 Implementation of THALASSA within Chapter 4

The Path block generates the trajectories of the spacecraft. These orbits can be obtained either by solving the ordinary differential equations (ODEs) of a chosen dynamical framework with some opportune initial conditions or from historical data (e.g., benchmark datasets). The Path block yields the nominal reference trajectory, where nominal refers thereafter to the true states of the spacecraft. A subset of points along this trajectory within the observational timeframe represents the states where sensor acquisitions are operated by the associated sensor blocks – see Figure 3-5.

The Sensor block simulates acquisitions of a sensor. These acquisitions are constructed by fusing information on position and attitude with a sensor model and information on the environment.

The Estimation block processes a single type of data, augmented with ancillary information. For example, in Chapter 4, the process is limb-fitting operated on camera outputs. Each of these outputs is augmented using information about the relative attitude of the Sun, the target, and the probe. It is worth noting that

Sensor/Estimation blocks can easily be simulated as noise models applied on the nominal trajectory.

The Filter block is deployed with the goal of integrating multiple information to estimate an unknown set of variables. Using again Chapter 4 as an example, a batch Nonlinear Least Square (NLS) algorithm [127] is employed to reconstruct the orbit the probe is moving on, from the Cartesian positions retrieved using the Estimation block. The NLS method was chosen because of its ease of implementation, and because its behaviour is well-understood. The latter allows focusing on the effects of the geometric approximation rather than on secondary effects induced by the filtering process.

3.3.1 Path

All the orbits studied in this work are obtained solving a system of ODEs with given initial conditions $\boldsymbol{x_0} = [x_0, y_0, z_0, v_x^0, v_y^0, v_z^0]$, without any additional perturbation or actuation. Hence, within the thesis all the orbits are ballistic, and defining a motion model equals to defining a mass distribution inducing a gravitational potential. The vector $\boldsymbol{x_0}$ defines the nominal or true trajectory and is propagated to a specific time t_{end}. Observations are then taken and processed at a time interval Δt, starting at the initial t_0.

3.3.1.1 Motion Model

In particular, the chosen gravitational model is one with a constant density: therefore, the trajectory generation process is a function of the geometry of the target and its orientation in the inertial space. The analysis focuses on a three-axis ellipsoid geometry with some properties equivalent to that of the complexly shaped asteroid Bennu. This can either be rotating or fixed.

From a gravitational model and boundary conditions descend motion models. Within the process it is possible to identify two motion models: the “real” one, defining the nominal trajectory of the probe, constructed within the Path block, and forecasted ones, determined by Estimation and Filter blocks. In this case both employ a constant-density, three-axis ellipsoid gravitational model.

Within the Path block this was selected over less approximate solutions, like a polyhedron model, to avoid introducing any discrepancy between the mathematical structure underlying the real trajectory and the forecasted one.

For Chapter 4, this choice would have mirrored the one analysed in this work for the Estimation block, potentially introducing secondary effects that could have interacted with the results within the scope of this study. For Chapter 5 this degree of realism would be unnecessary. This approximation, however, remains to be addressed in future work.

The chosen equivalent geometry was Bennu's Dynamically Equivalent Equal Volume Ellipsoid (DEEVE) [128], which is defined as the ellipsoid with constant density having the same volume of the body it describes, and moments of inertia in the same proportions. This is the three-axis ellipsoid shown in Figure 3-2.

In this specific case, the radii of the DEEVE have an extension of $R_x = 259\,m$, $R_y = 251\,m$, $R_z = 234\,m$, and its density, which is constant, has a value equal to $1190\,kgm^{-3}$. Moreover, the rotational period τ is considered equal to the real one of Bennu, $\tau = 4.29\,h$ [129]. The reference frame of this ellipsoid is defined consistently with that of the CAD model retrieved from a NASA/Goddard/University of Arizona repository [122]. This has the shortest radius aligned along the Z axis and the longest radius along the X axis. The definition of these parameters is sufficient to construct a closed-form gravity potential approximating the gravity well of the target [4] Given a point $\boldsymbol{s}$ outside the ellipsoid, the set of equations describing the potential in the point $U(\boldsymbol{s})$, written using a notation analogous to the one employed by Scheeres [4], is

$$\begin{cases} U(\boldsymbol{s}) \;=\; -\dfrac{3}{4}\mu\displaystyle\int_{\lambda(\boldsymbol{s})}^{\infty}\varphi(\boldsymbol{s},\kappa)\dfrac{d\kappa}{\Delta(\kappa)} \\ \varphi(\boldsymbol{s},\kappa) = \dfrac{S_x^2}{R_x^2+\kappa}+\dfrac{S_y^2}{R_y^2+\kappa}+\dfrac{S_z^2}{R_z^2+\kappa} \\ \Delta(\kappa) = \sqrt{(R_x^2+\kappa)\big(R_y^2+\kappa\big)(R_z^2+\kappa)} \end{cases} \qquad (3\text{-}1)$$

The only value not yet explained in the above set of equations is $\lambda(\boldsymbol{s})$, which represents the largest real root of the equation $\varphi(\boldsymbol{s},\kappa) = 0$. The partial derivatives of $U(\boldsymbol{s})$, needed to define accelerations and to fill state transition matrices can easily be retrieved from Scheeres [4] and will not be reported here.

3.3.2 Sensor

The Sensor block receives inputs from Path blocks. These inputs are used to generate readings at every point of interest of the trajectory for the chosen sensor, which in this case is a monocular visible camera.

3.3.2.1 Sensor: Blender

Blender, as mentioned before is the graphics engine utilised within the second iteration of the THALASSA package. This section describes how workflows are implemented within Blender.

The first step of the data construction process requires defining all the reference frames employed within the study, to fully describe the poses of all the objects involved. Let $\boldsymbol{f}(t_k)$ be the point in $\mathbb{R}^3$ representing the camera position at the k-th time step, and $\boldsymbol{p}(t_i)$ be the point in $\mathbb{R}^3$ towards which the camera is pointing its boresight at the i-th time step. Within this work the camera will be assumed to be consistently pointing towards the centre of the target; therefore $\boldsymbol{p}(t_i) \equiv \{0, 0, 0\}$ $\forall i$; therefore, the boresight is aligned instant by instant with $-\boldsymbol{f}(t_k)$.

There are three basic reference frames of interest within a Blender sensor block this work. These three are:

- The Target-Centered, Target-Fixed ($TCTF$) frame
- The Visual Reference Frame (VRF)
- The Camera Reference Frame (CRF)

To these, a second CRF, labeled CRF_2, is added, for convenience during the visual simulations. A superscript indicates the reference frame a generic quantity H is defined in. Therefore, H^T is defined in $TCTF$; H^V in VRF; H^C in CRF and H^{C_2} in CRF_2 Moreover, generally VRF

VRF is the global coordinate system intrinsically defined within the simulator employed for the Sensor module, in this case Blender v2.82. Since all the construction data from the Path stage imported in the visual simulator need to be defined in VRF to be used, it is convenient to explicitly maintain this ancillary system. Moreover, generally VRF can be treated as a proper inertial reference frame and used as a reference for all the moving and rotating elements.

The target-centered, target-fixed frame $TCTF$ is defined as the reference frame having the origin coincident with the origin of the mesh used to represent the target, and axes rigidly attached to the geometry of this mesh. As the 3d models representing the targets are often products supplied or retrieved from an external provider, for the sake of reproducibility there is little interest at this stage in altering the barycentre position and the axial orientation from the native ones. Therefore the one pre-defined by the provider and attached to the STL file [122] will be accepted, considered as standard axes, and labelled according to the convention mentioned in the previous subsection. The motion of the satellite is defined and studied in this coordinate system. It is important to observe that in these conditions the illumination is changed through the apparent motion of the Sun-like light source.

Lastly, for convenience in developing the workflow, two reference frames are associated to any camera, let them be CRF and CRF_2. The former, CRF, represents the camera reference frame employed in the estimation algorithm, the latter within the Sensor block. For the $m-th$ time-step, CRF is defined by three axes $\{\boldsymbol{x}^C, \boldsymbol{y}^C, \boldsymbol{z}^C\}$. These axes are defined as:

- $\boldsymbol{y}^C$ represents the transversal component of the reference frame. It is aligned with the orbit normal, and is obtained from the unit position vector $\hat{\boldsymbol{x}}_r(t_m)$ and the unit velocity vector $\hat{\boldsymbol{x}}_v(t_m)$ as $\boldsymbol{y}^C(t_m) = \hat{\boldsymbol{x}}_r(t_m) \times \hat{\boldsymbol{x}}_v(t_m)$;
- $\boldsymbol{z}^C$ represents the radial component of the reference frame and is aligned the direction of the boresight. Since $\boldsymbol{p}_i$ is considered to be identically equal to $\{0,0,0\}$, $\boldsymbol{z}^C(t_m) \equiv -\hat{\boldsymbol{f}}(t_m)$;
- $\boldsymbol{x}^C$ represent the out of plane component of the reference frame, completes the right-handed tern of vectors and is obtained as $\boldsymbol{y}^C(t_m) \times \boldsymbol{z}^C(t_m)$.

When a simulated camera is added to a graphic engine, the former is introduced as an object with its own local reference frame, to exploit the full 6 DoF of rigid body motion. This intrinsic orientation is engine-specific and even with all the rotation angles set to zero cannot be expected to match either CRF or VRF.

For example, Blender v2.82 creates a camera with the boresight aligned with its $-Z^V$, while Unreal Engine 4 generates a camera with a boresight along its $+X^V$, with both VRF completely differing from each other. Therefore CRF_2 is introduced to operate on the “as-spawned” cameras, and to retrieve the transformations necessary to rotate the camera object in the graphic simulation in such a way that the optical and the dynamical data consistently model the same scenario. For the reference frames employed in this work, to transform a column vector from CRF to CRF_2 in Blender it is sufficient to employ the rotation matrix

$$\boldsymbol{R}_C^{C_2} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \qquad (3\text{-}2)$$

This rotation matrix can be thought as a global adapter between the complete process and the specific visual simulation environment, the associated code portion as a THALASSA adapter block.

<u>*Observations*</u>—For each trajectory an observation is performed every Δt hours, starting from a time t_0 and up to a time t_{end}.

The environment has two parameters of interest, the surface characteristics of the target, and the light source. The former represents the simulated response to light of the material and its colour within the graphics engine. These were maintained to the default values assigned by Blender upon importation. Regarding the latter, even though Blender could potentially fully accommodate a more complex Sun behaviour, within this work it was chosen to constrain the position of the source of illumination within the scene to the equatorial plane. Moreover, the orientation of the bundle of parallel light rays is forced to be parallel to the segment connecting the light source to $\mathbf{0}^V = \{0, 0, 0\}^V$.

From these assumptions it was possible to define the motion of a Sun like light source within Blender. For simplicity, the axial tilt of Bennu was set to 0°, therefore the light source is constrained to move within Bennu's equatorial plane. This means that instant by instant the position of the Sun position is well-defined by only one parameter, its angular position on the equator ϑ.

Ultimately, from the dense nominal trajectory supplied by the path module, a subset of points is extracted employing the above time-based sampling condition. These w points define observation conditions and are introduced in Blender using a .csv file. This file is constructed so that all the w rows have the structure shown in Table 3-1.

Table 3-1 - A generic row of the .csv input file

Column	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Element	X	Y	Z	α_X	α_Y	α_Z	ϑ

The first three columns represent the Cartesian coordinates of the camera, in meters, the columns from 4 to 6 contain the angles describing the attitude of the sensor in radians, and, lastly, the seventh column contains the angular position of the Sun.

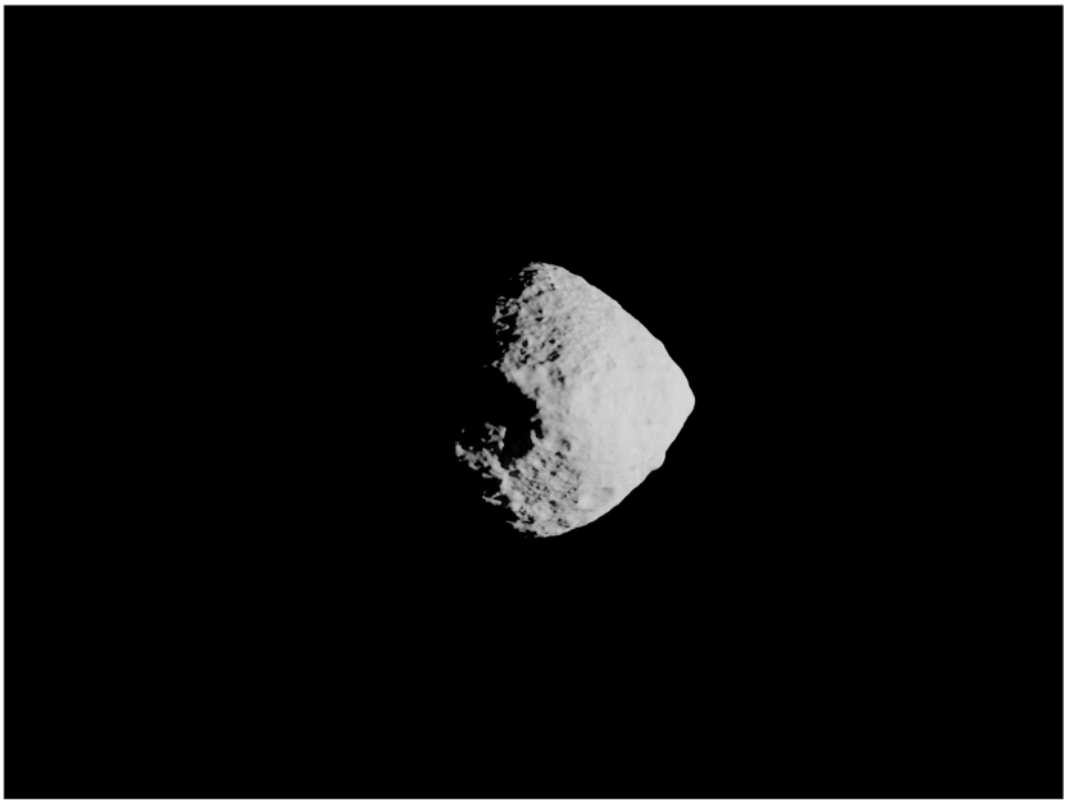


Figure 3-6 - A view of the target object in Blender

3.3.2.2 Sensor: Unreal Engine 4

The version of the Sensor block presented in this subsection represents a component of THALASSA’s first version (Beta). Compared to the Sensor block deployed using Blender, this version represents a much less sophisticated tool. The Blender-based block allows to perform most operations automatically through the initialisation of a few initial parameters. On the other hand, within the UE4 block most operations are performed by the user. This, however, is not due to the limitations of UE4, but rather a design choice driven by the different volume of image needed when the two blocks were developed.

In particular, the Sensor block employing UE4 was developed to analyse the effect of NAV-Landmarks within a Visual Odometry framework. The limited amount of test trajectories, and the few images per trajectory implied that this choice was not cumbersome on the experimenter.

It is worth noting that the reference frame defined within MATLAB or Blender differs from the one defined within Unreal Engine: namely, whereas the formers are right-handed, the latter is left-handed.

Within the first iteration of the Sensor block, three equally plausible development alternatives were evaluated: PANGU [130], the Simulink 3D Animation toolbox [131] and Unreal Engine 4. The choice, ultimately, fell on Unreal Engine 4 [132] (specifically, UE4.22.3). At first this selection might appear counterintuitive: PANGU is optimized to simulate scenes of planets and asteroids, and the 3D Animation toolbox can be easily integrated with the rest of the MATLAB-developed code. However, UE4 offers complete and direct control over the scenario and the cinematography, thus allowing more precise experimental operations. Moreover, within the context of this application UE4 is the only software of the three which is completely free. Thus, using it to develop tools and models promotes to the rest of the community a more accessible and flexible tool to construct, improve or expand their own models.

The visual model representing Bennu was constructed again starting from the NASA-Goddard-University of Arizona CAD model with a resolution of 75 centimetres [122].

It was deemed superfluous to model the background stars, as within the scope of this work, given the closeness to the surface, the magnitude of the target dominates the visual field, masking all the other, faint, objects. The same principle was applied within Blender-based studies.

Lighting, however, plays an important role in this simulation. Since the Sun can be considered a light source at infinity, it can be represented as a bundle of parallel light beams and modelled within the simulation as a “Directional Light”.

The direction of this light is chosen originating at $-\infty$ and parallel to the X axis defined in the Unreal Engine reference frame, here labelled X^{UE}. Therefore, the terminator will be oriented along Y^{UE}, with the sunlit area belonging to the negative X^{UE} region. The light intensity is selected heuristically, concurrently with some characteristics of the asteroid surface, trying to keep the target chromatically and visually like the one observed in pictures. Lastly, to obtain the colour of the light, the concept of colour temperature, mapping the temperature of a black body expressed in K into the RGB colour space is employed. The effective temperature of the Sun 5780 K [133]. A table listing Kelvin to RGB conversions [134] lists the hex value #fff0e9 as the colour associated to 5800 K, the temperature in the table which better approximates the Sun's 5780 K. The representation of #fff0e9 in RGB is the triplet $(255, 240, 233)$, shown in Figure 3-7.

At the moment there does not seem to exist a comprehensive package validated by the community mapping surface properties. Therefore, they are treated as a set of smaller heuristic problems. The first one addressed is represented by the radiative properties of the surface. From Lauretta et al. [129] it can be observed that the geometric albedo of Bennu is extremely low, having a value of $4.4\% \ (0.04)$. Therefore, the surface is modelled as a matte material (Roughness factor $= 1$), completely lacking specular reflections (Specular factor $= 0$). The albedo value, moreover, offers an important information to heuristically constraint the colour. The value of 0.04 is in fact roughly that of fresh asphalt [29]. This suggests a very low grayscale value, which was selected as $1/16$ of the 8-bit scale. This is the RGB value of $(15,15,15)$, corresponding to a normalised RGB triplet of $(0.058, 0.058, 0.058)$. The colour obtained from $(15,15,15)$ can be observed in Figure 3-8. Ultimately, the visual result of the complete scenario is the one seen in Figure 3-9.

The landmarks require a similar process to be visually characterized. In particular, they are assumed to be expandable structures with an aluminium coating. From an online colour repository [135] it is possible to retrieve a RGB value associated to Aluminium, equal to $(214, 214, 214)$, which, normalised, results in $(0.839, 0.839, 0.839)$, seen in Figure 3-10. This material will be characterized as metallic (Metallic factor $= 1$) with Specular and Roughness factors $= 0.5$.

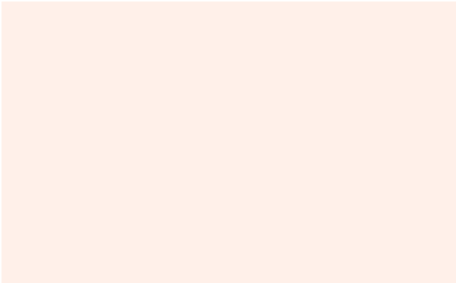


Figure 3-7 - The colour chosen for Sun illumination, RGB = (255, 240, 233)

Figure 3-8 - The colour chosen for the asteroid surface, RGB = (15, 15, 15)

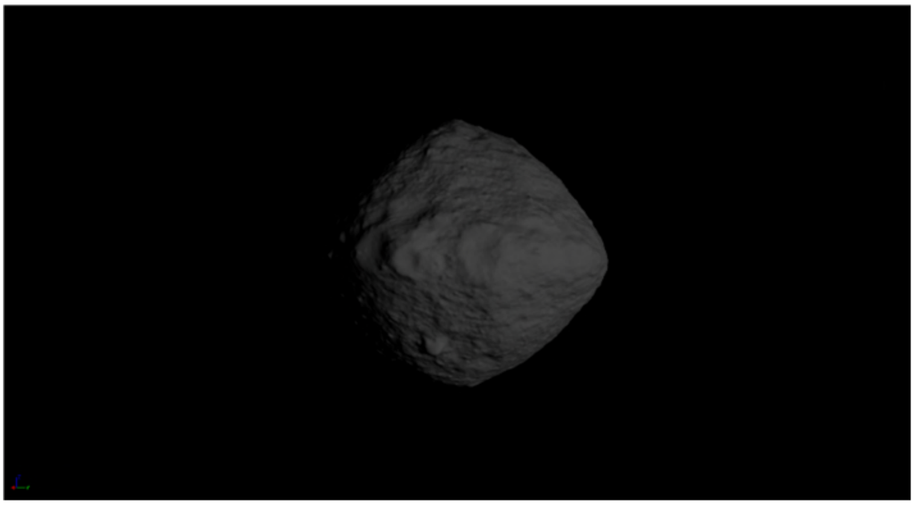


Figure 3-9 - The look of the asteroid in the simulation

Figure 3-10 - The colour chosen for the landmarks, RGB = (214, 214, 214)

3.3.3 Estimation

The Estimation block is tasked with operating on the images generated by the Sensor block. Algorithms within the Estimation block retrieve an estimation of the state vector of the probe (or a portion of it) or extract cognitive insights from the sensor data.

Chapter 4 and Chapter 5 contain in-depth descriptions of the corresponding Estimation blocks. The estimation algorithms of Chapter 4 employ a limb-fitting algorithm, whereas the estimation processes of Chapter 5 consider the use of feature-based methods.

3.3.4 Filter

Within the scope of this work, the Filtering block pools multiple sensor observations, and their corresponding levels of confidence, generating an estimation of the probe trajectory.

This block is absent from the techniques of Chapter 5. This is in line with the principle that algorithmic optimisation techniques should be replaced by methods intrinsically stabilised by appropriate observable selection. Moreover, navigation techniques such as visual odometry can perform a visual reconstruction incrementally, without having to fit observations to a model through a filtering method.

In Chapter 4 the estimations are collected in a Nonlinear Least Squares (NLS) framework [127]. The scope of the Filter block in Chapter 4 is to transform a collection of independent position estimates into a trajectory within the chosen gravitational model.

The NLS process requires an initial guess for the trajectory of the spacecraft, to begin the iterative process of residual minimization. The Filtering block generates a first guess, or a-priori knowledge of the spacecraft motion, by corrupting the true nominal trajectory $\boldsymbol{x_0}$ with error randomly extracted from a Gaussian distribution. This first guess is then referred as the reference trajectory $\boldsymbol{x_{ref}}$, which is iteratively improved in accuracy through the NLS process. Below the algorithm and its implementation within this work are presented, consistently using a notation borrowed from Wiesel [127].

3.3.4.1 Overview of NLS Algorithm

Out of all the w camera acquisitions, a number $w_R \leq w$ is used within the filtering process. This is in line with the contribution of this work, suggesting that only a subset of the acquired data points should be used.

For each considered time step k, the input data estimated using camera observations are organised in a vector, labelled $\boldsymbol{z}$. In this case, this is the $3x1$ array $\tilde{\boldsymbol{x}}_r^C(t_k)$ returned by the OpNav process. For each of these $\boldsymbol{z}(t_k)$, a $3x3$ covariance matrix $\boldsymbol{Q}(t_k)$ is assumed available. $\boldsymbol{Q}(t_k)$ is here obtained analytically from the estimation model.

The NLS process is initialized by propagating the initial reference trajectory $\boldsymbol{x}_0(t_o)$ to the times k of each observation, thus obtaining $\boldsymbol{x_{ref}^V}(t_k)$ and the state transition matrix $\boldsymbol{\Phi}(t_k, t_0)$ for each observation time t_k.

For each time step k the difference between the estimated data and the reference observation transformed to the same reference frame of $\boldsymbol{z}(t_k)$ defines the so-called residual vector $\boldsymbol{r}(t_k)$, where

$$\boldsymbol{r}(t_k) \; = \; \boldsymbol{z}(t_k) - \; G(\boldsymbol{x}_{ref}(t_k), t_k) \qquad \textbf{(3-3)}$$

To summarise, for each instant where a camera observation is both acquired and used to correct the available reference trajectory, five elements are required as input data. These are: the array of reference values at that instant $\boldsymbol{x}^{V}_{\boldsymbol{ref}}(t_k)$; the corresponding array of observed values $\tilde{\boldsymbol{x}}^{C}_{r}(t_k)$ and the uncertainties associated to them $\boldsymbol{Q}(t_k)$; the state transition matrix relative to the reference trajectory $\boldsymbol{\Phi}(t_k, t_0)$ and, lastly, all the parameters required to transform the reference values into a mathematical form enabling a direct comparison with the observations. In this case, the last input is simply the rotation matrix $\mathrm{G}(\cdot) = \boldsymbol{R}^{C}_{V}$ converting data from VRF – in which the reference states are defined – to CRF.

3.3.4.2 NLS Process

This section briefly explains how the set of inputs of the NLS process is used to construct its outputs. The NLS process returns a correction to the initial state vector for the desired model, and the covariance associated to this updated initial state. Let these be, respectively, $\boldsymbol{x}_{ref+1}(t_0)$ and the covariance matrix $\boldsymbol{P}_0$.

To obtain $\boldsymbol{x}_{ref+1}(t_0)$ and $\boldsymbol{P}_0$ it is necessary to construct two ancillary elements for each timestep t_k. These are the residual vectors (Equation (3-3)) and the observation matrix $\boldsymbol{T}_k = \boldsymbol{H}_k\boldsymbol{\Phi}(t_k, t_0)$. Also employed here are the covariance matrices $\boldsymbol{Q}(t_k)$ providing information on the level of confidence of the measurement at t_k.

Once the three sets have been structured for every considered timestep, $\boldsymbol{P}_0$ is obtained as

$$\boldsymbol{P}_0 = \left(\sum_k \boldsymbol{T}_k{}^T \boldsymbol{Q}_k^{-1} \boldsymbol{T}_k\right)^{-1} \tag{3-4}$$

$\boldsymbol{P}_0$ is then used to determine the correction to the initial state vector, through the formula

$$\Delta\boldsymbol{x}_{ref} = \boldsymbol{P}_0 \sum_k \boldsymbol{T}_k{}^T \boldsymbol{Q}_k^{-1} \boldsymbol{r}_k \tag{3-5}$$

Which ultimately leads to

$$\boldsymbol{x}_{ref+1}(t_0) = \boldsymbol{x}_{ref}(t_0) + \Delta\boldsymbol{x}_{ref} \qquad \textbf{(3-6)}$$

While the values in Equation (3-4) and Equation (3-6) represent the sought outputs, it is important to also store the residuals of the process. This enables to verify that if convergence is achieved the solutions are not merely a result of the mathematical process but physically meaningful.

3.3.4.3 Implementation Details

As it can be seen from the Equations from (3-4) to (3-6) the implementation of the Nonlinear Least Squares algorithm is rather straightforward. Therefore, the main concern for implementation appears to be on setting up the input data structures, in particular the state transition matrix $\boldsymbol{\Phi}(t_k, t_0)$. To obtain $\boldsymbol{\Phi}(t_k, t_0)$ it is necessary to solve a matrix differential equation involving the Jacobian of $\dot{\boldsymbol{x}}$ – that is, the matrix $\boldsymbol{A}(t_k)$. $\dot{\mathbf{x}}$ contains the velocity and acceleration components of the probe. The already available information allows to obtain the gravitational potential and the linear acceleration experienced by the platform in the considered point, and their first derivatives with respect to the components of $\underline{\boldsymbol{z}}$. This because these are function of the standard gravitational parameter and of the position of the object only, both of which are assumed known. It is worth noting that this work assumes a perfect knowledge of the standard gravitational parameter, whereas in real operations this would most likely not be true.

This work is performed within an asteroid-fixed asteroid-centered reference frame, co-rotating with the target. When the chosen reference frame is non-inertial, as in this case, it is necessary to add additional forces to the potential terms. Therefore, it is necessary to also consider within the model the Coriolis acceleration, let it be $\boldsymbol{a}_{C1}$, and the centrifugal acceleration, let it be $\boldsymbol{a}_{C2}$. The asteroid will be assumed to be in uniform rotation along its Z axis, aligned with the inertial Z axis, with a constant rotational speed ω ($\boldsymbol{\omega} = \omega[0,0,1]$; $\dot{\omega} = 0$). As shown below, this assumption simplifies both non-inertial accelerations to

$$\boldsymbol{a}_{C1} = 2\boldsymbol{\omega} \times \boldsymbol{v} = 2\omega[-v_y, v_x, 0] \tag{3-7}$$

$$\boldsymbol{a}_{C2} = \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{r}) = \omega^2[-x, -y, 0] \tag{3-8}$$

Therefore, the complete expression of $\dot{\boldsymbol{x}}$ is

$$\dot{\boldsymbol{x}} = \begin{bmatrix} \boldsymbol{v} \\ \boldsymbol{a} \end{bmatrix} = \begin{bmatrix} \dot{\mathrm{x}} \\ \dot{\mathrm{y}} \\ \dot{\mathrm{z}} \\ \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} v_x \\ v_y \\ v_z \\ U_x - a_{C1_x} - a_{C2_x} \\ U_y - a_{C1_y} - a_{C2_y} \\ U_z - a_{C1_z} - a_{C2_z} \end{bmatrix} \tag{3-9}$$

From Equation (3 - 9), it is possible to retrieve $\boldsymbol{A}(t_k)$

$$\boldsymbol{A}(t_k) = \begin{bmatrix} 0_{3x3} & \boldsymbol{I}_{3x3} \\ \boldsymbol{A_r} & \boldsymbol{A_v} \end{bmatrix} \tag{3-10}$$

$$\boldsymbol{A_r} = \begin{bmatrix} U_{xx} + \omega^2 & U_{xy} & U_{xz} \\ U_{yx} & U_{yy} + \omega^2 & U_{yz} \\ U_{zx} & U_{zy} & U_{zz} \end{bmatrix} \tag{3-11}$$

$$\boldsymbol{A_v} = \begin{bmatrix} 0 & 2\omega & 0 \\ -2\omega & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{3-12}$$

Where $\boldsymbol{I}_{3x3}$ is a 3x3 identical matrix.

Knowing $\boldsymbol{A}(t_k)$ finally allows to solve the matrix differential equation returning the sought state transition matrix, $\boldsymbol{\Phi}(t_k, t_0)$

$$\dot{\boldsymbol{\Phi}}(t_k, t_0) = \boldsymbol{A}(t_k)\boldsymbol{\Phi}(t_k, t_0) \tag{3-13}$$

Once $\boldsymbol{\Phi}(t_k, t_0)$ has been obtained all the required elements are available. Ultimately, this is the model that is integrated to reconstruct a trajectory from the observations.

It is worth noting that this model neglects any form of perturbation to the spacecraft. An improved approximation, for example, could include solar radiation pressure (SRP). A first order model to do so can be found in Jean, Ng and Misra [136], who assuming a perfectly spherical spacecraft simplify a general model for the SRP acting on spacecraft operating in the proximity of binary asteroids to:

$$a_{SRP} = \frac{2P_{SRP}}{B}\hat{u} \tag{3-14}$$

Where P_{SRP} is the pressure of the solar radiation on a spacecraft, B is the mass to area ratio of the spacecraft and $\hat{u}$ is the vector aligned with the Solar radiation.

This is the so-called “cannonball” model, which assumes a purely reflecting spacecraft. A cannonball model removes attitude considerations from the model, and represents the highest force exerted by SRP over the object. Being this the max acceleration that can be induced by the SRP this value can be used to test if the gravity acceleration is consistently larger than $max(a_{SRP})$ or if the two accelerations are comparable.

González et al. [137] report that for 3U, 6U and 12U platforms the mass to (frontal) area ratio is 3.33×10^{-3}. As these configurations are the most common this will be the value introduced in the equation.

P_{SRP} can be modelled as

$$P_{SRP} = \frac{P_0}{c}\left(\frac{R_0}{R}\right)^2 \tag{3-15}$$

With P_0 being the solar flux at 1 AU, c the speed of light, R_0 the value of 1 AU in meters and R is the distance between the spacecraft and the Sun – which can be confounded with the distance between the target asteroid and the Sun

By substituting all the values, it is possible to obtain:

$$P_{SRP} = \alpha\left(\frac{1}{R}\right)^2 = \alpha_0\left(\frac{R_0}{R}\right)^2 \tag{3-16}$$

Reasonable boundaries to map this value for a quick reference, are $R_{max} = 1.3$ AU, as per definition of NEO, and $R_{min} = 0.046$ AU, the expected perihelion of the Parker Solar Probe [138] which will be the closest artificial object to the Sun.

4 HORIZON BASED PERCEPTION AND NAVIGATION

4.1 Introduction

This chapter introduces the first experimental results of the thesis. It presents ideas and results relating to a set of altitudes where the target NEO is significantly more prominent than background stars in the image, yet fully contained within the field of view of the camera. Within these ranges, optical navigation can use as its observable the edge of the target illuminated by sunlight. This altitude is here considered to be between roughly 1 km and 2 km, either through purely demonstrative acquisition meshes (e.g., spheres sampled with constant angular spacing) or circular orbits.

As noted in Chapter 1 and 2, NEOs present a large variety in 3D shapes, and therefore in the geometry of their edges as seen within the on-board camera. In line with the arching scope of the thesis, the aim of this chapter is to study invariant aspects at the altitudes of interest. Again, an invariant is a common underlying element shared by many targets that can serve as the observable for a standardised navigation method. Identifying invariants enables developing algorithms that function with numerous targets while requiring, at most, a restricted number of parameters or information to be provided as a prior.

The invariance targeted within this chapter concerns using a position estimation algorithm always operating on a simple edge shape: an ellipse. Specifically, this work presents methods to extend the usage of this position estimation algorithm exploiting a standard shape to perform trajectory reconstruction around NEOs. In particular, the novelty proposed in thesis concerns how to integrate environmental information to make this method more robust without altering the algorithm itself. This algorithm was initially developed by Christian and Robinson [139] for bodies well approximated by three-axis ellipsoids, but is here employed in the proximity of target shapes more complex from a computer vision perspective, through apt management of the visual data acquisition strategy. This positioning algorithm is computationally light, and it appears that for NEOs it can be brought to a camera-only condition.

The trajectory reconstruction process reduces a set of unconnected estimations to a state vector that can be propagated within the on-board gravitational model. This condensed model must possess an uncertainty coherent with the desired navigation accuracy. Using an invariant-targeting method builds upon the hypothesis that approximating single-lobe convex objects to three axis ellipsoids induces an error that is generally within acceptable navigation thresholds. More specifically, it has been observed that regions inducing large errors are well-predicted by information about illumination and coarse information about the shape of the target.

Therefore, the hypotheses enabling this approach and tested within this work are:

1. The error induced by the reduction to a much simpler shape is generally well explained by both the coupling of the reduced and the real shape, and the solar illumination conditions
2. Large errors appear in spatially-bounded clusters.

This allows predicting critical acquisition regions, which would disrupt or corrupt the orbit reconstruction process, and eventually removing data acquired within them. This removal ideally maintains all the errors below a desired threshold.

Modulating the data acquisition strategy over avoidance of critical clusters, potentially opens the way to the repeated use of a plug-and-play algorithm for multiple missions. This work individuates, spatially maps, and discusses some of the conditions inducing large errors. Three acquisition parameters appear fundamental in regulating the uncertainty of the output. The first of these is the response of the reconstruction algorithm to the observation sampling frequency, for observations with given uncertainties. The second is the response of the position estimation model to changes in the position of the platform, which is non-trivial for operations around irregular bodies. The third and last is the effect of solar illumination on the localisation process, seen as a function of the Sun phase angle φ.

Propagating this information about error clusters could enable the computationally light, camera based and non-specific algorithm to be redeployed without any significant increase in complexity into resource constrained platforms operating around asteroids.

It is evident that the optical estimation algorithms discussed in this section differ starkly from the ones discussed in Section 2.2. Indeed, processes close to the ground tend to be feature-oriented, while the processes of chapter 4 can be seen closer to far-range methods for navigation employed while orbiting large gravity bodies [140]. The methods of chapter 4 remain relevant for landing because:

- The lack of an atmosphere around small celestial objects does not allow to distinguish clear approach phases such as Entry, Descent and Landing (EDL). Therefore, landing can be intended as extending well beyond the last few dozens of meters in the proximity of the ground.
- Given the small scale of the celestial bodies of interest, techniques exploiting the full body as a visual observable can be used up to quite close to the surface.

Within the landing framework presented in this thesis, outputs from the set of altitudes discussed within this chapter are particularly important for two reasons:

1. The first is that since monocular methods are up to scale and not global, this output can serve as a starting point to propagate scaled information within a global reference frame.
2. The second reason is that if the orbital information is sufficiently accurate its propagation can be used to trigger altitude or velocity-based events.

4.2 Simulation and Experimental Design

The simulation is operated within a THALASSA 1.0 framework, specifically with the structure shown in Figure 3-5.

4.2.1 Path

The Path block employs a three-axis ellipsoid gravitational model to generate orbits with a circular initial condition. The orbits of interest for each testing case will be discussed in the relevant result sections.

4.2.2 Sensor

The simulated camera has a focal length of $f = 9.6\,mm$, a null skew value, and square pixels with a side of $2.7 \times 10^{-5} mm$. The images returned in output by this simulated sensor have a size of 1280x720 pixels. Ultimately, the intrinsic matrix of the camera, $\boldsymbol{K}$ is completely constrained from these values. The values of f and the size of imaging plane have been retrieved from the datasheet of the C3D CubeSat Camera, which is a CMOS camera with a TRL of 9 that has already flown on CubeSat missions [141].

All the parameters of the surface have been left unchanged from the STL model as introduced in Blender, while the 'Energy' parameter of the Sun simulated within the computer graphics engine has been set to 30.

4.2.3 Estimation

The Estimation block is tasked with operating on the images generated by the Sensor block and retrieve a Cartesian estimation of the position from which the acquisition was performed. This information is retrieved through, an Optical Navigation (OpNav) algorithmic process, originally presented by Christian and Robinson [139]. This is a model-based approach exploiting the border of the sunlit surface of the target, which constitutes an extremely robust, and stable observable. Additionally, this algorithm does not require multiple iterations to return an output, and because of this it appeared well-suited for resource-constrained missions.

Except for peculiar spatial arrangements, the presence of the Sun generates over every object situated within the Solar System an illuminated region, in stark contrast with the black background of empty space. The edge of this sunlit region, appearing as a curve in the image, represents a discontinuity, either between open space and the sunlit region, or between the unilluminated and the sunlit regions. The former will be addressed as limb, while the latter as terminator. The limb has a broad extension, is easily detectable, and provides a large wealth of information.

At these altitudes, methods exploiting point-like local visual features (e.g., stellar parallax) were neglected because the space regions surrounding small bodies, can potentially be dust rich and could lead to ambiguities [139].

4.2.3.1 Algorithm

The chosen OpNav algorithm exploits the limb of the target and was developed for spherical or ellipsoidal objects. It operates an estimation of the position of the probe, using the outputs of a camera and additional information about the target and its attitude with respect to the probe. The algorithm requires a set of priors – such as known size and ephemerides of the target and knowledge of the relative attitude of the probe. Ultimately, the OpNav bridges the projection of the target observed in the image plane of the camera, approximated by an ellipse, to a quadric surface describing the object.

The process is further discussed by the LUMIO team for its mission design [142,143] and by Liounis [144]. Furthermore [144] Liounis analyses different approaches to perform this estimation, testing it even on two irregular bodies – from a computer vision perspective, including Bennu, and presenting a more specific approach extending horizon-based techniques to non-ellipsoidal bodies. This method does not match the paradigm discussed in thesis, as it would reintroduce a high degree of specificity in the workflow.

It is now worth noting that beyond inducing couplings which are difficult to handle within this estimation process, the complex shape of asteroids could also have beneficial effects on the overall framework.

Indeed, it could help transforming the estimation process presented here into a camera-only algorithm. The OpNav algorithm requires ancillary information, as the attitude, which is considered not retrievable from the camera itself. This must be assumed provided by an additional external sensor, as for example a star tracker. In the case of position reconstruction from an ellipsoid the attitude information is fundamental to introduce constraints reducing the number of solutions from infinite to one [145] owing to the infinite degrees of symmetry of the shape. Asteroids generally possess a much smaller degree of symmetry when compared to ellipsoids. In addition to this, local features as craters, boulders or ridges can further help differentiate between similar portions of the surface. Therefore, in the future it would be possible to envision systems providing attitude reconstruction through fusion of visual recognition of known local landmarks and the instantaneous illumination conditions.

Using the same notation as [139], the algorithm, for each observation, can be summarised as follows:

a. The shape matrix $\boldsymbol{A}^T$ is defined to succinctly describe the quadric equation defining the ellipsoid in the TCTF reference frame

$$\boldsymbol{A}^T = \begin{bmatrix} {}^1\!/\!_{R_x^2} & 0 & 0 \\ 0 & {}^1\!/\!_{R_y^2} & 0 \\ 0 & 0 & {}^1\!/\!_{R_z^2} \end{bmatrix} \qquad (4\text{-}1)$$

The matrix of Equation (4-1)**Error! Reference source not found.** is then converted to CRF through the appropriate rotation matrix $\boldsymbol{A}^C = \boldsymbol{R}_T^C \boldsymbol{A}^T \boldsymbol{R}_C^T$. The newly obtained $\boldsymbol{A}^C$ is then decomposed into a product of triangular matrices using a Cholesky factorisation process, such that $\boldsymbol{A}^C = \boldsymbol{U}^T \boldsymbol{U}$. The triangular matrix $\boldsymbol{U}$ ultimately serves as the basis to transform the problem into a more convenient representation subspace, namely one where the navigation problem can be solved around a unit sphere.

b. A set of M limb points is extracted from the ellipse approximating the target observed within the acquired image. The coordinates of those pixels, $(x_i, y_i), i = 1, \dots, M$, are stored in column arrays with the form $\boldsymbol{s}_i^T = [x_i, y_i, 1]$. These are pre-multiplied row by row by the matrix $\boldsymbol{U}$; normalised; transposed and ultimately concatenated in a $M \times 3$ matrix, named $\boldsymbol{H}$.

c. The matrix $\boldsymbol{H}$ is used in a linear least squares problem to retrieve the ancillary vector $\boldsymbol{n}$, with $\boldsymbol{Hn} = \underline{\mathbf{1}}$. The vector $\underline{\mathbf{1}}$ is a column array with M rows containing only entries identically equal to 1. Once $\boldsymbol{n}$ has been retrieved, it can be used to construct the position vector $\boldsymbol{r}$ through the equation $\boldsymbol{r} = -(\boldsymbol{n}^T\boldsymbol{n} - 1)^{-0.5}\boldsymbol{Un}$.

4.2.3.2 Implementation Details

Through the shape matrix of Equation (4-1)**Error! Reference source not found.** the Cholesky factorisation process receives in input a three-axis ellipsoid.

However, since the target is not an ellipse there is no univocal choice for the three parameters within the matrix (R_x^2, R_y^2, R_z^2). Therefore, in this case within the matching problem also lies an optimum problem: finding the set of three values which leads to the best fit.

Nevertheless, addressing this issue is out of the scope of this paper, and the same values used to define a 3-axis ellipsoid in the Path stage (Bennu's DEEVE) are used again in Equation (4-1).

Given a sequence of images, the program loops over each image, and performs a rather standard sequence of instructions to retrieve an ellipse equivalent to the horizon outline in the image. If defined in a RGB space the images are converted to grayscale. The absence of background stars and synthetically induced noise make the use pre-processing algorithms weakly relevant in this work.

The first operation performed on an image is a rotation, defined such that that the bundle of parallel rays emitted from the Sun would enter in the image from its left edge. To perform this rotation, it is necessary to retrieve the angle representing the direction from which solar illumination enters the image, let it be γ_{Im}^S. To obtain γ_{Im}^S the first step is to transform the Sun direction defined within $TCTF$, let it be $\boldsymbol{su}_{dir}^T$, to a vector defined in CRF. This is done as

$$\boldsymbol{su}_{dir}^C = \boldsymbol{R}_T^C \boldsymbol{su}_{dir}^T \qquad (4\text{-}2)$$

Simply using a rotation suffices, as in this case only the orientation of $\boldsymbol{su}_{dir}^C$ is of interest. $\boldsymbol{su}_{dir}^C$ is then a complete $3x1$ vector, defined in CRF. To transform it to image coordinates, each of its components is divided by the third component, obtaining an array of homogeneous coordinates, let it be $\boldsymbol{su}_{dH}^C$. At this point, only the ratio of the first two components of $\boldsymbol{su}_{dH}^{C}$ is needed. Therefore, these elements of $\boldsymbol{su}_{dH}^{C}$ are not pre-multiplied by the focal length of the simulated camera, albeit that would be the formally correct procedure.

Finally, γ_{Im}^{S} can be retrieved as

$$\gamma_{Im}^{S} = atan2(su_{dH_y}^{C}, su_{dH_x}^{C}) \tag{4-3}$$

Since the oblateness of the target is relatively low, and given the distances at play, it can be confidently supposed that the rotation phase does not crop off any useful region of the image. By rotating the image by $-\gamma_{Im}^{S}$ the direction of the Sun illumination within the image is aligned with the vertical axis; an additional rotation by $\pm\frac{\pi}{2}$ returns the configuration desired for the edge searching algorithm.

The ± sign is necessary due to a residual ambiguity resulting from the observation geometry. To avoid this ambiguity ever appearing in the process, a heuristic safeguard was added to the code. The safeguard acts to understand whether after the first rotation the lit region is pointing afterwards or downwards. Specifically, once the first rotation has been performed, the code halves the image in two equally sized horizontal blocks. It then searches whether the top or the bottom block has a higher cumulative intensity. The block with the highest overall intensity contains the bulk of the target. Therefore, once this intermediate orientation (up or down) is known, the second rotation can be performed accordingly.

Once the image has been rotated, the edge of the overall sunlit region of the object is detected using a Sobel edge detector [146] in the rotated image. To see an example of this configuration, with the edge extracted, see Figure 4-1. After the edge has been extracted, a search is performed a search is performed starting from the left edge, which retains only the first point encountered along that row and discards all the others. Given that the points on the left are ones facing direct solar illumination, barring singular configurations in this way only the horizon points are retrieved.

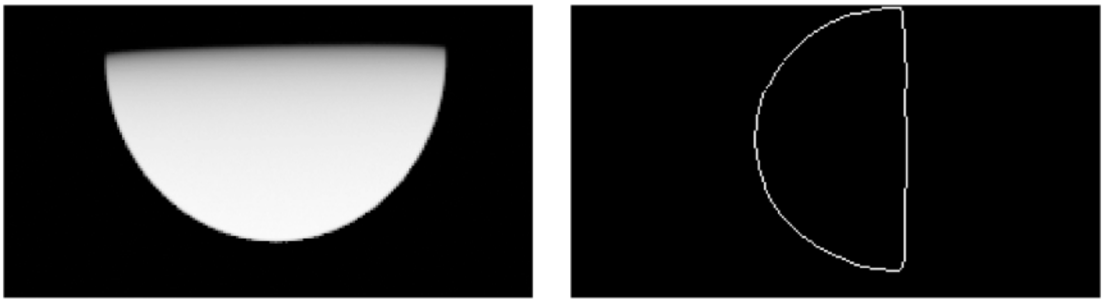


Figure 4-1 - The image of an ellipsoid and its Sobel edge in the rotated image.

Two things were observed concerning the image rotation process. The first is that different image rotation methods would generally lead to different navigation accuracies. The second is that in the ellipsoid test-case, after being rotated by some angles (even as small as 1 degree) some targets would get heavily distorted by the rotation process. Namely, the smooth edge would be divided in 'slabs' (see Figure 4-2) introducing artefacts that would translate to discontinuities in the detected edge.

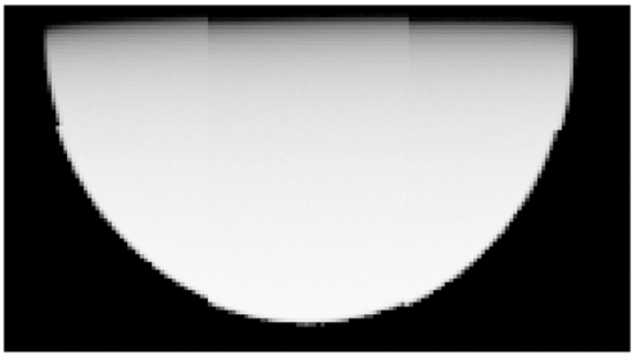


Figure 4-2 - A three-axis ellipsoid distorted by a 1° rotation

The first point would then fall on the terminator, leading to largely underestimated ellipses, and extremely large navigation errors. This required the addition of a second heuristic safeguard to remove these spurious points, which would degrade the estimation to the point of uselessness in the affected observations. This filter removes all the points where the derivative of the X value of the edge assumes a value larger than $\mu_{X_e} - 3\sigma_{X_e}$.

The minus is consistent with the structure of the observation, with the terminator always more on the right than the horizon.

The edge points are then cleaned up of the spurious points using the differential method described above, de-rotated, brought back to their original orientation, and used to fit an ellipse. The fitting is operated through the direct ellipse fit method proposed by Fitzgibbon, Pilu and Fisher [147] with the implementation provided by Chernov [148]. This method was chosen because of its tendency to underfit the ellipse. This choice might appear dangerous, as underestimating the ellipse equals to overestimating the distance, thus making the target appear further than it is. However, when specialised to Bennu's shape it felt more appropriate to have the fitted shape generally lying within the imaged target, rather than outside. It remains to be seen if a general solution to this issue can be found through an optimal choice of an equivalent ellipsoid. The ellipse fitting function returns an ellipse that is then sampled by linearly spacing the angle in a parametric ellipse representation, using M points (M = 600 in this case). This extraction choice might lead to an uneven point distribution along the perimeter and approaches to equally space the points along the curve need to be tested in the future.

These M ellipse points constitute the $\boldsymbol{s}_i^T$ in the algorithmic scheme described before. Given that the rotation matrix is supplied from the Path block, and that the shape matrix is a user-supplied information, at this point all the elements to operate the algorithm are available. It is worth noting that the noniterative method returns both the position and an analytical estimation of its covariance. To retrieve the latter, it is necessary to introduce two parameters within the code describing the optical system. These are the pixel pitch d_x, which is the distance between centers of contiguous pixels, and the σ in pixels associated to the acquisition of a point belonging to the limb. It is generally possible to observe good accordance, up to a scale factor, between the predicted covariance and the error. This, however, appears to cease to be true in eclipse conditions and, at times, in direct solar illumination conditions.

4.2.4 Filter

The filter block collects all the standalone results of the Estimation outputs and uses to reconstruct a trajectory using the three-axis ellipsoid gravity model. This orbital reconstruction process is operated through a Nonlinear Least Squares (NLS) method [127].

This method was chosen due to being well-known and well-understood. The work in the thesis capitalises on the integration of multiple approximated models, with a focus on optical navigation. Integrating exploratory methods and experiments in optical navigation within well-known workflows allows to focus primarily on the result of the processes of interest, without having to consider by-effects and interactions from more complex or less explainable methods.

4.3 Results

The results of this chapter present the implications of treating the real-world complex shape of a spin-top asteroid as an invariant element (a 3-axis ellipsoid) on the overall trajectory reconstruction framework.

4.3.1 Nonlinear Least Square performance results

This section studies of the effects of a specific observation process on the final orbit reconstruction. The Sensor block is here removed from the complete process. The Estimation block, instead, is replaced with a process corrupting the nominal position values in the state vector $\boldsymbol{x}_r^C(\, t_k)$with fixed gaussian white noise. These corrupted values represent the observables used in the NLS Filter. For this procedure, four σ levels along each axis were selected from plausible values, reported in Table 4-1. A circular orbit at 10xR$_a$ (2480 *m*) is used in this section as fixed nominal trajectory for the spacecraft.

Table 4-1 - Error levels for simulated observations

	L1	**L2**	**L3**	**L4**
σ [m]	5	30	75	115

$\mathbf{3\sigma}\ [\mathbf{m}]$	15	90	225	345
$\boldsymbol{\sigma}/\mathbf{10R_a}$	0.002	0.012	0.030	0.046

For each level acquisitions are extracted for 10 hours using three different simulated acquisition frequencies. These are set, respectively, to half an hour, one hour and two hours leading, respectively, to 21, 11 and 6 observations. These simulated observations are then used to correct through a NLS process an initial reference trajectory. The initial reference trajectory is generated by corrupting frame the initial state of the nominal orbit using values extracted from normal distributions with a standard deviation σ_{pos} for position components and σ_{vel} for velocity components.

The value of σ_{vel} has been set to be the escape velocity at the orbital distance of the trajectory considered, therefore it is the escape velocity at $10R_a$, where R_a is the average radius of Bennu, $R_a = 248\,m$. Therefore, $\sigma_{vel} = v_e^{10R_a} = 0.0628\ \mathrm{ms}^{-1}$. σ_{pos} on the other hand, is set for each error level to be equal to the error of the simulated observation divided by $\sqrt{3}$.

The correction process is carried out until the difference of the residuals between subsequent NLS iteration is less than one meter, up to a maximum of 10 iterations. The corrected initial state vectors found at the end of this process are then propagated for 10 hours, therefore $t_{end} = 10\ h$.

Let now ΔR be the norm of the difference between the estimated position state components and the nominal position state components at a given time step i, and let Δv be the equivalent parameter for velocity.

$$\Delta R_i = \left\|\boldsymbol{x}_{r_i}^{estimated} - \boldsymbol{x}_{r_i}^{nominal}\right\| \qquad \textbf{(4-4)}$$

$$\Delta v_i = \left\|\boldsymbol{x}_{v_i}^{estimated} - \boldsymbol{x}_{v_i}^{nominal}\right\| \qquad \textbf{(4-5)}$$

Table 4-2 and Table IV report, respectively the 1σ retrieved at $t_0 = 0\ h$ and $t_{end} = 10\ h$ for every combination of σ and Δt through a Monte Carlo analysis with 1000 samples per case.

Table 4-2 - 1σ error levels for corrected simulated observations at t = 0 h

$1\sigma(\Delta R^{Corr})$	**L1**	**L2**	**L3**	**L4**
$\Delta \boldsymbol{t_A}$ **= 0.5 h**	2.29 m	13.3 m	35.1 m	51.9 m
$\Delta \boldsymbol{t_B}$ **= 1 h**	3.13 m	18.5 m	45.2 m	68 m
$\Delta \boldsymbol{t_C}$ **= 2 h**	3.94 m	23.7 m	59 m	88.9 m

Table 4-3 - 1σ error levels for corrected simulated observations at t = 10 h

$1\sigma(\Delta R^{Corr})$	**L1**	**L2**	**L3**	**L4**
$\Delta \boldsymbol{t_A}$ **= 0.5 h**	2.25 m	14 m	33.8 m	51.4 m
$\Delta \boldsymbol{t_B}$ **= 1 h**	3.07 m	18.1 m	45.8 m	69.5 m
$\Delta \boldsymbol{t_C}$ **= 2 h**	3.83 m	22.9 m	58.2 m	90 m

It can be clearly observed that the values in both tables are almost identical. The values presented in Table 4-2 and Table 4-3 appear sufficient to describe the behaviour of the algorithm because it has been observed through extensive experimentation that in these conditions maxima in ΔR appear either in the initial or the final state of the trajectory. An example of error evolutions along trajectories with the four error levels of Table 4-1 are shown in Figure 4-3.

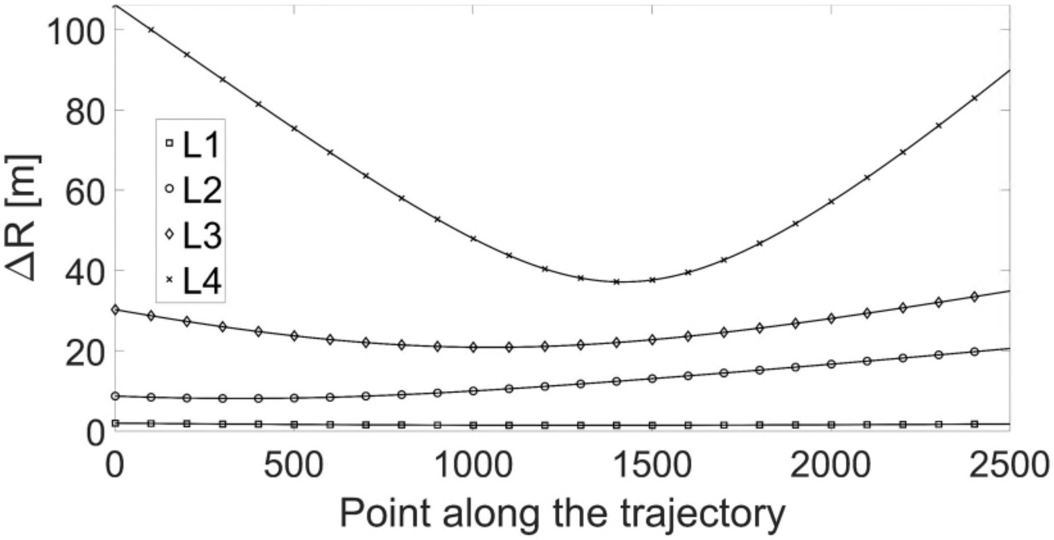


Figure 4-3 - Error evolutions along dense trajectories with the four chosen error levels

Being t_{end} itself an acquisition point, by looking at Table 4-2 and Table 4-3 it might be argued that there is only a weak advantage in adding the computational and algorithmic burden of a filter. However, it is worth noting a few things. First, there is an important correction effect on the velocity. Second, it is possible to further improve the accuracy by decreasing Δt – i.e., increasing the number of available samples. Figure 4-4 reports the 1σ at t_0 for $L4$ cases with $\Delta t = [0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1, 2, 5]$, obtained using 600 trajectories per Δt. As evident from it, this process appears to evolve as a power law. Lastly, through the NLS process it is possible to reconstruct a trajectory within the chosen gravitational model from a set of unconnected position estimations. This is as opposed to incremental algorithms like visual odometry, which require some overlap between consecutive acquisitions.

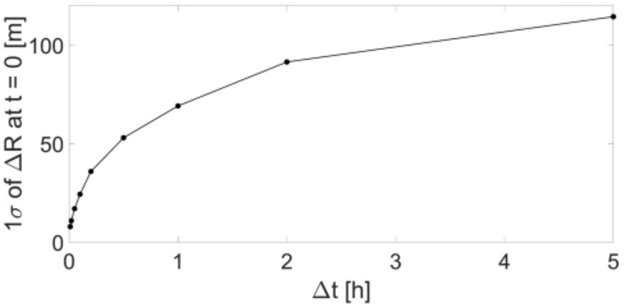


Figure 4-4 - 1σ at t = 0 for L4 cases

By defining the number of observations associated to a choice of t_0, t_{end} and Δt as N, using the same data of Figure 4-4 it is possible to obtain

$$\Delta R = \Delta R_0\,(N)^{-0.463}, \Delta R_0 \in \mathbb{R} \tag{4-6}$$

From Equation (4-6), the exponent of N is close enough to -0.5 to generally consider

$$\Delta R = \Delta R_0/\sqrt{N} \tag{4-7}$$

4.3.2 Limb-based Navigation around an ellipsoidal asteroid model

The testing iteration following the purely numerical workflow includes simulated visual data, and the complete estimation process.

This configuration differs from the one defining a complete workflow for two reasons. The first is the testing trajectory, which in section 4.3.2 is a segment aligned with the X^V axis spanning from $4R_a$ to $18R_a$. along which 501 linearly spaced observation points are defined.

The second is represented by the choice of the optical targets. Within section 4.3.2 the Sensor block is used to simulate two three-axis ellipsoids. Employing the OpNav algorithm's native target enables observing some of the properties associated to it without having to account for by-effects induced by the difference between the model and the effective target. The two targets constructed are:

1. The Dynamically Equivalent, Equal-Volume Ellipsoid (DEEVE) associated with Bennu
2. An ellipsoid with the same proportions as that employed within the Christian and Robinson reference paper [139]. Namely, this has the largest axis with an extension two times that of the smallest, and the intermediate axis spanning the average of the other two. By keeping the major axis fixed to Bennu's DEEVE this gives an object with dimension $R'_x = 259\, m$, $R'_y = 194.25$, $R'_z = 129.5$.

Figure 4-5 shows a comparison between them at a distance of $4R_a$.

The axes observed in this simulated descent are R_y and R_z. Whereas in the first case the observed ellipse has an eccentricity equal to $e = \sqrt{1 - \left(\frac{234}{251}\right)^2} = 0.36$ in the second case this is increased to $e = 0.74$.

For each of the 501 poses observed along the trajectory the algorithm generates an image in Blender through a centre-pointing camera having the same properties as that described in section 4.2.2.

The results shown below are obtained by decomposing in its axial components the estimated VRF position and comparing them to the nominal along-axes values. In particular, given an axis $x_i, i = 1, 2, 3$, the error along the x_i axis is defined as

$$\Delta x_i = x_i^{estimated} - x_i^{nominal} \qquad \textbf{(4-8)}$$

Figure 4-6 shows the axis-by-axis the difference between the nominal and the reconstructed position in VRF at each observation position for x, y, and z for both targets. All the plotted quantities are expressed as a function of the DEEVE's average radius ($R_a = 248\, m$).

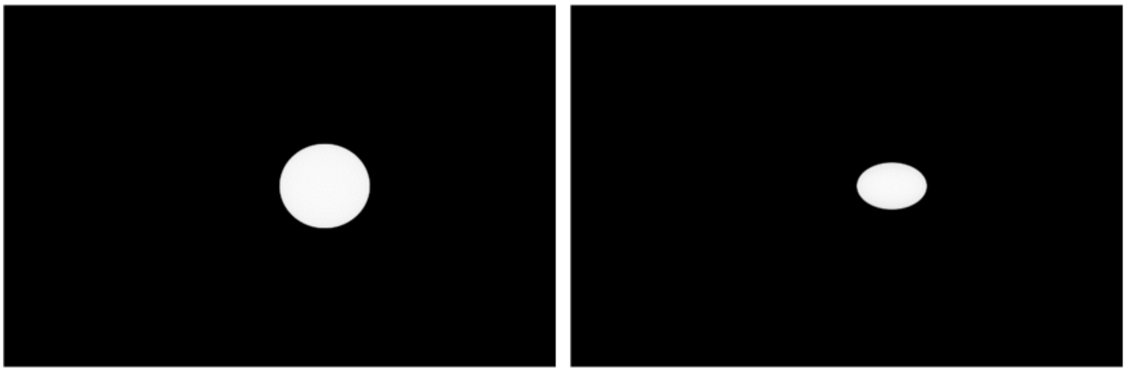
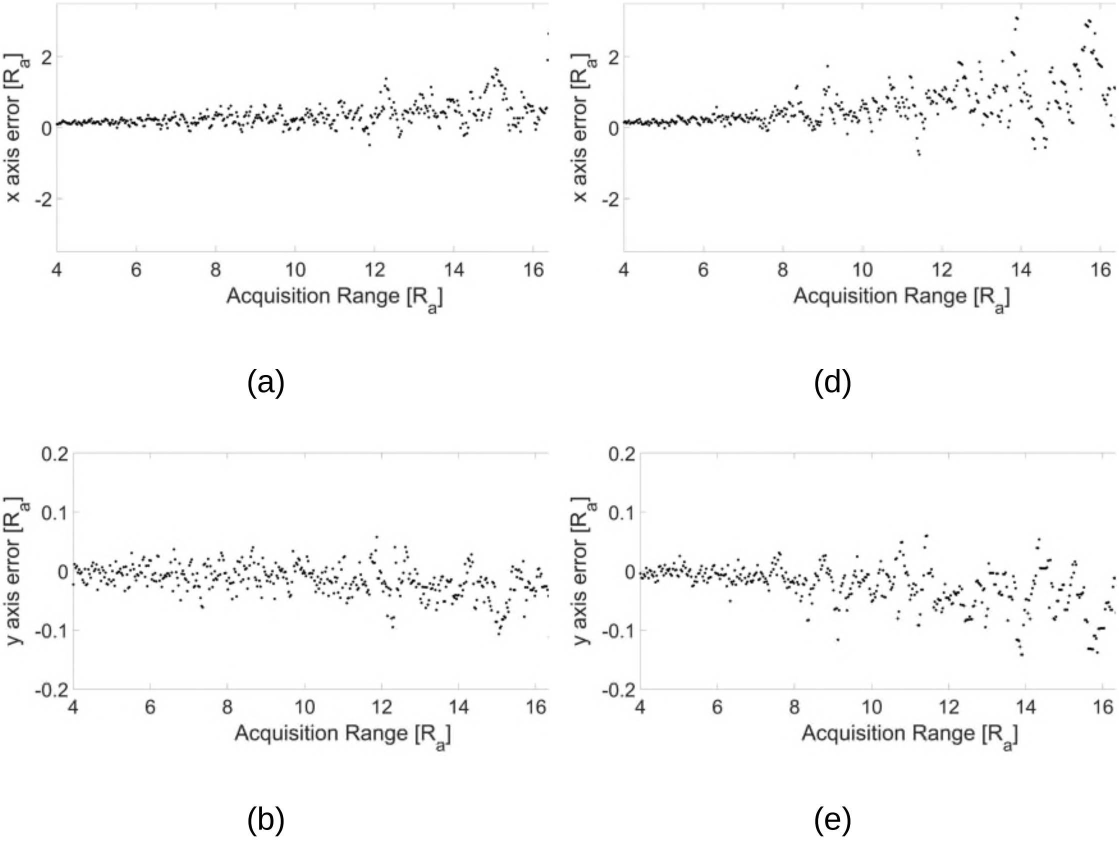


Figure 4-5 - The two target ellipsoids described in section 5.2. On the left the DEEVE, on the right the reference-like.

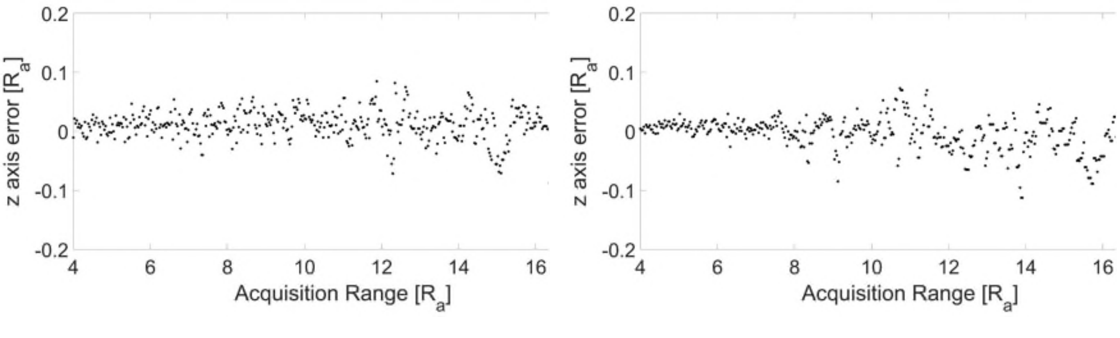


(a)

(d)

(b)

(e)



(c) (f)

Figure 4-6 - Error along each axis for the two test cases studied

It is possible to observe that the error appears to be predominantly distributed along the X^V axis, which is the along-boresight axis. This is explained with the main source of error being an erroneous scale of the fitted ellipse.

Figure 4-6(a) and Figure 4-6(d) show that the x axis error appears to be directly correlated to the range and biased towards slightly positive values, i.e., generally $\Delta x > 0$. This implies, from Equation 4-8, that $x^{estimated} > x^{nominal}$ – i.e., that the observed position tends to be larger than the nominal one. This is consistent with the behaviour of the ellipse-fitting algorithm, which is prone to underfitting (see section 4.2.3.2). An underfitted ellipse has a smaller scale than the one effectively present in the image. Under the same acquisition conditions, a smaller ellipse implies a larger orbital radius, and therefore a positive along-boresight error component, consistently with the shown data.

The seemingly larger error with respect to those observed by Christian and Robinson can be traced back to a few factors. Among these, worth mentioning are: potentially different simulated camera systems; difference in target's albedo, texture, structure and relative attitude; lack of a dedicated limb extraction method [149] and effects of the rotation algorithms, in particular at a critical Sun phase angle.

It is worth spending a final word of caution regarding the employed image rotation method. From the tests performed on the 'Bicubic' and 'Nearest' built-in MATLAB methods, Bicubic appears to be the best choice for Bennu, while Nearest works better for ellipsoids. This constitutes experimental work, which will have to be further refined in future work.

4.3.3 Observation Process

Due to the overall trajectory reconstruction process largely relying on the performance of the optical navigation algorithm, it is critical to possess a thorough understanding of the response of optical navigation to different interactions of camera pose, target, and illumination. This is done to address conditions inducing effects that could disrupt or corrupt the process. By selecting a reasonably low amount of acquisition points per trajectory it is easy to detect conditions that put a strain on the process.

For example, doing so in a trajectory with only three observation points, see **Figure 4-7**Figure 4-7, highlights three different estimation behaviours. Of these, only the central one (labelled 2), is acceptable, while the other two present non negligible errors. These are induced by the coupling of different factors.

The code fits an ellipse to the extracted set of points, employing the chosen model. However, since the shape from which the points are extracted is not a three-axis ellipsoid, the strength of this approximation depends on the relative pose of the camera. In the leftmost acquisition (labelled 3, see Figure 4-8 for a close-up), the observable points are asymmetrical with respect to the equatorial plane of the asteroid.

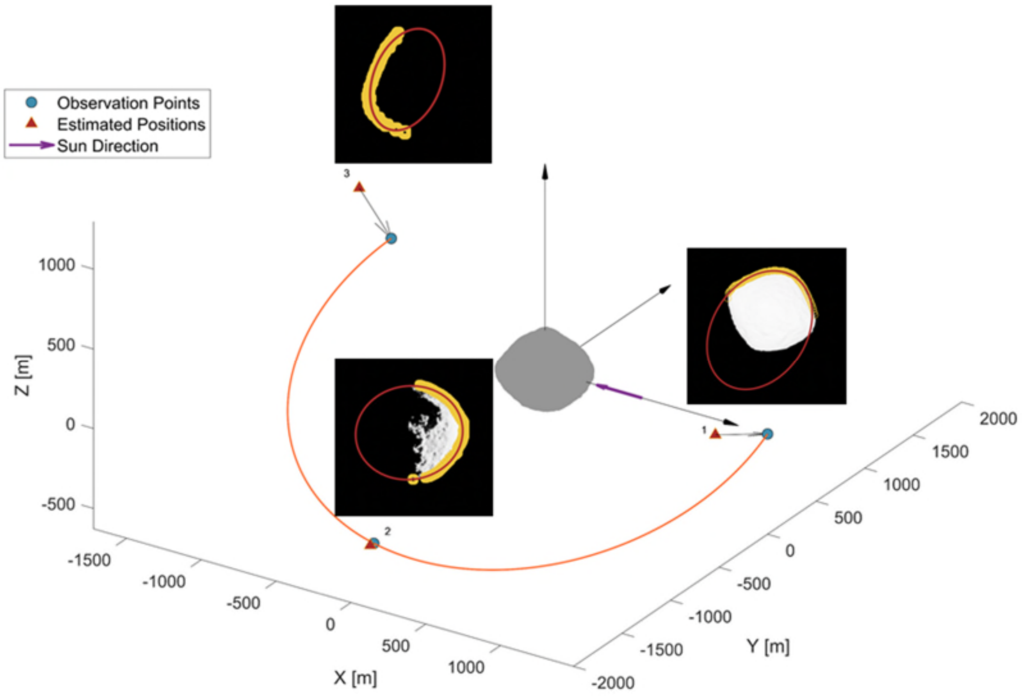


Figure 4-7 - A test trajectory with three observation points

This leads to a longer segment and a shorter segment stemming from the central angle. Since the two are not balanced, the ellipse-fitting process assumes them to be, respectively, a distribution along a major axis and a distribution along a minor axis of the ellipse. The ellipse fitted to these points strongly matches these extensions and orientations and is therefore much smaller than the target's dimensions in the image (underestimated ellipse). This underestimation leads the target position to be estimated further than its real one (overestimated distance). The opposite happens in the rightmost acquisition (labelled 1, see Figure 4-9 for a close-up) which represents an underestimated position estimate caused by an overestimated ellipse. Namely, being the estimated ellipsoid much larger than the target object in the image the estimation algorithm suggests a position erroneously closer to the target than the actual one.

Whereas it might appear that this is caused by the points covering almost three ‘sides’ (there are a couple of points crawling above the leftmost corner), this effect is most likely induced by the sharp almost rhomboidal appearance of the target for images acquired in proximity of the equatorial plane.

A testing approach with such a small number of images is however not optimal for a few reasons. It does not tell anything about the permanence of the behaviour in the regions surrounding the critical spots, it mostly relies on luck to find anomalous configurations and it is rather time consuming. A better exploratory method appears to be foregoing the heuristics of individual acquisitions to focus on the group behaviours.

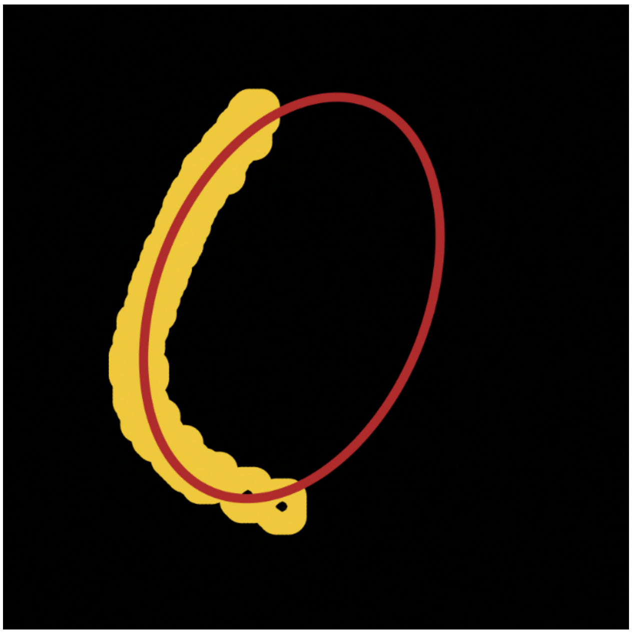


Figure 4-8 - A close-up of the acquisition labelled 3

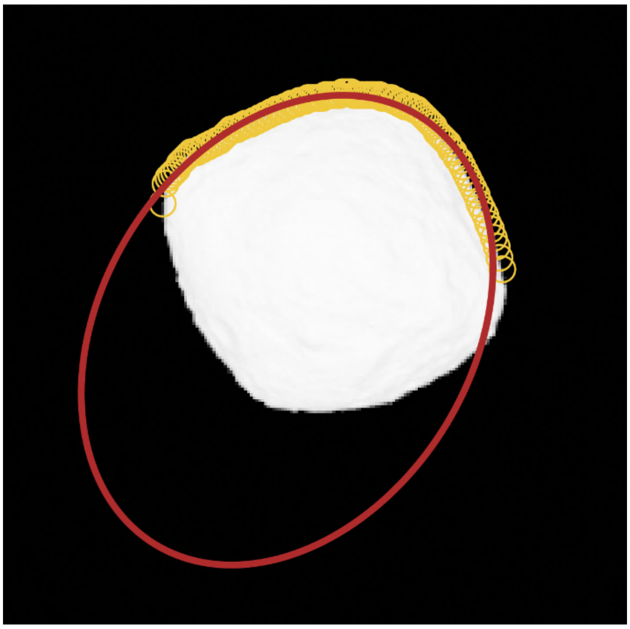


Figure 4-9 - A close-up of the acquisition labelled 1

Since the camera orientation law and the camera intrinsic matrix are fixed, with respect to camera factors the output data are a function of its position alone. Therefore, to obtain knowledge on the spatial response of the process it is possible to construct acquisition surfaces and map the response of the algorithm along the surfaces. For now, all environmental properties (e.g., illumination, target's attitude, etc.) are maintained constant given an acquisition surface. Therefore, once all parameters have been selected, the test case is uniquely defined.

Using this test approach, 5 test cases for the estimation process were defined labelled ETC (Estimation Test Cases). To study ETCs it was necessary to build spherical meshes of acquisition points around the target at a radius R with an azimuthal angular spacing of Δ_{Az} and an angular spacing in elevation of Δ_{El}. Five factors were taken into account when developing these test cases. These are R, Δ_{Az}, Δ_{El}, the target object and the direction of the Sun, ϑ. This last factor is important because Bennu's complex geometry leads to complex interactions between the shape and the lighting, thus inducing different navigation behaviours under different illumination conditions. Ultimately, however, these are reduced to four by choosing $\Delta_{Az} \equiv \Delta_{El} = \Lambda$. The five ETCs were defined as shown in Table 4-4.

Table 4-4 - Observation Test Cases

	R	$\Delta_{Az} \equiv \Delta_{El}$	$\boldsymbol{Target}$	$\boldsymbol{\vartheta}$
ETC1	$5R_a$	3°	Ellipsoid	0°
ETC2	$5R_a$	3°	Bennu	0°
ETC3	$5R_a$	3°	Bennu	120°
ETC4	$10R_a$	3°	Bennu	0°
ETC5	$10R_a$	10°	Bennu	0°

Comparing ETC1 and ETC2 allows observing the effect of the incoming Sun direction on the results. Comparing ETC1 and ETC3 allows to understand the effect of the shape of the target on the results. Comparing ETC1 and ETC4 allows observing the effect of distance on the results. Lastly, ETC4 and ETC5 are used to understand the information loss induced by downsampling Λ. Overall, observing the results returned by ETCs (Figure 4-10) allows to obtain a good grasp of the performance of the algorithm.

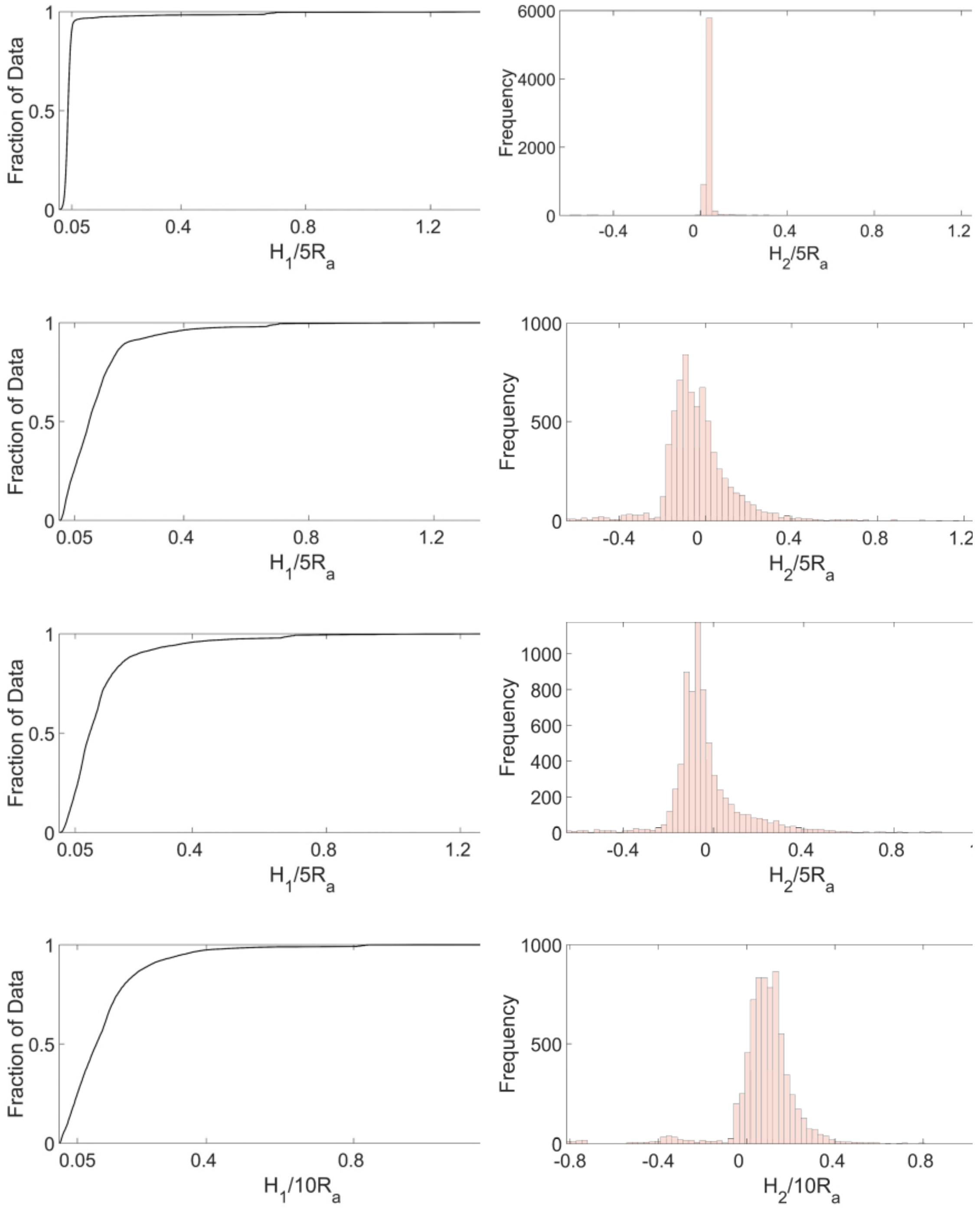
The first results presented in relation to ETCs are the distributions of two scalar values describing deviations of the estimated position from the nominal position. These two values, H1 and H2, are used to discover whether there is any evident trend or bias within the data. For the i-th acquisition, let us label the nominal position as $\boldsymbol{x}_{r_i}^{nominal}$ and the associated estimated position as $\boldsymbol{x}_{r_i}^{estimated}$, and let $||\boldsymbol{\tau}||$ be the norm of a generic vector $\boldsymbol{\tau}$. The first of the values is the norm of the differences between the estimated and the nominal position vector

$$H1 = \left\|\boldsymbol{x}_{r_i}^{estimated} - \boldsymbol{x}_{r_i}^{nominal}\right\| = \Delta R \tag{4-9}$$

While the second value represents the differences of the norms of the estimated and the nominal position vector

$$H2 = \left\| \boldsymbol{x}_{r_i}^{estimated} \right\| - \left\| \boldsymbol{x}_{r_i}^{nominal} \right\| \quad \textbf{(4-10)}$$

Below, in Figure 4-10, the histograms of $H2$ and the empirical cumulative distribution function (CDF) of $H1$ are shown for each ETC.



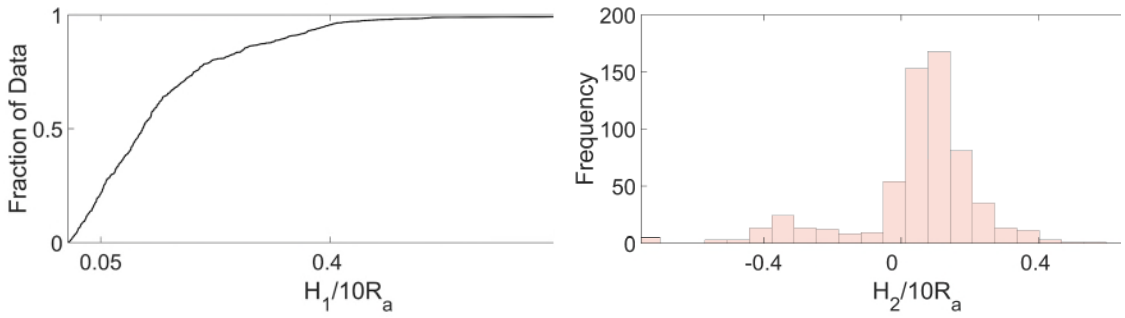


Figure 4-10 - On the left side, in descending order, the empirical CDFs of H1 for ETC1, ETC2, ETC3, ETC4 and ETC5. On the right side the histograms of the corresponding H2 values.

4.3.4 Navigation Charts

The error described in the previous section (4.3.3), has a strong spatial component, which is difficult to characterise employing only scalar values. This is addressed by adding another dimension to the graphs and localising the error values in a 2d maps presenting $\Delta R = f(Az, El)$, thus enabling the analysis of regional behaviours of the algorithm in the previous ETCs. These visualisation tools are labelled ‘Navigation Charts’ and map the response of the estimation algorithm as a function of position. Navigation Charts allow the identification of spatial regions critical for operational conditions, and the design of orbits minimising the localisation uncertainty for autonomous probes. Given the large number of points within these acquisition surfaces (more than 7000 for $\Lambda = 3°$), the distribution of error values can effectively be considered close to a continuum. For future developments it could therefore be conceivable to use fuzzy logic [150] or continuous empirical laws over the retrieved spectrum to select optimal trajectories. These could assign an operational approach chosen from a finite number of pre-defined ones using meta-information about the operational state. Additionally, Navigation Charts could be used to construct Boolean classification tools of the type Go/No Go or Acquisition/No Acquisition. Figure 4-11 shows the plot of H1(Az, El), i.e., the navigation chart for H1 for all the five ETCs analysed.

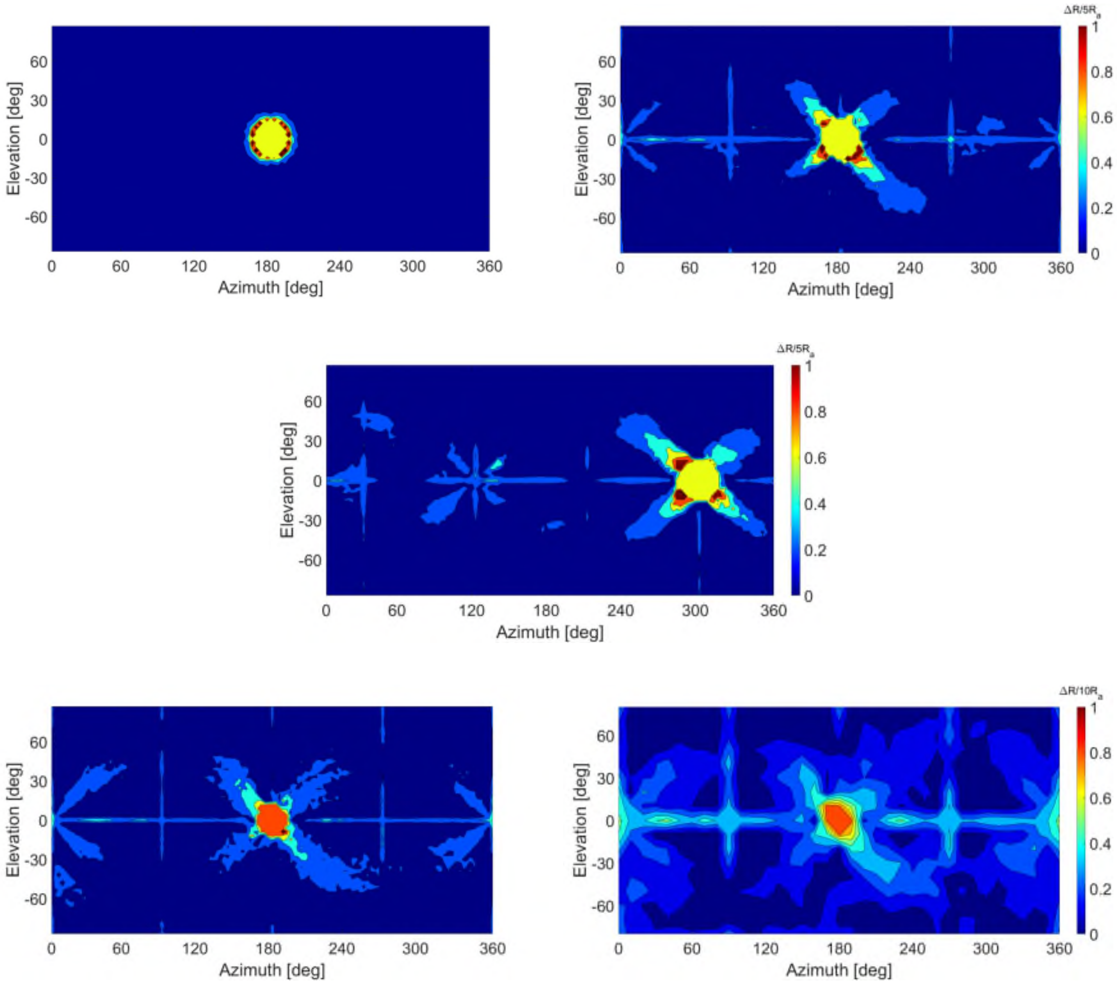


Figure 4-11 - The navigation charts mapping H1 for all the five ETCs. From left to right, the upper row contains ETC1 and ETC2, the middle row ETC3, and the last row ETC4 and ETC5.

From these charts, it is possible to observe the emergence of zones of invariant response. These zones can be included within a general navigation algorithm and provide contextual information regardless of the target's configuration. For regions in eclipse, where visible cameras struggle to operate it is possible to assume that the reconstructed trajectories are sufficient to provide triggers to stop and restart acquisitions. Some of these critical regions appear to be specific to Bennu, while some others appear to be more general and induced by illumination.

4.3.5 Trajectory Reconstruction

This section reports the importance of invariant conditions in the perception model when these are propagated within the full trajectory reconstruction framework.

As expected, for Gaussian errors in the estimation process an increase in the number of observations leads to a reduction in the uncertainty of the reconstructed trajectory (Equation 20). However, section 4.3.3 and section 4.3.4 have shown that the navigation error around a non-ellipsoidal target is a strong function of the position of the probe. The error is a function of position because the appearance of the target within the camera output determines how closely the target resembles an ellipse in the given acquisition. The appearance of the target within the sensor, in turn, is a function of the relative pose of the Sun, the target and the sensor itself. This position-connected bias translates to a non-Gaussian distribution of errors (Figure 12).

To test whether the same scaling mechanism of Equation 20 emerges from visual data acquired around a simulated Bennu, a test trajectory with circular initial conditions is defined at $r = 10R_a$. Within this trajectory an observation arc is defined using, $t_0 = 0$ and $t_{end} = T/5$, where T is the orbital period at R around an object with the same standard gravitational parameter μ of Bennu. Ultimately $T = 95.8\,h$.

This arc is used to generate Monte Carlo experiments both with $\Delta t = 1\,h$ and $\Delta t = 15\,s$ ($n = 600$ in both cases). Once again, the experimental process consists in using a fixed set of observations to correct an initial prior orbital determination. In this case, this prior is randomly generated by corrupting the position components of the initial state vector with $\sigma_{pos} = 115/\sqrt{3}\,m$ and the velocity components with $\sigma_{vel} = 0.0628/\sqrt{3}\ \mathrm{ms}^{-1}$. Figure 4-13 and Figure 4-12 report some outcomes of these tests allowing to discuss three noteworthy results.

The first, shown in Figure 4-12 is that when the results do not diverge to an escape trajectory, they converge to a level of at least 2σ to the same estimated trajectory. A condition for this convergence is that the estimations used do not pose conflicting constraints within the gravitational model.

This allows to present Figure 4-13 as a meaningful result. Indeed, Figure 4-13 shows ΔR as a percentage of the orbital radius within a reconstructed trajectory. For this trajectory a random initial noisy state is corrected always using the same set of observations. This correction converges with a probability slightly higher than 2σ to the same orbit, as supported by Figure 4-12.

In particular, Figure 4-13 shows point by point the difference between the reconstructed trajectory and the nominal one. The second result of this section shown in this figure, is that through the reconstruction pipeline it is possible to obtain uncertainties as low as roughly 1% of the orbiting radius, and that it is possible to do so with a low acquisition frequency – in line with resource constraints. The third result is that, albeit slightly in this case, increasing the number of observations increases the uncertainty of the reconstructed trajectory. This third conclusion, which might appear counterintuitive, emerges because increasing the number of acquisition points does not introduce estimations with Gaussian errors, but, rather, errors with a spatial bias varying with continuity (Figure 4-11). Hence whereas the error can remain generally low in favourable observation geometries, increasing the number of observed points can lead to moving through regions of higher uncertainty, and therefore worse reconstruction accuracies.

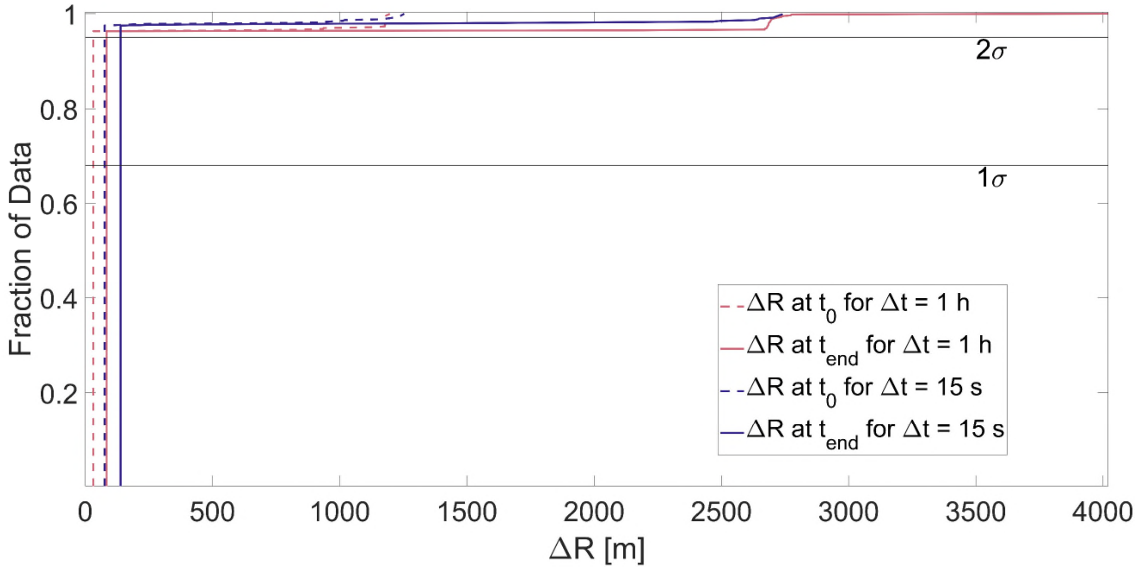


Figure 4-12 – Empirical Cumulative Distribution Function of Monte Carlo results (n = 600) of ΔR for Δt = 1 h and Δt = 15 s in the initial and final acquisition point (i = 45°, E = -30°, Ω = 90°).

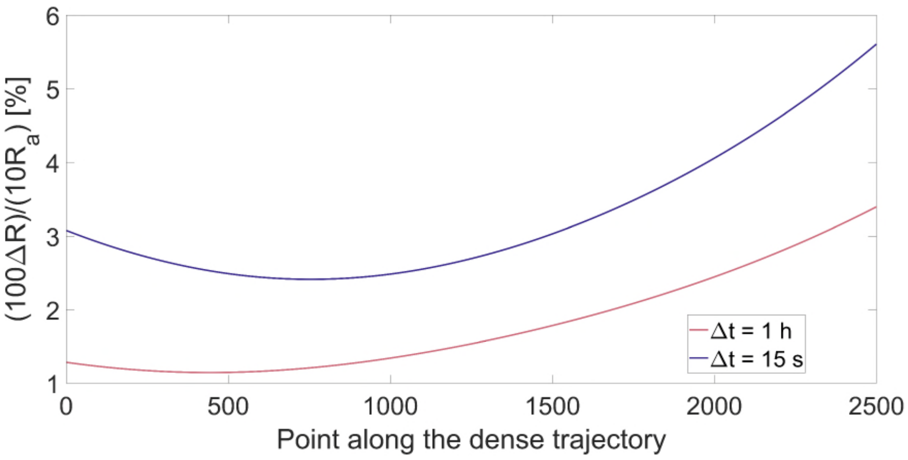


Figure 4-13 - ΔR along a dense trajectory for Δt = 1 h and Δt = 15 s

Concerning the spatial bias, from Figure 4-11 it is possible to distinguish the presence of three main critical regions for this navigation approach around Bennu. These are, respectively:

1. Eclipse conditions
2. Direct illumination conditions
3. Planes generated by two of the three largest axes of the target (in this case aligned with X, Y, Z).

Eclipse conditions are induced purely by illumination-related effects, and as a result, this region appears regardless of the shape of the target (Figure 4-11) and can be considered purely invariant. The effects in direct illumination conditions are induced by a coupling between the shape of the target and the direction of illumination, and generally appear only along specific inclinations. Lastly, critical planes are induced by the shape of the target alone. Indeed, when looking at the target from one of the planes containing its major axes, the target's shape appears sharp-edged, again leading to overestimated ellipses and thus underestimated position estimates (Figure 4-14Figure 4-14).

Experiments have shown that pruning acquisitions performed within these regions from the inputs of the reconstruction process generally improves its performance through two mechanisms, one direct and one indirect. The first of these is simply connected to a reduction of the 1σ level of error, induced by the pruning of bad data. As shown in Table 4-2 and Table 4-3 a lower average error reduces the error level of the overall reconstruction. The second, on the other hand, is connected to the implementation of the covariance algorithm found within. The returned covariance matrices generally capture extremely well the uncertainty of the points associated to them. However, in critical regions some of the exceedingly bad acquisitions have been observed to sometimes be associated to almost null covariances. This is for example the case for observations with scarce visible target points. When introduced in the NLS process, the small covariance is interpreted by the algorithm as an exceptionally good estimation, leading the algorithm to over-trust these points which end up acting as attractors for the orbit.

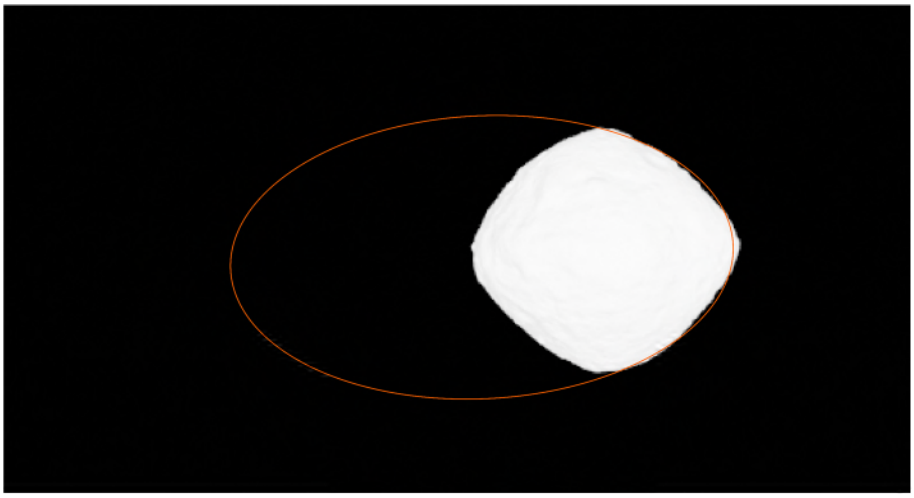


Figure 4-14 - An Equatorial acquisition and the associated estimated ellipse

Given that the Sun phase angle is assumed to be known at every acquisition, a first heuristic filter to prune data acquired within regions of the first and second type can simply be constructed by exploiting information about φ. Figure 4-15 shows $\Delta R(\varphi)$ for a generic trajectory. The acquisitions in direct Sun conditions performs discretely well in this case. However, the conditions in proximity of eclipse in Figure 4-15 are consistently critical and tend and converge towards a single error value for a range of φ. This implies that for a certain interval of values of φ the process becomes detached from the data and the results can be considered systematic effects.

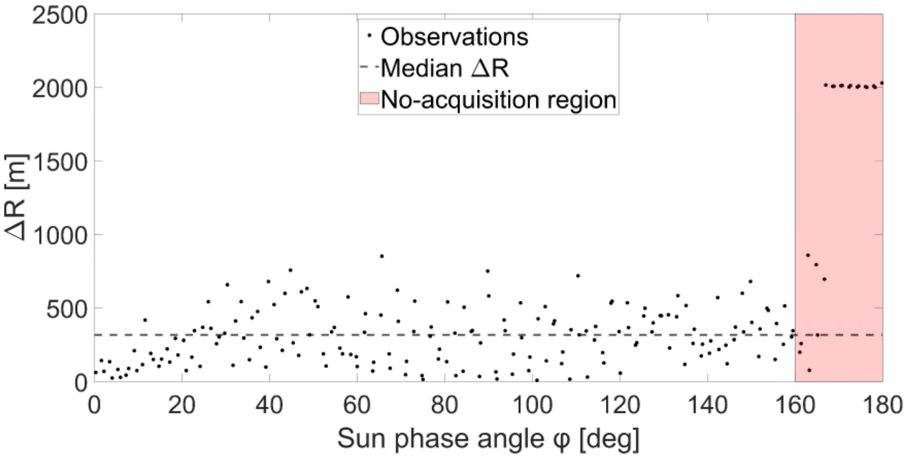


Figure 4-15 - ΔR(φ) for a complete circular orbit with i = 75°; Δt = 0.5 h.

However, being these critical points clustered as a function of φ, a first solution is to discard all acquisitions performed in $\varphi > 180° - \lambda_1, \lambda_1 \in \mathbb{R}$. In the figure, the region highlighted as critical is such that $\lambda_1 = 20°$.

An additionally conservative version of this filter could require the pruning of data with φ such that $\varphi < \lambda_2$, possibly also depending on the inclination value. Implementing a pruning mechanism for the third critical region is out of the scope of this work. This is because operating on the third critical region would either require prior information on position, making the task not invariant-oriented, or a way to discern the closeness of the target in the scene to an ellipse, which is considered future work.

As for the reconstruction process, all the factors concerning critical acquisition regions could be managed by factors determined through the camera, the ephemerides of the target, and local asymmetries and features of the target object. This would lead to a fully camera-only trajectory reconstruction framework, both in the data processing and in the data acquisition strategy. This camera-only algorithm would operate at ranges where scaling can be supplied through three parameters: the axes of an ellipsoid.

Obtaining scaling from such a restricted number of parameters has important implications on feature-based monocular OpNav following the whole-target phase. Specifically, through trajectories defined in an asteroid-fixed, asteroid-centered reference frame it is possible to scale monocular approaches and place them in a global reference frame. This information on scale and global position can simply be injected through propagation of the reconstructed trajectory. Moreover, provided that the accuracy of the reconstruction is sufficiently high, the reconstructed trajectory can be used to estimate critical conditions such as impact and set range and velocity triggers.

Ultimately, whole-target body represent a fundamental technology to develop full camera-only navigation for landing probes targeting NEAs.

5 FEATURE BASED PERCEPTION AND NAVIGATION

5.1 Introduction and Scope

The previous chapter discussed an approach to optical navigation exploiting the edge illuminated by sunlight of the target NEO. The invariance exploited in chapter 4 builds upon the reduction of complex geometries to a simple shape. Specifically, it is assumed that a certain set of shapes can be approximated with a small error to a three-axis ellipsoid, at least in some regions around the body. Thus, a single navigation method operating on three-axis ellipsoids can be developed. However, this navigation process has a critical intrinsic limitation, which constrains the process to have a minimum operational altitude. There is indeed an altitude from which the limb extension in pixels becomes so large that it is difficult to maintain it consistently and fully within the FoV of the camera. This limit distance is determined by the size of the celestial body and by the parameters of the spacecraft camera. Once again, owing to the broad variety in possible landing conditions it is not possible to provide a priori a precise numerical value for this distance. A coarse indicator of this altitude, however, could be retrieved using the ground sample distance (GSD). GSD represents the metric value of an object having the same extent of the pixel pitch d_x, the distance between the centre of a pair of adjacent pixels. Despite being a theoretical, ideal, quantity GSD can undoubtedly be significant in preliminary design. Assuming perfectly square pixels it is possible to write that

$$GSD = \frac{d_x}{f} h \qquad (5\text{-}1)$$

Where f represents the focal length and h the altitude of the camera, which can be confounded with that of the spacecraft. A threshold altitude can be defined as the point where a reference length of the body divided by the GSD returns values close to a characteristic size of the focal plane. When h_{lim} is reached, it is thus necessary to switch to a different navigation mode, employing observables that can still regularly be seen within the scene. A plausible navigation method in this context is feature-based navigation.

In feature-based navigation, the pose estimation process exploits opportunistic patterns detected within visual data. To return an estimation of the motion of the platform, it is generally necessary to track these patterns, appropriately identified and labelled, across multiple views. Since visibility issues for the limb arise as the probe gets close to the surface, it is possible to assume that in proximity of the limit altitude objects and patterns on the ground become resolvable within the sensor. Moreover, it is also reasonable to assume that their extent in pixel is non-negligible. If the characteristic timescales of motion (Action in Chapter 3) and perception are similar, and much faster than that of the target's motion (Scenario in Chapter 3) it is also possible to assume that the appearance of these visual patterns is stable enough to identify them across subsequent views.

Therefore, adding feature-based capabilities to a modular navigation package enables model-free optical navigation at close range. Although feature-based methods extend the operational range, they are still vulnerable to adverse effects generating uncertainties. For example, the environment might be devoid of useful features, or the patterns available might not be optimal for the feature detector implemented in the design phase. Moreover, algorithms exploiting monocular cameras tend to return an estimation up to scale. To introduce a scale factor to this trajectory it could be possible to propagate prior information about invariants occupying the environment (as will be the case for NAV-Landmarks), or from the limb-based navigation stage, which returns a scaled trajectory.

This chapter will address some of these issues by introducing and studying three non-computational optimisers that can increase the robustness of feature-based processes. Similarly to chapter 4, the goal is to obtain workflows that are robust to external variability and perturbations without introducing heavy computational overheads. And again, similarly to chapter 4, the development of these ancillary tools is connected to the construction of invariants.

5.2 Feature-Based Non-Computational Optimisers

This section introduces three invariant-based concepts that were studied to improve the robustness and stability of perception processes without the need for additional computational tools.

5.2.1 NAV-Landmarks

The scope of this subsection is to present an approach enabling robust camera-only navigation in small satellites deployed by a mothership. The approach is based on introducing standardised Non-cooperative Artificial Visual landmarks (NAV-Landmarks) on the surface of the target NEO as a mean to augment the capabilities of small satellites. These external elements can be effectively regarded as an infrastructure forming an extension of the navigation system. They can be assumed to be deployed by the mothership with the same mechanism used for the small satellite, and then able to expand on the surface. NAV-Landmark would be deployed with two principal scopes: provide robust, stable observation targets even in feature-starved environments, and act as a mean to recover scale information at close range. Their standardisation allows treating them as invariant and to tailor visual navigation strategies over them.

5.2.1.1 Introduction

Let assume for now the resource constrained platform to be a CubeSat. CubeSats are small satellites designed complying with a particular standard, defined by Bob Twiggs and Jordi Puig-Suari in 1999 [151]. The design prescription requires this spacecraft to be a small collection of basic “Units” (1U), defined as cubes with sides measuring $10.0\ cm$ and a mass smaller than $3.00\ lbs$ ($1.33\ kg$). Typical architectures are 1U, 2U, 3U, 6U and 12U. This restricted volume limits the number of resources available on board, substantially constraining the level of autonomy achievable by these small satellites. This shortcoming might not be a problem when orbiting in proximity to the Earth, since human controllers or other forms of navigational support might compensate for it. However, this issue becomes critical for deep space missions.

While it might be difficult for a CubeSat to carry out a complex mission by itself, a winning paradigm to employ this spacecraft might lie in integrating their capabilities with either other satellites or some form of infrastructure. This could mean either extending the capabilities of CubeSats through cooperation or interaction with other entities, or further augmenting more complex probes through small satellites employed as external subsystems. An example of the latter is represented the two 6U orbital modules making up the MarCO mission [152]. These two, MarCO-A and MarCO-B, were used as an additional detached communication subsystem for the InSight lander, acting as beyond-LOS orbital relays during its entry, descent and landing phase.

Ultimately, this section explores this idea of CubeSats operating as part of a hybrid distributed system, encompassing the landing probe and a set of passive beacons serving as navigational aids. In particular, the hypothesis under test in this section is that the addition of artificial features to the surface of the landing target enhances the accuracy of optical estimations.

The effectiveness of the deployable markers must be objectively proved and quantified to justify their non-negligible impact on the critical mass budget. This work will present an analysis of their effect on optical navigation, focusing on deployments on NEOs. This because the reduced size of these celestial bodies allows for the implementation of a dispersion configuration covering the complete target.

A whole-target coverage approach can be regarded as an inverse-GNSS architecture, with an orbiting receiver and a constellation-like infrastructure on the ground. The scenario studied contemplates the ballistic deployment of a number n of small markers by a mothercraft, implemented before the deployment of a CubeSat lander. It is assumed that the markers are foldable devices thus having a limited impact on the mass and volume budget while being transported.

Using artificial markers (also called fiducial markers, or fiducials) as an aid while performing visually controlled tasks is a rather common strategy for a broad range of operations, either crewed or autonomous. Indeed, typical applications may span from on-orbit proximity operations [153] to UAVs landing [154], from navigation of terrestrial robots [155] to underwater applications [156]. The function of fiducials is to provide consistent, standardised information and cues, used to augment the environment or a target into a cooperative form. A cooperative entity provides to the observing payload operative information that need little or no processing to be used.

The presence of fiducials implies that the operative methodology can be calibrated on mainly exploiting these elements, known a priori. Fiducial based approach, however, have been largely neglected by the space scholars and engineers for applications beyond Earth's orbit. There, coherently with the rest of the computer vision community, the research trends appear to be oriented towards highly centralised, marker-less solutions (e.g. the ALHAT package [157]) or towards high environmental specialisation (e.g. a crater oriented solution [158]). The constraints over volumetric and mass budgets are rarely as severe as in the space domain and transporting markers could significantly impact them.

Ultimately, three factors are expected to influence the realisation of fiducial-based schemes. These are the mass-to-intelligence costs ratio, the trends concerning space exploration strategies, and the development of expandable structures [159,160].

5.2.1.2 State of the Art

A key to far-reaching, cost-effective space operations could lie in warehousing [161]. To establish a warehouse means to structure a repository of modular solutions that can easily be integrated and deployed. This work investigates a paradigm shift in warehousing, where the repository contains a set of infrastructures used to structure the operational environments as default spaces. In this initial iteration, this architecture is realised through reference guideposts scattered on the target surface of interest: these are the NAV-Landmarks found in the title.

These features can be deployed at any stage, meaning that the operative methodology can be calibrated on operating using principally these elements, known a priori.

Despite the constraints over mass and volume budgets, there have been two instances in the last few years where fiducial landmarks have been employed in a space mission. Namely, variants of this technology have already been used on the JAXA probe Hayabusa [162] and its successor Hayabusa2 [45]. Both these probes carried on board, respectively, three and five highly reflective orbs, to enable close-range navigation for a high accuracy (pinpoint) autonomous landing. For additional robustness, both probes were equipped with a flashlight (FLA instrument) to highlight the markers on the surface. The programmed markers release altitude had been set to roughly 30 meters in both cases. However, the deployment was realized at an altitude of 40 meters for Hayabusa [163] and reported to happen at roughly 10 meters from the ground in Hayabusa2 sources [164]. The features injection strategy envisions them deployed not all at once, but iteratively, with the already deployed markers used to increase the deployment accuracy of the subsequent ones, ultimately forming a pseudo-cluster around the desired landing area. The augmentation provided by these objects can then be integrated in a wide array of close-range Guidance and Navigation strategies [163,165].

5.2.1.3 Landmark Design

This infrastructure approach is believed to be particularly relevant for CubeSats due to two main reasons. The first reason is that non-algorithmic optimizers would have a much more profound impact on low-budget objects, capable of reaching the desired accuracy only through assistive technology. The second reason is that a deploying mechanism for CubeSats could be reused for features, if their sizes have a comparable order of magnitude. This second argument implies that the technological complexity of the mission is not excessively increased.

An important role of the deployment process is ensuring that the landmarks touch down on the surface of the target with a speed less than a critical rebound speed.

The design process of the landmarks is mainly driven by mediating between two competing constraints:

1. Stability and reliability in optical acquisitions
2. Budget constraints

The former is a factor of merit in improving navigational performance and pushes for larger features size; the latter considers the additional load on the mass and the volume budget by the features and calls for smaller dimensions. Within the scope of this work the presence of surface patterns on the sides of the landmarks will be neglected.

The push for reliability is driven by the need for consistent information for navigation. It is possible to subdivide this condition into several secondary requirements. In particular:

- **Restricted symmetry**: the landmarks should be built with a limited number of axes of symmetry. This condition is important to avoid ambiguity: a textureless sphere hardly provides any directional information. However, a certain degree of symmetry is desirable, to passively reduce the risk of landing configurations of an asymmetrical object which might be sub-optimal or ineffective for estimation. Nevertheless, keeping the volume of an object constant, as the degree of symmetry of the object increases, the length of its sides (features in themselves) generally decreases, generating ambiguity
- **High visibility**: this driver concerns the capability of the landmark to be distinguished from its surroundings within visual data. This characteristic can be achieved both in active and passive ways. The former requires mechanisms capable of generating light or changing the colour or shape of the landmark in reaction to environmental effects reducing visibility. Passive methods are achieved through robust design. High visibility does not necessarily equal high reflectivity: for example, on a pale-tinted asteroid a VANTAblack [166] feature might be the best solution

- **Stability to displacing perturbations**: to be effective as a navigational feature, the landmarks must move as little as possible with respect to the surface when perturbed. Important parameters controlling this requirement are the contact surface at a given instant and the distance between the center of mass and the edge of said surface
- **Convexity**: the NAV-Landmarks should have convex shapes. These geometries both reduce the amount of dark, shadowed, zones and avoid the accumulation of surface material which could “camouflage” the landmark
- **Shape Retention**: features should not be affected by unpredictable changes in their appearance upon impact since this would lead template matching or scale reconstruction algorithms to failure.

Budget constraints require more advanced solutions to be properly addressed: for example, the expandable structures mentioned in beforehand. It is however out of the scope of this work to discuss them.

Aside from having been used at a different range and within a different architecture, Hayabusas’ target markers deviate from the above conditions because they do not seem to possess easily observable sides or features. Therefore, while at their operational distances they could hypothetically be employed for optical scale reconstruction, their gibbous globe-like form might generate ambiguity in this process. Cubes, on the other hand, satisfy all the above-mentioned constraints. Being Bennu a B Tholen type asteroid [19], to achieve high-visibility reflectivity is to be preferred to patterning or colouring. Therefore, the simulated features will be highly reflective cubes. The resulting object, rendered using the Unreal Engine 4 graphics engine, can be observed in Figure 5-1 and Figure 5-2.

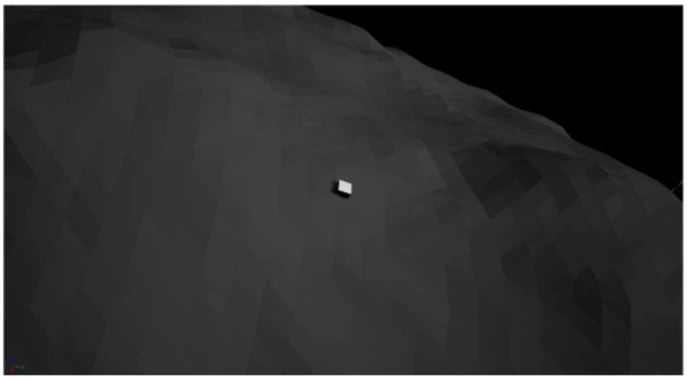


Figure 5-1 - A 1.5 m cube on the surface in the simulation.

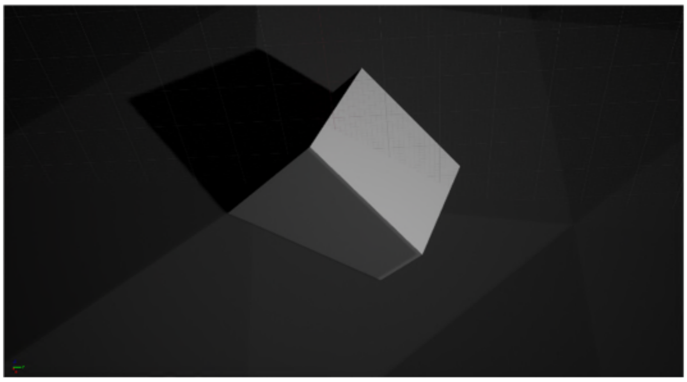


Figure 5-2 - Close-up of cube on the surface with its shadow. For this example, the light has been directed so as not to hit any face along its normal.

5.2.1.4 Simulation and Experimental Design

The experiments to study the propagation within the perception process of NAV-Landmarks have been performed in the beta version of THALASSA. In the beta version, as discussed in Chapter 3, Unreal Engine 4 (UE4) is used as a graphics engine. It is worth mentioning again that UE4 employs a left-handed reference frame, with a specific initial orientation for a simulated camera.

Three THALASSA blocks have been used within the experiments: Path, Sensor and Estimation.

*Path-*The Path block generated a trajectory by defining a landing condition and backpropagating it within the three-axis ellipsoid gravitational model. This trajectory is then followed from its furthest point from the ground to a certain altitude. In particular, the landing condition of interest is a vertical, equatorial landing around a nonrotational asteroid. The chosen landing position for the probe is $\boldsymbol{s}^L = \left(s_x^L,\ s_y^L,\ s_z^L\right) = (-259, 0, 0)$. The direction of the landing velocity, $\boldsymbol{V}^L$, is aligned with the local normal at $\boldsymbol{s}^L$, let this normal be $\boldsymbol{n}^L$, due to the condition of vertical landing. In these conditions the problem ultimately reduces itself to a one-dimensional motion, fully described by the behaviour along the X-axis.

A condition on the magnitude of the landing velocity can be implemented through V_c^L (Equation (2-6)), which must not be exceeded if the probe must remain on the surface. Thus, a reduction factor β is added to not operate with $\boldsymbol{V}^L$ values too close to the critical ones, and $V^L = \beta V^L, \beta \in \mathbb{R}^+, \beta < 1$. Additionally, it has been observed that the longer a trajectory takes to bring the probe to its designated target, the more the error committed by the on-board navigation grows. Ultimately, $\beta = 0.67$ - roughly $(2/3)$ - seems a good compromise between estimation stability and the presence of a safety margin, therefore the selected landing speed is $V_L = (0.67)V_L^c$.

With reference to Equation (2-6), it is possible to see that using the values of $R_M = 282.36\,m$ [129] (the equatorial semi-extension for the asteroid – its largest dimension), $m = 7.329 \times 10^{10}\,kg$ retrieved from [129], and $\epsilon_{crit} = 0.6$ [167]. Performing all the required operations it is possible to find that $V_L^c = 0.310\,ms^{-1}$. Therefore $V^L = 0.67(0.310) = 0.208\,ms^{-1}$, implying $\boldsymbol{V}^L = V^L\,\boldsymbol{\hat{n}^L} = (0.208, 0, 0)$.

These conditions lead to an extremely slow descent: it takes roughly 8 hours to move from about 3 kilometers away from the center of the asteroid to its surface. The motion of the capsule is followed from $||\boldsymbol{r}_0|| = 2021.75\,m$ (at $t_0 = 0$) to $||\boldsymbol{r}_{\boldsymbol{end}}|| = 1107.40\,m$, for a total time t_{end} of roughly 2 hours and 35 minutes ($t_{end} = 9257\,s$). The time step between two contiguous acquisitions is, therefore, $\Delta t = 1028\,s = t_{end}/(Steps - 1)$.

_Sensor-_Within the Sensor block, the directional light has been selected as coming from the $-X^{UE}$ direction. This implies that the landing trajectory observes a side of the object under direct solar illumination, and therefore fully illuminated.

Every image $\boldsymbol{I}$ in the discrete sequence is uniquely identified by a natural number, $k,$ and indicated as $\boldsymbol{I}_k$. As a total of 10 images are acquired through the sequence, the indexes will range from $\boldsymbol{I}_1$ to $\boldsymbol{I}_{10}$, with $\boldsymbol{I}_1$ being the image acquired furthest from the ground.

Cubes of two sizes are tested: respectively, cubes with a $l = 1.00\,m$ side (test condition "$s\,\bullet$") and cubes with a $l = 1.50\,m$ side (test condition "$S\,\bullet$"). The number of landmarks employed during these simulations is 4 in one characterization, and 6 in another. To these, a condition with 0 cubes is added, as a form of control experiment. Ultimately, this leads to 5 possible combinations of conditions to be tested, namely $\boldsymbol{tc} = \{S0, s4, S4, s6, S6\}$. It is worth noting that albeit experiments with $l = 1.50\,m$ might not be realistic, they provide important information to understand the effect of the cubes on navigation. The lower bound to the dimension of the landmarks within the simulation is provided by the resolution of the target's CAD model, which, again, is 0.75 m. This because it is of interest that a landmark always correlates to structure rather than noise.

Concerning the landmarks' position, Çelik et al. [168] have investigated, albeit for a 3-body system, the deployment accuracy for passive ballistic landings on small asteroids. In that study the same landing conditions (vertical, equatorial) and backpropagation method led to a "butterfly"-like pattern, characterized by a central dispersion ellipse and two slightly asymmetric "wings". The region covered by this pattern is roughly of the same order of magnitude of the asteroid's radius itself. Namely, this area stretches for 120° along the longitudinal axis and 20° along the latitude extension. This distribution is approximately centered on the nominal landing position, with the spread along the latitude increasing with an increase in longitudinal distance from the nominal landing position.

Since all the geometric and dynamical values involved are comparable in both instances, these angular diameters are imported to simulate a landing distribution. Considering the worst case and thus ignoring the central build-up, landing spots for markers are constrained to lie in one of the “wings”, approximated as isosceles triangles with angular size equivalent to the one mentioned above. 10 random markers positions have been extracted from equivalent planar triangles and then projected on the surface, obtaining the results seen in Figure 5-3.

The simulated sensor is a camera with a focal length of $f = 9.6\,mm$. The simulated sensing plane is constructed to be equivalent to a $1240x1080$ one (1.3 MPix), with square pixels with a $5.3\,\mu m$ side. Therefore, the sensor height will be set to $h = 6.05\,mm$ and the sensor width to $w = 6.94\,mm$. The effective resolution number of the generated images is, however, 664×578 pixels. The above values have been modelled, with some slight difference, again upon the C3D CubeSat Camera [141] also employed in Chapter 4.

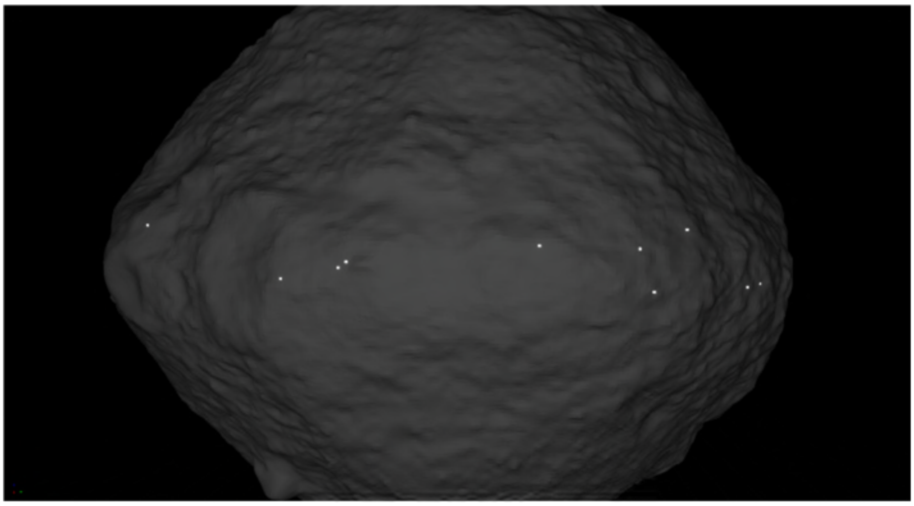
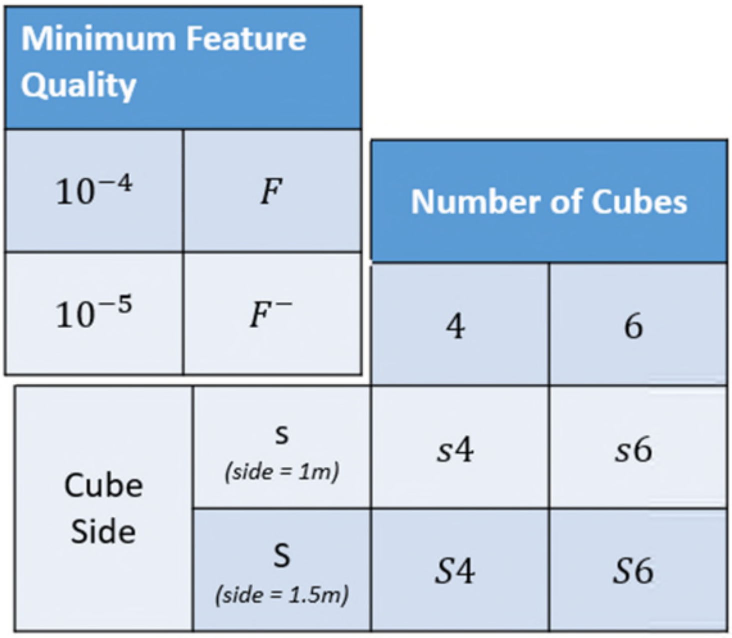


Figure 5-3 - The landmarks (l = 1.50 m) as they appear on the asteroid.

*Estimation-*Lastly, the Estimation block performs Visual Odometry through a standard implementation. When the images are fed to the processor, Harris-Stephens corners [84] are detected, described and matched over subsequent images.

Homologous points in consecutive images are then used to construct the fundamental matrix through the normalised eight-point algorithm [103]. The robustness of this process is further increased by introducing a RANSAC (RANdom SAmple Consensus) routine. Lastly, the fundamental matrix is reconstructed to estimate the unscaled relative translation (and rotation) between the two views. Rotation and translation are left as they are, and not refined through neither a local nor a global optimizer. This process, iterated sequentially, provides an estimation of the complete trajectory of the platform as unscaled motion direction. It is worth noting that introducing the RANSAC makes the process non-deterministic.

A last testing condition addresses the threshold associated to corners acceptance. This is set either to 10^{-4} or 10^{-5}. When this is set to 10^{-5} the associated test case is labelled with a – minus superscript, e.g., $S6^{-}$.Figure 5-4 provides a summary of all the testing conditions.



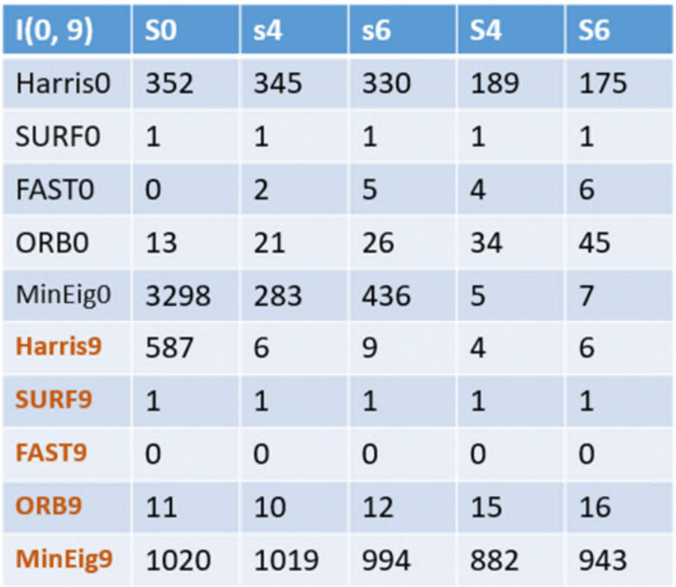
Minimum Feature Quality	
10^{-4}	F
10^{-5}	F^{-}

		Number of Cubes	
		4	6
Cube Side	s (side = 1m)	$s4$	$s6$
	S (side = 1.5m)	$S4$	$S6$

Figure 5-4 - Test conditions for NAV-Landmarks

It was previously observed that Harris-Stephens corners may not represent an ideal feature choice from an engineering point of view within scale-changing contexts. However, choosing them is scientifically significant, as this selection allows to introduce an analysis and a discussion about extrema in computer vision.

Figure 5-5 shows the number of features detected within the first and the last image using a few common feature detectors. When a detector's name is followed by 0 the detection is performed in the first acquisition, i.e., the one furthest from the ground. When a detector's name is followed by 9, the detection is performed within the 10th acquisition, i.e., the one closest to the ground.



I(0, 9)	S0	s4	s6	S4	S6
Harris0	352	345	330	189	175
SURF0	1	1	1	1	1
FAST0	0	2	5	4	6
ORB0	13	21	26	34	45
MinEig0	3298	283	436	5	7
Harris9	587	6	9	4	6
SURF9	1	1	1	1	1
FAST9	0	0	0	0	0
ORB9	11	10	12	15	16
MinEig9	1020	1019	994	882	943

Figure 5-5 - Features detected in the first and the last image using a few common detectors

Figure 5-5 shows that SURF and FAST always perform badly, while ORB and MinEig consistently return an operative number of features.

It is interesting to note that within the last image the number of detected Harris-Stephens corners appear to converge towards the number of NAV-Landmarks. Indeed, for the case with a small side ($s\ \bullet$) the number of detected features is similar to that of deployed NAV-Landmarks. These numbers are identical for the case with a long side ($S\ \bullet$). This, issue, however, stops appearing when the minimum threshold is lowered from 10^{-4} to 10^{-5}. This behaviour is explained in Section 5.4.

5.2.1.5 Results

The output of each step of the VO process are, within the scope of this work, a unit vector representing the direction of the estimated displacement, and a rotation matrix. This is because the scaling problem has been for now neglected. The simulated motion profile is one-dimensional, with a unit value for each time step and evolving only along the X axis. Hence, a perfect estimation from the time-step k to the time-step $k+j$ would return a displacement only along X, with a value equal to $(k+j)-k=j$.

This means that in any estimation within this model and reference frame, components along the Y axis and the Z axis are entirely spurious. Hence, estimation errors in translation can be associated to these components: if they are present, the estimator is not behaving as it should. Moreover, non-roll rotations propagate on the same Y and Z components, which can therefore be also seen as indicators of rotational errors. From these considerations, it is possible to define an error metric, named Δ_e^s, and characterized as

$$\Delta_e^s = (e-s) - X_e^{Global} \qquad (5\text{-}2)$$

Where e and s are the indices of the first and the last image of the sequence taken into account, and X_e^{Global} is the cumulative X value of the sequence, with the zero placed in the I^s condition.

$\sqrt{\Delta_e^s}$ is an indicator of the effective error components; therefore normalizing $\sqrt{\Delta_e^s}$ by $(e-s)$ would return the percentage uncertainty in estimation over the trajectory of interest.

Assuming implicitly $s = 1$, the values Δ_5 $(e = 5)$ and Δ_{10} $(e = 10)$ are presented for each case. Since, as already observed, the process contains source of randomness, it would be misleading to simply present the outputs for one instance: rather, statistical descriptors must be introduced. Namely these will be the mean, the average, and the standard deviation of these parameters, computed over populations containing 300 samples. The results for $\{S0,\ s4,\ S4,$

$s6\}$ are shown in Figure 5-6 and Figure 5-7 and some statistical indicators are reported in Table 5-1 and Table 5-2.

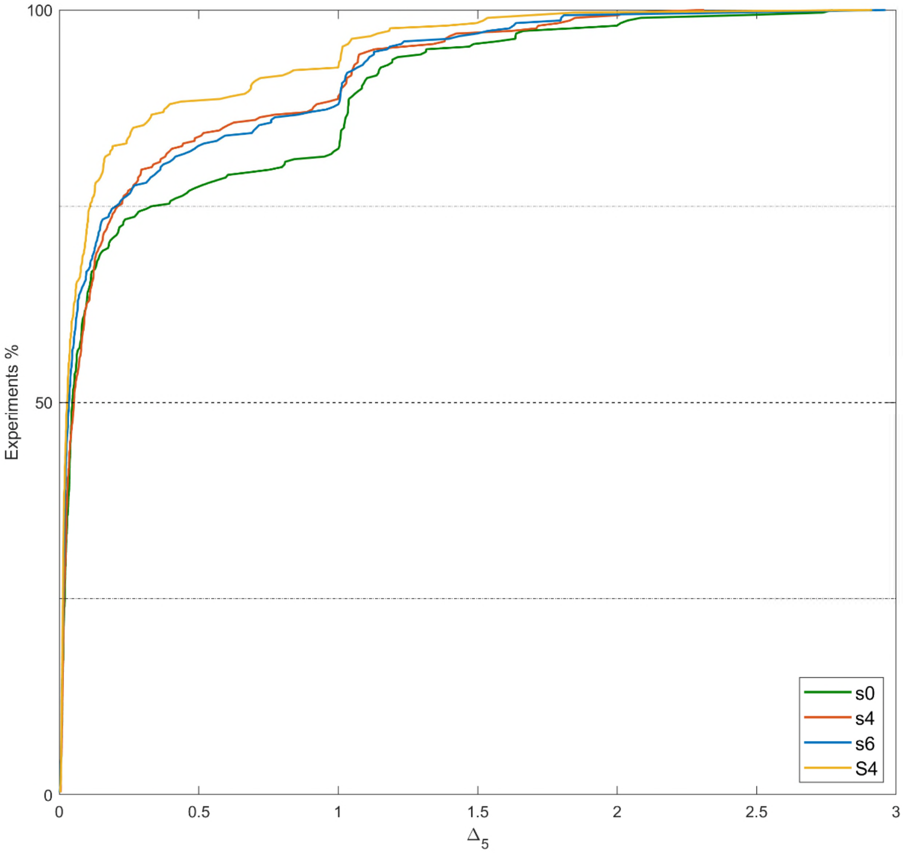


Figure 5-6 - Cumulative Δ5s for s0, s4, s6 and S4.

Table 5-1 - Statistical indicators for Δ5s for s0, s4, s6 and S4.

Test Case	Δ_5		
	Mean	***Median***	***Standard Deviation***
S0	0.3235	0.0467	0.5317
s4	0.2531	0.0527	0.4388
S4	0.1769	0.0285	0.3686
s6	0.2530	0.0351	0.4572

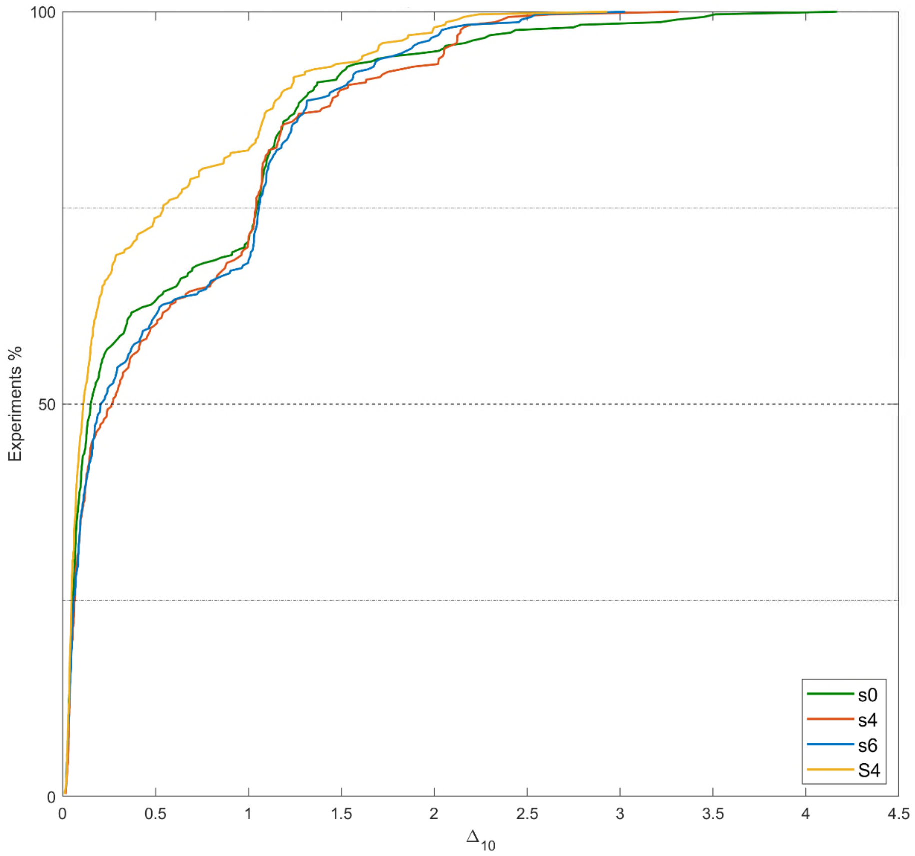


Figure 5-7 - Cumulative Δ10s for s0, s4, s6 and S4.

Table 5-2 - Statistical indicators for Δ10s for s0, s4, s6 and S4.

Test Case	Δ_{10}		
	Mean	***Median***	***Standard Deviation***
S0	0.5564	0.1529	0.7179
s4	0.5878	0.2685	0.6597
S4	0.3970	0.1125	0.5524
s6	0.5726	0.2106	0.6359

The nontrivial shape of the curve with a sharp step around a $\Delta_5, \Delta_{10} = 1$ value can be explained by analyzing the various errors affecting the estimation. The most common type of large-scale errors for this process appears to be not estimating at all a translation along the X axis. A single missed estimation of this type induces a unitary Δ error, i.e., $\Delta = 1$. The accumulation of this type of error over multiple tests forms the plateau observed in the figures. Therefore, the extended vertical regions around natural numbers roughly indicates the likelihood to happen of discrete non-detection events. The onset in the curve, instead, relates to the general probability of this phenomenon happening at least once.

From the above figures two things can be observed. The first, is the regularity of the profile of the curve. The second is how extreme values are generally comparable. This means that the algorithm is behaving consistently in every test case. However, the various curves do not collapse to a single profile, but, rather, present the same shape translated vertically. The translation is induced by a variation of the number of test cases achieving at least an error of Δ. The more results have a higher error, the more the curve shifts downwards. Thus, the position along the abscissae of the first onset of an exponential behavior is a good indicator of the performance of the algorithm. In other words, the more a curve is to the left, the better the algorithm is performing.

As it can be inferred from both the tables and the curves, the $S4$ landmarks provide the best improvement from the $S0$ case, both for Δ_5 and Δ_{10}. On the other hand, $s4$ and $s6$ appear to induce an identical improvement for Δ_5, halfway between the $S0$ and the $S4$ test case, but then present a worse performance than $S0$ for the Δ_{10} case.

The test case S6 has not been shown above because using the algorithm with the same parameters, would, inevitably, make the program unable to operate, and lead to crashes. This can be explained using Figure 5-3 and remembering that the VO algorithm requires at least 8 points to operate.

This effect is most likely induced by the stark differences in grayscale between the markers and the background, coupled with a higher number of landmarks in extremely favourable positions, improving the quality distribution of the best Harris-Stephens corners. This, in turn, implies that for a broad range of acceptance thresholds on the quality of the features there are fewer features satisfying the score requirements. In turn, this leads the cardinality of the features to be tracked to be critically low, close to the minimum number of points necessary as inputs to the algorithm. If this is coupled with a, for any reason, perceptionally troublesome points distribution, ultimately not enough features are able to be matched between consecutive views to estimate the fundamental matrix. This problem, however, might be addressed using a controller operating on the required minimum quality of the features.

To test this hypothesis, the minimum quality parameter is reduced manually from 10^{-4} to 10^{-5}. This means that for a feature to be accepted by the algorithm its quality score is at least the best feature's score multiplied by 10^{-5}. These are the test cases labelled $S6^{-}$, $S0^{-}$ and $S4^{-}$. These are, respectively, $S6,$ $S0$ and $S4$ with the same reduced value of minimum feature quality. The results are presented in Table 3 and 4 and.

To bridge the two sets of results in an easier way, Figure 5-8 and Figure 5-9 show how the $S0^{-}$ and $S4^{-}$ set perform, respectively, against $S0$ and $S4,$ and $S0$ and $s6$. The tables show that reducing the minimum quality parameters improves the Δ_5 behaviour of the marker-less case $S0$ to a $s4/s6$ level. This advantage, however, is entirely lost in the Δ_{10} case.

In general, what can be observed from these tests is that using lower quality features greatly reduces the effect of NAV-Landmarks on navigation performances. This is because this reduction introduces many spurious features in the process, which corrupt the matching phase and degrade the overall accuracy level. From an ASP perspective, this work shows the effects on Perception of actions on the structure of the Scenario.

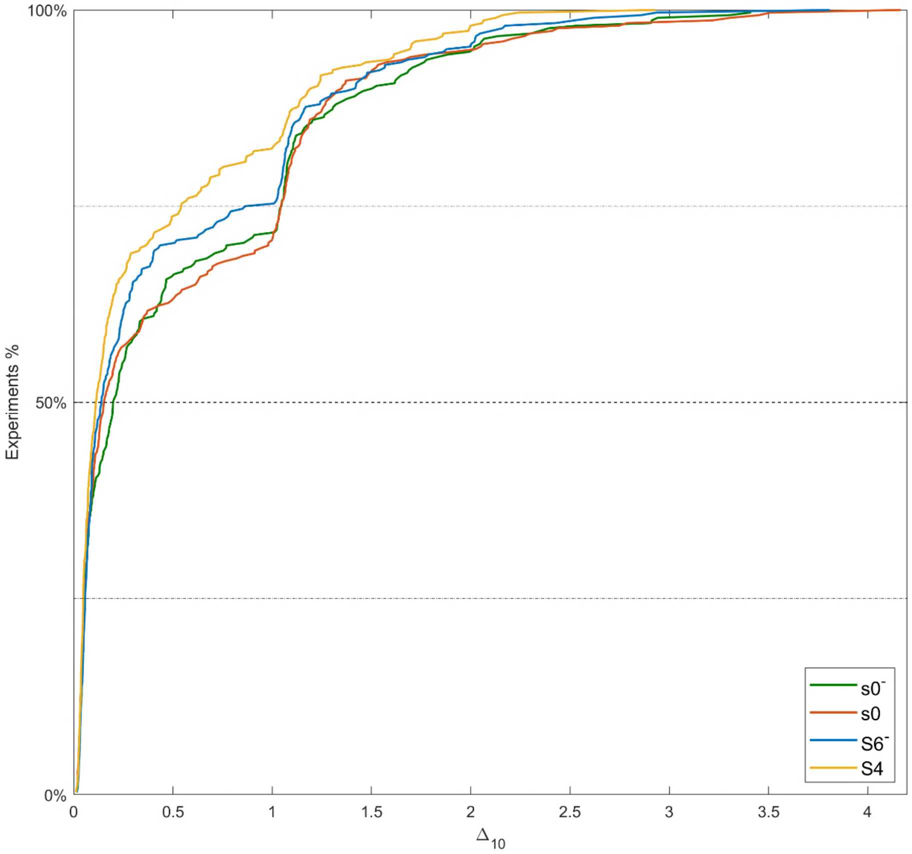


Figure 5-8 - Cumulative Δ10s for s0-, s0, S6- and S4.

Table 5-3 - Statistical indicators for Δ5s for S0-, S4-, S6-.

Test Case	Δ_5		
	Mean	*Median*	*Standard Deviation*
$S0^-$	0.2649	0.0452	0.4387
$S4^-$	0.2362	0.0344	0.4766
$S6^-$	0.2626	0.0412	0.4738

Table 5-4 - Statistical indicators for Δ10s for S0-, S4-, S6-.

Test Case	Δ_{10}		
	Mean	*Median*	*Standard Deviation*
$S0^-$	0.5555	0.1997	0.6979
$S4^-$	0.4925	0.1305	0.7305
$S6^-$	0.4832	0.1397	0.6488

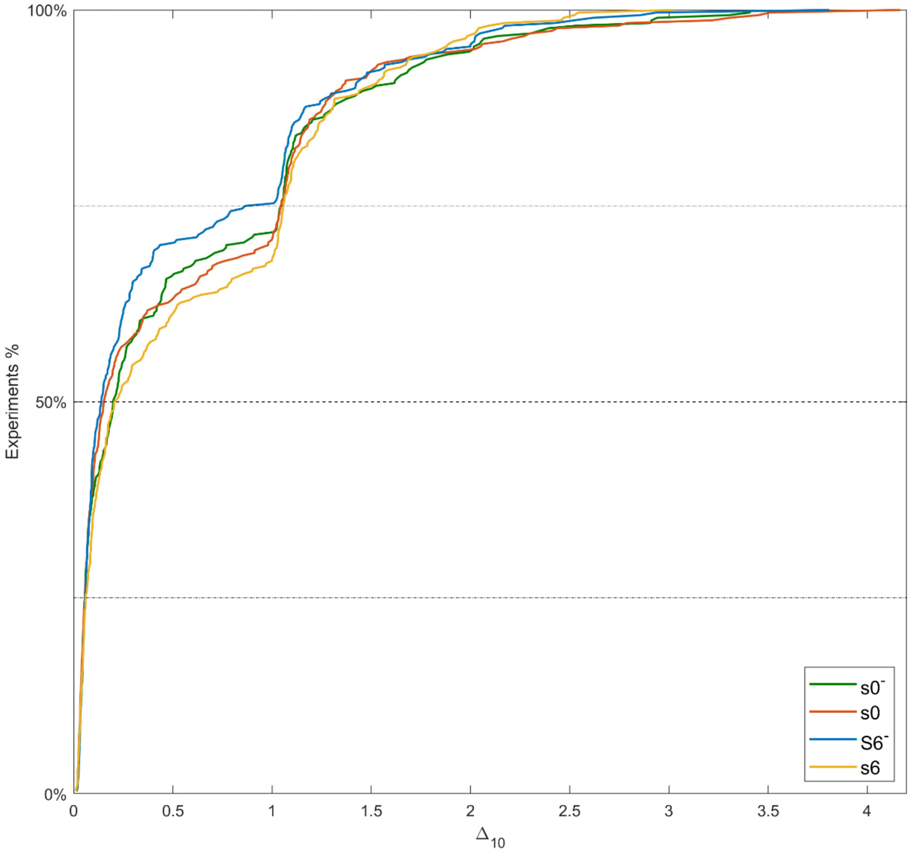


Figure 5-9 - Cumulative Δ10s for s0-, s0, S6- and s6.

5.2.2 Extrema in Computer Vision

This section presents invariants concerning perception models themselves. In particular, the perception result shown concerns the patterns strongly exciting feature detection models. These extremely good features are a fixed property of the algorithm: therefore, knowing them would allow both to select optimal detectors for the given operations and to avoid situations that would be critical for the underlying perception model.

5.2.2.1 Case study

In the previous section it was observed that under certain circumstances there may be an undesired by-effect caused by augmenting the environment through a restricted number of artificial elements. This is because these observables are specifically designed to induce a strong response in feature detectors. Therefore, optical infrastructure has the potential to completely lead the process within detection algorithms, sometimes with disruptive effects. Since the adverse effects are connected to the strong response, to study these effects equals studying the extrema value in a detection process.

This translates to the principle that a small set of physical objects can be propagated within the computational detection and interpretation process to manipulate, improve, set back, or neutralise the operations of an autonomous platform.

The possibility to exploit these effects was at first presented through a possible defence application. This use case was an electro-optical countermeasure affecting unmanned aerial platforms invading a restricted space. The unmanned platform is assumed to use feature-based navigation mediated through only one feature detector and is affected through its estimation process.

In this work, an optical attractor (OA) is defined as the collection of pixels inducing an exceptionally strong response for a target feature detector, and optical attractor kits (OAK) define the physical objects generating OAs. Finally, the complete countermeasure approach operating on this principle is termed, as per title, Countermeasures Leveraging Optical Attractor Kits (CLOAKs). From this, a physical object inducing these pixel structures for defence purposes was defined a CLOAK: *Countermeasure Leveraging Optical Attractor Kits*.

In particular there are two hypotheses tested within this section: that it is possible to retrieve optical attractors, and that they have a significant effect when introduced in an image. These hypotheses were tested through static tests on SURF implemented with default parameters within MATLAB.

CLOAKs act at a feature level. It has been noted before that most computer vision processes begin with the detection of features. Thus, the whole optical workflow depends on interpreting and manipulating these computational objects. Hence, the interpretational process of the artificial intelligence can be manipulated by diverting the focus of the detection towards specific regions and then acting on this restricted set of observables. Using optical attractors, any optical based algorithm relying on features extraction can potentially be disrupted, in a completely passive and nondestructive fashion. Within this work this is envisioned happening in at least two ways, but many more could follow.

5.2.2.2 Background

For a given image $\boldsymbol{I}_k$ acquired during ordinary operative conditions, a feature detector generally associates a score p_{jk} to each feature X_{jk} detected. The value p_{jk} characterizes the quality of the detection.

After the M features satisfying the boundary conditions have been retrieved, let us order their scores in a decreasing order. Let the number i, $i = 1, \dots, M; i \in \mathbb{N}$ indicate the rank of the feature with the score q_{ik}. The feature with the highest score is labelled “pivot”, and its score is indicated with $Q_{0k} = q_{1k}$. Experimentally, it was observed that it is generally possible to find three parameters such that $q_{ik}(i)$ is well approximated by a power law over a range of values

$$q_{ik}(i) \cong c_1 i^{c_2} + c_3 \qquad \textbf{(5-3)}$$

As an example, running a SURF detection with all default values over the built-in MATLAB demo image termed “concordorthophoto.png“(see Figure 5-10) it is possible to retrieve 22435 points, that when ordered have the distribution shown in Figure 5-11. To help in visualising where these points might be, Figure 5-12 shows the 80 strongest SURF features detected within the image of a remarkably fluffy cat. A noteworthy exception to this distribution is represented by images containing fractals structures where the distribution tends to be stair-like.

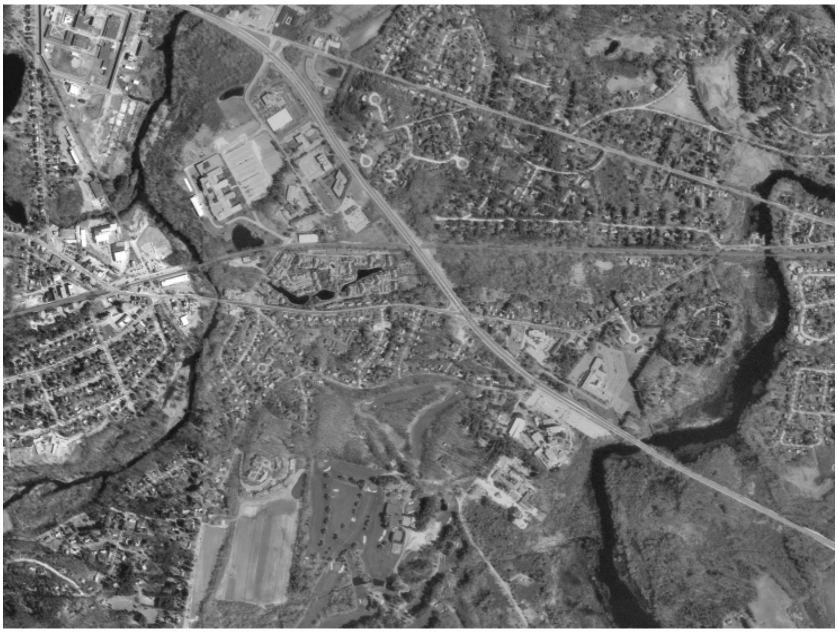


Figure 5-10 - The picture "concordorthophoto.png". Image from MATLAB

Scores and pivots are important because their values drive the acceptance or rejection of pixel patches as features. Let us illustrate this through the MATLAB 2019b [169] implementation of SURF and Harris-Stephens corners. It is again worth remembering that heuristically the two detectors look for different objects (blobs and corners, respectively), both from an algebraic and data perspective; however, this will be not a concern within this work.

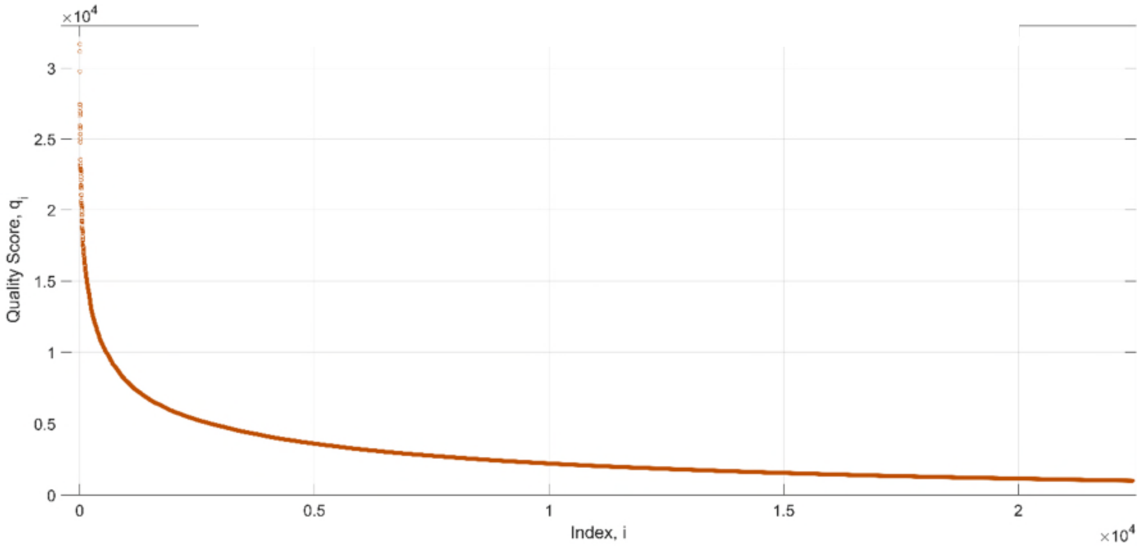


Figure 5-11 - Distribution of the Ordered Quality Score qi of SURF Features in "concordorthophoto.png"

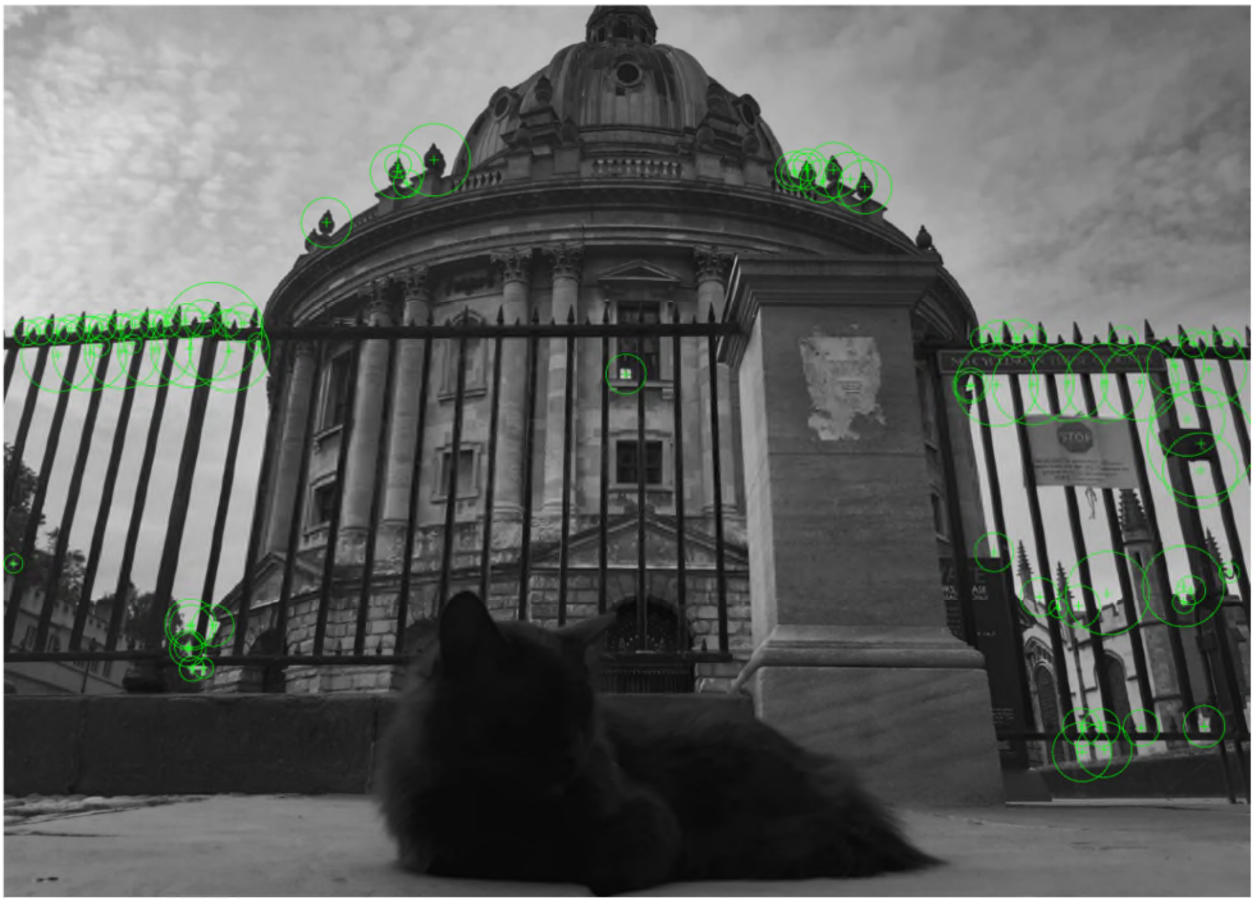


Figure 5-12 - The 80 strongest SURF features (green crosses and circles) detected within the underlying image.

Let T be a threshold. Moreover, let the subscript k be removed. Without adding computational safeguards, within MATLAB Harris-Stephens' threshold operates in such a way that a detected pattern with a score p_j is accepted as a feature if $Q_0 \geq p_j > T_H Q_0$, with $T_H \ll 1$.

On the other hand, for SURF $T_s \gg 1$ and a generic pattern with a score p_j is accepted if $Q_0 \geq \ p_j > \ T_S$. This means that the score of a feature is higher than the fixed threshold, and, by definition, lower than or equal to the pivot. Features with a rejection process of the first type are defined Pivot-Led (PL) and features with a rejection process of the second type are defined Pivot-Independent (PI).

These behaviours can easily be visualised by plotting the ordered score values q_i given two different thresholds for a PL detector and two thresholds for a PI detector. Figure 5-13 shows the results for the detection of SURF points with $T_S = 100$ and $T_S = 1000$. On the other hand, Figure 5-14 shows the results for the detection of Harris-Stephens corners with $T_H = 0.001$ and $T_H = 0.01$.

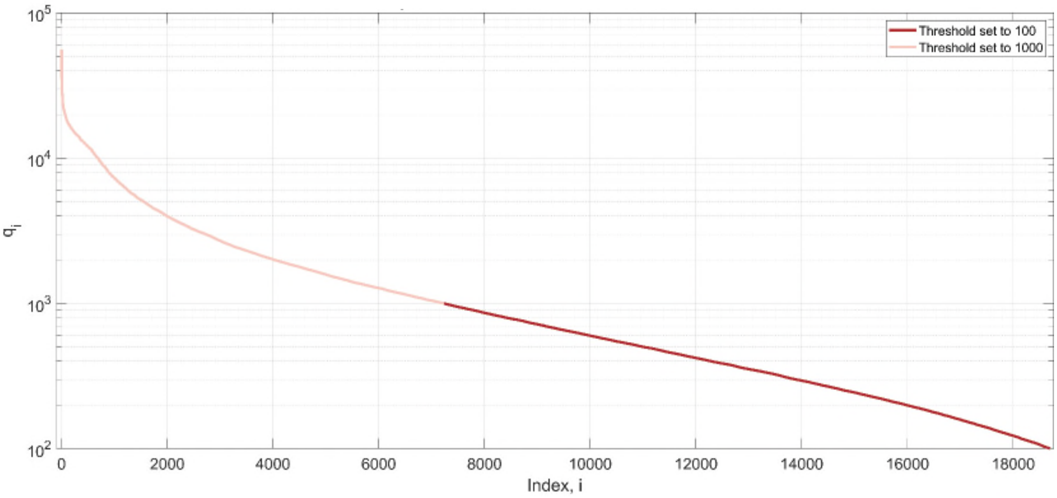


Figure 5-13 - Semi-logarithmic plot of qi for the detection of SURF points with two different thresholds

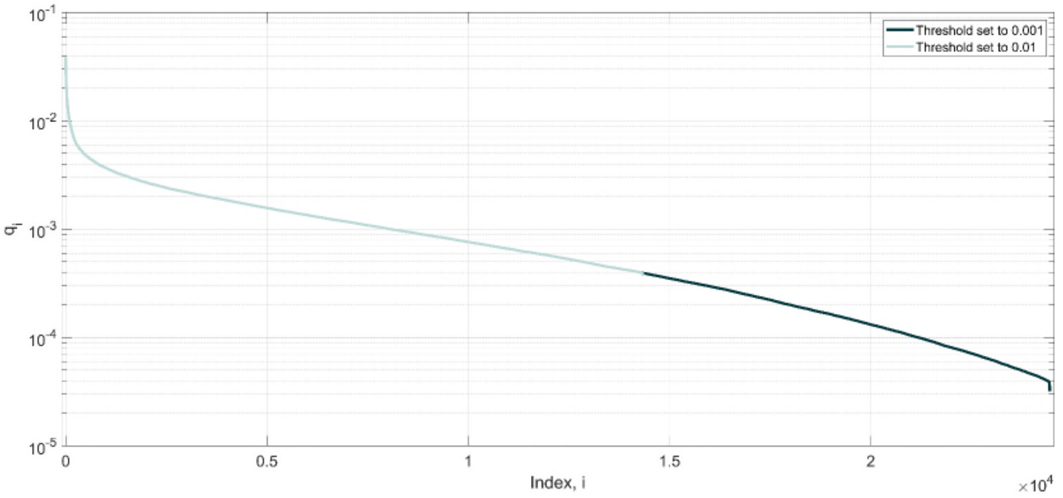


Figure 5-14 - Semi-logarithmic plot of qi for the detection of Harris-Stephens corners with two different thresholds

5.2.2.3 Affecting Visual-based Processes through Optical Attractors

Using the definition of PL and PI processes, it is ultimately possible to introduce the first two approaches enabling the disruption of a visual based process. First, a PL process gives the ability to shift and manipulate the whole rejection interval by introducing an OA. It can be added that if the rest of the environment is built in such a way that the number and quality of the remaining features is kept low, it is effectively possible to consistently induce a scarcity of detected features. This approach was defined “starving”.

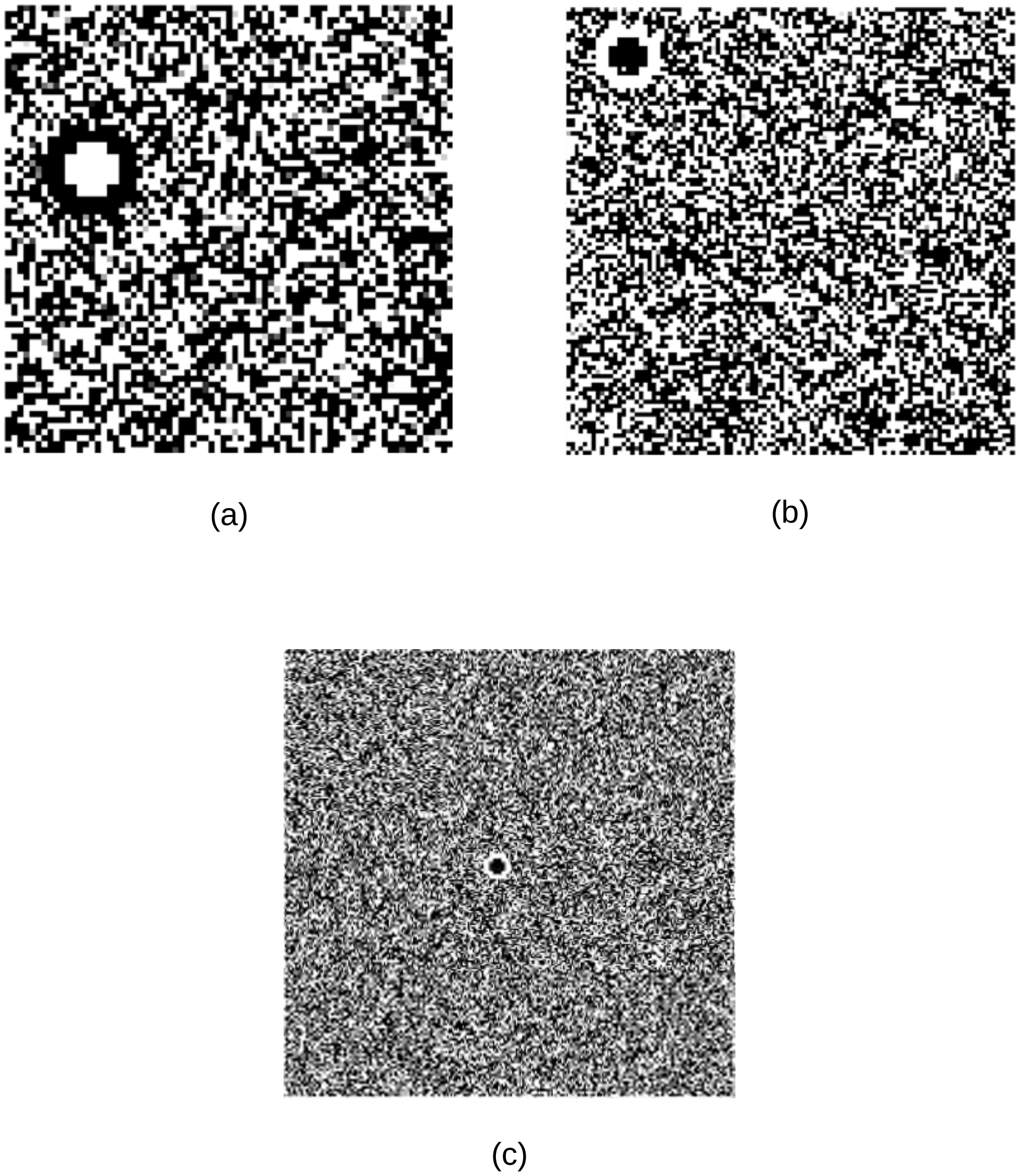
On the other hand, since the lower boundary in a PI process cannot be externally manipulated, it is not possible to exploit it in the same way. Therefore, PI processes require a more complex approach, which is defined “herding”.

Whereas starving is a static technique, herding, is a dynamic technique, requiring the scene to change between images. That being said, herding may still be considered to be a passive system when there is no activation prompted by an external signal. Neglecting for now thermal effects as only solutions for visible cameras are considered, as a first iteration it is possible to consider CLOAKs to be screens capable of showing a grayscale image. This enables considering different OAs interchangeably and is consistent with a dynamic application.

Therefore, ultimately, herding could consist in continuously moving OAs between the screens, structuring their motion pattern in such a way that the same descriptor ends up applying to objects which are continuously shifting position in space. As these features represent by design the ones with the highest scores, the detector rarely rejects them, thus constantly integrating disruption-inducing elements in the process. Beyond the thresholding methods, there are two main processes involved in the deployment of a CLOAK. These are the “Extraction” and “Injection” phases. The former, extraction, refers to determining optical attractors for the given configuration of a feature detector, through either heuristics, or optimization techniques. Afterwards, in the injection stage, these are introduced within the field of view of the sensor, in such a way that process disruption is maximized.

Within the scope of this study a naïve full version of this workflow is developed and analysed, again in MATLAB (v. R2019b). For the extraction stage, search and optimization processes known as genetic algorithms [170] are used to “evolve” an ideal optimal response to the analysed parametrization of the feature detector.

Figure 5-15Figure 5-15 - The Optical Attractors emerged in the 75x75, 105x105 and 255x255 square. shows the outputs of this search process with the retrieved optical attractors for square pixel patches of different sizes (75x75, 105x105, and 255x255). OAs can be clearly seen in all the three figures as the cross-like structure surrounded by a continuous circle of the opposite colour.



(a)

(b)

(c)

Figure 5-15 - The Optical Attractors emerged in the 75x75, 105x105 and 255x255 square.

It is worth noting that although it might appear that the OAs shrink with a larger image side in pixels, this impression is simply an artefact of the way the features are presented, where pictures with an increasing side length in pixels were kept at a fixed metric size.

The shape of the OA depends on the mathematical model underlying the detector. As an example, Figure 5-16 shows optical attractors for KAZE features and FAST features. KAZE is a multiscale feature detector. The name of this detection algorithm is not an acronym, but, rather, the Japanese word meaning "*wind*" [171]. KAZE's optical attractor is represented by the object looking like a "Moon with a halo". FAST's optical attractor is the object resembling a question mark in the right-hand side of the picture.

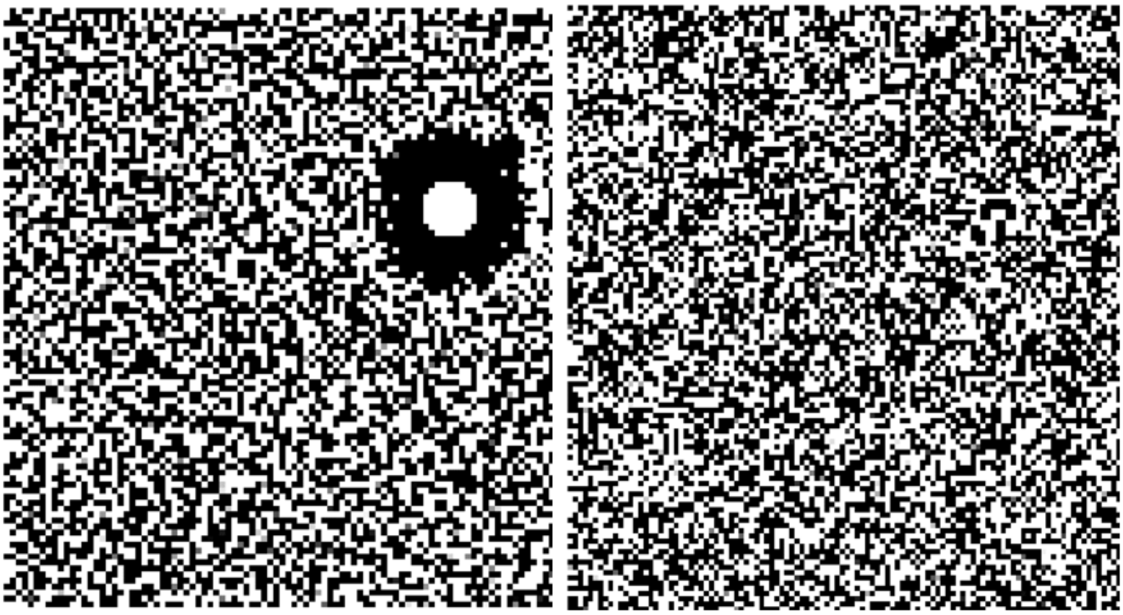


Figure 5-16 - Optical attractors for KAZE and FAST features

5.2.2.4 Results

Figure 5-17 shows the q_i obtained within the pixel patches in Figure 5-15 using a SURF detector. It is immediately possible to see the three OAs values the in the upper left corner.

To test the injection stage, the 15x15 pixels patch containing the OA from Figure 5-15c is extracted. As per Figure 5-17 this OA is the one with the highest Q_0 among the three OAs and obtain a transferable OA. This is an approximation, because the shape of the OA appears to be circular, while the selected patch is a square. However, it has been observed that this choice does not appear to affect the Q_0 in a significant way.

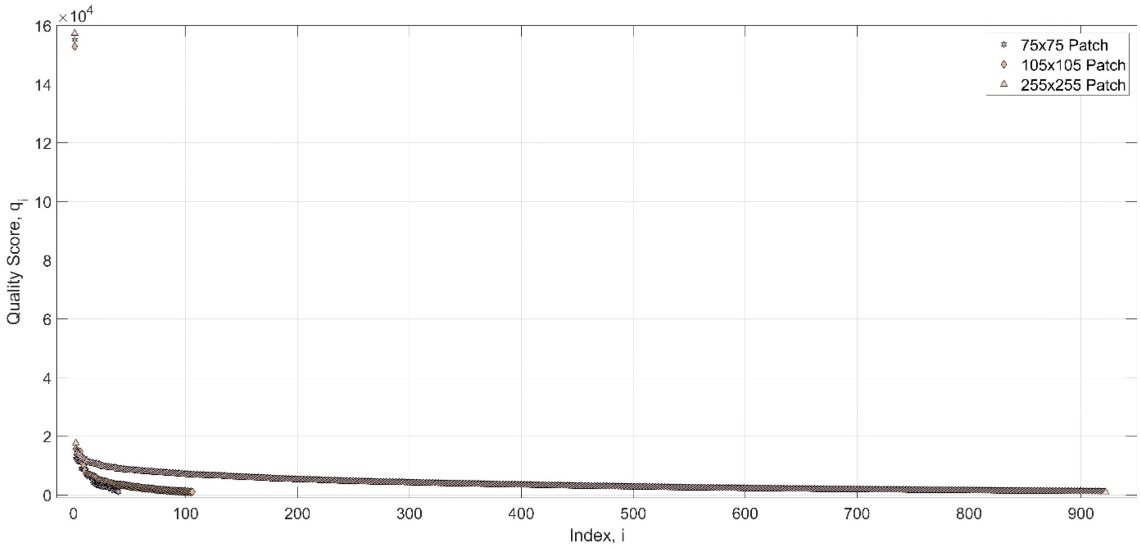


Figure 5-17 - Distribution of the Ordered Quality Score qi of SURF Features in GA-Evolved Patches

By performing a first proof of concept test, pasting the square OA in random places within the test image of Figure 5-10 it is possible to observe two things:

a. The Q_0 value remains essentially constant in whatever picture and position the OA is found. This has not been tested for rotational invariance

b. If the OA is placed on the edge of the image Q_0 reverts to a value detected within the picture not containing the OA.

5.2.2.5 Discussion

This section introduced a set of ancillary concepts to discuss the effect on perception of extrema values in computer vision. These maxima in the perceptual response are labelled Optical Attractors, OAs. From an ASP triangles point of view, this work showed that when extrema values in optical processes are to be expected in the Scenario, appropriate safeguards should be implemented in the design of the Perception model.

These concepts were discussed through an application, labelled Countermeasures Leveraging Optical Attractor Kits (CLOAKs). These countermeasures target and exploit visual based autonomous estimation processes. CLOAKs are based on the concept of OA.

Two important phases in the operative deployment of optical attractor kits are recognised: extraction and injection. Extraction concerns the retrieval of the relevant pixel patterns, injection their introduction within a scene. Using a simulated extraction/injection process it is shown that it is possible to retrieve an optimisation process, and that this feature strongly affects detection. Through these results it is possible to introduce an invariant: the digital object strongly exciting a feature detector. These pixel patterns can then be used for multiple scopes in space applications, from choice of the optimal feature given the characteristics of the environment to safeguarding against critical observations.

5.2.3 Perception Fields

5.2.3.1 Scope

The investigations pursued in Section 5.2.2 and Section 5.2.3 stem from issues highlighted by the results of the NAV-Landmarks section (5.2.1). Section 5.2.2 addresses the problem of the number of detected visual features at times progressively decreasing as the probe gets closer to artificial landmarks designed for maximum visibility. This reduction has been observed to be due to the presence of perceptual ‘sinks’, strongly attracting the attention of a detector. To explain these sinks, Section 5.2.2 lays the groundwork for a theory of extrema detections within feature detection processes. However, for some feature types (e.g., SURF features) that effect does not explain the observed scarcity of detections. Hence, there must be some other process at play inducing this second behaviour.

Observing the heuristics of the pixel patterns inducing a strong response in a SURF detector (Figure 5-15), it is possible to observe that these are distributed patterns, while Harris-Stephens are strongly excited by corners and isolated points. The experiments concerning NAV-Landmarks are performed around a nonrotating asteroid, and the probe performs a vertical descent with a nadir-pointing camera. Therefore, the observed side of the asteroid is always under the same solar illumination conditions: within the field of view, the target only expands.

Moreover, the constant solar illumination conditions are conditions of direct illumination. The studied object is textureless, thus all the features observed come from shading, i.e., from the interaction of illumination and the structure of the target. Therefore, when the Sun is right behind the probe, it does not induce any widespread shadow in the field of view. Therefore, the low number of detections of extended features (as SURF or ORB) could be explained by the probe moving along the direction of Sun beams, and never reaching a pose where structural features from shadow are available.

Therefore, starting from that observation, it can be hypothesised that by increasing the Sun phase angle, the number of available features for a SURF-like would increase. This hypothesis, if verified, could suggest a new way of interpreting features detected in images generated by visible cameras in the proximity of small celestial bodies, in particular as a way to passively measure φ. Ultimately, the hypothesis tested within this section is that trends in the quantity of available optical features can be correlated to variations in the angular distance from the source of illumination.

From this, the discussed approach is based on treating properties related to these detected optical features as readings of a field – labelled “perception field”. These fields are assumed induced by the coupling of the environmental conditions and the state of the sensing device.

Being induced by illumination, perception fields are considered invariant to the spectrum of characteristics of NEO targets. Therefore, they represent self-contained ancillary tools providing context and an evaluation of estimators’ performance while using the least number of priors. These characteristics, make perception fields extremely significant around NEOs. This preliminary study presents an analysis of the occurrences of perception fields observed in a planar motion around two asteroids, 101955 Bennu and (8567) 1996 HW1 in visual data simulated within THALASSA 1.0. Ultimately, the comparison of three different feature detectors shows distinctive trends in the distribution of the detected optical features, directly correlated to the spacecraft-target-Sun angle, confirming the hypothesis.

5.2.3.2 Usage

At any given stage of a mission, selecting optical features to track that are optimal in terms of availability and quality strongly depends on multiple operational conditions. Developing a framework automatically performing this selection for operations in the proximity of asteroids is a particularly complex task.

The opportunistic features observable in operations around NEOs are likely to be ambiguous (similar to each other) or unstable (self-similarity not preserved over time) due to the nature of the landscape on the surface of an asteroid. Moreover, the generally fast spin rates induce continuous changes in illumination, further aggravating the lack of stability in the visual appearance of the surface features within sequences of images. On the other hand, well-defined and large unique geological features such as craters cannot be assumed generally available. Therefore, building a complete visual pipeline based only on these elements would greatly reduce the generality and applicability of the navigation process.

Following these premises, this work suggests a counterintuitive approach which can be used to increase the robustness of feature-based methods, while providing an additional navigation tool in resource-constrained conditions. Accordingly, bulk information on features is interpreted in a way that is analogous to readings of fields for physical quantities, like magnetic or gravitational fields. These were defined as ‘perception fields’, readings of quantitative values defined for each point in the space surrounding a target body. Perception fields are considered induced by the interaction of the perception model operating on a platform, the state of the platform and the state of the operational environment. These readings can be used to determine the optimal feature given an orbital position, structure ancillary navigation functions, or validate the outputs of concurrent processes.

The rationale behind this idea emerged from the following consideration: while lighting conditions can have deleterious effects on individual local structures by changing their appearance over time, they conversely induce seemingly regular distributions of collective feature properties. This section focuses on presenting a correlation induced by illumination observed between the spacecraft-Sun-target- angle, or Sun phase angle φ, and curves generated by enumeration of available features per detector, among the selected ones. Specifically, two assumptions were tested, fundamental to understanding if there is the potential to integrate this information in navigation processes:

- For a given set of fixed target conditions, properties associated with optical features are distributed as a function of the Sun phase angle.
- It is possible to predict the availability of optical features using in the first instance only information about φ and one picture in direct solar illumination

Moreover, as shown in the previous section, the shape of the optimal feature can be predicted either analytically or through search/optimisation methods. Therefore, once that shape is known, it is possible to infer if a local variation in the time series corresponds to the presence of an optimal-like feature. Therefore, anomalies in the feature distribution are potentially helpful discriminants to resolve ambiguities.

5.2.3.3 Simulation

The The simulation process in this case is needed to generate sets of visual data enabling the determination of the curves showing the quantitative distribution of features for each observing station. This is done by employing three blocks of THALASSA 1.0. These three blocks are a Path block, a Sensor block, and an Estimation block.

The tests are operated around two simulated asteroids representing plausible NEO targets, with different shapes and levels of surface resolution. These are once again the model of 101955 Bennu with a resolution of 75 cm [122], and a coarse model of the asteroid (8567) 1996 HW1 which was generated from ground-based observations [123]. These are respectively representative of spin-top-shaped bodies and contact binary asteroids.

To avoid defining a lower observational limit considering the point where the size of voxels becomes comparable to the distance sampled on the ground for each pixel, a smoothing of the models is performed within Blender. This allows avoiding the emergence of spurious features induced by voxels occupying more than 1 pixel. This could be a solution also in a case like the NAV-Landmarks experiments, where a lower bound for operations had to be set. The Path block is used to generate circular sets of acquisition points around the two targets, while an Estimation block is used to perform feature detection and quantify the number of features available at a given angular station.

Most of the design choices concerned the Sensor block. In both cases, the surface material was set to be a Diffuse BSDF with an RGB value of R=15, G=15, B=15 and a null roughness value. Within Blender the models have the dimension reported in Table I, both scaled to have the largest dimension equal to Bennu's largest axis, where for Bennu $R_{max} = 282.5\,m$. This provided for both cases the ability to equally configure acquisition distances, Sun orientation and camera models. The final output consisted of images like the ones illustrated in Figure 5-18 to Figure 5-23. Due to the lack of albedo or texture distributions, it was only possible to consider features generated from the interaction of structure and illumination. Accepting this condition appears justified by assuming that features induced by textural elements are uniformly distributed. Therefore, their effect on the shape of the illumination-related feature distribution curve would be limited to variations in amplitude, rather than in shape.

All the test cases have been generated for static Sun conditions. The field labelled ‘energy’ of the simulated stellar light source, defining its intensity within Blender, is set to 5, specifically to maintain a modestly powerful level of illumination.

The simulated image acquisition processes are performed at 6 and 12 times R_{max}, where $6R_{max} = 1695\ m$ and $12R_{max} = 3390\ m$, with an angular step of 1.5°, from 0° to 358.5°, by centre-pointing cameras moving in circles in the X-Y plane of the target asteroids. The simulated camera is based on the design of the CMOS-based GomSpace NanoCam C1U [172] assumed to be equipped with a lens having a focal length of 35 mm, and a focal plane with a resolution of 2048 x 1536 pixels. Three feature detectors were employed as observables in this study: Harris-Stephens corners [84] and SURF [173], which are well established in computer vison and aerospace domains and respectively representative of pivot led and pivot independent features.

Table 5-5 - Targets' Dimensions

	X	Y	Z
Bennu	565 m	536 m	499 m
HW1	565 m	222 m	243 m

The large number of parameters involved led to choosing only three simple test cases. They were used to present preliminary results as a foundation for more ambitious future tests in more challenging configurations. The first test case has Bennu as a target body; the second test case has HW1 with its longest axis aligned with X^b (configuration A); the third test case has HW1 with its longest axis aligned with Y^b (configuration B).

In all three sets of test conditions the Sun direction is set to be coincident with the direction of the X^b axis, coming from +X^b.

Moreover, in all cases, the Sun direction is coplanar with the plane containing the motion.

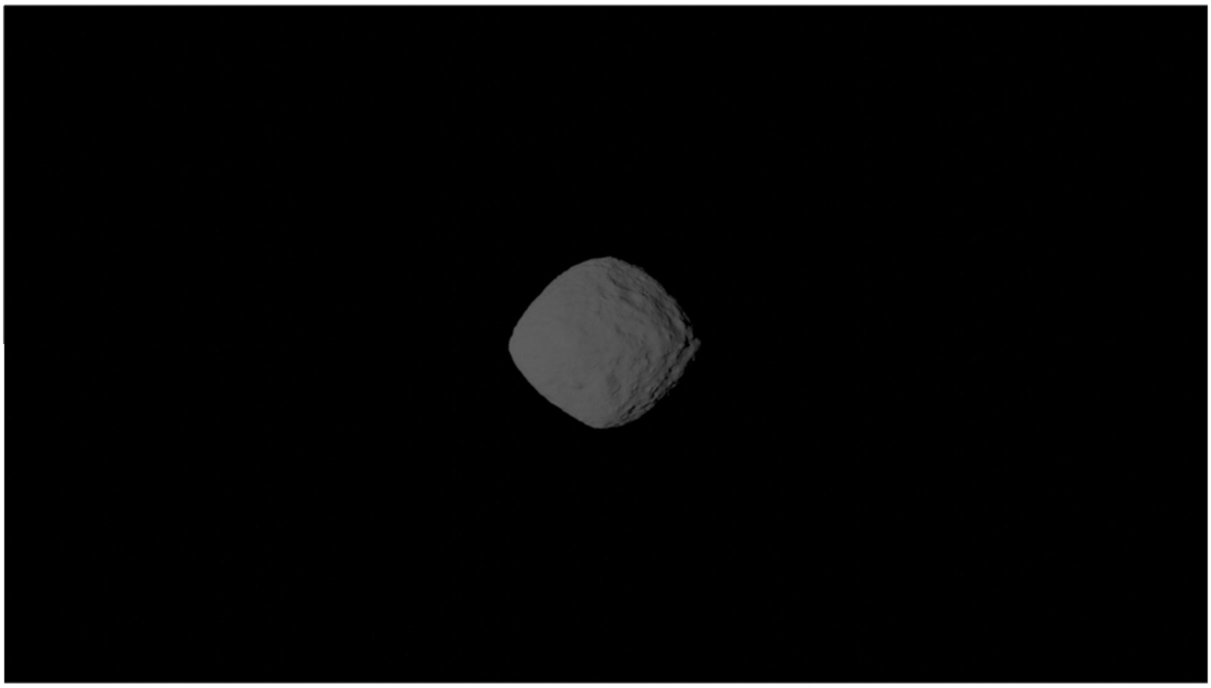


Figure 5-18 - A simulated Bennu at $12R_{max}$ at an angular station equal to 25.5 deg

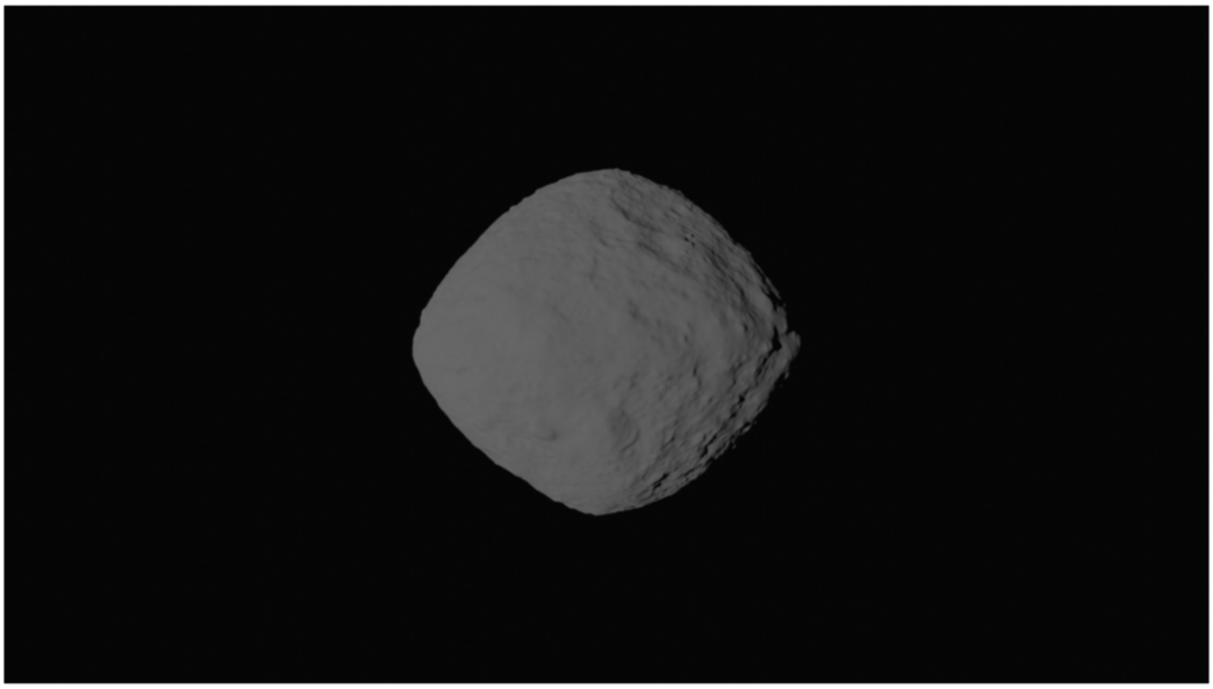


Figure 5-19 - A simulated Bennu at $6R_{max}$ at an angular station equal to 25.5deg

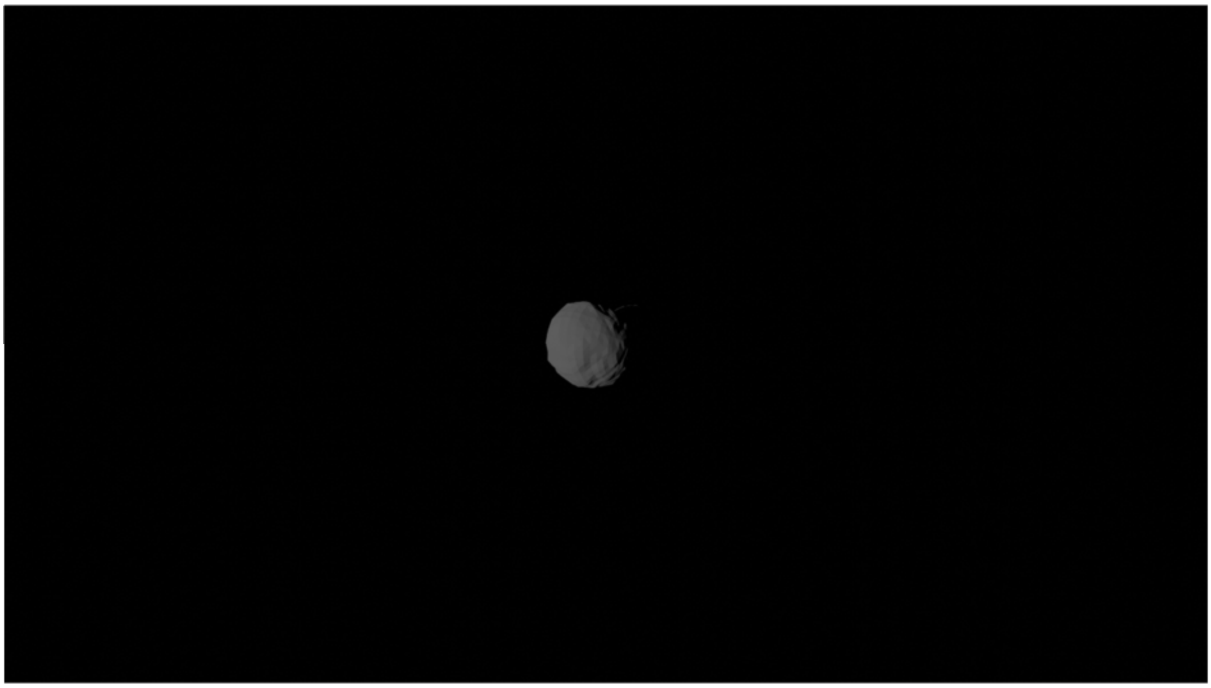


Figure 5-20 - A simulated HW1 in configuration A at $12R_{max}$ at an angular station equal to 25.5deg

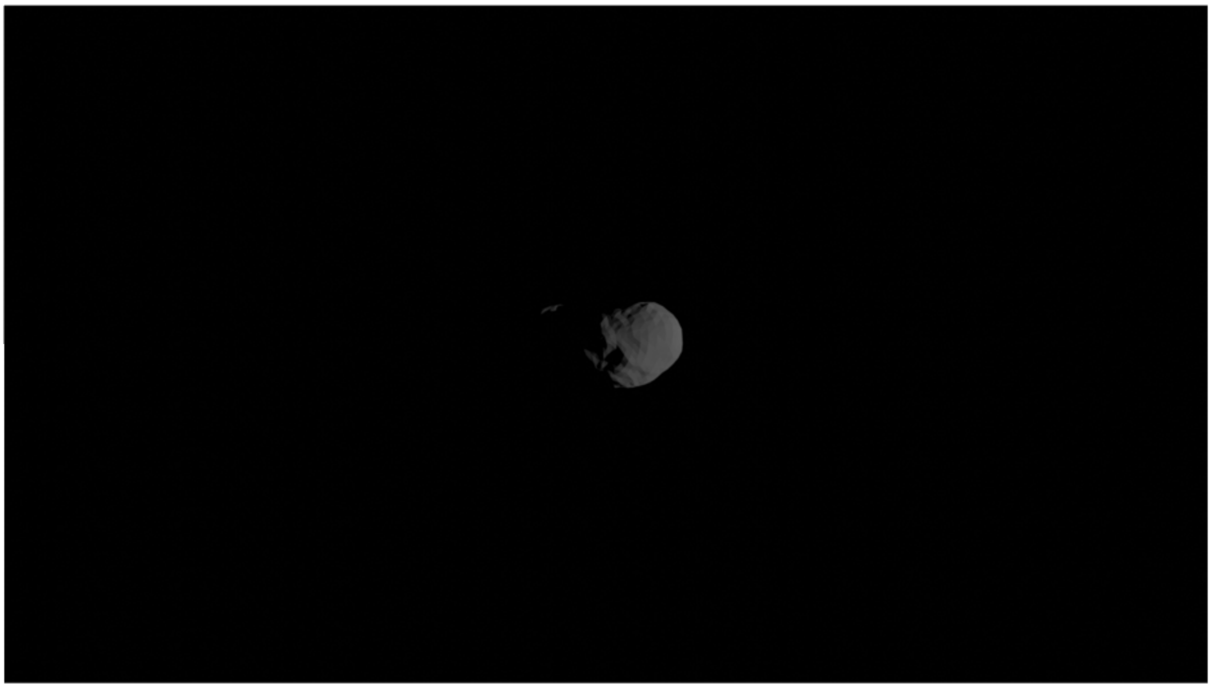


Figure 5-21 - A simulated HW1 in configuration A at $12R_{max}$ at an angular station equal to 315deg

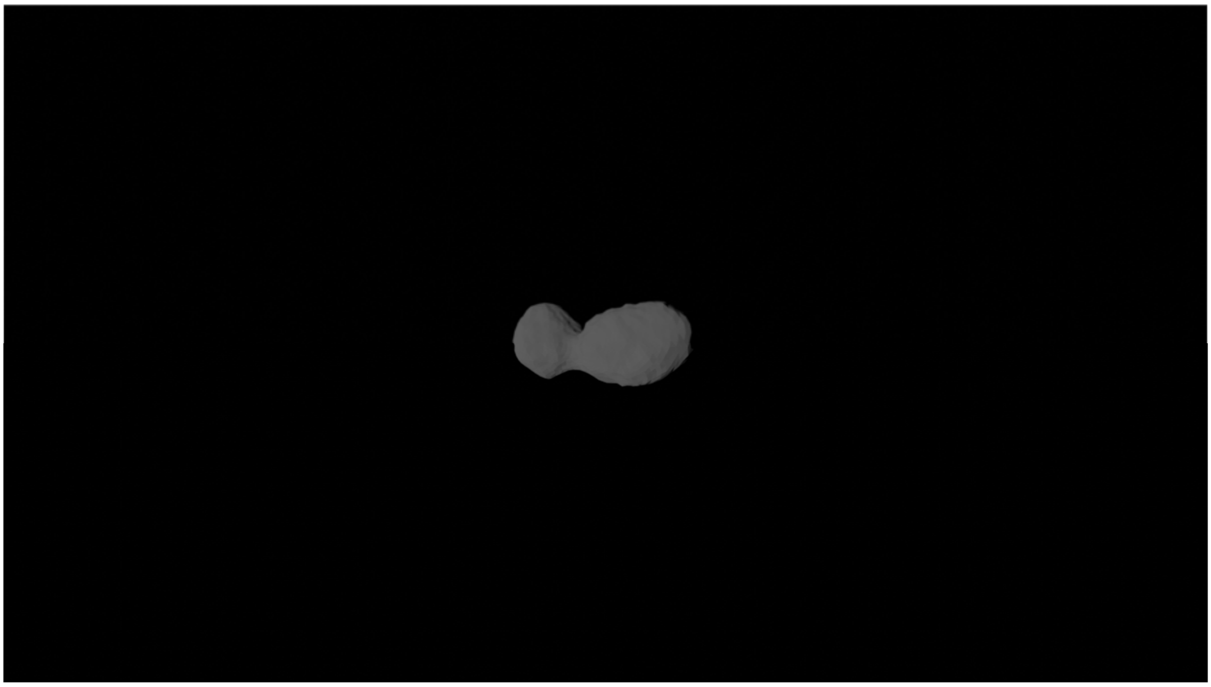


Figure 5-22 - A simulated HW1 in configuration B at $12R_{max}$ at an angular station equal to 25.5deg

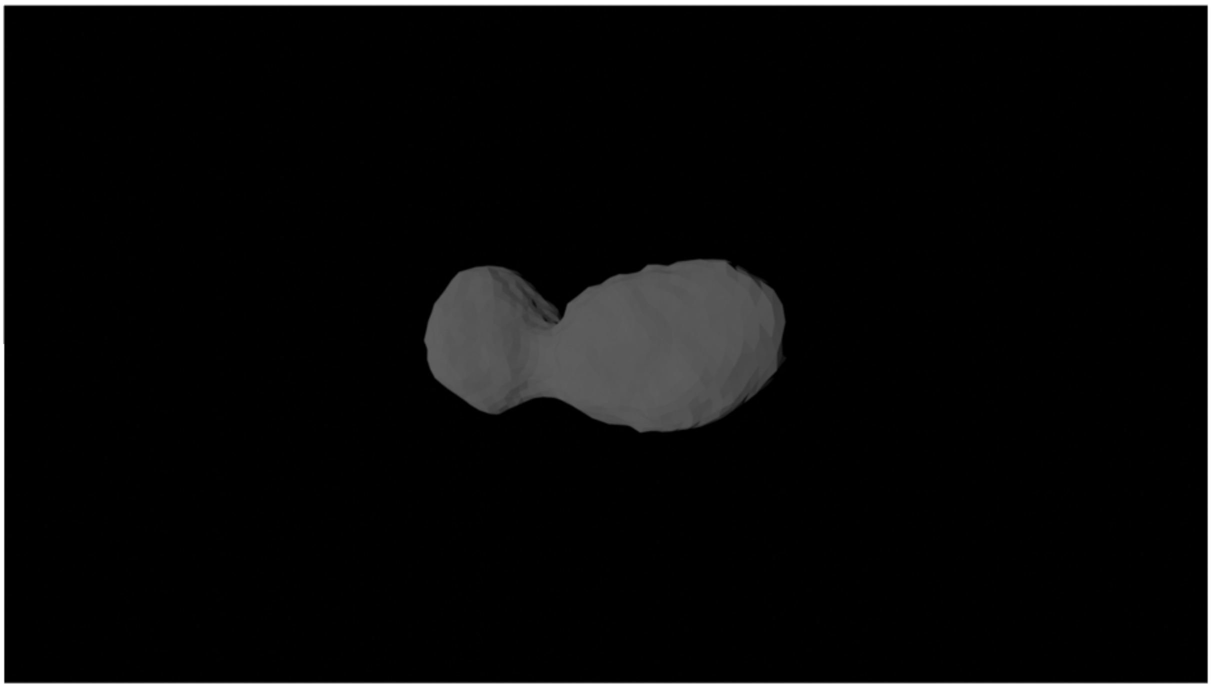


Figure 5-23 - A simulated HW1 in configuration B at $6R_{max}$ at an angular station equal to 25.5deg

5.2.3.4 Results

The The results were generated for Harris features [84] and SURF features [85], for Bennu, and for HW1 in two configurations. All the asteroids have been oriented with their spin axis normal to the X^b-Y^b plane.

Harris' implementation in MATLAB is representative of pivot led features, while SURF's represents pivot independent features. Due to this mechanism, the presence of good features within the scene raises the lower threshold for Harris corners, pruning bad features. If the general distribution is heavily shifted towards containing lower-quality features, the good quality features could induce a significant reduction of features available for operations, up to extreme case of insufficient features for the task.

For pivot led features, as the probe progresses from direct Solar illumination towards the terminator it is possible to observe two competing mechanisms. On one side the shadow generally creates more complex local patterns, inducing a larger number of stronger features. On the other side, these stronger features lead to the rejection of lower quality features, reducing the number of available features.

Pivot independent features only see the increase in good quality features, and therefore generally increase as the probe moves towards the terminator and decrease as the probe moves away from it.

The threshold T_S was set to 1000, while the threshold T_H was set to both 0.001 and 0.4. From the even-numbered figures from Figure 5-24 to Figure 5-34 it is immediate to observe that the higher the T_H, the larger portion of the distribution is removed, therefore the more susceptible the process becomes to variations in the pivot.

It can also be noticed that the value of the best metric score is a weak function of the distance, but the amount of available features is reduced by a factor of roughly 3 for Bennu and 2 for HW1.

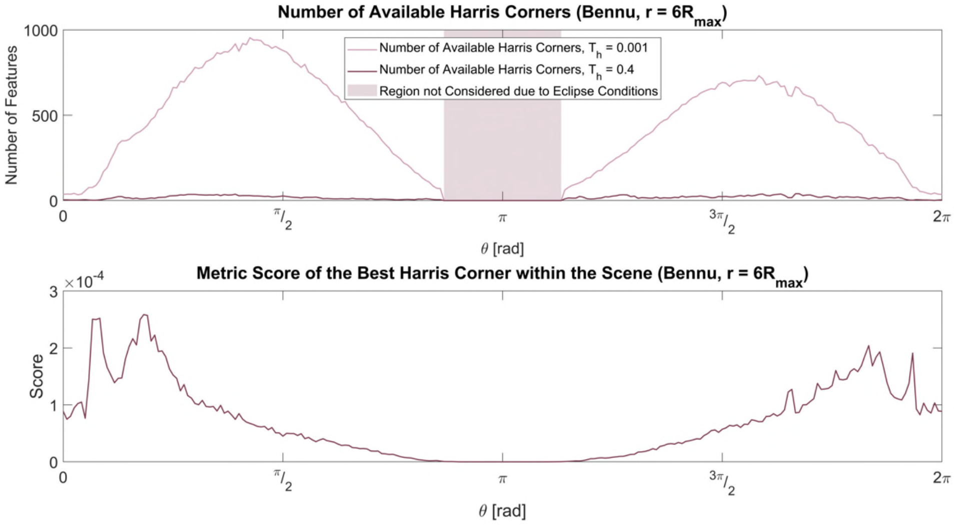


Figure 5-24 - Harris features around Bennu ($r = 6R_{\mathrm{max}}$)

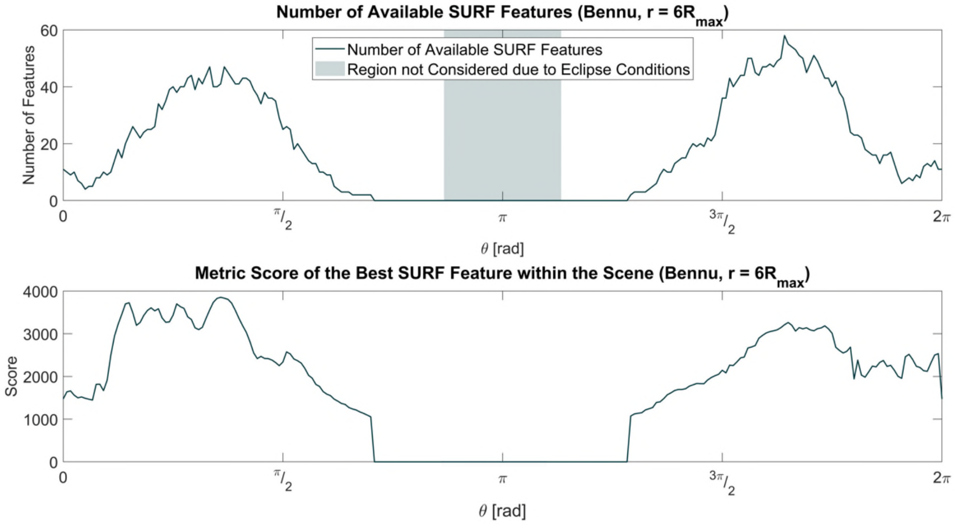


Figure 5-25 - SURF features around Bennu ($r = 6R_{\mathrm{max}}$)

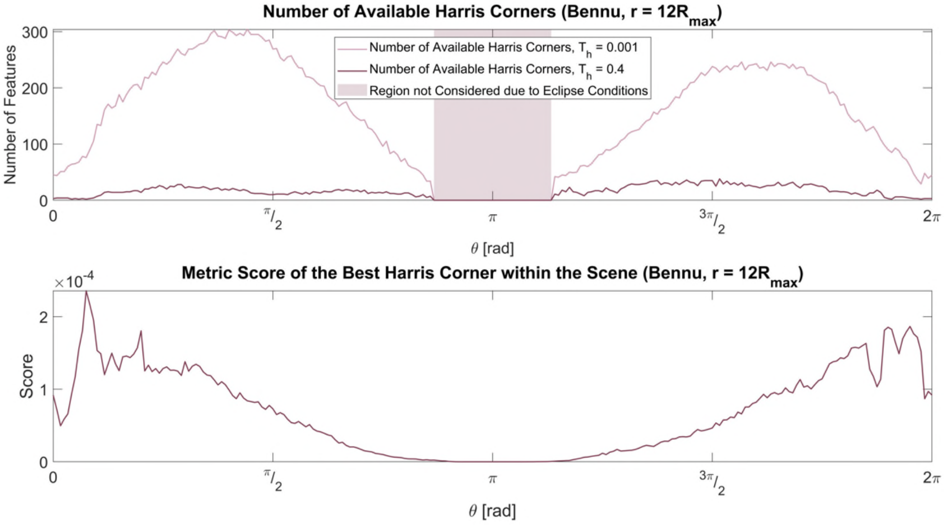


Figure 5-26 - Harris features around Bennu ($r = 12R_{\mathbf{max}}$)

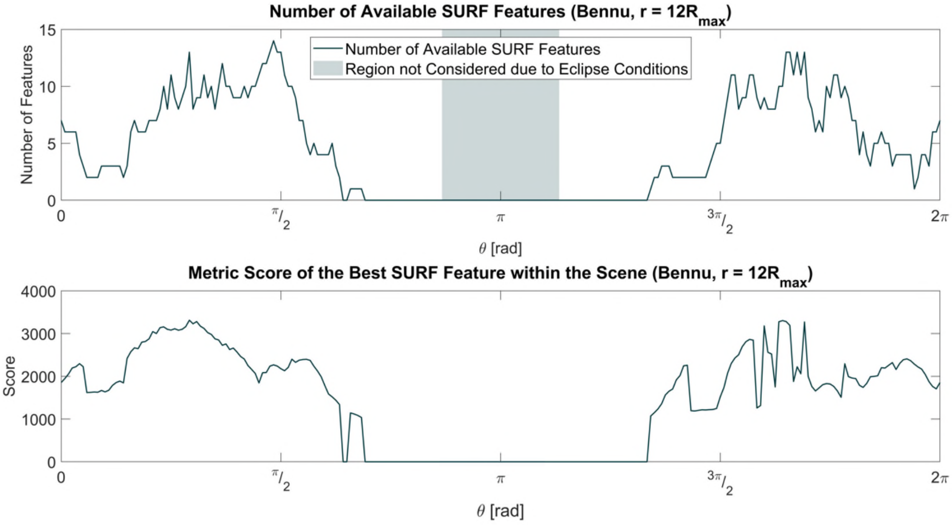


Figure 5-27 - SURF features around Bennu ($r = 12R_{\mathbf{max}}$)

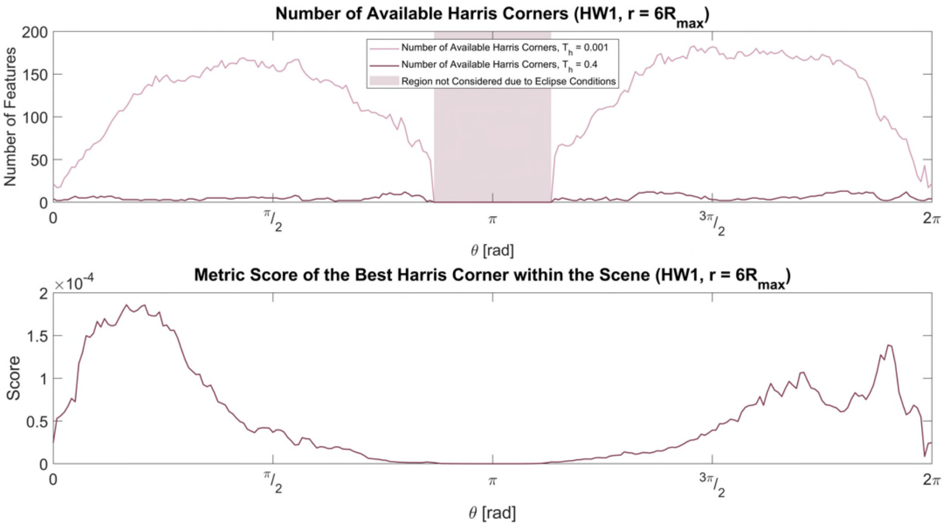


Figure 5-28 - Harris features around HW1 ($r = 6R_{\max}$, configuration A)

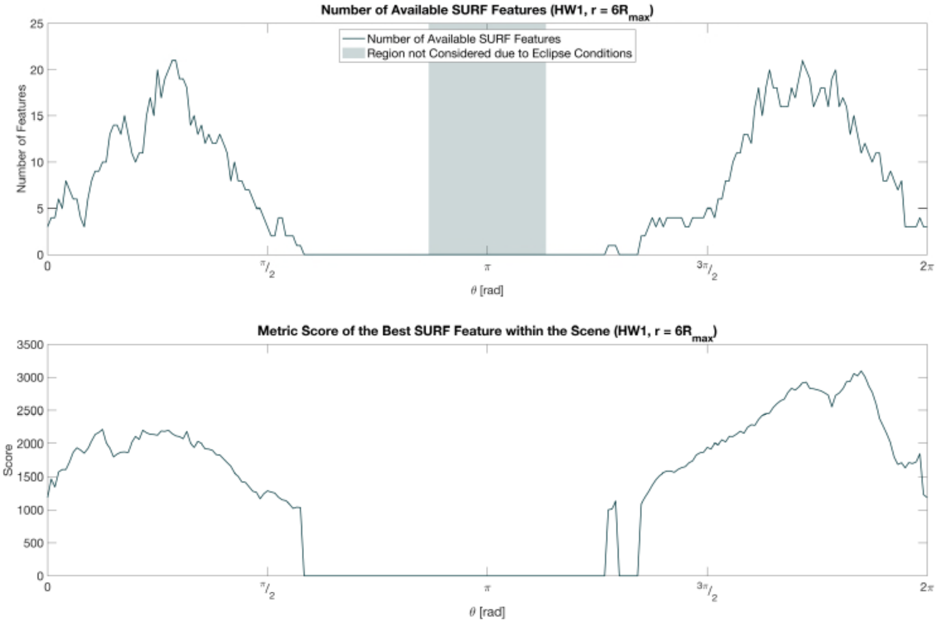


Figure 5-29 - SURF features around HW1 ($r = 6R_{\max}$, configuration A)

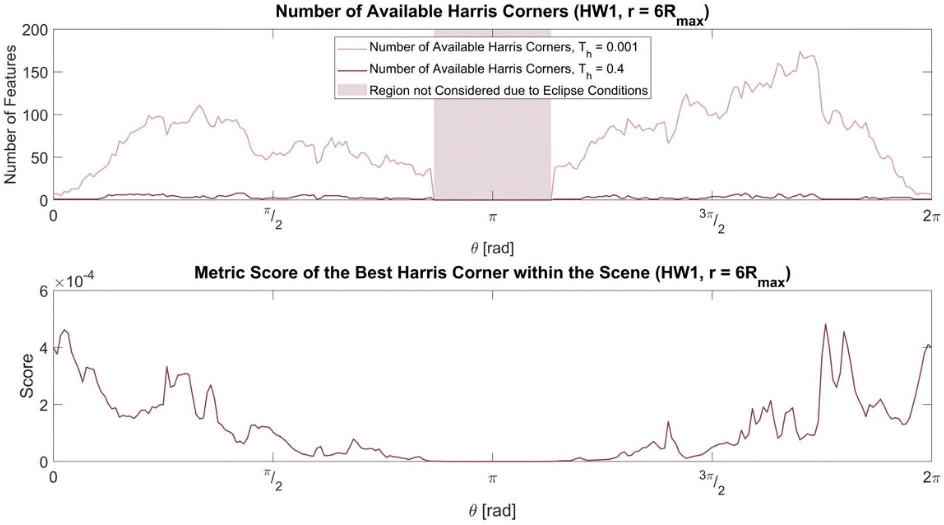


Figure 5-30 - Harris features around HW1 ($r = 6R_{\max}$, configuration B)

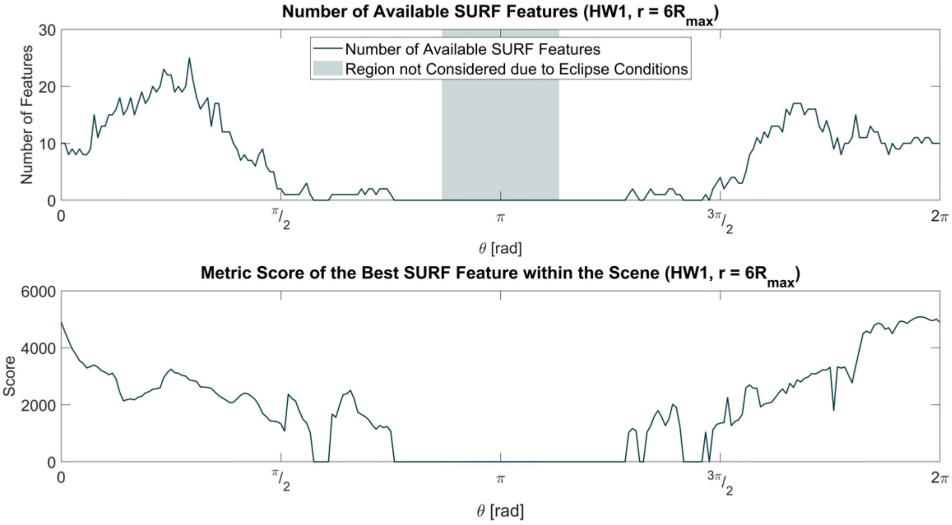


Figure 5-31 - SURF features around HW1 ($r = 6R_{\max}$, configuration B)

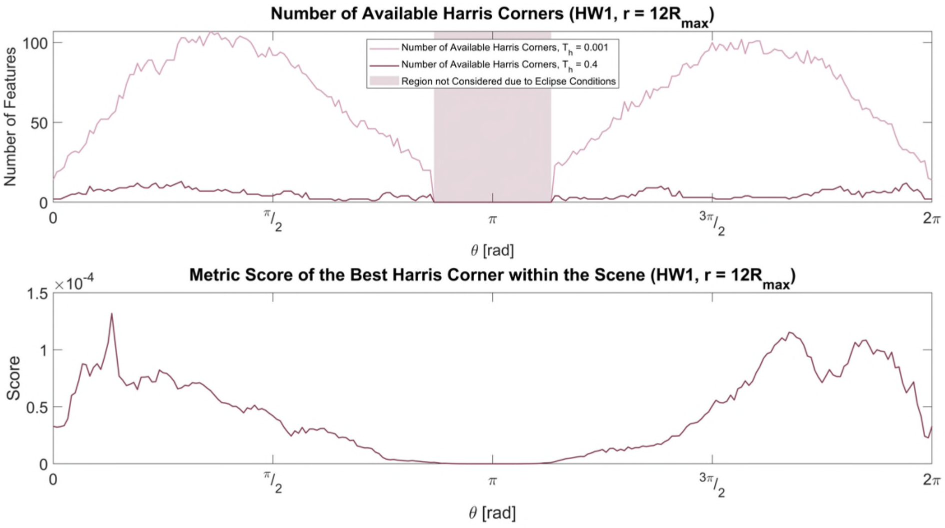


Figure 5-32 - Harris features around HW1 ($r = 12R_{\max}$, configuration A)

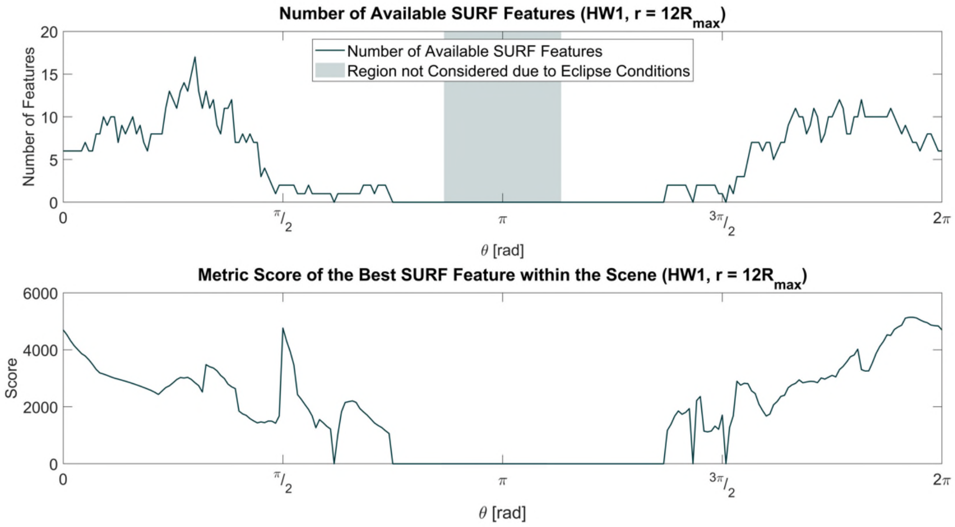


Figure 5-33 - Harris features around HW1 ($r = 12R_{\max}$, configuration A)

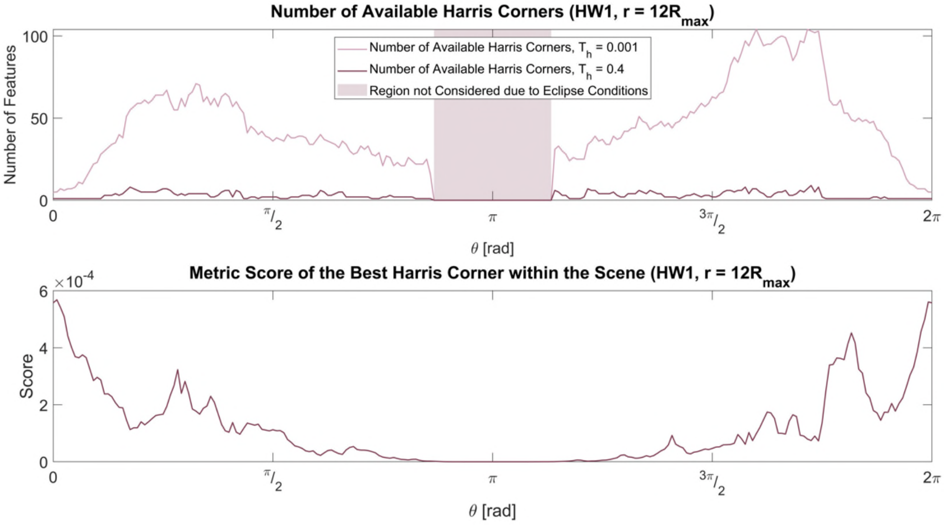


Figure 5-34 - Harris features around HW1 ($r = 12R_{\max}$, configuration B)

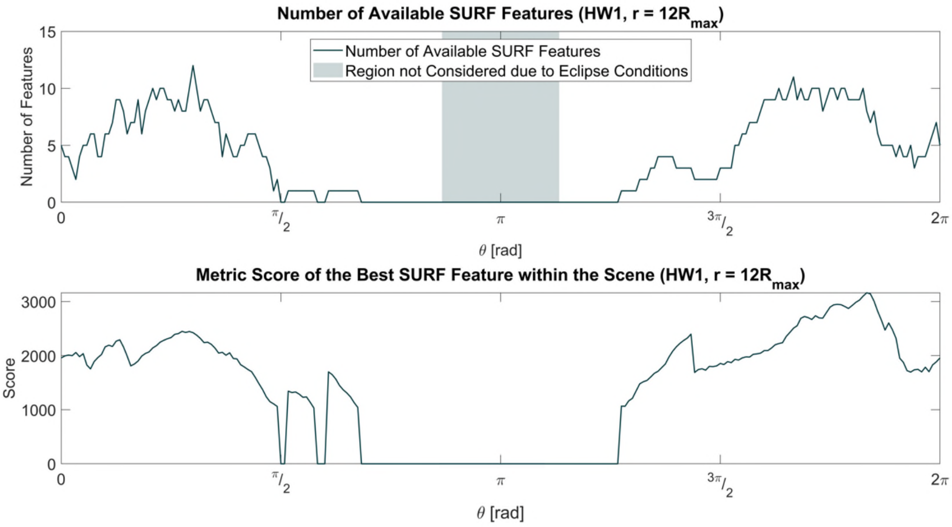


Figure 5-35 - SURF features around HW1 ($r = 12R_{\max}$, configuration B)

The implications of this process are that for pivot led features these mechanisms require a case-by-case evaluation. In this instance, the Harris' Pivot score peaks close to direct Solar illumination conditions and the best feature is a corner on the limb of the asteroid (see Figure 5-36). This potentially implies that at this distance, or in this configuration, the features on the ground become less relevant, and the stark contrast between night and the lit face of the asteroid provides the dominant set of features leading the Pivot-led process.

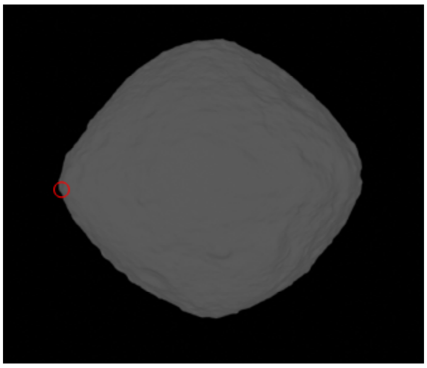


Figure 5-36 - The location of the strongest Harris feature for Bennu at 1.5 deg, $6R_{\max}$

However, it can still be appreciated how the local geomorphology induces some pivot metric spikes in the HW1 and HW1-Configuration B cases. In some cases, these correlate to a drop in the number of available features, while in others this does not seem to disturb the underlying distribution. In the first case, the distribution of features has to be shifted towards low quality features, and therefore the strong pivot introduces pruning. In the second case, the feature distribution relies on strong features with high metric scores and is not affected by the Pivot score increase.

HW1 in the first configuration shows results similar to Bennu, as the target generally appears as a single-lobe object. However, HW1 in configuration B presents peaks associated to both lobes, and this can be observed in the associated perception field curve. As a rule of thumb, it could be assumed that it is possible to observe one peak per visible lobe.

5.2.3.5 Discussion

By fixing a sensor's attitude law and internal model, the response of a perception model can be considered a function of the operational conditions and the sensor's position. This means that for fixed environmental conditions and sensor characteristics this response is a function of the sensor's position alone. This section has demonstrated that by moving the sensor around, the response can be mapped as if it were a physical field, similarly to a gravitational one. The method suggested is an application of perception fields - i.e., treating some quantifiable property of onboard perception models as readings of a physical field. Perception fields are seen as a key enabler of robust camera-only autonomous spacecraft operations, and thus, cost-effective space

Considering this, the navigation charts described in section 4.3.4 can be interpreted as a particular perception field, induced by a specific estimation (or perception) model. The primary components of perception fields appear to be a function of illumination. Since illumination does not depend on the target, it can be considered a strong invariant element for navigation.

Feature detectors return features which are essential in most computer vision pipelines, such as optical navigation. Therefore, ensuring that features are always detectable and of good quality is essential when computer vision operations are part of the mission plan.

This work presented some criteria to select a feature detector given information about the Sun phase angle. In general, optimality is a strong function of the mission objectives and platform. Here, however, it will be defined in terms of feature availability. This becomes even more critical in an architecture where monocular cameras represent the only sensing element, mediated through software payload.

In particular, results have shown that for pivot led features there are two competing mechanisms determining the number of available features and that it is possible to identify three regions:

Regions where extremely good features can be detected are generally more feature-rich, and that drives up the number of available features. However, in some instances, the good feature might be surrounded by low-quality features (e.g., a high contrast corner in the limb in direct solar illumination conditions) and drive down the number of available features. These effects can be amplified or reduced by setting an appropriate threshold

For pivot independent features, feature availability is directly proportional to the quality of the best feature in the scene

The approach discussed can serve two scopes:

a) Select optimal visual features given the Sun phase angle and some information about the target's shape.
b) Reconstruct the position of the probe from the time series of pivot independent and pivot led features acquired with known frequency.

6 CONCLUSIONS AND FUTURE WORK

This research aimed to identify and explore pathways to develop navigation methods assisting soft landings on Near-Earth Objects performed by resource constrained platforms. Landing on the surface of these distant celestial bodies is extremely challenging and consequently requires an elevated degree of autonomy. As noted by Amini et al. [174], autonomy is a necessity for spacecraft operating in unknown environments when the timescale of a task is orders of magnitudes smaller than the time required for the signal from a human operator to reach the probe. Some aspects of landing on NEOs can be rather benign from a navigation point of view: the low gravity implies relaxed timescales of motion, and low velocities for ballistic trajectories. However, NEOs are often found at distances from Earth having one-way light times of the order of magnitude of minutes. Therefore, in critical phases such as landings, robust autonomous navigation remains a strong functional requirement due to the substantial ranges at play. Moreover, robust autonomous navigation is a design choice also pushed by economic consideration, as handling missions through human controllers induces significant operational costs.

Inherently, resource constrained spacecraft cannot afford to employ complex sensor packages or deploy computationally intensive algorithms. This intrinsic limitation on the on-board resources which can be deployed is critical for missions targeting NEOs. NEOs can hardly be characterised from the ground and present an extreme variety of characteristics that therefore require to be handled on-line. Due to the constrain on resources, safety and robustness must be introduced in navigation processes through mechanisms that do not rely on design complexity.

Addressing these requirements by developing ad-hoc solutions for each target would imply massive R&D costs for each mission and would be inconsistent with the paradigm explored in this thesis.

The hypothesis addressed in this thesis is that it is possible to manage the conflicting constraints of resource constrained platforms through two parallel processes. The first process is the appropriate selection of the sensing payload. Within the scope of this work, this sensing capability was limited to a single visible camera, i.e., implementing a (monocular) camera-only approach.

The second process is the identification of abstract observables or patterns enabling vision-based processes that change very little between different targets – addressed as “invariants” throughout the thesis. Computational cognition processes operating on invariants have been labelled “Invariant-oriented perception”.

Ultimately, the core research question investigated in this thesis is whether camera-only invariant-oriented perception is a viable strategy for navigation. A positive answer would enable complex operations even for probes with limited on-board capabilities, like CubeSats. Through this approach to navigation, the introduction of robustness would be shifted away from the addition of computational methods towards designs that are intrinsically stable to the coupling of factors.

To illustrate and investigate these principles, multiple results and tools have been studied and built capitalising on the underlying arguments mentioned above. A central contribution was constructing the modular prototyping environment named THALASSA. This software package, developed from scratch throughout the doctorate, allowed to perform complex testing procedures over simulated visual data. The use of THALASSA provided the ability to study the suitability of various invariants introduced at different stages of various navigation processes and analyse the effects of invariants on the overall process.

Limited results have been investigated, yet they have showed promising results. In particular, these concern the applicability of at least one of the core concepts (perception fields) to the most common types of NEOs shapes, spin-tops and contact binaries. These results have opened interesting new research questions and have led to insights which could have important implications on astronautical engineering, such as standardised navigation packages for low-cost spacecraft.

The first invariant analysed is associated to the overall shape of the target asteroid. The tests involved using a light localisation method designed for targets with an ellipsoidal shape to perform trajectory reconstruction around convex asteroids made up by a single lobe. By investigating the OpNav algorithm it has been observed that the estimation uncertainty is a strong function of the relative pose of the Sun, Bennu, and the autonomous platform.

This implies that optimal perception can be achieved by modulating the acquisition patterns to target only optimal zones. Implementing this OpNav approach demonstrated the ability to achieve navigation accuracies up to an order of magnitude of 0.01 orbital radii. As the shape exploited in the algorithm does not change, this approach can be considered invariant-oriented.

The acquisition regions performing poorly emerge from interactions between the estimation model, illumination, and the shape of the body. For Bennu, these appeared to mainly be the regions in the proximity of eclipse conditions, direct Sun illumination, and the planes generated by two of the major axes of the target. For both these first two conditions of interest, a Sun sensor would suffice to provide the information to account for these data in the workflow. For a three-axis ellipsoid, instead, these would only depend on the Sun-phase angle. The latter, however, could be retrieved from a perception-field like approach.

At this stage, the development and the heuristic corrections implemented in this first set of results require a monocular visible camera, a Sun sensor, and the ephemerides of the target. However, the behaviour of illumination can be expected to remain universally similar. This means that a robust machine learning approach extracting the angular distance from the terminator could consistently provide a first bearing estimation. Coupling the bearing with information on the local asymmetries - generally abounding on asteroids, a camera and the target's ephemerides could suffice to perform navigation duties. Additional future work concerns studying the statistical distribution of errors returned by this model, which tends to be non-normally distributed. This implies that to perform further Monte Carlo analyses over equivalent systems, suitable statistical populations must be determined.

Moving down from ranges where the full target can be imaged towards the ground, within camera-only navigation the navigation observables must transition to features on the local terrain. One of the strongest drawbacks of navigation techniques mediated through monocular cameras is scale ambiguity. This means that motion can be reconstructed only up to a scale factor, unless additional sensors or estimations provides context information resolving the ambiguity.

In limb-based navigation, this information is represented by the asteroid size, and its ephemerides. However, the reconstructed trajectory could eventually be propagated to the point where observables are changed, thus resolving, at least initially, this core issue for the subsequent close-range phase.

Closer to the ground, scale information in camera-only operations could be introduced again through elements with known sizes. In line with the need for invariance, in this work these are assumed to be standardised deployable infrastructures augmenting the environment to a cooperative state. Thus, the second set of results reports the implications on perception of such invariant visual elements. For landings on small celestial objects this approach was pioneered by the Hayabusa probes [45]. Within the scope of this thesis, this idea has been explored by analysing the effects of the introduction of artificial passive landmarks on the surface of a small asteroid on visual odometry performed by a landing probe. These landmarks are labelled “NAV-Landmarks”, which stands for Noncooperative Artificial Visual Landmarks, and have specific design requirements. The effect of NAV-Landmarks has been studied on a camera-only probe at a range comprised between two and one kilometres, and again employs only a visible camera to navigate. An early version of THALASSA was used in this case to simulate the introduction of the markers on the surface of the Bennu model.

The results of the NAV-Landmarks study were promising. Indeed, it has been observed that the landmarks improve the unscaled results of visual odometry in most tested cases. More research on the topic is however needed to clarify and explain some unexpected behaviour. Moreover, issues related to perception, as the influence of extreme values in the scene were highlighted. Subsequent research has therefore focused on dealing with these issues and evaluating the optimal observation conditions for a feature detector. Future work will address implementing scale reconstruction and comparing the performance of these “physical” optimizers with computational optimizers.

In particular, the analysis of values inducing a strong response is introduced through a practical application. This use case is the development of a set of EO countermeasures disrupting visual based interpretation processes on-board autonomous drones. These countermeasures are called Countermeasures Leveraging Optical Attractor Kits (CLOAKs), where an optical attractor is defined in this work as the collection of pixels inducing an exceptionally strong response for a target feature detector. To properly discuss CLOAKs, it was necessary to introduce a wealth of ancillary connected to extrema values in computer vision, and to search methods to identify these extrema. Within this work this search was operated through genetic algorithms.

There are multiple lines of work connected to extreme values in perception that are expected to be further developed and analysed in the future. The first one concerns retrieving optimization-based optical attractors for all the feature detectors for which this problem is well-posed. The second one is structuring an accurate mathematical search of the highest possible extrema, to validate the genetic search, or, eventually, find improved solutions. The third one is developing a score to understand how well the given feature detector operates in an environment. Lastly, the fourth one is examining if from data on the motion and the observable payload and of the target it is possible to determine the structure of the on-board AI.

Building upon findings from NAV-Landmarks and optical attractors, the last research topic explored in the thesis introduced navigation strategies relying on quantitative distributions of visual. In particular, the hypothesis tested was that trends in the quantity of available optical features can be correlated to variations in the angular distance from a source of illumination. Initial results obtained for circular, equatorial orbits, demonstrated that the quantity of structural features detected around small celestial objects is not random, but rather, follows a distribution, primarily induced by the Sun phase angle, and only secondly by the local topology. The number of available features therefore defines instant by instant a scalar field labelled “Perception field”. This allowed to validate that sequential observations of the collective features’ behaviour can be exploited to assist either in the retrieval of the probe’s localisation information, or in the selection of an optimal detector. Therefore, it appears legitimate to further pursue the development of methods where these readings are used to complement or improve approaches considering features as distinct units requiring accurate labelling and tracking.

Future work connected to perception fields will involve generating response maps using spherical acquisition patterns, as an extension of circular ones. The information contained in these maps could then be fed to machine learning algorithms, which would be able to reconstruct the altitude and the pose of the sensor by comparing the number of available features, the number of available surface pixels and the ephemerides of the target. Ultimately, this would potentially enable a camera-only, target agnostic solution. Moreover, the response maps could be used as charts to select the optimal feature detector given a trajectory and illumination conditions.

The process through which perception fields emerge also has another implication. When the response of perception is a function of three motions interacting, and when two of these are known (e.g., through ephemerides) it is possible to filter the priors out from the resulting time series, ultimately relaying navigation information.

Perception fields represent the last use case, and contribution, of the thesis, and define an important steppingstone in the study of optical invariants. The reason for their importance lies in the fact that every perception model could be used to define a perception field. Therefore, invariant elements could be studied as an emergent property in this secondary space. Moreover, this field structure, if properly constructed, can be well-defined algebraically, thus allowing the use of standardised methods of analysis borrowed from other disciplines.

All the methods and scenarios explored in this work fit within a broad ecosystem of solutions and ideas. The common purpose behind the introduction of these concepts and tools is laying the groundwork towards the appropriate selection of optical features enabling camera-only solutions for autonomous navigation. To restrict sensing to such a simple passive sensor, the underlying perception models and what is observed by them need to be carefully tailored to elements that are largely available, unambiguous and provide multiple layers of information, about the sensing platform and its operational environment. To avoid a costly development process for every new probe, an additional requirement would be for these elements to be shared among most of the plausible operational targets.

Within this work elements that possessing all the characteristics above are labelled invariants. The perception of invariants can serve either as standalone solutions or be the basis for software with higher data-driven decision-making capabilities, such as machine or deep learning.

Thus, camera-only autonomous probes appear to be an important enabling technology for deep space operations. The core reasons for this importance are:

- From a commercial perspective, camera-only autonomy greatly reduces development, deployment, and operational costs. Moreover, these methods would be compatible with small low-cost probes, allowing smaller players and developing countries to gain access to the surface of distant small objects.

- Camera-only methods introduce resilience against the failure of more complex sensing mechanisms.

However, the implications of these approaches could extend well beyond the space field. Today’s world is increasingly moving towards full connection and integration of cyber and physical elements. The computing requirements for deep space operations are not too dissimilar from those of small computational units operating as standalone units in unstructured environments, as for smart city devices, wearables, or drones. When moving computing to the edge, in paradigms such as Internet of Things these technologies could play a pivotal role in making embodied intelligence pervasive and available everywhere.

6.1 Future Work

All the Future work appears to be aligned along three directions. These are:

1. Improving THALASSA and the other computational tools developed during the thesis and structure them into an open-source project.
2. Extending the knowledge on the methods and approaches presented in the thesis and define tests to bring them to higher TRLs.
3. Introducing novel technologies, algorithms and methods capitalising on the criteria and design tools developed in the thesis.

Improving the developed computational tools will be pursued by operating mostly on three factors:

- Robustness: including computational safeguards against computational failures and ambiguous cases, and monitors for data quality and model performance quality.
- Accessibility: generating a better UI, where the user can interact at a higher level, streamline the integration between computer graphics generation and analytics engine, move all the parts of the pipeline to free software in line with an open science philosophy.
- Functionalities: include additional modules and models within the software. These could be less approximate gravitational potential models (polyhedron, …), a repository of various flight heritage camera models, stronger filtering capabilities (e.g., Kalman), a collection of plug-and-play navigation and pose estimation methods, hazard detection modules (e.g., slope identifier or dangerous boulders identifier). Potentially this could be done in partnership with other entities which could supply information and code, and overall structure the project as a large-scale open source one.

Most of the elements presented within the thesis are laying the groundwork for further developments at higher TRLs. There are technologies which are currently missing to deploy a fully camera-only navigation toolbox; yet to deploy any of the functions described in the thesis even before that stage it will be at least necessary to perform:

- Obtaining highly accurate covariances through large scale Monte Carlo analyses, considering more parameters, operated on improved phenomena modelling and larger samples. For example, analysing navigation charts at different distances, with different cameras models and different outlier rejection mechanisms.
- Integrate models and tools within complete pipelines, and for each pipeline and set of target conditions return means, covariances and other statistical descriptors of the errors that can be used for testing and simulation.
- Perform sensitivity analyses for all the experiments, analyse the implications and the propagation of design choices and uncertainties from various sources through consider covariances, and retrieve configurations which optimise performance indicators and figures of merit of interest.
- Perform searches of optimal features for all the existing feature detectors, potentially using also different search and optimisation methods.
- For each navigation pipeline, obtain statistical descriptors about reliability, failure rates, and failure modes, and continuously try to introduce safeguards against them from a systemic perspective.
- Experiment with increasingly less approximate models and try to obtain charts detailing computational load required vs. accuracy. These charts could be useful to rapidly identify the best set of tools given a mission's constraints and objectives.

To obtain a fully camera-only navigation suite it will be necessary to develop multiple novel technologies capable of either replacing capabilities currently handled by other sensors (e.g., a vision-only rangefinder instead of a laser one).

Some ideas in that direction are:

- Test whether the distance from the object can be predicted as a perception field employing the number of visible pixels and information about the Sun phase angle.

- Explore the possibility to employ ground-generated shape models to obtain coarse perception fields that can be used in orbit with little corrections.
- Identifying flexible mathematical models with a small set of parameters offering a good compromise between the computational simplicity of ellipsoids and the accuracy of complex templates.
- Trying to understand whether NAV-Landmarks could be created as origami-like structures, and what would their constraints be in that sense (both for transporting and detection).
- Exploring the usage of Deep Learning algorithms for tasks ranging from non-geometric visual odometry to object detection focused on specific observables.
- Adding astrodynamical considerations to the studies, such as: are there orbits maxisiming/optimising the quality of available features for a task?
- Exploring the usage of novel developments in the camera world, ranging from AI-mediated lensless cameras [175] to event-based cameras [176].
- Exploring the usage of novel mission configurations, for example, employing tandems or swarms of satellites, either communicating or aware of each other through other sensing mechanisms.

BIBLIOGRAPHY

1. Klaus K. Near Earth Object Characterization, Exploration & Exploitation. Space Resources Roundtable VI Program and Abstracts. 2004. p. 25.

2. Martin G. NewSpace: The emerging commercial space industry. 2017;

3. Salt D. NewSpace-delivering onthedream. Acta Astronautica. Elsevier Ltd; 1 December 2013; 92(2): 178–186. Available at: DOI:10.1016/j.actaastro.2012.08.020 (Accessed: 7 May 2021)

4. Scheeres DJ. Orbital Motion in Strongly Perturbed Environments. Orbital Motion in Strongly Perturbed Environments. 2012. Available at: DOI:10.1007/978-3-642-03256-1

5. JPL. NEO Basics. Center for Near Earth Object Studies of the Jet Propulsion Laboratory. 2017. Available at: https://cneos.jpl.nasa.gov/about/neo_groups.html (Accessed: 9 May 2021)

6. European Space Agency. Near-Earth Objects - NEO Segment. Available at: http://www.esa.int/Safety_Security/Near-Earth_Objects_-_NEO_Segment (Accessed: 9 May 2021)

7. Lupishko DF., di Martino M., Binzel RP. Near-Earth objects as principal impactors of the Earth: Physical properties and sources of origin. Proceedings of the International Astronomical Union. Cambridge University Press; 2006; 2(S236): 251–260.

8. Trilling DE., Valdes F., Allen L., James D., Fuentes C., Herrera D., et al. The size distribution of near-earth objects larger than 10 m. The Astronomical Journal. IOP Publishing; 2017; 154(4): 170.

9. Dotson JL., Mathias D., Ostrowski D. Density Distribution of Near Earth Asteroids. AAS/Division for Planetary Sciences Meeting Abstracts# 48. 2016. pp. 221–222.

10. Pajuelo M., Birlan M., Carry B., DeMeo FE., Binzel RP., Berthier J. Spectral properties of binary asteroids. Monthly Notices of the Royal Astronomical Society. Oxford University Press; 2018; 477(4): 5590–5604.

11. DeMeo FE., Carry B. Solar System evolution from compositional mapping of the asteroid belt. Nature. Nature Publishing Group; 2014; 505(7485): 629–634.

12. Gregg J. Space Mining. The Cosmos Economy. Springer; 2021. pp. 145–152.

13. Rumpf CM., Lewis HG., Atkinson PM. Asteroid impact effects and their immediate hazards for human populations. Geophysical Research Letters. Wiley Online Library; 2017; 44(8): 3433–3440.

14. Chapman CR., Mulligan BM. NEO impact hazard scales in the context of other hazard scales. AAS/Division for Planetary Sciences Meeting Abstracts# 34. 2002. pp. 14–15.

15. Gal-Edd J., Cheuvront A. The OSIRIS-REx Asteroid Sample Return Mission operations design. IEEE Aerospace Conference Proceedings. 2015. Available at: DOI:10.1109/AERO.2015.7118883

16. Yoshikawa M., Kawaguchi J., Fujiwara A., Tsuchiyama A. Hayabusa sample return mission. Asteroids IV. 2015. Available at: DOI:10.2458/azu_uapress_9780816532131-ch021

17. Cheng AF., Rivkin AS., Michel P., Atchison J., Barnouin O., Benner L., et al. AIDA DART asteroid deflection test: Planetary defense and science objectives. Planetary and Space Science. 2018; Available at: DOI:10.1016/j.pss.2018.02.015

18. Sitepu E., Cullen DC. Towards a 3U CubeSat-payload for in-situ resource utilisation demonstration at C-type near earth asteroids. Proceedings of the International Astronautical Congress, IAC. 2018.

19. Zacny K., Chu P., Paulsen G., Hedlund M., Mellerowicz B., Indyk S., et al. Asteroids: Anchoring and sample acquisition approaches in support of science, exploration, and in situ resource utilization. Asteroids: Prospective Energy and Material Resources. 2013. Available at: DOI:10.1007/978-3-642-39244-3_12

20. Ross SD. Near-Earth Asteroid Mining Space Industry Report. 2001.

21. Yun S., Lee YJ., Sung S. IMU/Vision/Lidar integrated navigation system in GNSS denied environments. 2013 IEEE Aerospace Conference. 2013. pp. 1–10.

22. Tiemann J., Schweikowski F., Wietfeld C. Design of an UWB indoor-positioning system for UAV navigation in GNSS-denied environments. 2015 international conference on indoor positioning and indoor navigation (IPIN). 2015. pp. 1–7.

23. Duckworth GL., Baranoski EJ. Navigation in GNSS-denied environments: Signals of opportunity and beacons. Proceedings of the NATO Research and Technology Organization (RTO) Sensors and Technology Panel (SET) Symposium. 2007.

24. Sullivan J., Lovell TA., D’Amico S. Angles-only navigation for autonomous on-orbit space situational awareness applications. AAS/AIAA Astrodynamics Specialist Conference, Snowbird, UT. 2018.

25. Parreira B., Vasconcelos JF., Oliveira R., Caramagno A., Motrena P., Dinis J., et al. Performance Assessment of Vision Based Hazard Avoidance During Lunar and Martian Landing. Proceedings of 7th International Planetary Probe WorNshop. Barcelona, Spain. 2010;

26. Cheng Y. Real-time surface slope estimation by homography alignment for spacecraft safe landing. 2010 IEEE International Conference on Robotics and Automation. IEEE; 2010. pp. 2280–2286.

27. Hoag DG. Apollo Guidance and Navigation-A Problem in Man and Machine Integration. second AIAA-NASA Manned Spaceflight Meeting. 1963.

28. Goodfellow I., Bengio Y., Courville A. Deep learning. MIT press; 2016.

29. Cheng AF., Rivkin AS., Michel P., Atchison J., Barnouin O., Benner L., et al. AIDA DART asteroid deflection test: Planetary defense and science objectives. Planetary and Space Science. Elsevier; 2018; 157: 104–115.

30. Reddy VS. The spacex effect. New Space. Mary Ann Liebert, Inc. 140 Huguenot Street, 3rd Floor New Rochelle, NY 10801 USA; 2018; 6(2): 125–134.

31. Di Fraia MZ., Chermak L., Cuartielles J-P., Felicetti L., Scannapieco AF. NAV-Landmarks: Deployable 3D Infrastructures to Enable CubeSats Navigation Near Asteroids. IEEE Aerospace Conference Proceedings. 2020. Available at: DOI:10.1109/AERO47225.2020.9172720

32. Di Fraia MZ., Chermak L. Countermeasure Leveraging Optical Attractor Kits (CLOAK): Interpretational disruption of a visual-based workflow. Proceedings of SPIE - The International Society for Optical Engineering. 2020. Available at: DOI:10.1117/12.2574099

33. Di Fraia MZ., Feetham L., Felicetti L., Cuartielles J-P., Chermak L. Perception fields: analysing distributions of optical features as a proximity navigation tool for autonomous probes around asteroids. 8th IEEE International Workshop on Metrology for AeroSpace, MetroAeroSpace 2021- Proceedings.

34. Di Fraia M. ASP Triangles: Sketching the Artificial Intelligence of a Mobile Platform. Cranfield Online Research Data (CORD); 2020;

35. di Fraia MZ., Cuartielles J-P., Felicetti L., Chermak L. Autonomous Visual Trajectory Reconstruction in the Proximity of Small Celestial Objects. 7th IAA Planetary Defense Conference. 2021. p. 164.

36. Di Fraia MZ. Monitoring Dust Motion Around Airless Celestial Bodies: Characterizing Suitable Landing Zones. 2018 British and Irish Conference on Optics and Photonics, BICOP 2018 - Proceedings. 2019. Available at: DOI:10.1109/BICOP.2018.8658355

37. Overview | Our Solar System – NASA Solar System Exploration. Available at: https://solarsystem.nasa.gov/solar-system/our-solar-system/overview/ (Accessed: 13 July 2021)

38. A'Hearn MF., Belton MJS., Delamere WA., Kissel J., Klaasen KP., McFadden LA., et al. Deep Impact: Excavating comet Tempel 1. Science. American Association for the Advancement of Science; 14 October 2005; 310(5746): 258–264. Available at: DOI:10.1126/science.1118923

39. Dutta S., Way DW. Comparison of the effects of velocity and range triggers on trajectory dispersions for the Mars 2020 mission. AIAA Atmospheric Flight Mechanics Conference. 2017. p. 245.

40. Blackmore L. Autonomous precision landing of space rockets. Frontiers of Engineering: Reports on Leading-Edge Engineering from the 2016 Symposium. 2016. pp. 15–20.

41. NASA NSSDCA Spacecraft details (Luna 9). Available at: https://nssdc.gsfc.nasa.gov/nmc/spacecraft/display.action?id=1966-006A (Accessed: 12 June 2021)

42. Greeley R., Iversen JD. Wind as a geological process: on Earth, Mars, Venus and Titan. CUP Archive; 1987.

43. Opik EJ. Physics of meteor flight in the atmosphere. Courier Corporation; 2004.

44. Sagan C., Bagnold RA. Fluid transport on Earth and aeolian transport on Mars. Icarus. Elsevier; 1975; 26(2): 209–218.

45. Kikuchi S., Terui F., Ogawa N., Saiki T., Ono G., Yoshikawa K., et al. Design and reconstruction of the Hayabusa2 precision landing on Ryugu.

Journal of Spacecraft and Rockets. American Institute of Aeronautics and Astronautics; 2020; 57(5): 1033–1060.

46. Veverka J., Farquhar B., Robinson M., Thomas P., Murchie S., Harch A., et al. The landing of the NEAR-Shoemaker spacecraft on asteroid 433 Eros. Nature. 2001; Available at: DOI:10.1038/35096507

47. Yano H., Kubota T., Miyamoto H., Okada T., Scheeres D., Takagi Y., et al. Touchdown of the Hayabusa spacecraft at the muses sea on Itokawa. Science. 2006; Available at: DOI:10.1126/science.1126164

48. Riedel JE., Wang T., Werner R., Vaughan A., Myers D., Mastrodemos N., et al. Configuring the Deep Impact AutoNav System for Lunar, Comet and Mars Landing. AIAA/AAS Astrodynamics Specialist Conference and Exhibit. 2008; (August): 1–21. Available at: DOI:10.1007/s10946-012-9296-5

49. Boehnhardt H., Bibring JP., Apathy I., Auster HU., Finzi AE., Goesmann F., et al. The Philae lander mission and science overview. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences. 2017. Available at: DOI:10.1098/rsta.2016.0248

50. Tsuda Y., Saiki T., Terui F., Nakazawa S., Yoshikawa M., Watanabe S ichiro. Hayabusa2 mission status: Landing, roving and cratering on asteroid Ryugu. Acta Astronautica. 2020; Available at: DOI:10.1016/j.actaastro.2020.02.035

51. Lauretta DS., Balram-Knutson SS., Beshore E., Boynton W V., Drouet d'Aubigny C., DellaGiustina DN., et al. OSIRIS-REx: Sample Return from Asteroid (101955) Bennu. Space Science Reviews. 2017. Available at: DOI:10.1007/s11214-017-0405-1

52. Barucci MA., Perna D., Popescu M., Fornasier S., Doressoundiram A., Lantz C., et al. Small D-type asteroids in the NEO population: New targets

for space missions. Monthly Notices of the Royal Astronomical Society. 2018; Available at: DOI:10.1093/mnras/sty532

53. McNutt L., Johnson L., Clardy D., Castillo-Rogez J., Frick A., Jones L. Near-earth asteroid scout. AIAA SPACE 2014 Conference and Exposition. 2014.

54. CNEOS - Discovery Statistics. Available at: https://cneos.jpl.nasa.gov/stats/totals.html

55. Balázs A., Baksa A., Bitterlich H., Hernyes I., Küchemann O., Pálos Z., et al. The central on-board computer of the Philae lander in the context of the Rosetta space mission. Ada-Europe International Conference on Reliable Software Technologies. 2015. pp. 18–30.

56. Ulamec S., Fantinati C., Maibaum M., Geurts K., Biele J., Jansen S., et al. Rosetta lander--landing and operations on comet 67P/Churyumov--Gerasimenko. Acta Astronautica. Elsevier; 2016; 125: 80–91.

57. ESA - Three touchdowns for Rosetta's lander. Available at: https://www.esa.int/Science_Exploration/Space_Science/Rosetta/Three_touchdowns_for_Rosetta_s_lander (Accessed: 18 July 2021)

58. Jia P., Andreotti B., Claudin P. Giant ripples on comet 67P/Churyumov–Gerasimenko sculpted by sunset thermal wind. Proceedings of the National Academy of Sciences. National Academy of Sciences; 7 March 2017; 114(10): 2509–2514. Available at: DOI:10.1073/PNAS.1612176114

59. Krüger H., Seidensticker KJ., Fischer H-H., Albin T., Apathy I., Arnold W., et al. Dust impact monitor (SESAME-DIM) measurements at comet 67P/Churyumov-Gerasimenko. Astronomy \& Astrophysics. EDP Sciences; 2015; 583: A15.

60. Chabot NL., Ernst CM., Denevi BW., Harmon JK., Murchie SL., Blewett DT., et al. Areas of permanent shadow in Mercury's south polar region

ascertained by MESSENGER orbital imaging. Geophysical Research Letters. Wiley Online Library; 2012; 39(9).

61. Bussey DBJ., Lucey PG., Steutel D., Robinson MS., Spudis PD., Edwards KD. Permanent shadow in simple craters near the lunar poles. Geophysical Research Letters. Wiley Online Library; 2003; 30(6).

62. Platz T., Nathues A., Schorghofer N., Preusker F., Mazarico E., Schröder SE., et al. Surface water-ice deposits in the northern shadowed regions of Ceres. Nature Astronomy. Nature Publishing Group; 2016; 1(1): 1–6.

63. ESA - Philae found! Available at: https://www.esa.int/Science_Exploration/Space_Science/Rosetta/Philae_found (Accessed: 20 July 2021)

64. Cheng AF., Stickle AM., Fahnestock EG., Dotto E., della Corte V., Chabot NL., et al. DART mission determination of momentum transfer: Model of ejecta plume observations. Icarus. Elsevier; 2020; 352: 113989.

65. Michel P., Kueppers M., Sierks H., Carnelli I., Cheng AF., Mellab K., et al. European component of the AIDA mission to a binary asteroid: Characterization and interpretation of the impact of the DART mission. Advances in Space Research. Elsevier; 2018; 62(8): 2261–2272.

66. Goldberg HR., Karatekin Ö., Ritter B., Herique A., Tortora P., Prioroc C., et al. The juventas cubesat in support of ESA’s hera mission to the asteroid didymos. 2019;

67. Ferrari F., Franzese V., Pugliatti M., Giordano C., Topputo F. Preliminary mission profile of Hera’s Milani CubeSat. Advances in Space Research. Elsevier; 2021; 67(6): 2010–2029.

68. Zhang X., Huang J., Wang T., Huo Z. ZhengHe-a mission to a near-earth asteroid and a main belt comet. 50th Annual Lunar and Planetary Science Conference. 2019. p. 1045.

69. Jin WT., Li F., Yan JG., Andert TP., Ye M., Yang X., et al. A simulated global GM estimate of the asteroid 469219 Kamo ‘oalewa for China’s future asteroid mission. Monthly Notices of the Royal Astronomical Society. Oxford University Press; 2020; 493(3): 4012–4021.

70. Levison HF., Olkin CB., Noll KS., Marchi S., Bell III JF., Bierhaus E., et al. Lucy mission to the Trojan asteroids: Science goals. The Planetary Science Journal. IOP Publishing; 2021; 2(5): 171.

71. Hart W., Brown GM., Collins SM., Pich MDS-S., Fieseler P., Goebel D., et al. Overview of the spacecraft design for the Psyche mission concept. 2018 IEEE Aerospace Conference. IEEE; 2018. pp. 1–20.

72. Ozaki N., Yamamoto T., Gonzalez-Franquesa F., Gutierrez-Ramon R., Pushparaj N., Chikazawa T., et al. Mission design of DESTINY+: Toward active asteroid (3200) Phaethon and multiple small bodies. Acta Astronautica. Elsevier; 2022; 196: 42–56.

73. McNutt L., Johnson L., Kahn P., Castillo-Rogez J., Frick A. Near-earth asteroid (NEA) scout. AIAA Space 2014 Conference and Exposition. 2014. p. 4435.

74. Howell SM., Chou L., Thompson M., Bouchard MC., Cusson S., Marcus ML., et al. Camilla: A centaur reconnaissance and impact mission concept. Planetary and Space Science. Elsevier; 2018; 164: 184–193.

75. Smith GE. Nobel Lecture: The invention and early history of the CCD. Reviews of modern physics. APS; 2010; 82(3): 2307.

76. Fossum ER. CMOS image sensors: Electronic camera-on-a-chip. IEEE transactions on electron devices. IEEE; 1997; 44(10): 1689–1698.

77. colour - The visible spectrum | Britannica. Available at: https://www.britannica.com/science/color/The-visible-spectrum
(Accessed: 30 July 2021)

78. Waltham N. CCD and CMOS sensors. Observing Photons in Space. Springer New York; 2013. pp. 423–442. Available at: DOI:10.1007/978-1-4614-7804-1_23

79. Szeliski R. Computer vision: algorithms and applications. Springer Science \& Business Media; 2010.

80. Cocaud C., Kubota T. SURF-based SLAM scheme using octree occupancy grid for autonomous landing on asteroids. Proceedings of the 10th International Symposium on Artificial Intelligence, Robotics and Automation in Space. 2010;

81. Takao Y., Tsuda Y., Saiki T., Ogawa N., Terui F., Kawaguchi J. Optical navigation for asteroid explorer by use of virtual feature points. AIAA/AAS Astrodynamics Specialist Conference, 2016. 2016. Available at: DOI:10.2514/6.2016-5213

82. Li S., Cui P., Cui H. Autonomous navigation and guidance for landing on asteroids. Aerospace Science and Technology. 2006; Available at: DOI:10.1016/j.ast.2005.12.003

83. Bhaskaran S., Nandi S., Broschart S., Wallace M., Alberto Cangahuala L., Olson C. Small body landings using autonomous onboard optical navigation. Journal of the Astronautical Sciences. 2011. Available at: DOI:10.1007/BF03321177

84. Harris C., Stephens M. A Combined Edge and Corner Detector. Proceedings of the fourth Alvey Vision Conference (ACV88). 1988.

85. Bay H., Tuytelaars T., van Gool L. SURF: Speeded up robust features. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Springer, Berlin, Heidelberg; 2006. pp. 404–417. Available at: DOI:10.1007/11744023_32 (Accessed: 3 May 2021)

86. Lowe DG. Object recognition from local scale-invariant features. Proceedings of the IEEE International Conference on Computer Vision. IEEE; 1999. pp. 1150–1157. Available at: DOI:10.1109/iccv.1999.790410

87. Lindeberg T. Scale-space theory: A basic tool for analyzing structures at different scales. Journal of Applied Statistics. Carfax Publishing Company ; 1 January 1994; 21(1–2): 225–270. Available at: DOI:10.1080/757582976

88. Rublee E., Rabaud V., Konolige K., Bradski G. ORB: An efficient alternative to SIFT or SURF. Proceedings of the IEEE International Conference on Computer Vision. 2011. Available at: DOI:10.1109/ICCV.2011.6126544

89. Rosten E., Drummond T. Machine learning for high-speed corner detection. European conference on computer vision. 2006. pp. 430–443.

90. Yu M., Cui H. Robust hazard matching approach for visual navigation application in planetary landing. Aerospace Science and Technology. Elsevier Masson SAS; 2015; 47: 378–387. Available at: DOI:10.1016/j.ast.2015.09.028

91. Johnson A., Ivanov T. Analysis and testing of a lidar-based approach to terrain relative navigation for precise lunar landing. Aiaa guidance, navigation, and control conference. 2011. p. 6578.

92. Johnson AE., Montgomery JF. Overview of Terrain Relative Navigation approaches for precise lunar landing. IEEE Aerospace Conference Proceedings. 2008. Available at: DOI:10.1109/AERO.2008.4526302

93. Tian Y., Yu M. A novel crater recognition based visual navigation approach for asteroid precise pin-point landing. Aerospace Science and Technology. Elsevier; 2017; 70: 1–9.

94. Scaramuzza D., Fraundorfer F. Tutorial: Visual odometry. IEEE Robotics and Automation Magazine. December 2011; 18(4): 80–92. Available at: DOI:10.1109/MRA.2011.943233

95. Christian JA., Hong L., McKee P., Christensen R., Crain TP. Image-Based Lunar Terrain Relative Navigation Without a Map: Measurements. Journal of Spacecraft and Rockets. American Institute of Aeronautics and Astronautics; 2021; 58(1): 164–181.

96. Moravec HP. Obstacle avoidance and navigation in the real world by a seeing robot rover. 1980.

97. Nistér D., Naroditsky O., Bergen J. Visual odometry. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 2004. Available at: DOI:10.1109/cvpr.2004.1315094

98. Maimone M., Cheng Y., Matthies L. Two years of Visual Odometry on the Mars Exploration Rovers. Journal of Field Robotics. John Wiley & Sons, Ltd; 1 March 2007; 24(3): 169–186. Available at: DOI:10.1002/rob.20184

99. Achtelik M., Achtelik M., Brunet Y., Chli M., Chatzichristofis S., Decotignie JD., et al. SFly: Swarm of micro flying robots. IEEE International Conference on Intelligent Robots and Systems. 2012. pp. 2649–2650. Available at: DOI:10.1109/IROS.2012.6386281

100. An L., Zhang X., Gao H., Liu Y. Semantic segmentation--aided visual odometry for urban autonomous driving. International Journal of Advanced Robotic Systems. SAGE Publications Sage UK: London, England; 2017; 14(5): 1729881417735667.

101. Derpanis KG. Overview of the RANSAC Algorithm. Image Rochester NY. 2010; 4(1): 2–3.

102. Nistér D. An efficient solution to the five-point relative pose problem. IEEE transactions on pattern analysis and machine intelligence. IEEE; 2004; 26(6): 756–770.

103. Hartley RI. In defense of the eight-point algorithm. IEEE Transactions on pattern analysis and machine intelligence. IEEE; 1997; 19(6): 580–593.

104. Sünderhauf N., Konolige K., Lacroix S., Protzel P. Visual odometry using sparse bundle adjustment on an autonomous outdoor vehicle. Autonome Mobile Systeme 2005. Springer; 2006. pp. 157–163.

105. Riedel JE., Wang TC., Werner R., Bayard D., Vaughan A., Mastrodemos N., et al. Configuring the deep impact AutoNav system for lunar, comet and Mars landing. AIAA/AAS Astrodynamics Specialist Conference and Exhibit. 2008. Available at: DOI:10.2514/6.2008-6940

106. Bhaskaran S., Nandi S., Broschart S., Wallace M., Alberto Cangahuala L., Olson C. Small body landings using autonomous onboard optical navigation. Journal of the Astronautical Sciences. Springer; 2011. pp. 409–427. Available at: DOI:10.1007/bf03321177

107. Riedel J., Bhaskaran S., Desai S., Han D., Kennedy B., McElrath T., et al. Using autonomous navigation for interplanetary missions: The validation of deep space 1 AutoNav. 2000;

108. Mastrodemos N., Kubitschek DG., Synnott SP. Autonomous navigation for the Deep Impact mission encounter with comet Tempel 1. Space Science Reviews. 2005; Available at: DOI:10.1007/s11214-005-3394-4

109. Bhaskaran S. Autonomous navigation for deep space missions. SpaceOps 2012 Conference. 2012. Available at: DOI:10.2514/6.2012-1267135

110. Campbell TS. A deep learning approach to autonomous relative terrain navigation. The University of Arizona; 2017.

111. Downes LM., Steiner TJ., How JP. Lunar terrain relative navigation using a convolutional neural network for visual crater detection. 2020 American Control Conference (ACC). 2020. pp. 4448–4453.

112. Lee H., Choi H-L., Jung D., Choi S. Deep neural network-based landmark selection method for optical navigation on lunar highlands. IEEE Access. IEEE; 2020; 8: 99010–99023.

113. Williams BG., Antreasian PG., Bordi JJ., Carranza E., Chesley R., Helfrich CE., et al. Navigation for NEAR Shoemaker: the first spacecraft to orbit an asteroid. 30 July 2001;

114. Dunham DW., Farquhar RW., McAdams J V., Holdridge M., Nelson R., Whittenburg K., et al. Implementation of the First Asteroid Landing. Icarus. Academic Press; 1 October 2002; 159(2): 433–438. Available at: DOI:10.1006/ICAR.2002.6911

115. Yoshimitsu T., Kubota T., Nakatani I. MINERVA rover which became a small artificial solar satellite. 2006;

116. Blume WH. Deep impact mission design. Space Science Reviews. Springer; 2005; 117(1–2): 23–42.

117. Kubitschek DG., Mastrodemos N., Werner RA., Kennedy BM., Synnott SP., Null GW., et al. Deep impact autonomous navigation: The trials of targeting the unknown. Advances in the Astronautical Sciences. 2006; 125: 381–406. Available at: DOI:2014/38755

118. Williams B., Antreasian P., Carranza E., Jackman C., Leonard J., Nelson D., et al. OSIRIS-REx flight dynamics and navigation design. Space Science Reviews. Springer; 2018; 214(4): 1–43.

119. Atchison JA., Dong CP., Jensenius MA. Double asteroid redirection test—Mission design and navigation. Proceedings of the International Syposium on Space Flight Mechanics, F. 2014;

120. Pugliatti M., Rizza A., Piccolo F., Franzese V., Bottiglieri C., Giordano C., et al. The Milani mission: overview and architecture of the optical-based GNC system. AIAA Scitech 2022 Forum. 2022. p. 2381.

121. Kitazato K., Milliken RE., Iwata T., Abe M., Ohtake M., Matsuura S., et al. The surface composition of asteroid 162173 Ryugu from Hayabusa2 near-infrared spectroscopy. Science. American Association for the Advancement of Science; 2019; 364(6437): 272–275.

122. NASA/Goddard/University of Arizona. Bennu shape model files. Available at: https://www.asteroidmission.org/updated-bennu-shape-model-3d-files/

123. Magri C., Howell ES., Nolan MC., Taylor PA., Fernández YR., Mueller M., et al. Radar and photometric observations and shape modeling of contact binary near-Earth Asteroid (8567) 1996 HW1. Icarus. 2011; Available at: DOI:10.1016/j.icarus.2011.02.019

124. Wright JR., Cooper JE. Introduction to aircraft aeroelasticity and loads. John Wiley \& Sons; 2008.

125. Collar AR. The expanding domain of aeroelasticity. The Aeronautical Journal. Cambridge University Press; 1946; 50(428): 613–636.

126. Coradeschi S., Saffiotti A. Symbiotic robotic systems: Humans, robots, and smart environments. IEEE Intelligent Systems. IEEE; 2006; 21(3): 82–84.

127. Wiesel WE. Modern orbit determination. Aphelion Press Beavercreek; 2010.

128. Lauretta DS., Bartels AE., Barucci MA., Bierhaus EB., Binzel RP., Bottke WF., et al. The OSIRIS-REx target asteroid (101955) Bennu: Constraints on its physical, geological, and dynamical nature from astronomical observations. Meteoritics and Planetary Science. 2015; Available at: DOI:10.1111/maps.12353

129. Lauretta DS., DellaGiustina DN., Bennett CA., Golish DR., Becker KJ., Balram-Knutson SS., et al. The unexpected surface of asteroid (101955) Bennu. Nature. 2019; Available at: DOI:10.1038/s41586-019-1033-6

130. PANGU - Planet and Asteroid Natural scene Generation Utility - STAR-Dundee. Available at: https://www.star-dundee.com/products/pangu-planet-and-asteroid-natural-scene-generation-utility/ (Accessed: 21 July 2021)

131. Simulink 3D Animation - MATLAB & Simulink. Available at: https://uk.mathworks.com/products/3d-animation.html?requestedDomain (Accessed: 21 July 2021)

132. The most powerful real-time 3D creation platform - Unreal Engine. Available at: https://www.unrealengine.com/en-US/ (Accessed: 21 July 2021)

133. Rouan D. Effective Temperature. Encyclopedia of Astrobiology. 2015; : 712.

134. What color is a blackbody? - some pixel rgb values. Available at: http://www.vendian.org/mncharity/dir3/blackbody/ (Accessed: 21 July 2021)

135. Hex Color Codes, Paint Matching and Color Picker. Available at: https://encycolorpedia.com/ (Accessed: 21 July 2021)

136. Jean I., Ng A., Misra AK. Impact of solar radiation pressure modeling on orbital dynamics in the vicinity of binary asteroids. Acta Astronautica. Elsevier; 2019; 165: 167–183.

137. González D., Cañas V., Roberts PCE. Modeling and simulation of very low earth orbit. 8th European Conference for Aeronautics and Space Sciences (EUCASS). 2019.

138. Venzmer MS., Bothmer V. Solar-wind predictions for the Parker Solar Probe orbit-Near-Sun extrapolations derived from an empirical solar-wind model based on Helios and OMNI observations. Astronomy & Astrophysics. EDP Sciences; 2018; 611: A36.

139. Christian JA., Robinson SB. Noniterative horizon-based optical navigation by cholesky factorization. Journal of Guidance, Control, and Dynamics. 2016. Available at: DOI:10.2514/1.G000539

140. Christian JA., Lightsey EG. Onboard image-processing algorithm for a spacecraft optical navigation sensor system. Journal of spacecraft and rockets. 2012; 49(2): 337–352.

141. C3D CUBESAT CAMERA | XCAM. Available at: http://www.xcam.co.uk/c3d-cubesat-camera (Accessed: 21 July 2021)

142. Speretta S., Cervone A., Sundaramoorthy P., Noomen R., Mestry S., Cipriano A., et al. Lumio: Achieving autonomous operations for lunar exploration with a cubesat. 15th International Conference on Space Operations, 2018. 2018. Available at: DOI:10.2514/6.2018-2599

143. Franzese V., Di Lizia P., Topputo F. Autonomous optical navigation for LUMIO mission. Space Flight Mechanics Meeting, 2018. 2018. Available at: DOI:10.2514/6.2018-1977

144. Liounis AJ. Limb Based Optical Navigation for Irregular Bodies.

145. Eberly D. Reconstructing an ellipsoid from its perspective projection onto a plane. Website. 2007;

146. Duda RO., Hart PE., Stork DG. Pattern Classiication and Scene Analysis 2nd ed. Part 1: Pattern Classiication. 1995.

147. Fitzgibbon A., Pilu M., Fisher RB. Direct least square fitting of ellipses. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1999; Available at: DOI:10.1109/34.765658

148. Chernov N. Ellipse Fit (Direct method). Ellipse Fit (Direct method). 2009. Available at: https://www.mathworks.com/matlabcentral/fileexchange/22684-ellipse-fit-direct-method (Accessed: 19 November 2020)

149. Christian JA. Accurate planetary limb localization for image-based spacecraft navigation. Journal of Spacecraft and Rockets. 2017; Available at: DOI:10.2514/1.A33692

150. Zadeh LA. Fuzzy logic. Computer. IEEE; 1988; 21(4): 83–93.

151. Swartwout M. The First One Hundred CubeSats: A Statistical Look. JSSat. 2013.

152. Schoolcraft J., Klesh A., Werne T. MarCO: Interplanetary mission development on a cubesat scale. Space Operations: Contributions from the Global Community. Springer International Publishing; 2017. pp. 221–231. Available at: DOI:10.1007/978-3-319-51941-8_10

153. Oda M., Kibe K., Yamagata F. ETS-VII, space robot in-orbit experiment satellite. Proceedings of IEEE international conference on robotics and automation. 1996. pp. 739–744.

154. Hui C., Yousheng C., Xiaokun L., Shing WW. Autonomous takeoff, tracking and landing of a UAV on a moving UGV using onboard monocular vision. Proceedings of the 32nd Chinese control conference. 2013. pp. 5895–5901.

155. Babinec A., Jurišica L., Hubinsk\`y P., Ducho\vn F. Visual localization of mobile robot using artificial markers. Procedia Engineering. Elsevier; 2014; 96: 1–9.

156. dos Santos Cesar DB., Gaudig C., Fritsche M., dos Reis MA., Kirchner F. An evaluation of artificial fiducial markers in underwater environments. OCEANS 2015-Genova. 2015. pp. 1–6.

157. Epp CD., Smith TB. Autonomous Precision Landing and Hazard Detection and Avoidance Technology (ALHAT). IEEE Aerospace Conference Proceedings. 2007. Available at: DOI:10.1109/AERO.2007.352724

158. Yu M., Cui H., Tian Y. A new approach based on crater detection and matching for visual navigation in planetary landing. Advances in Space Research. COSPAR; 2014; 53(12): 1810–1821. Available at: DOI:10.1016/j.asr.2013.04.011

159. Escrig F., Valcarcel JP. Geometry of expandable space structures. International Journal of Space Structures. SAGE Publications Sage UK: London, England; 1993; 8(1–2): 71–84.

160. Schenk M., Viquerat AD., Seffen KA., Guest SD. Review of inflatable booms for deployable space structures: packing and rigidization. Journal of Spacecraft and Rockets. American Institute of Aeronautics and Astronautics; 2014; 51(3): 762–778.

161. Krishnakumar KS., Stillwater RA., Babula M., Moreau MC., Riedel JE., Mrozinski RB., et al. A Proposed Strategy for the U.S. to Develop and Maintain a Mainstream Capability Suite ("Warehouse") for Automated/Autonomous Rendezvous and Docking in Low Earth Orbit and Beyond. 2012;

162. Kubota T., Hashimoto T., Kawaguchi J. Motion planning of intelligent explorer for asteroid exploration mission. Motion Planning. InTech; 2008. pp. 243–260.

163. Kubota T., Sawai S., Hashimoto T., Kawaguchi J. Collision dynamics of a visual target marker for small-body exploration. Advanced Robotics. Taylor \& Francis; 2007; 21(14): 1635–1651.

164. The Pinpoint Touchdown – Target Marker 1 (PPTD-TM1) operation | Topics | JAXA Hayabusa2 project. Available at: https://www.hayabusa2.jaxa.jp/en/topics/20190515e_PPTD-TM1/ (Accessed: 11 August 2021)

165. Ogawa N., Terui F., Kawaguchi J. Precise landing of space probe on asteroid using multiple markers. IFAC Proceedings Volumes. Elsevier; 2010; 43(15): 172–177.

166. Y\ilmaz A. Color of absence and presence: Reconsidering black in architecture. Color Research \& Application. Wiley Online Library; 2017; 42(3): 378–387.

167. Maurel C., Michel P., Biele J., Ballouz R-L., Thuillet F. Numerical simulations of the contact between the lander MASCOT and a regolith-covered surface. Advances in Space Research. Elsevier; 2018; 62(8): 2099–2124.

168. Çelik O., Karatekin O., Ritter B., Sánchez JP. Reliability analysis of ballistic landing in binary asteroid 65803 (1996GT) Didymos under uncertainty and GNC error considerations. 26th International Symposium on Spaceflight Dynamics. 2017.

169. The Mathworks, Inc. MATLAB, Version 9.6 2019. MATLAB 2019b - MathWorks. Www.Mathworks.Com/Products/Matlab. 2019.

170. Mirjalili S. Genetic algorithm. Evolutionary algorithms and neural networks. Springer; 2019. pp. 43–55.

171. Alcantarilla PF., Bartoli A., Davison AJ. KAZE features. European conference on computer vision. 2012. pp. 214–227.

172. NanoCam C1U Datasheet.

173. Bay H., Ess A., Tuytelaars T., Van Gool L. Speeded-Up Robust Features (SURF). Computer Vision and Image Understanding. 2008; Available at: DOI:10.1016/j.cviu.2007.09.014

174. Amini R., Azari A., Bhaskaran S., Beauchamp P., Castillo-Rogez J., Castano R., et al. Advancing the Scientific Frontier with Increasingly Autonomous Systems. arXiv preprint arXiv:2009.07363. 2020;

175. Pan X., Chen X., Takeyama S., Yamaguchi M. Image reconstruction with transformer for mask-based lensless imaging. Optics Letters. Optica Publishing Group; 2022; 47(7): 1843–1846.

176. Gallego G., Delbrück T., Orchard G., Bartolozzi C., Taba B., Censi A., et al. Event-based vision: A survey. IEEE transactions on pattern analysis and machine intelligence. IEEE; 2020; 44(1): 154–180.