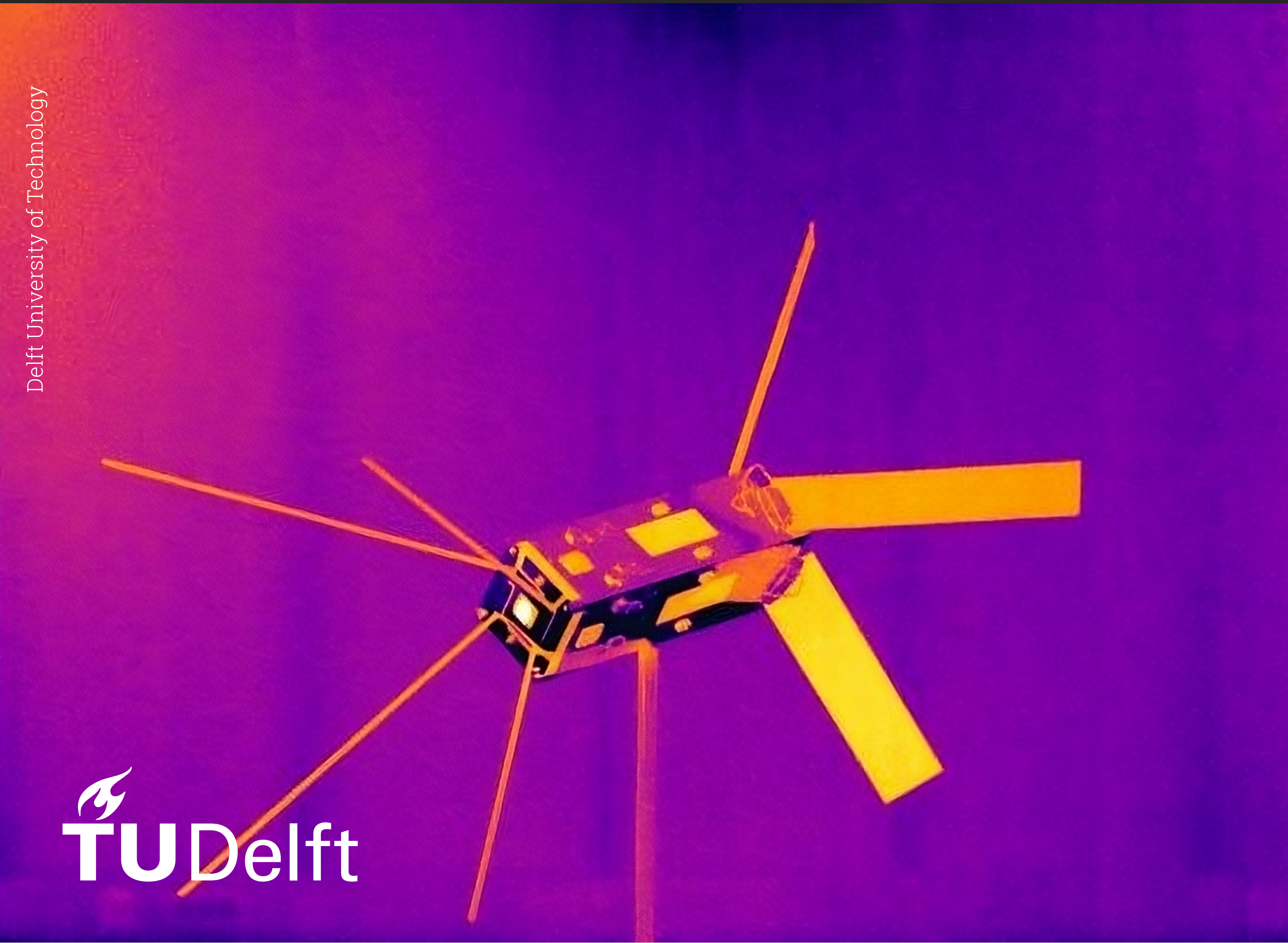
Pose Estimation of an Unknown Uncooperative Target Using LiDAR & Thermal Infrared Fusion

MSc Thesis in Aerospace Engineering

Conor Creagh



Delft University of Technology

TU Delft

Pose Estimation of an Unknown Uncooperative Target Using LiDAR & Thermal Infrared Fusion

by

Conor Creagh

A thesis submitted to the Delft University of Technology in partial fulfillment of the requirements for the degree of Master of Science in Aerospace Engineering.

To be defended publicly on 22 March 2024 at 13:00.

Student name:	Conor Creagh	
Student number:	4995325	
Project duration:	15 June 2023 – 22 March 2024	
Thesis committee:	Dr. J. Guo,	TU Delft, Supervisor
	Dr. A. Menicucci,	TU Delft, Chairperson
	Dr. Ir. E. van Kampen,	TU Delft, External Examiner

TU Delft

Abstract

In recent years, researchers have proposed a variety of approaches to tackle the problem of orbital debris. Debris targets are diverse, and prior knowledge may be limited, with unknown uncooperative debris targets being the most challenging category. A crucial portion of any debris capture scenario is the observation phase during the approach. During this phase, the chaser spacecraft attempts to learn as much as possible about the target by using remote sensing, of which relative pose is of particular interest, to enable the advancement of the mission towards eventual capture.

This research utilizes the fusion of data from a scanning LiDAR with a long-wavelength infrared camera to estimate the relative pose of an unknown uncooperative target. Two separate bespoke pose estimation algorithms, color-ICP and Feature Matching, were developed and tested with laboratory experiments mimicking the close-approach phase with a target under various lighting conditions and relative motion rates. The color-ICP algorithm uses a thermal infrared-infused color-assisted Generalized Iterative Closest Points method, while the Feature Matching algorithm uses computer vision on LiDAR point-infused thermal images to track BRISK feature points in each frame to estimate pose.

In general, the color-ICP algorithm delivered more accurate results throughout the range of experiments, though the fusion was slightly detrimental while the target is being heated or cooled. The Feature Matching algorithm contains a large amount of tunable parameters, making the estimation highly sensitive yet versatile, demonstrating that harsh lighting conditions can be mitigated with accurate features tracked after the implementation of image processing techniques. Overall, the end product shows promise as a light-agnostic remote sensing and pose estimation solution.

This research contributes to the advancement of active debris removal theory and explores two promising avenues for LiDAR-infrared sensor fusion for pose estimation, laying the groundwork for further iterations exploring this sensor pairing. The resulting use case is a conceivable scenario in which these sensors work together to supplement individual strengths and mitigate disadvantages throughout the approach phase of a debris removal mission.

Keywords: Active Debris Removal, Remote Sensing, Sensor Fusion, LiDAR, Thermal Infrared, Pose Estimation, Unknown Uncooperative Targets

Preface

This body of work is dedicated to my mother, father and brother. I would not be here without your continued support and belief in me. Mom, Dad and Troy, I love you.

There are many kind people who have helped me when they had absolutely no obligation to. I would like to take this moment to thank them in detail, which I think will also give a glimpse into this long and intense process.

To Professor Guo, thank you for your guidance and understanding during this entire process. I truly believe that we have made something we can both be proud of, and I will dearly miss our conversations. Liu Jingyi, thank you for your kindness and assistance during the initial phase of the literature study while I was lost in an ocean of possible research topics. Joshua Spaander, thank you so much for all your help and mentorship during the experiment process. Suddenly working with all the hardware was a steep learning curve and you always had time to answer questions and to help. Thank you to the technicians at the Flight Hall for your flexibility and allowing me to use the fantastic FLIR camera. Thank you Alejandro Jimenez del Toro for helping me retrieve lost data files when I accidentally deleted them. Thank you Saskia Peters for connecting me with Amogh Gudi, who I also want to thank for taking pity on someone getting started with computer vision and for helping me to divorce C++ and .dll hunting. Thank you to the kind employee at the Copie-Sjop in BK City who helped me print and fabricate a new calibration target. Thank you Sevket Uludag for making time for me when you were extremely busy and offering to help me make a rotation ring for a target. Thank you to Michael Van Rijn and Brandweerkazerne Slotlaan in Capelle for offering help in finding an infrared camera. Also the anonymous internet heroes of the computer vision world deserve my eternal gratitude for creating a vast expanse of open-source documentation and tools to do incredible things with these sensors.

I would also like to thank my Aunt Loes for housing me at two important phases of my studies. Thank you Troy and Gabby for giving me a place to stay at the end of this strange period of my life, and for being there for me when I needed it most. To my brother Troy, thank you for listening to my ramblings about algorithms, sparring ideas, improving my lackluster visual-aids, and for being my role model and greatest hero.

Thank you all. It takes a village to write a Master's thesis.

Conor Creagh
Houston, November 2023

Contents

List of Figures

List of Tables

Nomenclature

Abbreviations

Abbreviation	Definition
ADR	Active Debris Removal
AE	Aerospace Engineering
BRISK	Binary Robust Invariant Scalable Keypoints
CIE	International Commission on Illumination
CV	Computer Vision
CW	Continuous Wave
DOF	Degrees of Freedom
FOV	Field Of View
HDR	High Dynamic Range
imPt(s)	LiDAR point(s) projected onto an image
IR	Infrared
ISS	International Space Station
JDL	Joint Directors of Laboratories
KAZE	Non-linear pyramid-based Features
LiDAR	Light Detection and Ranging
LWIR	Long-Wavelength Infrared
MATLAB	MATrix LABoratory programming language
MEV	Mission Extension Vehicle
MSER	Maximally Stable Extremal Regions
MWIR	Medium-Wavelength Infrared
NIR	Near Infrared
OOS	On-Orbit Servicing
ORB	Oriented FAST and Rotated BRIEF
RGB	Red-Green-Blue (color values)
RPO	Rendezvous and Proximity Operations
RMSE	Root Mean Square Error
SIFT	Scale-Invariant Feature Transform
STS	Space Transport System
SURF	Speeded-Up Robust Features
SWIR	Short-Wavelength Infrared
tform	transformation
TOF	Time-Of-Flight
TriDAR	Triangulation + LiDAR
TUD	Technical University of Delft
UV	Ultraviolet
V&V	Verification and Validation

Symbols

Symbol	Definition	Unit
A	Transformation Matrix	transformation portion in [m]
C	Control array	transformation portion in [m]
E_{inc}	total sun-earth incident flux	[J]
K	Camera intrinsic matrix	[pixel]
O	Observed array	transformation portion in [m]
R	Rotation Matrix	[unitless]
r	Radius	[m]
$RMSE$	Root Mean Square Error	[unitless]
T	Translation Matrix	[m]
T_{deb}	Debris temperature	[K]
T_e	Equilibrium temperature	[K]
V	Velocity	[m/s]
w	Zhang’s Method scale factor	[unitless]
x	Pixel x-coordinate	[pixel]
X	World x-coordinate	[m]
y	Pixel y-coordinate	[pixel]
Y	World y-coordinate	[m]
Z	World z-coordinate	[m]
α	Scale factor in image x-axis	[unitless]
β	Scale factor in image y-axis	[unitless]
γ	Skew of image axes	[unitless]
ε	Emissivity	[unitless]
θ	Rotation Euler angle	[radians]
σ	Stefan Boltzmann constant	[J/K]
τ	Absorptivity	[unitless]

1
Introduction & Overview

This chapter will serve as an introduction to the research performed by introducing key concepts, topics and ideas relevant to the study. Section 1.1 will define the motivations for the research as well as the research questions to be answered as a direct result of the work done herein, as well as defining the scope and application of the research before providing a document outline. Section 1.2 will continue in introducing concepts that are essential to the research.

1.1. Introduction

This research follows a literature study regarding the state-of-the-art of the key concepts and technologies relevant to orbital debris removal. Before the lessons and insights gained from this study are discussed, it is useful to explain the motivation of this work while indicating to the reader what this research does and does not entail.

1.1.1. Motivation

Our role in space is evolving. This revolutionary step for humanity comes with unique challenges that must be addressed and mastered before the sector can advance further. The problem of space debris has garnered attention and there are multiple approaches being considered to tackle the issue, an issue which will require a diverse set of approaches to mitigate the worse case scenario, the so-called Kessler syndrome.[56] The skills and techniques required to address this issue are, however, applicable to a wider variety of tasks in orbit and deep space. Rendezvous in orbit was first accomplished by the American Gemini missions, the first of which was flown by Neil Armstrong and almost resulted in his and his crew-mate's death.[41] Since then, in-orbit rendezvous have become a routine part of space flight, most notably with the International Space Station (ISS) which is regularly resupplied and changes crew. However, these rendezvous have been *cooperative*, meaning that both the chaser (resupply craft for example) and the target (ISS for example) have been built with rendezvous in mind, having aids and references for precisely this task. In the scenario of docking with the ISS, the ISS can be categorized as a *cooperative known* target in that its parameters are known before the docking procedure and that it cooperates actively (communicating) or passively (reference points) to assist the chaser in its rendezvous and docking procedure.

Cooperative known targets are an ideal case. However, consider the task of rendezvous and capture of debris from an orbital collision. This piece of debris may once have been a well known and understood section of a larger system, however after the collision it has changed. Another scenario would be that of the unexpected failure of a satellite in orbit, such as was the case with the school-bus-sized Envisat[1], before it is able to discharge its batteries and pressurized internal contents, a process called passivation. If the satellite is not passivized before contact is lost, it is possible that the resulting loss of control will lead to heating beyond the design parameters resulting in an explosive breakup of the satellite.[14] To approach such a target necessitates the assumption that the target is *uncooperative* and *unknown* as the parameters are likely to have changed and it can not be assumed that any aids or references are available to assist the chaser. Another example would be the capture of an asteroid.

The *uncooperative unknown* scenario is the most challenging, and will be the sole scenario considered in this research.

1.1.2. Research Questions & Proposal

This research is preceded by a literature study preformed to understand the current state of the art in remote sensing for pose estimation, and to find any potential knowledge gaps worth exploring. This revealed a perceived lack of attention to pose estimation using thermal infrared sensors, and the fusion of infrared data with other sensors. Many existing pose estimation methods utilize monocular cameras in the visual spectrum, however these cameras are sensitive to the harsh lighting conditions in space. Fusing images with LiDAR point-clouds is a common way to resolve scale in an image, but fusing LiDAR and thermal infrared data would theoretically create a system independent of lighting conditions. Though LiDAR is able to determine pose on its own, the sensor often contains considerable inaccuracies and performs poorly specifically when viewing targets at unfavorable angles, such as head-on positions and when trying to detect small diameter objects such as antennae. By developing suitable LiDAR-infrared fusion and pose estimation algorithms and testing them on a diverse set of target scenarios, this research supports development in the field of spacecraft rendezvous and proximity operations and form the primary thesis objective: *Research the suitability of LiDAR-thermal infrared fusion to estimate the pose of an unknown uncooperative target*. With this objective in mind, the primary research question can be stated as:

How can data from LiDAR and thermal infrared sensors be fused to estimate the relative pose of an unknown uncooperative target?

The primary research objective and question are supported by the following sub-questions to provide a deeper and more thorough understanding, briefly clarified below:

RQ-1 What algorithm best suits the fusion and pose estimation goals?

a. At what point should the data fusion take place?

    Data fusion can take place in different phases of the pose estimation process. The choice of where the fusion takes place is dependent on the specific application and scenario and can have broader implications on the performance of the wider algorithm.

b. How can the fusion be calibrated?

    LiDAR returns 3D data while the thermal infrared sensor returns 2D images, and as they are not in the same location in space, the alignment of these two reference frames is crucial to obtaining accurate results.

c. What computer vision tasks can aid the process?

    Computer vision is used to gain insights from data obtained from electro-optical sensors such as LiDAR and thermal infrared. There is a wealth of computer vision and image processing tools available to assist any potential pose estimation algorithm, however some are more appropriate than others for this research.

d. How will the pose be estimated?

    The result of any pose estimation algorithm should fully describe the 12 degrees of freedom for relative pose numerically, and be able to be represented in different frames of reference.

RQ-2 How does the method developed perform?

a. How accurate is the pose estimation?

Though accuracy is not the primary goal, the estimation should be convincingly similar to the ground truth recorded during data gathering. If this is not the case then a detailed explanation of the cause of the inaccuracies will be required.

b. How can an experiment be designed to test the algorithm?

The algorithm must be supplied with data that is specifically gathered for the purpose of this research to be able to perform a comprehensive test of the method developed. Laboratory conditions should strive to represent the real-world use case for this research within reasonable means of terrestrial-bound testing.

c. What are the limitations of the LiDAR and thermal infrared sensors?

Both the LiDAR and thermal infrared sensors have advantages and limitations. Both of these will be thoroughly explored.

d. What scenario is this method best suited for?

After experimentation, testing and the resulting pose outputs, it should become clear what this method does well and what its limitations are. A subsequent use case scenario can then be formulated in which this research can be applied to.

e. What are the advantages and disadvantages of the method?

What are the strengths of this method and what were the contributing factors? The same analysis can be done on the weaknesses of this method.

This document is intended for researchers, academics and engineers working in the field of active debris removal, particularly those interested in multi-modal sensor fusion for relative pose estimation of unknown uncooperative targets. It is also valuable for those developing algorithms for LiDAR-infrared sensor fusion and those exploring light-agnostic solutions for remote sensing in space applications.

1.1.3. Research Scope & Applications

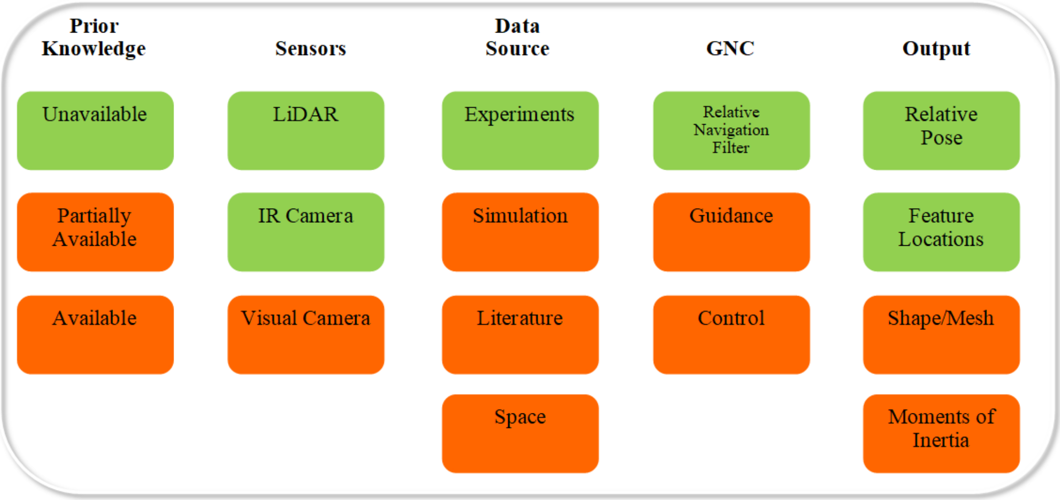


Figure 1.1: Items included (green) and not included (orange) within the scope of the research presented.

This research will develop and test a method of pose estimation by fusing data retrieved from a thermal infrared sensor with that of a LiDAR sensor. The targets will be unknown and uncooperative, and the method developed will operate under the assumption that no prior knowledge of the target's shape, size or composition will previously be known before sensing. This research focuses on a specific phase in the rendezvous process which would occur at relatively close distance between the chaser and target. The phases before this (such as angles-only approach) and after (such as capture) will not be considered. Figure 1.1 shows a visual overview of what is and is not included in this research's scope.

The discussion portion of this document, specifically Section 3.3.4 will describe a specific use case for the methods developed during this research as well as some recommendations for further development in Section 4.3.

1.1.4. Document Overview

Chapter 1 first detailed the background, motivation and main questions of this research, and will continue with a knowledge-primer for concepts and technologies relevant to the development and results of the research performed. Chapter 2 will then detail the development of the calibration, fusion and pose estimation algorithms before describing in detail the various iterations of the verification and validation experiments, with a special emphasis on the thought process behind the choices made in both. Chapter3 will then proceed to describe the results of the methods developed and continue to analyse and discuss the results as they pertain to the stated research questions, also including a discussion of a possible use case for these methods. Finally, Chapter 4 will revisit and answer the research questions posed, and draw some conclusions on the results presented as well as recommendations for further research.

1.2. Remote Pose Estimation Overview

This Section will serve as an introductory primer to the essential concepts utilized and discussed in Chapter 2 and Chapter 3. Some of these concepts go beyond the usual realm of aerospace subject matter, with computer vision being a prime example. Section 1.2.2 also serves to show the relevance of the methods and topics covered by describing missions in orbit that have utilized some of the instruments and technology integral to this research.

1.2.1. Rendezvous & Proximity Operations

The field of rendezvous and proximity operations (RPO) in orbit plays a pivotal role in the broader scope of space missions, encompassing critical maneuvers and delicate interactions between spacecraft. As the demand for satellite constellations, orbital servicing, and interplanetary exploration intensifies, mastering the art of precisely controlled RPO becomes indispensable. This section provides a foundational overview of the concepts, techniques, and significance of RPO in the realm of space exploration.

Rendezvous in orbit refers to the intricate process of bringing two spacecraft together at a designated location and time. This operation can serve various purposes, such as docking, satellite servicing, or the formation of satellite constellations. Proximity operations, on the other hand, involve the careful navigation of spacecraft in close proximity to each other without physical contact. Both rendezvous and proximity operations demand meticulous planning, precise control, and a thorough understanding of orbital mechanics.[23]

The importance of RPO in orbit is underscored by its diverse applications across different space missions. Docking maneuvers are crucial for spacecraft delivering cargo to the International Space Station (ISS) or assembling large structures in space. Satellite servicing missions leverage RPO techniques to extend the operational life of satellites, refuel spacecraft or repair malfunctioning components. Furthermore, the formation and maintenance of satellite constellations demand precise proximity operations to ensure optimal orbital configurations. Of particular interest to this research is the importance of RPO in Active Debris Removal (ADR).

RPO presents unique challenges due to the complex nature of orbital dynamics and the need for real-time adjustments. The vastness of space, combined with the high velocities of orbiting objects, requires advanced algorithms and control systems. Techniques involve utilizing sensors, such as LiDAR, radar and optical sensors, for relative navigation, as well as implementing autonomous control systems to adapt to unforeseen changes in the orbital environment.[22]

1.2.2. Electro-Optical Sensors & Computer Vision

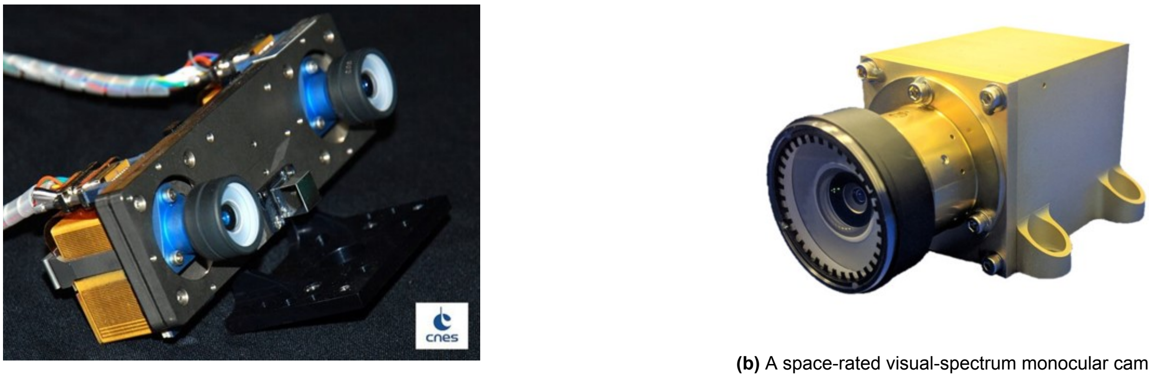
The term electro-optical sensor is used to refer to a diverse group of sensors capable of sensing electromagnetic radiation (either emitted or reflected) anywhere within a wavelength rage between 0.01 µm and 1000 µm, corresponding to the ultraviolet (UV), visible and infrared (IR) spectra ranges.[24] Electro-optical sensors have a wealth of flight heritage in space and there are a wide variety of commercial space-rated products available. Electro-optical sensors can be divided into two groups, passive and active, as discussed below.

Passive Electro-Optical Sensors

Passive sensors do not illuminate their targets, instead simply collecting the EM radiation that enters the sensor. They are characterized by their relative simplicity and low power consumption, with the added benefit of being able to be used in inspection tasks.[42]

Monocular cameras, as seen in Figure 1.2b, have one lens and can be sensitive to UV, visible or IR wavelengths, though almost almost all RPO literature referenced here refer to monocular cameras in the visible spectrum. As the literature study has shown, there has been much more research on visible-spectrum monocular cameras for RPO compared to those in other spectra. Monocular sensors return 2D images of the target and can be in color or grayscale. Visible-spectrum cameras benefit from having comparatively high resolution compared to other electro-optical sensors, however they are extremely sensitive to lighting conditions and unfavorable phenomenon such as artifacting. All monocular cameras represent the recorded photon wavelengths as pixel values, which is something integral regarding their suitability for an application.

Stereocameras, seen in Figure 1.2a are essentially two monocular cameras separated by a distance known as its baseline. This configuration allows for the sensor to output 3D images with the same high resolution benefit of monocular cameras, however stereocameras are strongly limited by the baseline as it determines the maximum operating range that it is able to collect 3D data from. Similarly to monocular cameras, it is also particularly vulnerable to lighting conditions.



(a) A space-rated stereovision system from MCSE[49]

(b) A space-rated visual-spectrum monocular camera system from MCSE[8]

Figure 1.2: Space-rated passive vision systems available for purchase

It is useful to mention that the above sensors can all be turned into active sensors by supplying their own artificial illumination. For example, a target in the dark can be illuminated with a flashlight and then observed in the visible spectrum.

When faced with the choice of passive electro-optical sensor selection, it is important to understand the source and behavior of electromagnetic radiation in various spectra. For the purposes of RPO and ADR, visible and UV sensors would detect the light in their spectrum reflected off of the target. IR sensors, however can detect radiation emitted by the target itself without having to rely on an external source such as the sun, as all objects emit radiation in the infrared spectrum based on their emissivity and temperature as given by Planck's Law. This is based on the so-called blackbody approximation, an idealized body that absorbs all incident electromagnetic radiation. Expanding further, Wein's Law defines the peak wavelength emitted by an object as being solely dependent on temperature.[24] For the purpose of ADR, targets are not likely to be ”white hot” so their emissivity will primarily be in the long-wavelength infrared (LWIR) spectrum, and therefore not require any illumination if observing with a sensor sensitive to this range. This will be discussed further using academic sources in Section 1.2.3. As shown in Table 1.1, the IR spectrum corresponds to black-body radiation associated with a specific temperature range. It is important to note that these temperatures correspond to the *peak* wavelength, and that an object will also radiate other wavelengths at the same temperature, but with a lesser intensity.

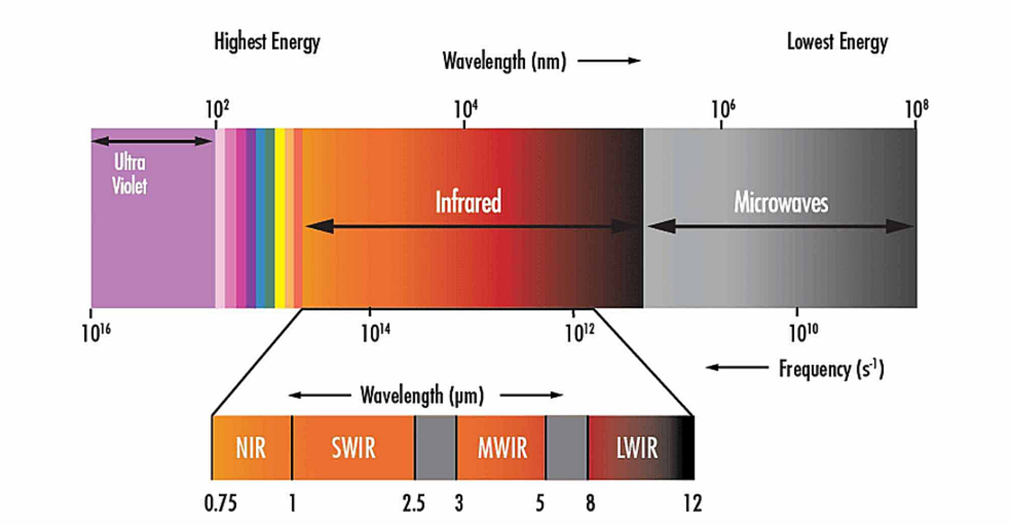
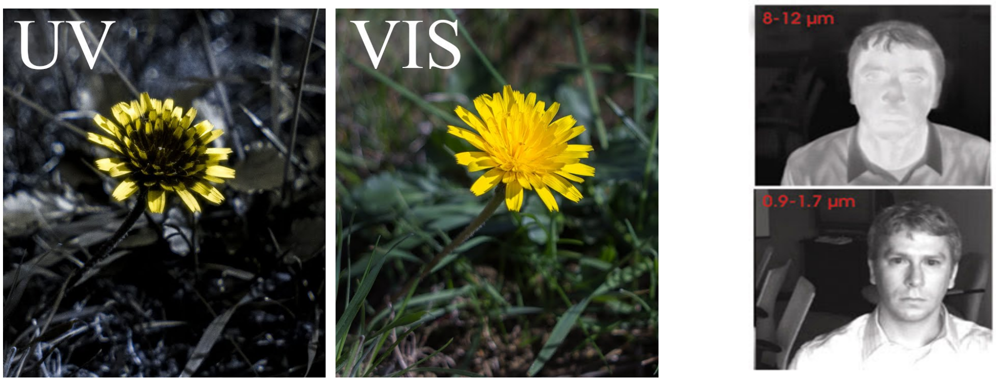


Figure 1.3: Infrared Spectrum [59]

Division	Abbreviation	Wavelength	Temperature
Near-infrared	NIR	0.75 to 1.4 μm	3591 to 1797 ° C
Short-wavelength infrared	SWIR	1.4 to 3 μm	1797 to 693° C
Mid-wavelength infrared	MWIR	3 to 8 μm	693 to 89° C
Long-wavelength infrared	LWIR	8 to 15 μm	89 to -80° C
Far infrared	FIR	15 to 1000 μm	-80.15 to -270.15° C

Table 1.1: Peak infrared wavelength corresponding to temperature definitions[38]

Visual spectrum sensors are integral to modern life, and are usually characterized as having a high resolution compared to other spectra of sensors. However, visual sensors are sensitive to lighting conditions, being blinded by strong light and darkness. IR sensors do not share this downside. LWIR sensors are effectively able to see in the dark as it collects the emitted radiation within its spectral temperature range. The downside of LWIR sensors is that they generally have lower resolution, though the need for very high resolutions may not be necessary for all applications. As LWIR sensors are highly agnostic to lighting conditions, they offer an attractive proposition for this research. Though other electro-optical sensors exist in other spectra, as shown in Figure 1.4, only LWIR offers the characteristic light-invariance at non-extreme temperatures.



(a) Image in UV and Visible spectra[43]

(b) LWIR (top) vs SWIR (bottom) images[24]

Figure 1.4: Imaging in different wavelengths

Active Electro-Optical Sensors

Active sensors emit their own EM radiation that they then collect and measure the backscatter of. These sensors typically use more power than their passive counterparts but have a number of unique benefits. One active sensor of particular interest is LiDAR.

LiDAR illuminates targets with an infrared laser source, usually in the Near-Infrared Range (NIR), and provides both 3D position and intensity data in the form of point clouds. LiDAR is known for generating relatively sparse point clouds compared to stereovision's dense 3D range images, however this leads itself to being less computationally burdensome. LiDAR is robust to lighting conditions and can operate in eclipse, with the sun or Earth in the scene, and can image the ”dark” side of the target without issue. As shown in Figure 1.6, there are two overarching categories of LiDAR, pulsed (sometimes called time-of-flight TOF) systems and continuous wave (CW) systems. As the name implies TOF systems operate on the principle of measuring distance through timing the radiation in reference to the speed of light. CW systems measure distance by measuring the phase difference between the emitted signal and the reflected echo, exploiting what is known as the heterodyne principle[42]. CW LiDAR is better suited for close ranges, however they can experience problems with blurring due to relative motion.[24]

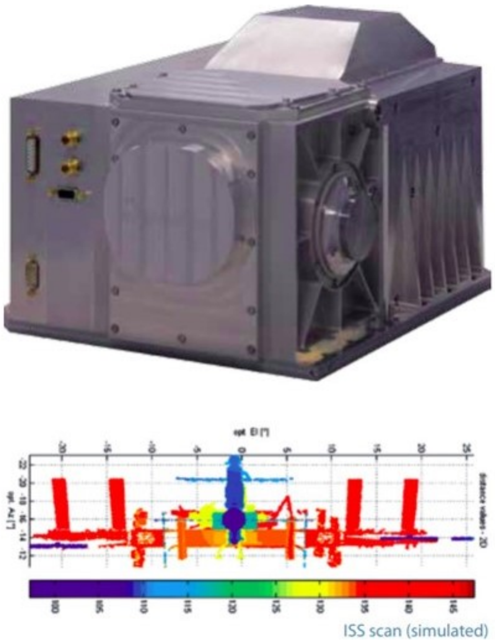


Figure 1.5: A space-rated LiDAR system from Jena-Optronik [46]

An important parameter of any LiDAR is the point density. This refers to the number of points per unit of area at a target distance. Many LiDAR sensors emit points at a diverging angle from its center line in order to view a larger scene at distance. However this means that the point density decreases with distance and, as it will be further discussed in Section 2.2.1, scanning pattern when applicable.

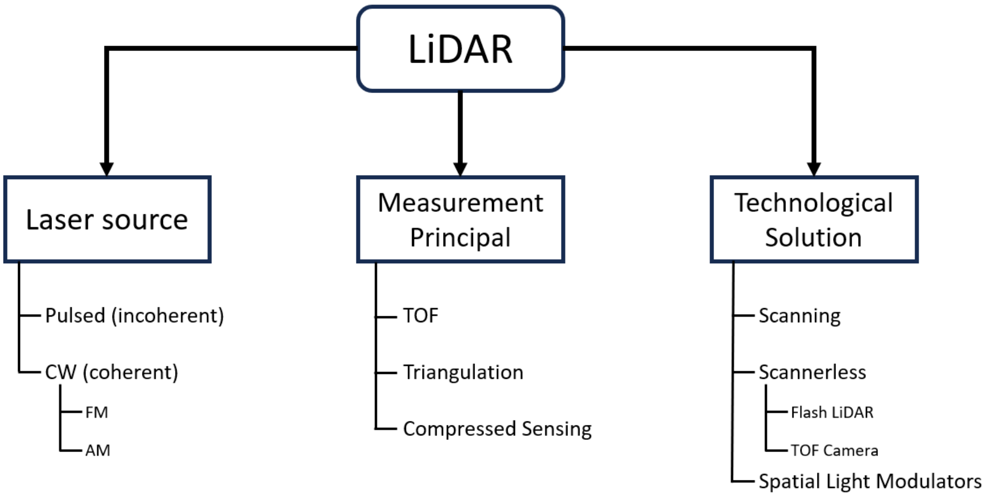


Figure 1.6: Taxonomy of LiDARs, recreated from [42]

As LiDAR sensors are able to retrieve 3D position information independent of lighting conditions, they are an attractive option for use in spacecraft, particularly with relative navigation. The choice of LiDAR type depends on the mission profile, however this research will be interested in TOF scanning LiDAR due to their high point density (points to m^2 related to distance), versatility in scanning pattern type, and wide availability.

Though there are many attractive strengths, LiDAR sensors can also suffer from small inaccuracies and noise, particularly with very small or translucent objects. Figure 1.7 below shows a point cloud in which the points corresponding to a target of interest blend into the space between the target and the background, while also omitting smaller portions of the target's geometry. In the following section, Figure 1.9e shows a LiDAR scan made of a target in orbit, but it appears to have large gaps in the middle of its solar panels, gaps which do not exist on the physical target. The author of the source article for this image does not give a reason for this error, however this effect will be seen again during the course of this research and seems to be a result of to the complicated nature of photons leading to omissions in the scan. Due to these reasons, LiDAR can benefit from an additional data source when sensing an unknown uncooperative target. This is a key aspect of this research and reinforces the value of sensor fusion.

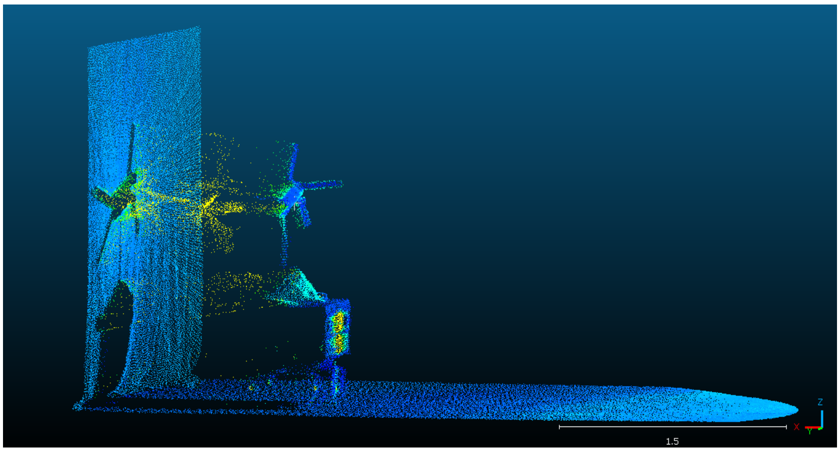
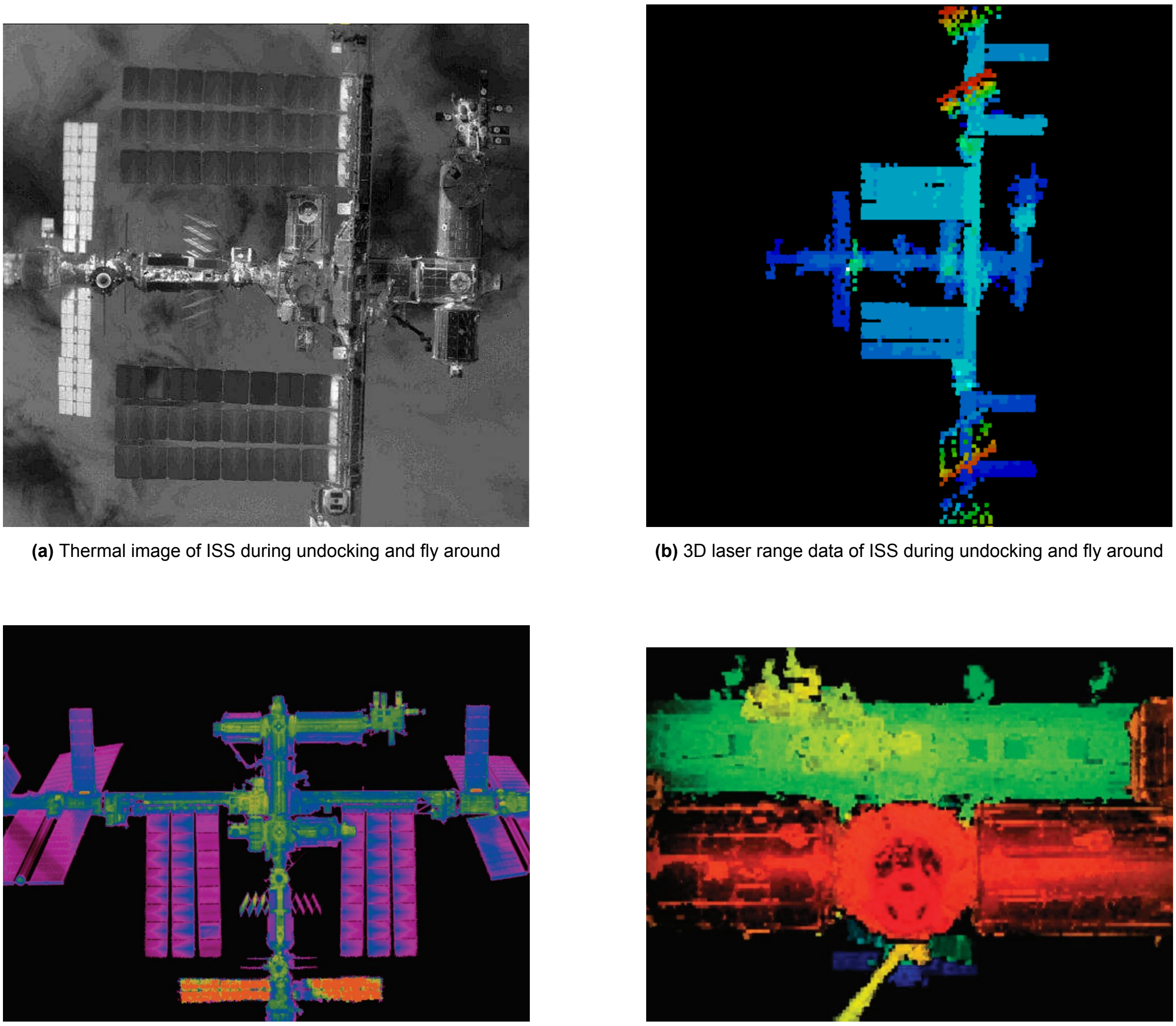


Figure 1.7: Noisy LiDAR recording with points blending from a target of interest into the background

The next section will examine some relevant examples of the use of electro-optical sensors and includes an example, MEV-1 & 2, which utilized the TOF scanning LiDAR shown above in Figure 1.5.

Electro-Optical Sensors in Orbit for Non-Earth Observation

There are many examples of spacecraft using electro-optical sensors to observe the Earth from orbit. As this research focuses on sensing other resident space objects, missions involving non-Earth observation (which to avoid confusion with Near Earth Orbit, will not be abbreviated) are of much more relevance. An interesting instrument with flight heritage is the TriDAR (Triangulation + LIDAR) Automated Rendezvous and Docking instrument flown on the Space Shuttle Discovery during STS-128 and STS-131, Space Shuttle Atlantis during STS-135,[28] and continues to fly on the Cygnus resupply craft to the ISS.[40] TriDAR uses two laser-based sensors and a thermal imager to assist in rendezvous and docking with the ISS. Figure 1.8 below shows real laser-range and thermal IR data obtained by the TriDAR instrument during STS-135. It is important for this research to note that TriDAR implementation for pose estimation and navigation relied on a model-based approach,[28] therefore it has only been used on cooperative known targets.



(a) Thermal image of ISS during undocking and fly around

(b) 3D laser range data of ISS during undocking and fly around

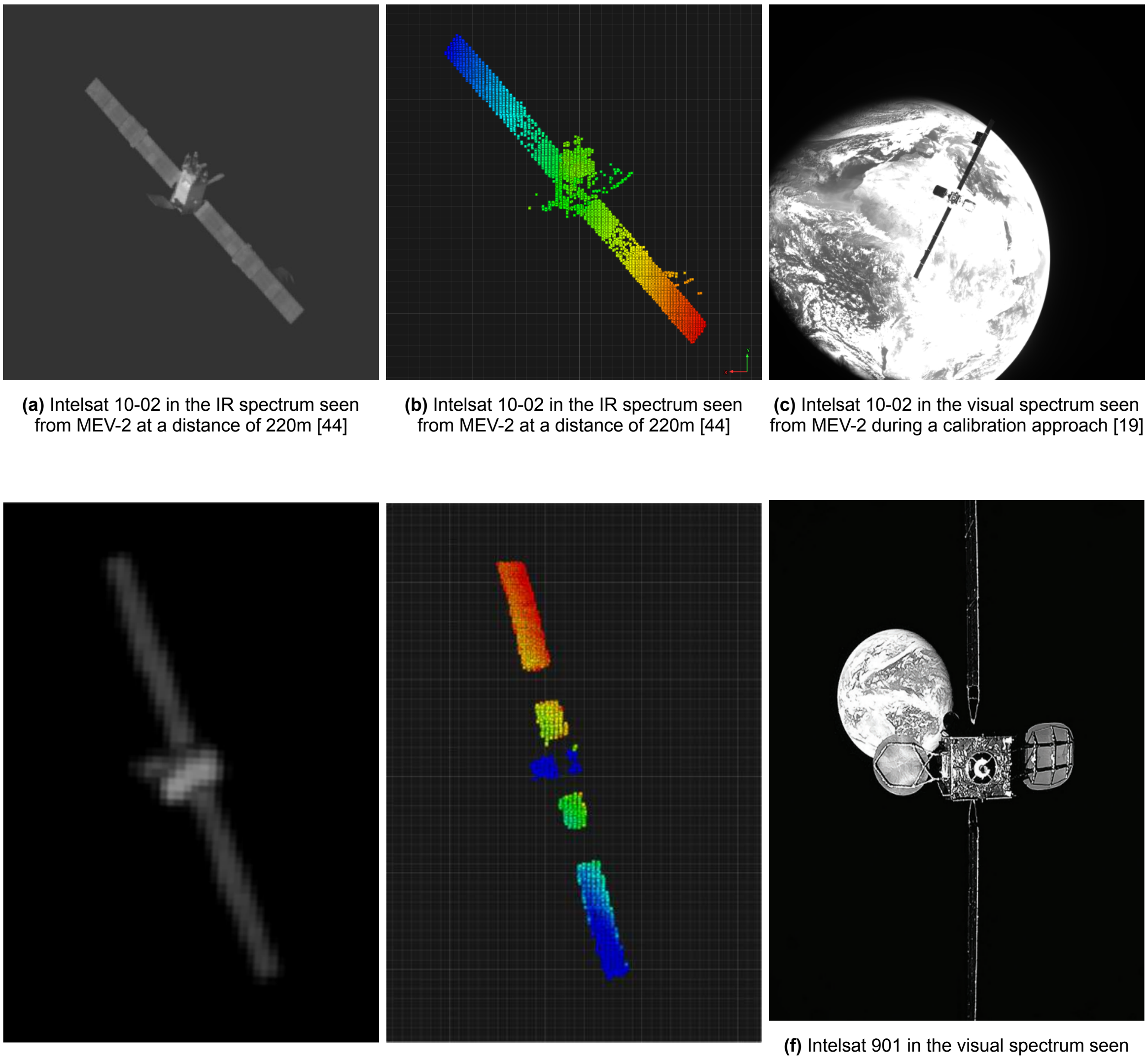
(c) False-color thermal image of ISS at 200m

(d) 3D laser range data of the ISS docking port

Figure 1.8: STS-135 TriDAR data obtained from [28]. Figures 1.8a and 1.8b are recorded at the same distance, though this distance is not specified.

The Mission Extension Vehicle (MEV) is an On-Orbit Servicing (OOS) proof of concept vehicle built by Northrop Grumman with so far two missions (MEV-1 & MEV-2) successfully rendezvousing and attaching to their targets (Intelsat 901 & Intelsat 10-02 respectively) in geostationary orbit. The MEV vehicle is designed to remain attached to its target and provide station-keeping propulsion for a duration of 5 years, before potentially continuing on to another target.[45] As the satellites targeted by MEV are operational and known a-priori, these missions fall into the known cooperative target category as well.

Figure 1.9 shows images captured during MEV-1 & MEV-2 in different phases of approach. MEV has a sensor suit including a visual spectrum camera set, LWIR camera set and a scanning LiDAR.[44] It is interesting to see real remote sensing images in orbit, particularly with the Earth in the background presenting a segmentation and recognition challenge for on-board computer vision tasks expected in such an OOS mission. Note that Figures 1.9d and 1.9e are a LWIR image and a LiDAR scan respectively at the same distance, as are 1.9a and 1.9b.



(a) Intelsat 10-02 in the IR spectrum seen from MEV-2 at a distance of 220m [44]

(b) Intelsat 10-02 in the IR spectrum seen from MEV-2 at a distance of 220m [44]

(c) Intelsat 10-02 in the visual spectrum seen from MEV-2 during a calibration approach [19]

(d) Intelsat 10-02 in the IR spectrum seen from MEV-2 at a distance of 2000m [44]

(e) Intelsat 10-02 point cloud seen from MEV-2 at a distance of 2000m [44]

(f) Intelsat 901 in the visual spectrum seen from MEV-1 at its ”far hold” position at a distance of 80m [45]

Figure 1.9: Images taken by MEV-1 & MEV-2 of their respective targets in geostationary orbit during approach

Both Tridar and MEV offer valuable real-world examples of the fusion between LiDAR and thermal infrared sensors, which is a key aspect of this research. Particularly with the MEV, the TOF scanning LiDAR used was able to return accurate range data at distances in excess of 2 km, and the uncooled microbolometer LWIR used allowed for the tracking of the target in excess of 10 km, with the target fully resolve at 2 km distance.[44] The performance of these sensor suits was outstanding and gave the MEV a level of flexibility in approaching the target that surpasses the use of any individual sensor alone and allowed the chaser to operate independent of lighting conditions. This shows how these sensors can be used for RPO in space. Though both the MEV missions and the use of TriDAR have solely been used for known cooperative targets, their LWIR-LiDAR fusion show great promise towards implementation in an unknown uncooperative ADR scenario. The MEV missions will also allow for a comparison between real-world and laboratory conditions, and will be reexamined in Section 3.3.4 once results are obtained.

Computer Vision

Computer vision (CV) is a category of remote sensing that enable the derivation of meaningful information from images and data collected from electro-optical sensors.[21] Some common CV tasks involve object detection and tracking, image processing, facial recognition, segmentation and stitching, classification, mapping and much more. The CV industry is seeing particular success in a number of areas, including autonomous vehicles and robotics, medical imagery, sports analysis, military applications, virtual reality and smartphone applications.[6]

One task in CV that is of particular interest to this research is local feature detection. This refers to the detection of localized patterns or structures within an image corresponding to a region that differs from its surroundings in terms of texture, color or intensity. These often correspond to edges, corners, points or any localized patch. A number of feature detection methods have been developed with different performance metrics in mind, such as detection speed for real-time use, robustness to lighting, accuracy above all else, or scale-change invariance. Not all of these methods detect the same types of features. Harris, FAST, ORB and Shi & Tomasi methods detect exclusively corners, while SIFT, SURF, and KAZE detect blob-features. Blob features refer to descriptors of color intensity and texture.[33] Choosing the right feature detection method depends of the application in consideration of the distinct characteristics of each. A number of feature detectors will be studied further and compared in Section 2.1.4.

It is important to note that these features can act as anchor points within each image and can be used to match and compare images based on matching features having the same descriptors. This will be referred to as Feature Matching and will become an integral part of this research. Features can be matched by their descriptors falling within a specified range to indicate similarity, as certain metrics might differ between images based on lighting, perspective or background for example. The tuning of both the detector and matcher is a tedious process, as false-matches can come about from using the wrong tuning metrics.

1.2.3. The Space Environment's Effect on Debris

All missions to space must contend with the harsh and unforgiving environment present above the Karman line. Characteristics of the space environment include ionizing radiation, temperature extremes and a hard vacuum as altitude rises. With electro-optical sensors in mind, it is useful for this research to examine the expected temperature ranges of an ADR target in orbit. As the discussion of electromagnetic radiation highlighted, objects emit radiation in the infrared spectrum at a peak wavelength based on their temperature, and can be viewed without external illumination by using the correct sensor corresponding to this wavelength.

Component or Subsystem	Operating Temperature (°C)	Survival Temperature (°C)
General electronics	-10 to 45	-30 to 60
Batteries	0 to 10	-5 to 20
Infrared detectors	-269 to -173	-269 to 35
Solid-state particle detectors	-35 to 0	-35 to 35
Motors	0 to 50	-20 to 70
Solar panels	-100 to 125	-100 to 125

Table 1.2: Typical spacecraft component temperature limits [3]

Table 1.2 shows common temperature ranges for spacecraft components, however this table was created for engineers to anticipate *operational* temperature limits in their thermal design. Operational spacecraft have electronics and moving parts that contribute to the overall thermal flux, along with thermal input from the sun and Earth, while always managing the fine thermal balance needed to operate properly. Targets for ADR will be inoperable and no longer able to control their thermal balance, but also do not receive heat from internal electronics or moving parts. These targets will likely experience a range of heating and cooling towards a passive thermal equilibrium, with temperatures depending on their particular orbit and illumination conditions as well as the absorptivity and emissivity of the object itself.

As a general reference, the temperature of a debris object T_{deb} as can be calculated using Kirchhoff's Law, the thermodynamic expression for the absorbed and emitted energy, and by assuming the absorbed and emitted energies are uniformly distributed from the entire surface:

$$\tau \pi r^2 E_{inc} = \varepsilon (4\pi r^2/3) \sigma T_{deb}^4 \tag{1.1}$$

where r refers to the object's radius, τ the absorptivity, ε emissivity, E_{inc} the incident flux from the sun and/or Earth, and σ being the Stefan-Boltzmann constant. The debris temperature can then be given by:

$$T_{deb} = \left[\frac{3\tau E_{inc}}{4\varepsilon\sigma} \right]^{1/4} \tag{1.2}$$

thus the temperature does not depend on size and only the ratio of absorptivity and emissivity. An equilibrium temperature T_e independent of object characteristics can be defined as:

$$T_e = \left[\frac{3E_{inc}}{\sigma} \right]^{1/4} \tag{1.3}$$

In this definition, the equilibrium temperature is only dependent on the incident flux, and therefore its orbital height. The debris temperature can be related to the equilibrium temperature with the following equation: [26]

$$T_{deb} = \left(\frac{\tau}{\varepsilon} \right)^{1/4} T_e \tag{1.4}$$

A Dynetics study on space debris characterization using thermal imaging systems found that debris at an altitude of 1000 km would have a maximum theoretical equilibrium temperature (independent of τ and ε) of 315°K (42°C) for sunlit and 255°K (-18°C) for non-sunlit conditions.[11] This is equilibrium temperature is well within the LWIR spectrum, which corresponds to temperatures within about -80°C to 89°C, referencing Table 1.1 from Section 1.2.2. Another study of LEO debris thermal characteristics experimented with varying absorptivity-emissivity ratios in different LEO orbits and found a minimum temperature of -90°C and a maximum of 40°C at both extremes.[60]

These temperature ranges clearly show an opportunity for debris observation in the LWIR spectral range, as it is the only passive sensor able to detect debris in the expected temperature range independent of illumination.

1.2.4. Sensor Fusion & Calibration

The integration of electro-optical sensors has emerged as a pivotal research area, driven by the demand for sophisticated sensing systems in applications ranging from robotics to autonomous vehicles. At the heart of this pursuit lies the challenge of fusing data from diverse optical sensors and ensuring their accurate calibration. This subsection provides an overview of the fundamental concepts, techniques, and challenges in electro-optical sensor fusion as a prelude to the subsequent research endeavors.

Sensor Fusion

Sensor fusion is not a new concept. Through evolution, humans have learned to incorporate the fusion of our own senses to survive, such as viewing, smelling and tasting a potential edible plant, or listening and looking through a forest for a predator. Furthermore, human sight from two eyes is another example of sensor fusion allowing for depth perception, but also can be undermined by irregularities such as astigmatism. This intuitive anatomical perspective of sensor fusion can be useful for understanding the electro-optical sensor fusion relevant to this research, as well as its benefits and challenges.

Electro-optical sensor fusion refers to the combination of information derived from disparate optical sensors, encompassing cameras, LiDAR, and other optical technologies as discussed in Section 1.2.2. The overarching objective is to harness the complementary strengths of each sensor type to overcome individual limitations, thereby enhancing the overall perceptual capabilities of the system. In doing so, the fusion process enables a more complete and accurate representation of the surrounding environment. Electro-optical sensor fusion sees widespread implementation in a variety of applications, from autonomous cars[7] to landmine detection for humanitarian de-mining[10], but also for remote sensing in orbit[50].

Sensor fusion can occur at different stages between the retrieval of raw data and the final application. The Joint Directors of Laboratories (JDL) Data Fusion Model developed an important standardized blueprint for data fusion widely used in defense applications as well as other non-military applications, and describes three main methods for data fusion: direct fusion of sensor data (Figure 1.10a), feature-based fusion of sensor data (Figure 1.10b), and decision-based fusion of sensor data (Figure 1.10c). Direct fusion assumes that the sensors are viewing the same entity (association) and combines the data at the earliest moment, with raw data or after some initial preprecessing. Feature-based fusion extracts key features from each sensor to match and align the data. Finally, decision-based fusion combines sensor data after the sensors have made a preliminary determination of the attributes of an entity in the scene, such as location, classification, identity and size.[4]

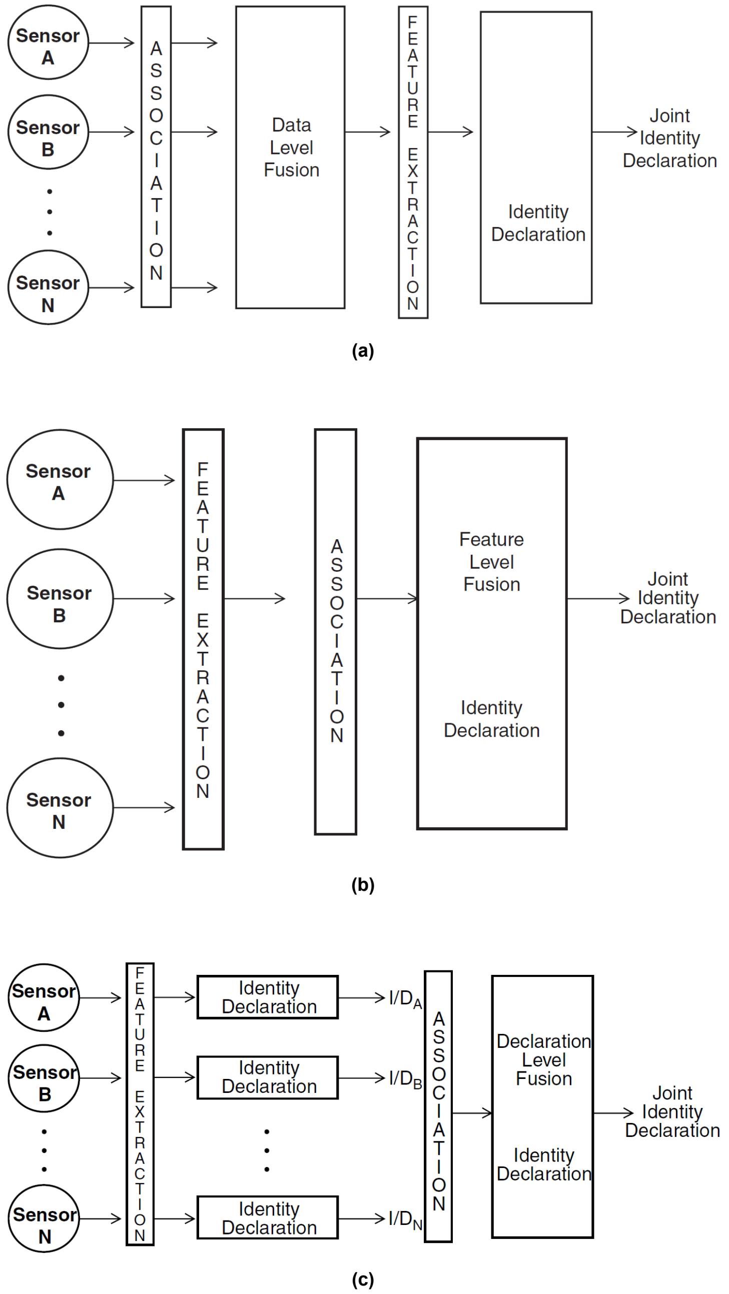


Figure 1.10: JDL Sensor Fusion Model and its various architectures[4]

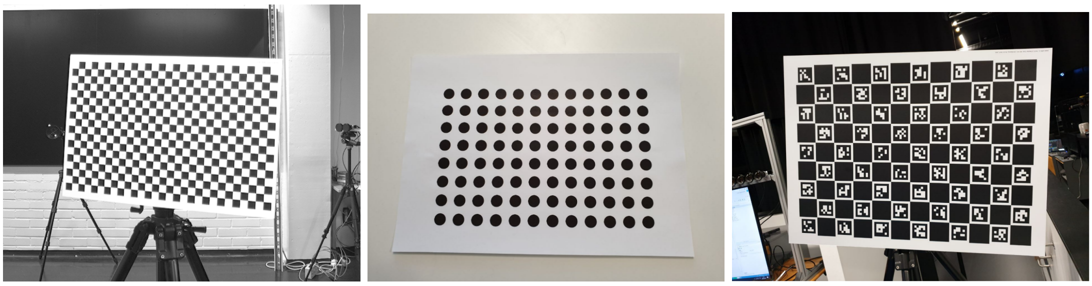
The choice between these three fusion architectures is dependent on the scenario and application in which the fusion is being used. Decision-based fusion is useful in very busy scenes in which only a portion sensed is of interest. In this architecture, the system first identifies objects in the scene before applying the fusion, effectively segmenting the data into zones of interest. For spacecraft applications, the expected scene in an RPO mission could have the Earth, Sun, Moon, stars or other small points of light in the background, however the object of interest will not be in a cluttered environment. For this reason, the decision-based fusion architecture would not be appropriate for RPO missions, and the remainder of this research will consider only direct (or data-level) fusion or feature-based fusion.

Calibration

The effectiveness of sensor fusion hinges upon meticulous calibration procedures that address the inherent differences between optical sensors. Calibration is multifaceted, encompassing intrinsic parameters such as focal length and lens distortion, and extrinsic parameters for aligning sensors in a common coordinate system. These calibration challenges underscore the need for sophisticated techniques to ensure the accuracy and reliability of the fused sensor data.

Intrinsic calibration focuses on defining the internal characteristics of each sensor, essential for accurate fusion. Techniques like Zhang's camera calibration algorithm play a crucial role in determining parameters such as focal length and principal point, while also mitigating lens distortions.[62]

Aligning sensors in the external world coordinate system is achieved through extrinsic calibration. Employing known patterns, such as the three common patterns shown in Figure 1.11, facilitates simultaneous calibration of multiple sensors, crucial for accurate fusion in real-world scenarios. As the sensors or not located in the same point in space, extrinsic parameters provide a way to define and compensate for this difference in perspective in the fusion.



(a) Checker/Chessboard calibration pattern **(b)** Circle-grid calibration pattern **(c)** ChArUco calibration pattern

Figure 1.11: Commonly used camera calibration patterns[57]

Accurate calibration is essential for this research and will be further discussed in Section 2.2.2.

1.2.5. Pose Estimation & Transformation Representation

Pose is defined as the combination of attitude and position of the target relative to the chaser. Relative pose consists of 6 degrees of freedom (DOF) parameters, 3 translational DOF (position) and 3 rotational DOF (attitude). Taking the first derivative of these parameters leads to 12 DOF which can be useful for describing rotational and translational velocities, each with 6 DOF. This can be visualized as describing an airplane in 3D space, its roll, pitch and yaw angles, and the change in these parameters with time for a full 12 DOF relative pose description.

Considering now the full range of relative navigation scenarios, such as docking, ADR, formation flying and orbital servicing, 4 different relevant pose estimation types can be identified:[42]

- Actively Cooperative
- Passively Cooperative
- Uncooperative Known
- Uncooperative Unknown

A target is considered to be cooperative if it was built to aid in pose estimation. An example of actively doing so would be docking with the ISS. The target (ISS) is actively transmitting attitude and position data to the chaser to aid in relative pose estimation. An example of passive cooperation would be if a target had light patterns or scale length markings designed to aid the chaser in pose estimation.

Conversely, uncooperative targets do not contribute to the relative navigation process. In this instance, the targets can be separated into categories of known and unknown. An example of an uncooperative known target would be a spent rocket stage that is still intact. It does not contribute to the relative navigation process, actively or passively, however it is known in the sense that its parameters (length, shape, mass) are known from its manufacturing and launch specifications. An example of an uncooperative unknown target would be the remains of a satellite after a collision. It is not able to cooperate with the chaser and its parameters have changed due to the collision.

Uncooperative unknown targets represent the biggest challenge to potential ADR missions. As detailed in the Research Scope, this category of target will be the focus of the research.

Relative Coordinate Systems

When describing the full 12 DOF relative pose, it is important to understand how the change in position and attitude is described mathematically in the different coordinate systems present in the chaser-target system. Two main coordinate systems will be relevant for the remainder of this research: the 3D chaser world coordinates and the 3D target intrinsic coordinates, a representation of which is shown below in Figure 1.12.

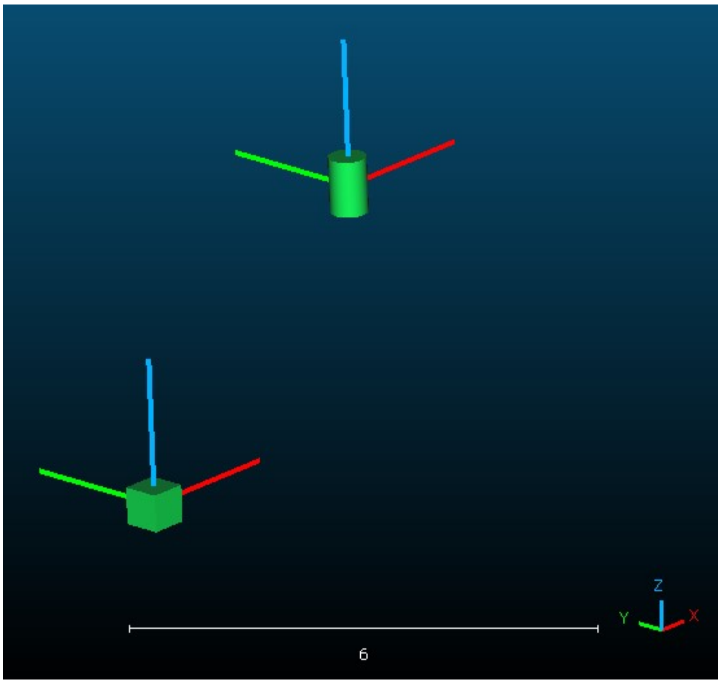


Figure 1.12: World coordinate axes of the chaser (box), and the intrinsic coordinate axes of the target (cylinder).

The world, or global, coordinate system has three orthogonal spacial position coordinates (x, y, z) with the origin in the center of the chaser. The LiDAR sensor used in the subsequent Chapter 2 defines the orientation of these axes in which positive x protrudes directly through the sensor in the direction that the sensor suite is facing. Positive y is horizontally to the left of the sensor suite, and positive z upwards vertically. A target in the top left quadrant of the sensor's field of view will have positive (x, y, z) position values, while a target in the bottom right quadrant will have positive x but negative y, z values. As the LiDAR used in this research is forward-facing, a target will not be in the field of view if its x coordinate is negative as it can not be sensed with this particular type of LiDAR, however there are many rotating LiDAR sensors with a 360° toroidal field of view capable of sensing targets "behind" $(-x)$ and "along side" (small $|x|$, $\pm y$) the sensor, and see frequent use in autonomous vehicles.

The target intrinsic coordinates also have these three orthogonal (x, y, z) spacial coordinates, but its origin is in the center of a bounding box containing its perceived volume. As the target is unknown and uncooperative, the orientation of these axes are aligned to the orientation of the chaser world axes, with each axis being parallel to their respective world axis. So if the target is in the top left quadrant of the chaser's field of view, the chaser's position in target intrinsic coordinates will have negative (x, y, z)

values. If the target is in the bottom right quadrant, the chaser's position will have negative x and positive y, z values.

The bounding box containing the target's perceived volume will likely not encompass the entirety of the target's actual volume. If the chaser observes a spherical target, it's initial view will only be that of a hemisphere. The other side hidden from view cannot be predicted in this initial observation. However, this center point of the bounding box, the origin of the target intrinsic coordinate system, can either be updated when more of the target is observed, or remain constant as it is an arbitrary point relating the two systems. An initial LiDAR scan of the target will return (x, y, z) world coordinate values, and the minimum and maximum of each axis can be used to compute the center point of the bounding box with the following equation:

$$\left(\frac{x_{max} - x_{min}}{2} + x_{min}\ ,\ \frac{y_{max} - y_{min}}{2} + y_{min}\ ,\ \frac{z_{max} - z_{min}}{2} + z_{min}\right) \tag{1.5}$$

The bounding box describes the initial position of the target relative to the chaser as well as its initial attitude, oriented to align with the world coordinate axes as mentioned. Though it may seem intuitive to orient the target intrinsic axes based on the target's geometry, it is useful to remember that the target has not yet been fully observed, and that an arbitrary initial axis orientation is just as useful if it is consistently tracked. Consider a spherical target, which would have no intuitive intrinsic axes. This example makes the decision to implement a consistent arbitrary definition of the target's intrinsic axes based on the world axes easier to understand.

Matrix Representation of Transformation

These two coordinate systems are essential for describing not only the position and attitude, but also the observed relative motion of the target. This observed motion can be described as the relationship between two target states at two separate observations. The transformation between these two states can be describe with a transformation matrix A consisting of a rotational matrix R and a translation matrix T, with $A = T \bullet R$. The rotation matrix depends on the plane in which the rotation occurs, with multi-axis rotation leading to a more complicated result. For single-axis rotation, the other two rotational matrices will simply be 4x4 identity matrices, so that the total rotational matrix will be equal to the rotational matrix of that axis.

Though the translation matrix is intuitive to interpret, the matrix representation of rotation can seem confusing and vague. Luckily it can easily be converted to Euler angles, axis angles and quarternions for a more intuitive visualization.

$$T = \begin{bmatrix} t_x & t_y & t_z \end{bmatrix} \tag{1.6}$$

$$R_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cosd(\theta_x) & -sind(\theta_x) \\ 0 & sind(\theta_x) & cosd(\theta_x) \end{bmatrix} \tag{1.7}$$

$$R_y = \begin{bmatrix} cosd(\theta_y) & 0 & sind(\theta_y) \\ 0 & 1 & 0 \\ -sind(\theta_y) & 0 & cosd(\theta_y) \end{bmatrix} \tag{1.8}$$

$$R_z = \begin{bmatrix} cosd(\theta_z) & -sind(\theta_z) & 0 \\ sind(\theta_z) & cosd(\theta_z) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{1.9}$$

$$R = R_z \bullet R_y \bullet R_x \tag{1.10}$$

$$A = T \bullet R \tag{1.11}$$

$$A = \begin{bmatrix} R(1,1) & R(1,2) & R(1,3) & t_x \\ R(2,1) & R(2,2) & R(3,2) & t_y \\ R(3,1) & R(3,2) & R(3,3) & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1.12}$$

For example, the chaser observes the target once, and then again after 5 seconds have past and deduces that the target has changed position by -1 meter in the world coordinate x-axis, 0.5 meters in the y-axis, and 0.25 meters in z. This can be described by a translation matrix T in which $T = [t_x, t_y, t_z]$, which in this case $T = [-1, 0.5, 0.25]$.

This becomes more complicated when considering the intrinsic and world coordinate frames, as the origin of the target intrinsic frame is dependent on its Euclidean distance to the origin of the world frame. For example, if a target rotates 5° about its z intrinsic body axis and translates -0.5 meters in the y-axis, this motion can be represented by the following intrinsic transformation matrix:

$$A_{intrinsic} = \begin{bmatrix} cosd(5) & -sind(5) & 0 & 0 \\ sind(5) & cosd(5) & 0 & -0.5 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

However this would not be the world transformation deduced by the observations of the chaser, as the target does not rotate around the world z-axis and the translation would be different to accommodate this perspective shift. The aligning translation matrix T_{align} is the translation required to position the origin of the intrinsic coordinate system at the origin of the world coordinate system. The total transformation matrix relating world coordinated and intrinsic coordinates can be defined as follows:

$$A_{intrinsic} = T_{intrinsic} \bullet R_{intrinsic} \tag{1.13}$$

$$A_{align} = T_{align} \bullet \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{1.14}$$

$$A_{world} = T_{align} \bullet A_{intrinsic} \tag{1.15}$$

$$A_{total} = A_{world} \bullet A_{align} \tag{1.16}$$

This total transformation matrix can be used to describe the change in position and orientation of the target, which with two observations in time, is used to estimate the full 12 DOF of relative pose.

2 Research Framework

As it has been shown in the previous sections, the fusion of a LWIR camera with a LiDAR sensor offers the enticing possibility to perform remote sensing for the relative pose estimation of an unknown uncooperative target completely independent of lighting conditions while providing the ability to resolve scale with both 2D and 3D data. From this prospect, as well as the main research questions defined in Section 1.1.2, a framework in which to test, challenge and answer these questions can be developed.

This chapter will first detail the algorithms developed for pose estimation of an unknown uncooperative target using fused LIWR-LiDAR data. After this, the various experiments designed for verification and validation (V&V) of the algorithms will be described. Special emphasis will be placed on the decision making process throughout both the algorithm and the experiments' development. Section 2.1 details the development and final form of the pose estimation algorithm before Section 2.2 thoroughly explains how the experimentation and data collection process evolved and improved to better supply relevant data to the algorithms.

As initial experimentation commenced and the hardware's abilities and limitations were explored, the algorithms were improved to better consider the realities of data gathering, instead of operating on purely theoretical concepts. Due to this, early hardware experimentation influenced many of the decisions made in the development of the algorithm, such as the tuning of algorithmic parameters, which was critically dependent on the scenarios of each experiment and could only be explored after some experimentation had taken place. It is for this reason that images from the experiments will be used to illustrate concepts from the algorithms. Computer vision requires vision after all.

Thus it is important to note that the algorithm developed was not implemented on-line, or live at the time of experimentation, but after data gathering stopped to be able to tune parameters to achieve accurate results. The implementation of on-line algorithms is beyond the scope of this research, but would be of critical importance to a real ADR mission in which autonomy is key.

2.1. Algorithms for Pose Estimation

Initial algorithmic experimentation, particularly with computer vision and point cloud manipulation, was done using C++ and Python, and included tools such as OpenCV, Point Cloud Library and CloudCompare. However MATLAB demonstrated its value to this research by offering a much more streamlined and thorough set of tools to achieve exactly what this research sets out to do. MATLAB also provided a wealth of intuitive documentation and scholarly sources for their tools that made a deeper understanding of the sources of the results possible.

Initial experimentation with calibration, fusion, lighting and target types led to major revelations that would shape the final product of both the developed algorithms and the experiments designed to verify and validate them. This Section details the growth and development of these algorithms as well as their final form, with the following Section 2.2 describing the hardware side of the learning process as well as the final experiments used for algorithm V&V.

Increasing familiarity with point clouds led to the initial formulation of an Iterative Closest Points (ICP) algorithm that would attempt to relate an IR-infused reference point cloud to a moving one repeatedly, effectively describing the change in position and attitude of the target in relation to the chaser over time. Sensor fusion would add an extra reference dimension to the matching and make for an interesting pose estimation solution. Once this was explored, it was decided that there could be another approach to consider in which the LWIR sensor provided more than just color information. This is when the Feature Matching algorithm was developed, focusing more on computer vision and image processing with the added benefit of LiDAR range information to track feature locations between images. Sections 2.1.3 and 2.1.4 will detail the inception and development of these algorithms in much greater detail.

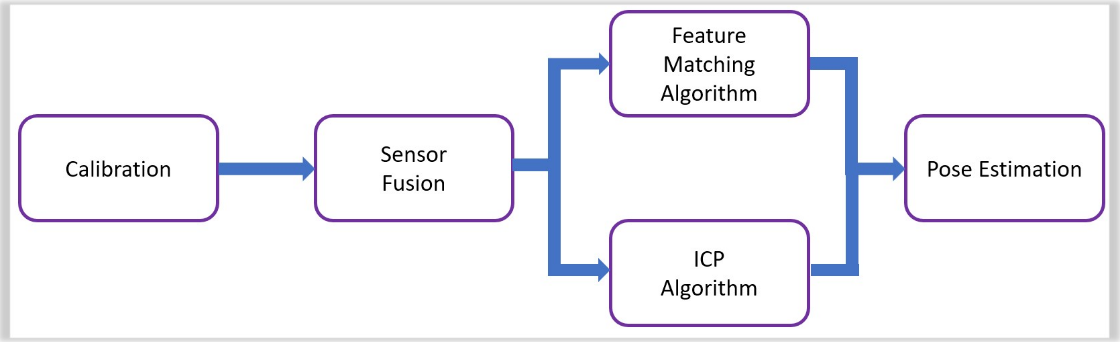


Figure 2.1: Process developed for pose estimation

As shown above in Figure 2.1, both algorithms use the same calibration and sensor fusion methods, and both return a pose estimation for the target. The calibration, fusion and both algorithms with their resulting pose estimation will be covered in detail in the following sections.

2.1.1. Calibration and Sensor Fusion

The calibration process developed for the LiDAR and IR sensors can be thought of as localized sensor fusion with subsequent fine tuning. First the intrinsic parameters of the IR camera are estimated as well as its extrinsic parameters relating to its position in space. As mentioned in Section 1.2.4, intrinsic parameters relate to the focal length, principal point, lens distortion and other undesirable effects to be corrected for. After this, the extrinsic parameters relating the IR and LiDAR sensors' positions in space are estimated. From this, an initial estimation of the relation between the IR pixels and LiDAR world points can be made resulting in fusion. The final step is a feature-based fine-tuning of the pixel-world point relation to achieve the best possible fusion and the most accurate relation between the LiDAR and image points. The fused data contains 3D world coordinates, pixel coordinates and IR color information as shown below in Table 2.1. It is important to note that the resolution of the two sensors are not equal and will result in image pixels without world-coordinates if there is no LiDAR point associated with that pixel location. This is dependent on the LiDAR point density at the target's distance and location in the LiDAR Field-of-View (FOV) and will be further discussed in relation to the hardware available in Section 2.2.1. Figure 2.2 below shows a workflow for the calibration and fusion steps, for which further detail will be given in the paragraphs immediately below.

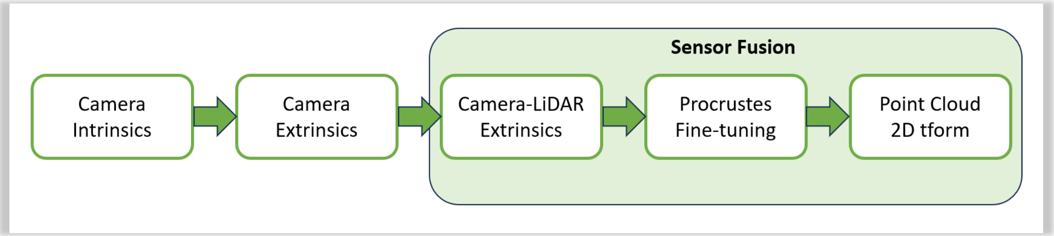


Figure 2.2: IR-LiDAR calibration and fusion process

Data Property	Source Sensor
X-world	LiDAR
Y-world	LiDAR
Z-world	LiDAR
X-pixel	IR
Y-pixel	IR
color	IR

Table 2.1: Properties of the resulting fused data points

In MATLAB the *detectCheckerboardPoints* function used in camera calibration returns a set of arguments detailing the detected checkerboard corner coordinates (cornerPoints) in the image as well as the details of the checkerboard in terms of the number of squares (boardSize). The boardSize as well as the size of the squares in mm can be used to create the worldPoints object representing the world coordinates of the squares. With cornerPoints and worldPoints as inputs, the *estimateCameraParameters* function returns a cameraParameters object in which the intrinsic, extrinsic and lens distortion parameters of the camera are stored.[29] The reprojection error per calibration image, as well as a visualization of the extrinsic parameters can be viewed with the *showReprojectionErrors* and *showExtrinsics* functions respectively.[31] The calibration function used in MATLAB is known as "Zhang's Method" developed for the Microsoft Corporation[62] and is a widely used standard with lower reprojection error compared to other methods such as Tsai's method.[61] Equation 2.1 shows Zhang's calibration algorithm relating a point in world coordinates (X, Y, Z) to the corresponding image pixel (x, y) used in the aforementioned MATLAB *estimateCameraParameters* function. Here, w is an arbitrary scale factor, K is the camera intrinsic matrix, R is the 3D rotation matrix of the camera and t is the translation of the camera relative to the world coordinate system.[29] Equation 2.2 defines the camera intrinsic matrix K with components (x_0, y_0) being the pixel coordinates of the principal point, α and β being the scale factors in the x and y axes respectively, and γ representing the skew of the two image axes.[62]

$$w \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = K \begin{bmatrix} R & t \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{2.1}$$

$$K = \begin{bmatrix} \alpha & \gamma & x_0 \\ 0 & \beta & y_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.2}$$

The next step in the calibration process is to calibrate the LiDAR and camera together. The LiDAR data acquired from the available Livox Mid-70 sensor is saved as a .lvx file, but can also be converted to a .las file in the Livox Viewer application. This then needs to be formatted into a file type that can be used by MATLAB, so the application CloudCompare can be used to convert the .las file into a .ply file. Now the retrieved LiDAR data can be used in the algorithm.

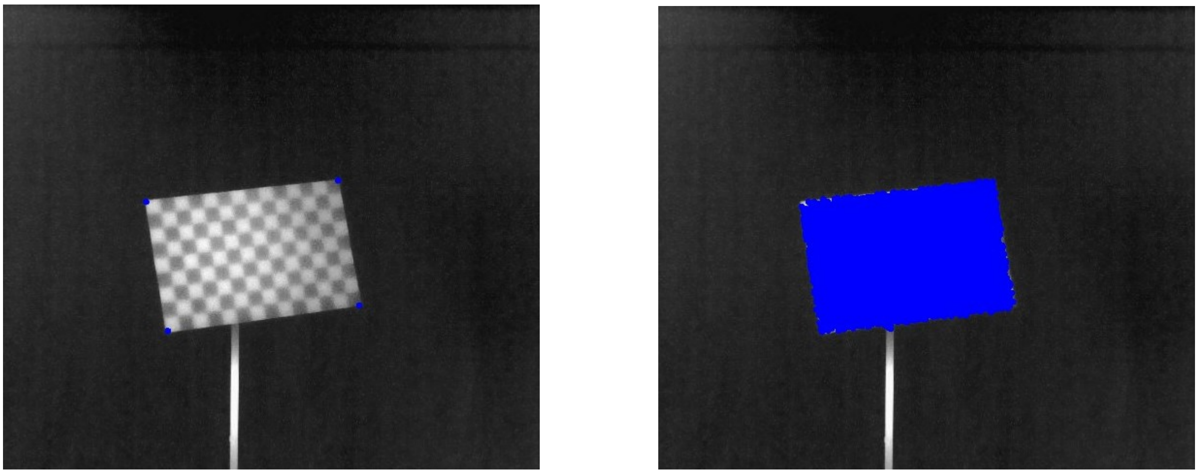
The LiDAR point-clouds, in practice, contain many irrelevant points that are of no concern to either the calibration process or the pose estimation process. The calibration of LiDAR with a camera relies on the detection of a rectangular plane (the checkerboard) in the scene, and in practice it was often confused by other rectangular planes in the scene such as walls and boxes. For this reason, the calibration process relied on manually selecting the section of the point-cloud containing the calibration target which could be done in MATLAB's Lidar-Camera Calibrator application, though this leads only to a ballpark estimate with the result only serving as a preliminary solution before further tuning for an accurate result.

Procrustes and Fine-tuning the Fusion

Procrustes analysis, also referred to as least-squares orthogonal mapping, gets its name from the mythical Greek inn owner Procrustes and is a mathematical technique used in statistics and geometric shape analysis to align, compare, and analyze the similarity between two sets of geometric shapes. The primary goal of Procrustes analysis is to find the optimal transformation (translation, rotation, scaling, or reflection) that minimizes the differences between corresponding points in two sets of shapes.[17]

For the purposes of this research, Procrustes analysis can be used to more accurately calibrate the LiDAR and IR sensors by selecting distinct features, such as corners, in the two coordinate systems and then estimating the transformation between the two sets of feature locations. This in practice leads to a more accurate transformation compared to the MATLAB LiDAR-Camera Calibration App and is also a good way to check that the calibration is accurate for each target and dataset. The result of an accurate calibration is a 2D affine transformation (affinetform2d) that relates each LiDAR point to its corresponding 2D image pixel coordinate. This can be used to project the LiDAR points onto the IR image which is something that will be critical in Section 2.1.4.

Checking the calibration with Procrustes analysis gives a good visual indication for successful calibration, but it also returns the procrustes distance, which is a measure of dissimilarity between the inputted points. This effectively provides a numerical validation of the accuracy of the calibration, with the minimization of the procrustes distance being the goal for accurate calibration.



(a) LiDAR corners matching image corners

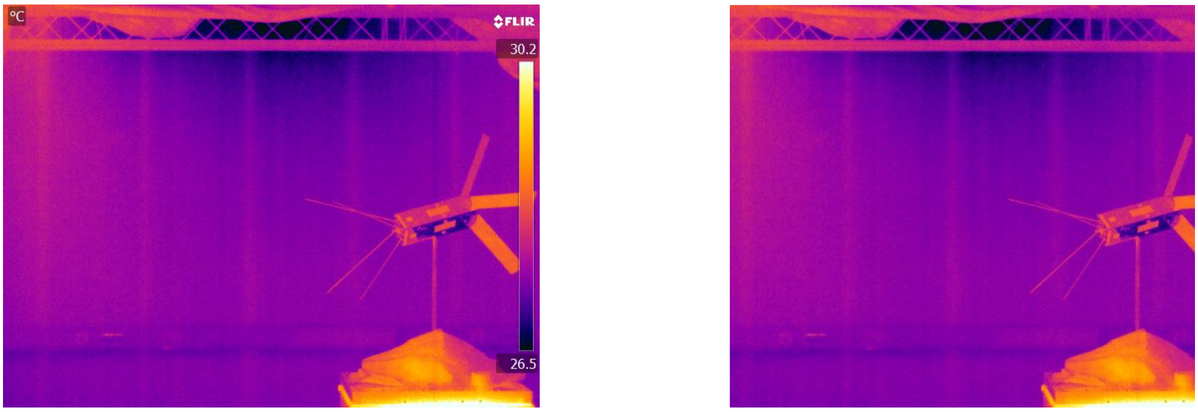
(b) The full LiDAR checkerboard can be projected

Figure 2.3: Accurate calibration using Procrustes Analysis

The calibration and initial reference fusion provides a template for data fusion during the experiments. With the LiDAR and LWIR coordinate frames matched, the algorithms are accurately able to relate the data obtained from both sensors; undistorted pixels to LiDAR coordinates and vice versa. A projection of the feature-based tuning and fusion displayed as a projection of LiDAR points on its corresponding IR image is shown above in Figure 2.3.

2.1.2. Data Preprocessing

The calibration process could be done with minimal preprocessing. The only problem was that the FLIR watermark and temperature scale were not able to be removed when collecting data. This means that every image had to be cropped to the same dimensions which unfortunately reduces the effective field of view. Cropping these undesirable overlays out reduced the image size from 640x480 to 555x480 pixels.



(a) Raw FLIR image

(b) Cropped FLIR image

Figure 2.4: The unfortunate loss of features due to cropping out the FLIR watermark and temperature scale

Another preprocessing step essential to the algorithms described in the following sections is the segmentation of the LiDAR data. This involves defining a region of space outside of which all other points in a point cloud will not be considered in the analysis. As the experiments were done indoors, this region would only contain the target and empty space and would remove the floor, ceiling, walls and anything else in the field of view. Ideally, the point cloud of a target in orbit would only contain points of the target. Even if there was backscatter from another body in the field of view, the vast distances in the space environment would make segmentation a simple step to automate in that scenario.

The Delfi-n3xt target used in some of the experiments was connected to a stand via a transparent plastic rod that tended to reflect very little of the LiDAR's NIR radiation back to the sensor, effectively making it almost invisible in the point clouds. The larger Boxsat target had a much more substantial stand that was very prominent in the point cloud, but this could easily be segmented out by setting the lower limit of the segmentation bounding box to just under the height of the stand connecting to Boxsat.

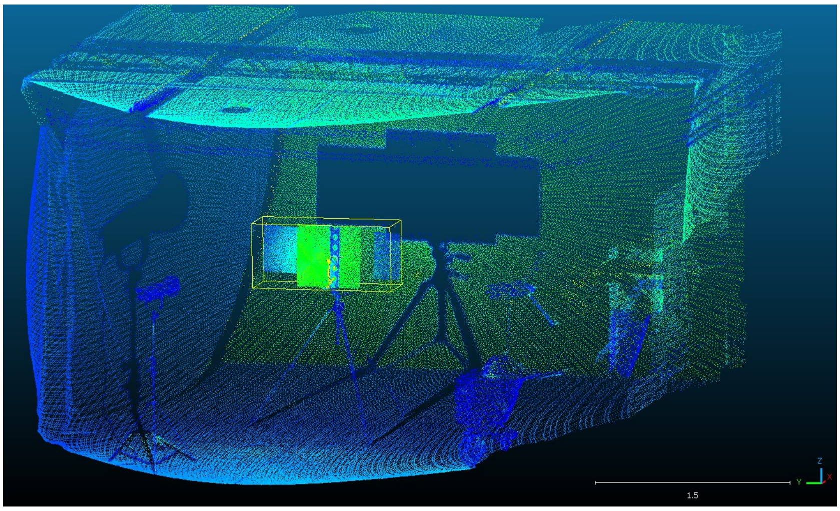


Figure 2.5: Resulting bounding box of points after defining the segmentation region

The preprocessing steps described here are generalized first steps to ready the data for use in the following sections describing the algorithms developed for pose estimation. The two algorithms below will make use of additional preprocessing techniques to achieve the best possible results and will be discussed further in their subsequent sections.

2.1.3. Color-ICP Algorithm

Accurate calibration returns a 2D affine transformation matrix relating each LiDAR point to its corresponding 2D image pixel coordinate. The resulting points are called imPts. By indexing these imPts and registering the corresponding RGB value of the pixel at their location in the image, an IR-colored point cloud is returned and can be used to determine pose. An example of the resulting IR-colored point cloud is shown below in Figure 2.6.

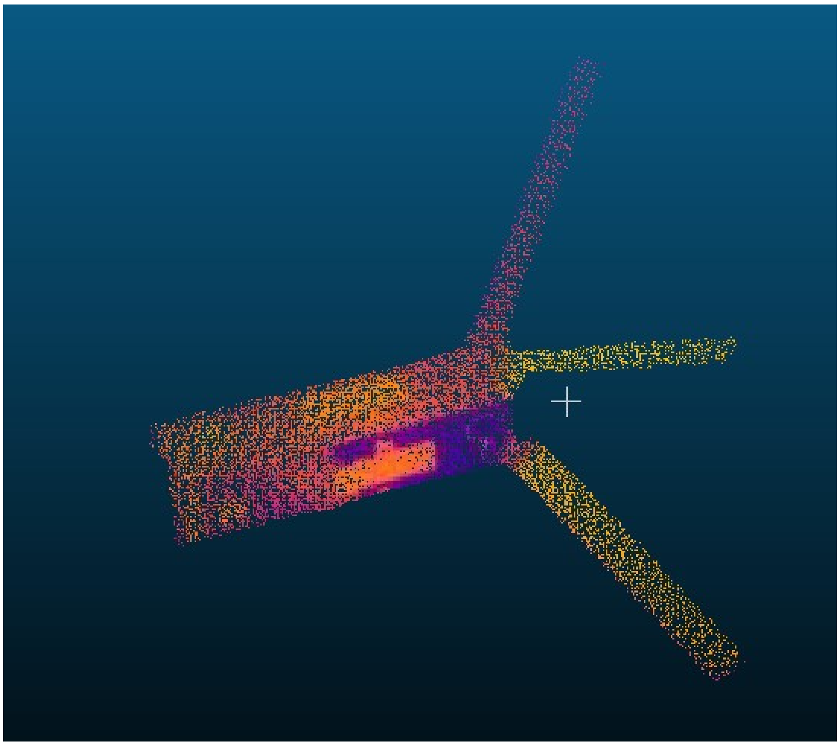


Figure 2.6: Fused point cloud of Delfi-n3xt during the Series J experiment

As it was outlined in Section 1.2.5, the pose of the target can be estimated by locating its position and orientation in 3D world coordinates and estimating the transformation matrix A describing how the position and orientation changes over time. One way to do this is by comparing the fused 3D data from the IR and LiDAR sensors over time with an Iterative Closest Points (ICP) algorithm.

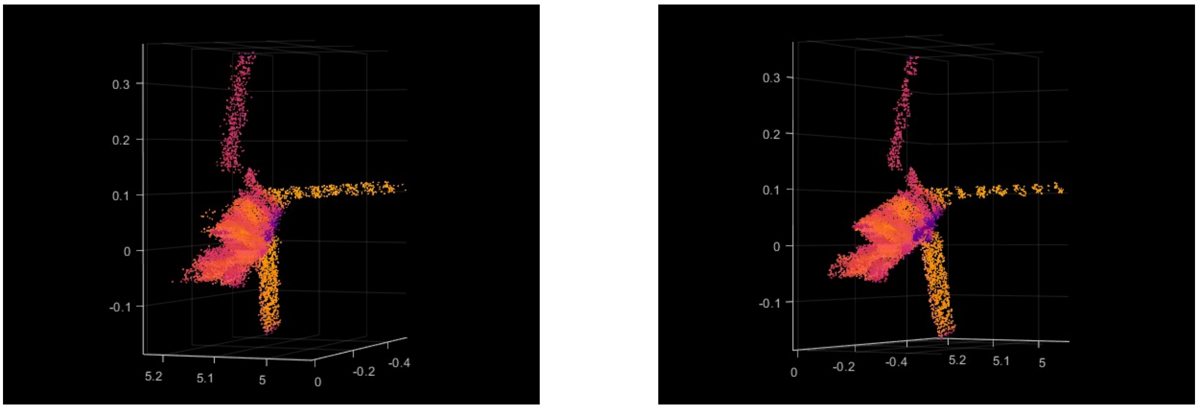
The ICP algorithm estimates the best transformation matrix needed that minimizes the distance between two point clouds (a fixed/reference point cloud and a moving point cloud), and iterates to reduce the error between the two.[9] In MATLAB's ICP function *pcregistericp*, the metric in which the distance between the two is minimized can be specified in different ways, two of which are relevant to the fused IR point clouds: point-to-plane with color, and plane-to-plane with color.[36] This means that the ICP algorithm that will be used in this research also takes color in to consideration in the minimization process and is not solely LiDAR sensor dependent, but actively being assisted by the IR sensor data.

The original ICP algorithm, the so-called Standard ICP, used a point-to-point minimization metric, in that it attempted to minimize the Euclidean distance between matching point pairs. This works well for identical point clouds in different locations in space, however it is not ideal for hardware applications in which point clouds are not identical and contain noise or small errors. To overcome this, new ICP algorithms were developed utilizing point-to-plane and plane-to-plane minimization metrics. These metrics take advantage of surface-normal information in the point cloud, which are vectors perpendicular to the surface at that individual point. The point-to-plane approach only considers the surface normals of the fixed point cloud, matching the moving points to the plane containing the fixed points. The plane-to-plane approach, known as Generalized ICP, considers the surface normals of both sets of points and matches the plane of the moving points to the plane of the fixed points. Further reducing the error, Korn et al. added color information to the point-to-plane and plane-to-plane approaches and demonstrated improved performance in a wide range of datasets.[25]

It is this color and surface-normal approach, plane-to-plane with color, that will be used in the bespoke Color-ICP algorithm developed for this research, as it gives the greatest amount of fused point cloud variables to be considered in the registration.

Further Preprocessing

After the initial preprocessing described in Section 2.1.2, some further steps are required to ready the fused data for the Color-ICP registration. Though the point clouds are already segmented, the target point cloud still contains some noise. The *pcdenoise* function can be tuned to remove outliers while maintaining the integrity of the more accurate points, though this step can, in theory, be skipped by tuning the Color-ICP function itself to ignore outliers.

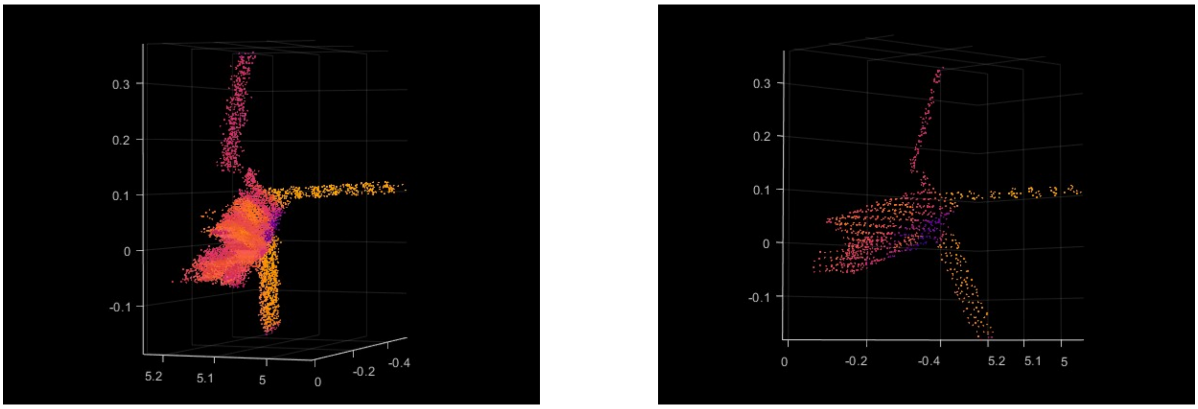


(a) Original colored point cloud with noise

(b) Colored point cloud denoised

Figure 2.7: Effect of the denoise step. The original point cloud has 6,225 points while the denoised point cloud has 5,716.

The next step is critical to the accuracy of the Color-ICP registration. Downsampling the point clouds effectively reduces the number of points by grouping points in a specified range into voxels, essentially small 3D boxes in space, and has the added benefit of computing the normals of each point in the new point cloud. Each voxel will contain only a single point computed based on the downsampling method selected. There are several downsampling methods, such as random, grid average, and non-uniform grid sample for the *pcdownsample* function in MATLAB. Grid average is used in this research, as it computes the voxel point value as the average of the locations, colors and normals of original points within the voxel, creating a more uniform shape and reducing unnecessary duplicate points. The non-uniform grid sample method does not tend to preserve shape while the random method picks a random point in the voxel as the sole voxel point.[35]



(a) Original colored point cloud.

(b) Colored point cloud denoised and downsampled

Figure 2.8: Effect of downsampling. The original point cloud has 6,225 points, while the denoised and downsampled point cloud has 869.

Figure 2.8 above shows the result of the Color-ICP preprocessing steps implemented. The number of points is decreased in relation to the denoising and voxelization steps, with the resulting point cloud organized into a uniform grid-like structure. With this step implemented on both fixed and moving point clouds, the algorithm can proceed to the main Color-ICP function.

Color-ICP Tuning and Implementation

MATLAB's *pcregistericp* function has multiple metrics to tune for greater accuracy, as shown in the list below. Inliers are accepted matching point pairs that contribute the the estimation of the resulting transformation matrix. This is particularly important in the case of a rotating target in which surfaces disappear between frames. If all points were to be considered as inliers, the algorithm would attempt to match these points between frames that have no match in reality. The acceptance of these matches as inliers is dependent either the Inlier Ratio or Inlier Distance Threshold as they cannot be used together in the function. The algorithm stops depending on which of the Tolerance or Max Iterations conditions are met first.

- **Minimization Metric** - As discussed above: point-to-point, point-to-plane, plane-to-plane, point-to-plane with color, plane-to-plane with color.
- **Max Iterations** - the maximum number of iterations estimating the transformation matrix, attempting to reduce the error between the two point clouds.
- **Inlier Ratio** - Specifies the percentage of matched points to be considered inliers. The default is 100% so it is critical to tune specifically with rotating targets or changes in perspective.
- **Inlier Distance Threshold** - the maximum distance between a fixed and moving point match to be considered as inliers.
- **Tolerance** - the tolerance between the rotation and translation matrices in consecutive iterations. The algorithm stops when three consecutive iterations are within tolerance.
- **Initial Transform** - useful if there is already an initial coarse transformation matrix estimate.

The result of the Color-ICP algorithm is a 3D rigid transformation object (*rigidtform3d*) containing the total transformation matrix A_{total} discussed in Section 1.2.5, as well as the final root mean square error (RMSE) between the matches that the algorithm attempted to minimize.

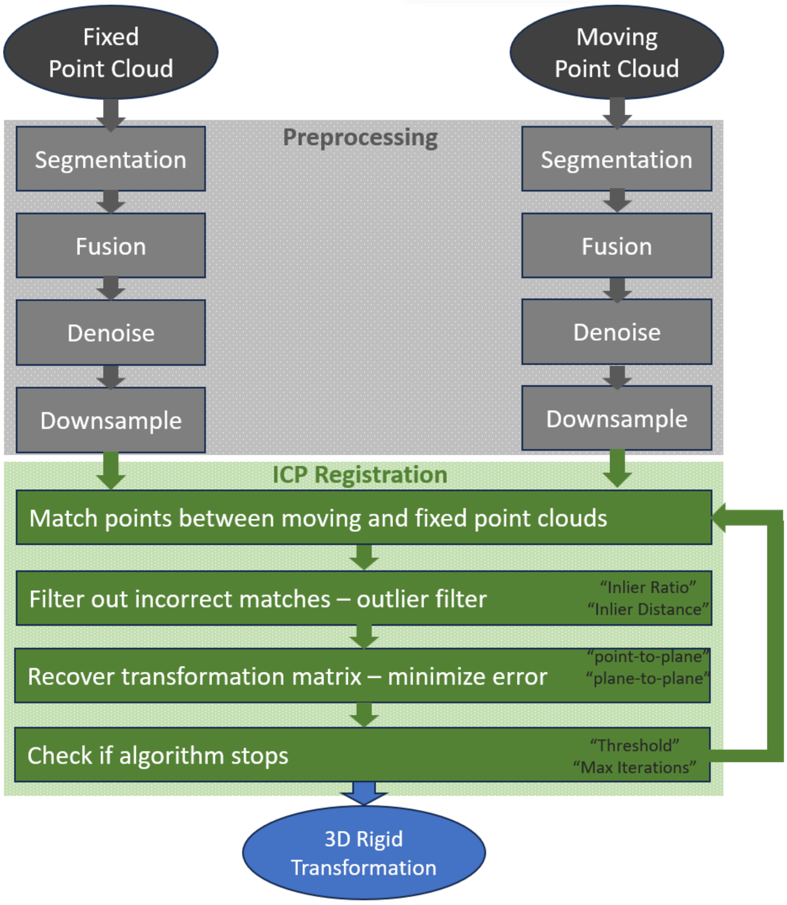


Figure 2.9: Color-ICP Algorithm work flow. Adapted from [36]

.

Figure 2.9 above shows a complete work flow of the Color-ICP algorithm from the process of point cloud segmentation to the final transformation output.

Control Transform and RMSE

Using the matrix methods of linear algebra described in 1.2.5, a control or ideal transformation matrix can be calculated based on the target's expected intrinsic translation and rotation documented in each experiment.

The control, or expected, transformation matrix and the observed transformation from the Color-ICP algorithm can be compared by calculating the root mean squared error (RMSE) between both the translation and rotation matrices. Separating the transformation matrix in this manner gives a better indication of what the algorithm is better estimating, translation or rotation. The formula for calculating RMSE is shown below, where C represents the control array, while O represents the observed array.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}|O_i - C_i|^2} \tag{2.3}$$

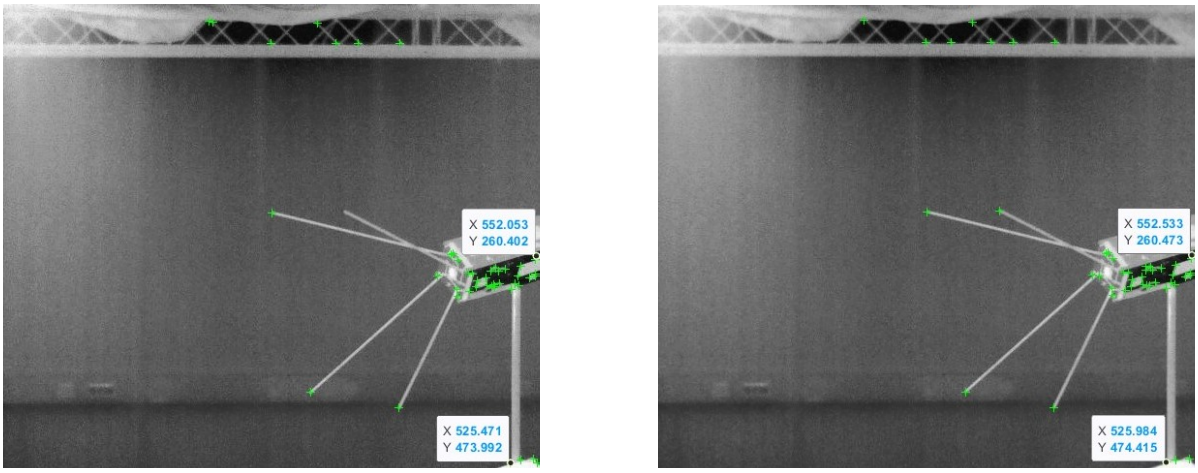
This can be directly implemented to the total translation matrix, however the rotational matrix requires an extra steps. First the rotation matrices are converted to Euler angles representing the rotation in each axis, done by using the *rotm2eul* function which returns a 1x3 matrix in the form of $[\theta_x\theta_y\theta_z]$ in radians. It is important to specify the order as "XYZ" as this is not the default and would otherwise lead to an inaccurate conversion. A preliminary check can be done using Equations 1.7-1.10 relating Euler angles to rotation matrices presented in Section 1.2.5. The Euler angle arrays can then be put into the RMSE equation above for comparison. A perfect match between the control and observed matrices would return $RMSE = 0$.

2.1.4. Feature Matching Algorithm

The Color-ICP algorithm benefits from sensor fusion by gaining an extra minimization metric with the addition of infrared color fused to each point. However this makes the Color-ICP algorithm much more dependant on the performance of the LiDAR sensor compared to the infrared sensor. Due to this, the decision was made to develop a new algorithm that would be more dependent on the infrared sensor and make better use of the fusion between the two.

The concept of the idea was to use image processing and computer vision to find features in the IR images and track them throughout the subsequent frames, which when fused with the LiDAR point clouds would mean that the (x, y, z) world coordinates of these matching features could be used to estimate the position, orientation and movement of the target to retrieve pose. As the target is rigid (not changing scale or shape in the real world), this movement can be estimated as a 3-D rigid geometric transformation (rigidtform3d) to estimate the rotation and translation matrices of the target, as was done in the Color-ICP Algorithm.

As this algorithm tracks features on the pixel level, an accurate calibration between the LiDAR and image coordinates is essential, the process of which was discussed in Section 2.1.1. When this is completed as accurately as possible, the algorithm outputs a 2-D rigid geometric transformation (geoform2d) of the LiDAR points in image coordinates, which will be called imPts. These 2D imPts are indexed with their corresponding 3D LiDAR points for a simple conversion between the two. The two sensors will likely not have the same resolution, therefore it is probable that not all pixels will have a corresponding imPt based on the LiDAR point density. As the algorithm depends on features being registered to imPts, a denser point cloud on the target will help to ensure that strong features are covered.



(a) Original Image

(b) Undistorted Image

Figure 2.10: The subtle difference in location of the same detected corners at the border in the original and undistorted images. This is a simple Harris corner detection.

With the IR camera intrinsic properties estimated during calibration, the first step is to undistort the two images being compared using the *undistortImage* function to rid the images of any negative effects such as distortion and stretching, as seen above in Figure 2.10. At this point, displaying the undistorted images and plotting the LiDAR imPts onto the image can show intuitively if the calibration in each case is sufficient. Remember that the IR camera and LiDAR are not in the exact same point in space, so the two differing views can lead to seemingly exposed areas on the overlay.

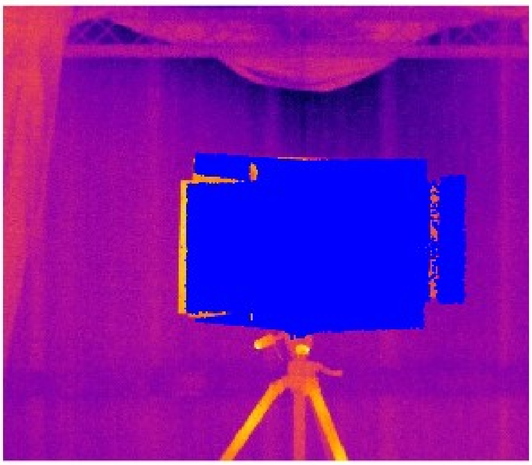
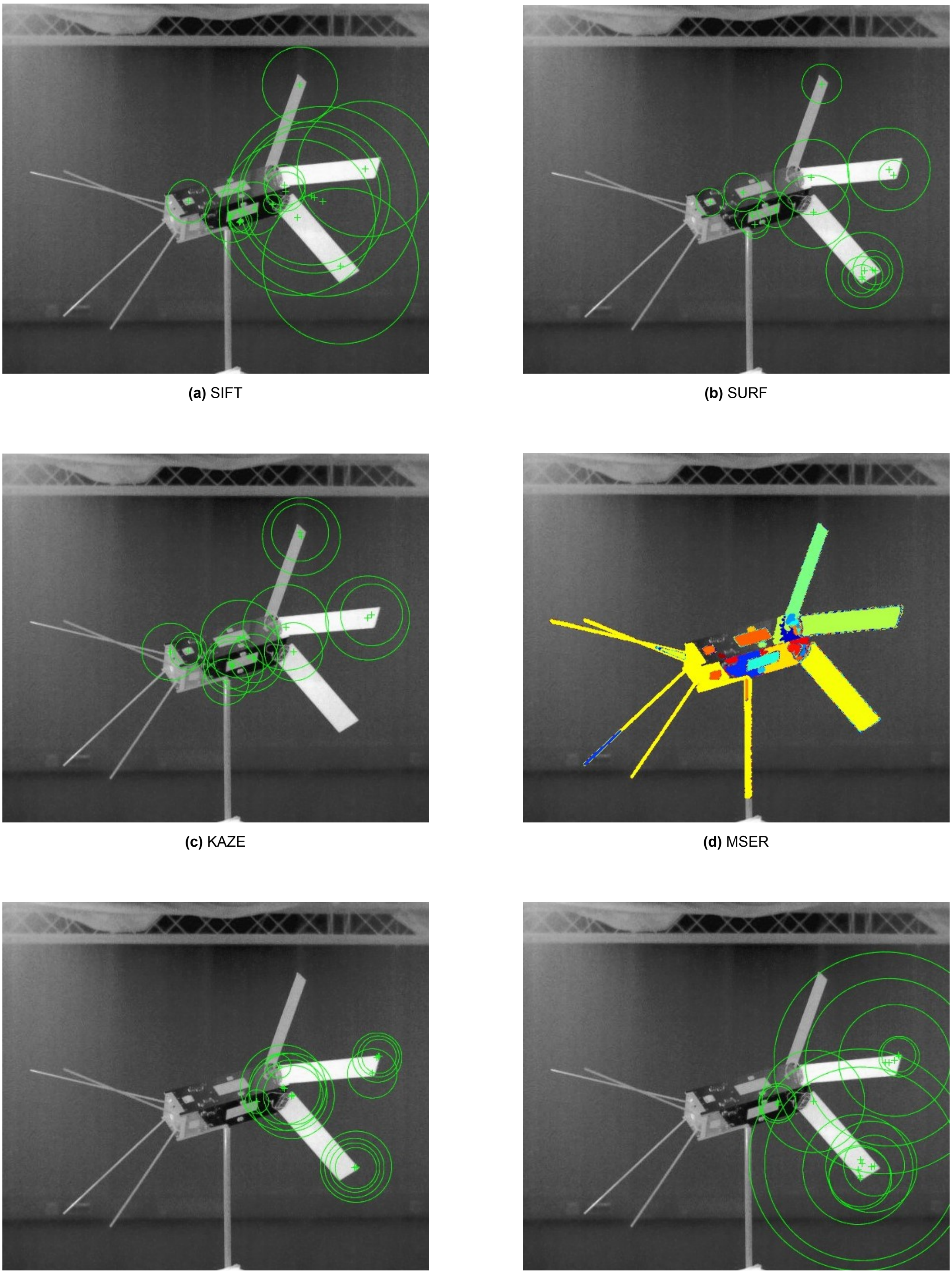


Figure 2.11: Well-aligned imPts projected onto Boxsat. The LiDAR sensor is to the right of the IR sensor and therefore sees a bit less of the edge of the left wing, and more of the right wing compared to the IR sensor.

Many feature detection algorithms require grayscale images for detection (as many are designed to work on a single color channel),[52] so the IR images are then converted to grayscale using the *rgb2gray* function which eliminates the hue and saturation information while maintaining the luminance in the image.[37] This was done in Figure 2.10 for the Harris corner detection. The images can easily be changed back to RGB images after detection and matching is complete, though a reverse conversion is unnecessary as the feature locations found in the grayscale images can simply be plotted onto the undistorted RGB images for visualization.

Feature Detection

The Feature Matching Algorithm was developed by experimenting with different approaches and evaluating the impact and effectiveness of each to choose the best possible approach for the application. MATLAB has a wealth of feature detection capabilities, with options for the detection and extraction of SIFT, SURF, KAZE, MSER, ORB and BRISK features.[32] Each of these have fundamental strengths, as discussed in Section 1.2.2, however the use case in this research is unique and demands an empirical comparison of each to find which method works well for infrared images of the target translating and/or rotating in direct light, occlusion, noisy scenes and in scenarios in which the target is very dim due to a strong infrared source in the background. In the list below, remember that scale change does not refer to a change of scale in an image, as in the target moving closer and farther, but it instead refers to the target changing size in the real world as a balloon would when it is inflated.



(a) SIFT

(b) SURF

(c) KAZE

(d) MSER

(e) ORB

(f) BRISK

Figure 2.12: Each of the feature detectors with default parameters displaying the strongest 15 matches. A large radius circle indicates a strong feature. Note that this image is a luxurious example in that the target is near and clear.

SIFT Scale-Invariant Feature Transform

> **Theory:** Robust to scale changes and is invariant to rotation and affine transformations. Works well for images with low contrast and is suitable for feature matching. Computationally expensive.[20]
>
> **Observation:** Highly tunable including a Gaussian filter to assist in low contrast images. Returned the most features but the accuracy was limited. Inability to specify ROI.

SURF Speeded-Up Robust Features

> **Theory:** Designed to increase efficiency compared to SIFT. It is robust to rotation, scale and illumination.[20]
>
> **Observation:** Offers good tunability, though not always intuitive. Struggled in the noisy occlusion scenarios and rotation was not ideal. Returned many features but was not as accurate.

KAZE Non-linear pyramid-based features

> **Theory:** Robust to scale and affine transformations. It can work well in low contrast and low-illumination scenarios. Not suitable for real-time applications.[2]
>
> **Observation:** Octaves are very tunable, but the impact of this is minimal. Options for different diffusion methods helps to easily switch the detection for different lighting conditions. The threshold parameter is the only other tunable metric.

MSER Maximally Stable Extremal Regions

> **Theory:** Useful for detecting stable regions in an image. May not provide distinct keypoints for matching and could be more suitable for region-based analysis.[39]
>
> **Observation:** Feature points describe regions leading to very inaccurate coordinates for this application. Minimally tunable.

ORB Oriented FAST and Rotated BRIEF

> **Theory:** Optimized for real-time applications and is robust to rotation and scale changes, but does not provide features as distinctive as other methods.[20]
>
> **Observation:** ORB returns the least amount of features, many of which are non-intuitive. Rotation is challenging. Limited parameters make it difficult to tune.

BRISK Binary Robust Invariant Scalable Keypoints

> **Theory:** Designed for real-time applications, and its binary descriptor is efficient. It is robust to challenging lighting conditions and has a more sophisticated keypoint estimator.[20]
>
> **Observation:** BRISK preforms particularly well with target rotation and returns accurate features in all lighting scenarios. Intuitive and highly tunable parameters to adapt to different lighting conditions.

As each feature detection method has different parameters, a comparison of each with constant parameters is difficult, however implementing each on a constant range of IR images obtained during laboratory experiments and tuning to the best possible accuracy for each led to the observations listed above. Based on the experimental comparison of IR images of targets to be used during the final experiments, the decision was made to implement the **BRISK** feature detection method due to its intuitive and high level of parameter tunability, feature point accuracy, and robustness in all lighting scenarios. BRISK was also evaluated to be the most accurate features specifically for target rotation, returning numbers that were closest to the expected values. The following list describes the parameters that are able to be tuned in the BRISK feature detection function:

- Minimum intensity difference between a corner and its surrounding region.
- Minimum accepted quality of corners.
- Number of octaves to implement for multi-scale detection.
- Region of interest in an image. [30]

Feature Matching

After the BRISK detection, the *extractFeatures* parameters validates each feature point's location in the image and returns a binaryFeature object which can then be used for feature matching between the two images.[32] The *matchFeaturesInRadius* function matches these features based on specific matching parameters relating the two sets of features. As the inputs are binaryFeature objects, the function uses the Hamming distance to compute the similarity metric between features.[34] One of the tunable parameters is the radius, which puts a limit on the Euclidean distance (distance in pixel coordinates) between the two matches, which is useful for filtering out unwanted matches particularly in challenging lighting conditions where features are not as pronounced. The other parameters used in the matching function are as follows:

- Matching threshold representing a percent of the distance from a perfect match. Used to filter out weaker matches.
- Ratio threshold used to reject ambiguous matches.
- ”Unique” flag, which indicates if a single feature can have multiple matches. In this application, all matches must be unique.[34]

For each experiment scenario, both the detection and matching parameters can be tuned to detect and match the correct correlating features between each frame. To tune the radius parameter, the algorithm prints the minimum, maximum and mean of the Euclidean distances between the set of matching points, giving a good indication of if the radius parameter is set correctly.

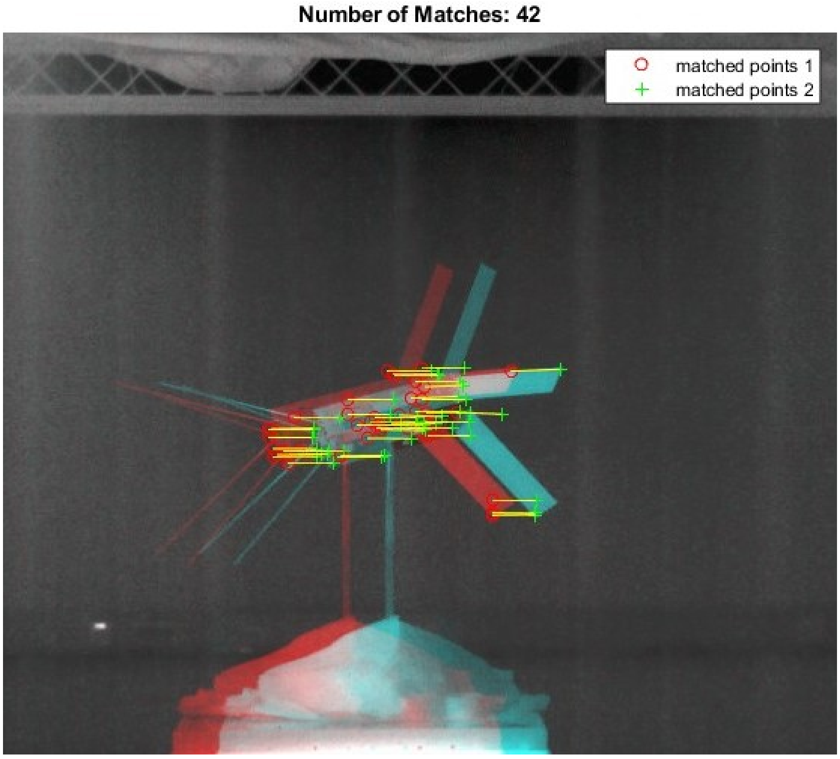


Figure 2.13: BRISK feature matches between image 1 & 2 in which the target moves 10cm from left to right

Matching Features and imPts

The locations of the matching features detected are in image coordinates and are recorded to the thousandth of a pixel, as are the LiDAR projected imPts. A thousandth of a pixel is an unnecessary level of accuracy that would lead to zero features corresponding to LiDAR imPts, so both the feature locations and the imPts are first rounded up to the nearest pixel. By indexing both the imPts and the feature locations, each feature can be matched to its corresponding imPt if they share the same pixel value, and those that do not have a corresponding imPt are discarded, as they then cannot be used in the following step if they do not have a measurement from the LiDAR sensor.

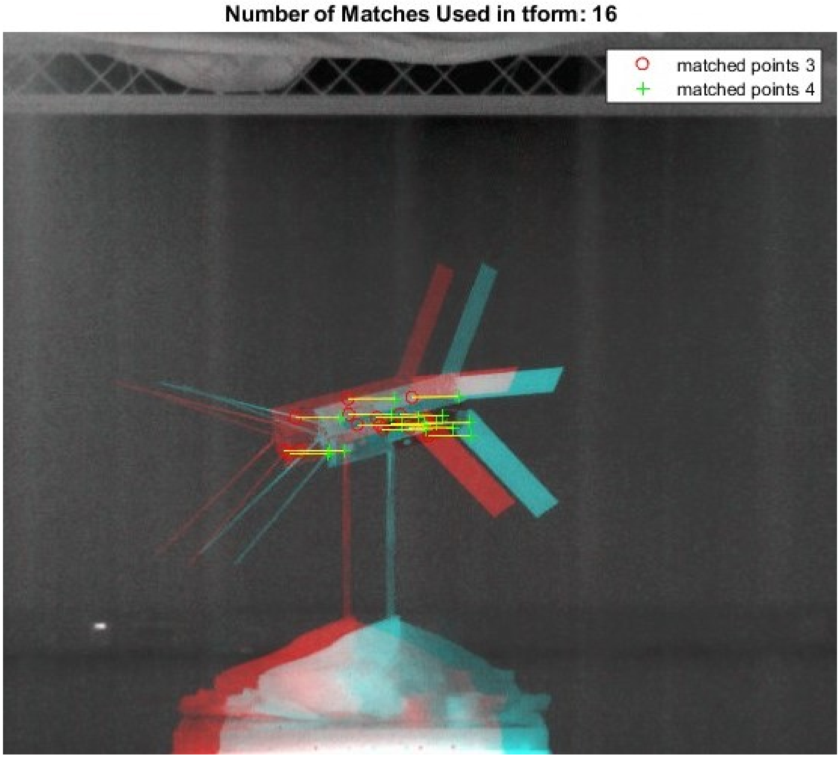


Figure 2.14: BRISK imPt feature matches

Indexing the imPts and the world coordinate LiDAR points allows for an easy transition between the 2-D image and 3-D world coordinates, however the same can be done with the *transformPointsInverse* function using the original affine rigid transformation used to convert the LiDAR points to imPts. Notice the number of matches listed on top of Figure 2.13 and Figure 2.14. Out of the 42 matching points in the images, only 16 had imPts registered to precisely the same pixel, again due to the difference in sensor resolution. Remember that both imPts and feature points were rounded up to the nearest pixel, and without this rounding there would be no matches. This ratio of matches found and matches used is why the density of the point cloud on the target is so critical to the success of this algorithm.

Estimating Transformation

With the matching features now translated to their 3-D world coordinates, the points in each image can be converted into a point cloud with the *pointCloud* function, and their transformation matrix estimated with the *estgeotform3d* function, specifying the transformation as being rigid, as the target does not deform between the two frames. The transformation matrix represents the rotation and translation between the two scenes. The function needs a minimum of 3 points to be able to estimate the transformation. LiDAR point clouds can contain small imprecisions in their measurements (specifically in the x distance axis), so having as many accurate matches as possible is a good way to filter out this effect and retrieve more accurate transformations.

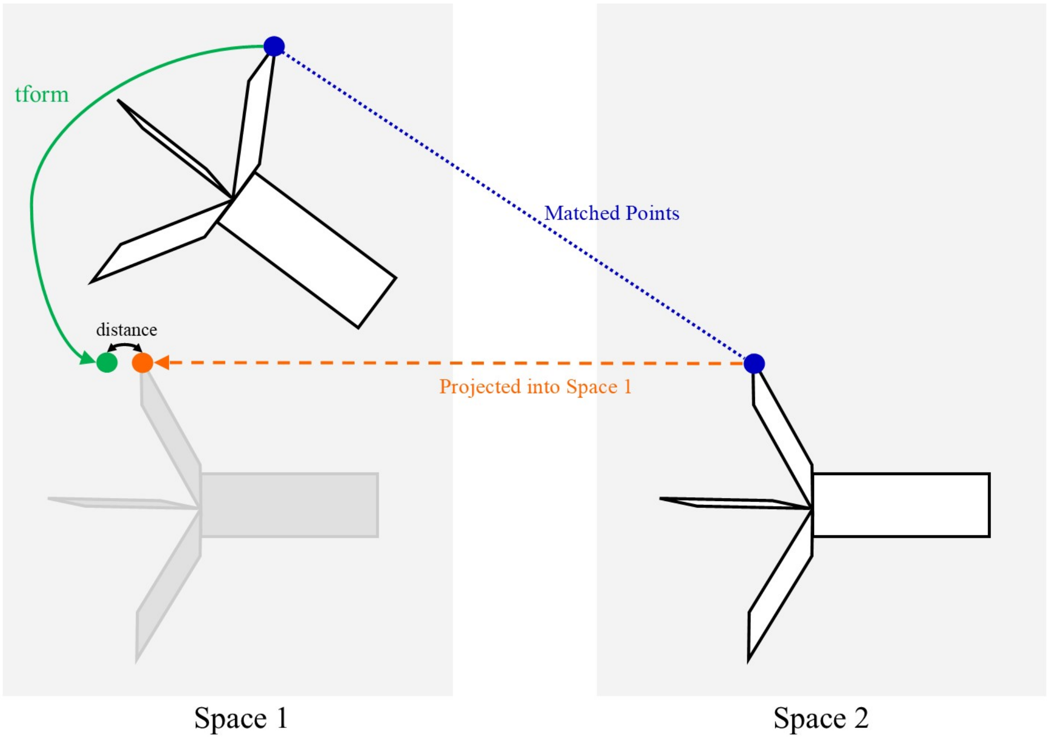


Figure 2.15: Threshold for matches to be considered inliers after transformation

As the matching points are not always perfect (recall the discussion of feature points located at the thousandth of a pixel), a threshold can be set in the form of a maximum distance between matches to be considered inliers, or valid matches to be considered in the calculation of the transformation matrix. Ideally, this distance should be as small as possible to ensure the most accurate transformation estimate. Figure 2.15 shows a visual representation of this maximum distance threshold. As in the Color-ICP algorithm, the result is a transformation matrix A relating the moving point cloud (second frame) to the fixed point cloud (first frame) and describes the target's pose over time.

Control Transform and RMSE

To check the result, a control transformation matrix can be defined. In the LiDAR world coordinates, the x axis points directly in front of the LiDAR, the y axis laying horizontal with positive y pointing to the left, and the z axis vertical. So for a target moving from left to right in the frame by 10 cm, the expected translation matrix would be [0 -0.10 0]. If the target is moving 10 cm directly towards the sensors, the expected translation matrix would be [-0.10 0 0]. The rotation matrix can also be anticipated for scenarios of rotation in different planes, with the scenario of rotation in the x-y plane being shown below. With the translation matrix T and rotation matrices R, the full control transformation matrix A can be built.

$$T = \begin{bmatrix} t_x & t_y & t_z \end{bmatrix}$$

$$R = \begin{bmatrix} cosd(\theta_z) & -sind(\theta_z) & 0 \\ sind(\theta_z) & -cosd(\theta_z) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$A = \begin{bmatrix} cosd(\theta_z) & -sind(\theta_z) & 0 & t_x \\ sind(\theta_z) & -cosd(\theta_z) & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The control, or expected, transformation matrix and the observed transformation from the feature matching algorithm can be compared by calculating the root mean squared error (RMSE) between both the translation and rotation matrices. Separating the transformation matrix in this manner gives a better indication of what the algorithm is better estimating, translation or rotation. The formula for calculating RMSE is shown below, where C represents the control array, while O represents the observed array.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}|O_i - C_i|^2} \tag{2.4}$$

This can be directly implemented to the total translation matrix, however the rotational matrix requires an extra steps. First the rotation matrices are converted to Euler angles representing the rotation in each axis, done by using the *rotm2eul* function which returns a 1x3 matrix in the form of $[\theta_x\theta_y\theta_z]$ in radians. It is important to specify the order as ”XYZ” as this is not the default and would otherwise lead to an inaccurate conversion. A preliminary check can be done using Equations 1.7-1.10 relating Euler angles to rotation matrices presented in Section 1.2.5. The Euler angle arrays can then be put into the RMSE equation above for comparison. A perfect match between the control and observed matrices would return $RMSE = 0$.

This RMSE step is the same for both the Color-ICP and Feature Matching algorithms. It has been detailed in both sections to provide a complete description of each algorithm developed independently.

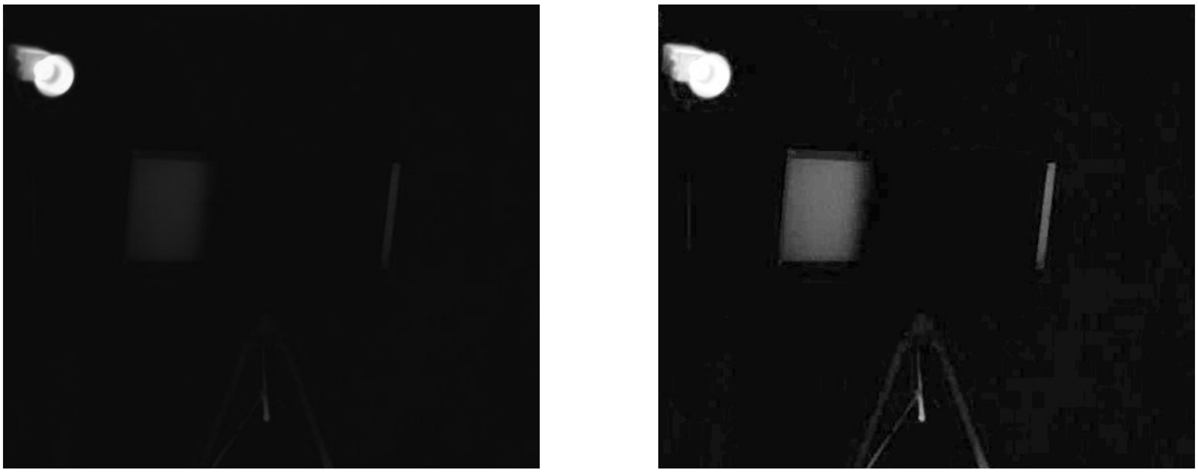
Preprocessing for Challenging Scenarios

Though LWIR cameras are much less sensitive to lighting conditions compared to visual spectrum cameras, there is a scenario in which the representation of wavelengths (temperatures) in the scene can make feature detection difficult. LWIR cameras represent the LWIR spectrum wavelengths as RGB color values in images based on the range of wavelengths present in a scene. High Dynamic Range (HDR) images store all of the wavelength values observed and allow the user to choose the range of interest, however this should not be assumed to be available as not all LWIR sensors have the HDR feature.

If the temperature range in a scene is small, subtle differences will be more visible (dependant on the sensitivity of the sensor) as the full color range is employed to cover a small temperature scale. In contrast, if the temperature range is large, subtle differences are lost as the image representation attempts to cover a large range.

For this research, the use case this is referring to is a scenario in which the chaser observes a target with either a large target temperature range, or in which the sun is visible in the frame. Though these two scenarios seem intuitive, there is more to discuss regarding the sun’s effect on particular LWIR sensors, which can be found in Section 2.2.1. For now, the sun in-frame scenario can be understood to drastically enlarge the range of temperatures observed. When this happens, the details of the entire target are obscured and dimmed. A preview of the experiment developed to simulate this can be seen below in Figure 2.16a for illustrative purposes. The spotlight in the top-left corner obscures the target almost entirely.

This is the worst-case scenario for the heavily IR and CV-dependent Feature Matching algorithm which, while also representing a real use case in the space environment, has the added benefit of testing the limitations of the algorithm.



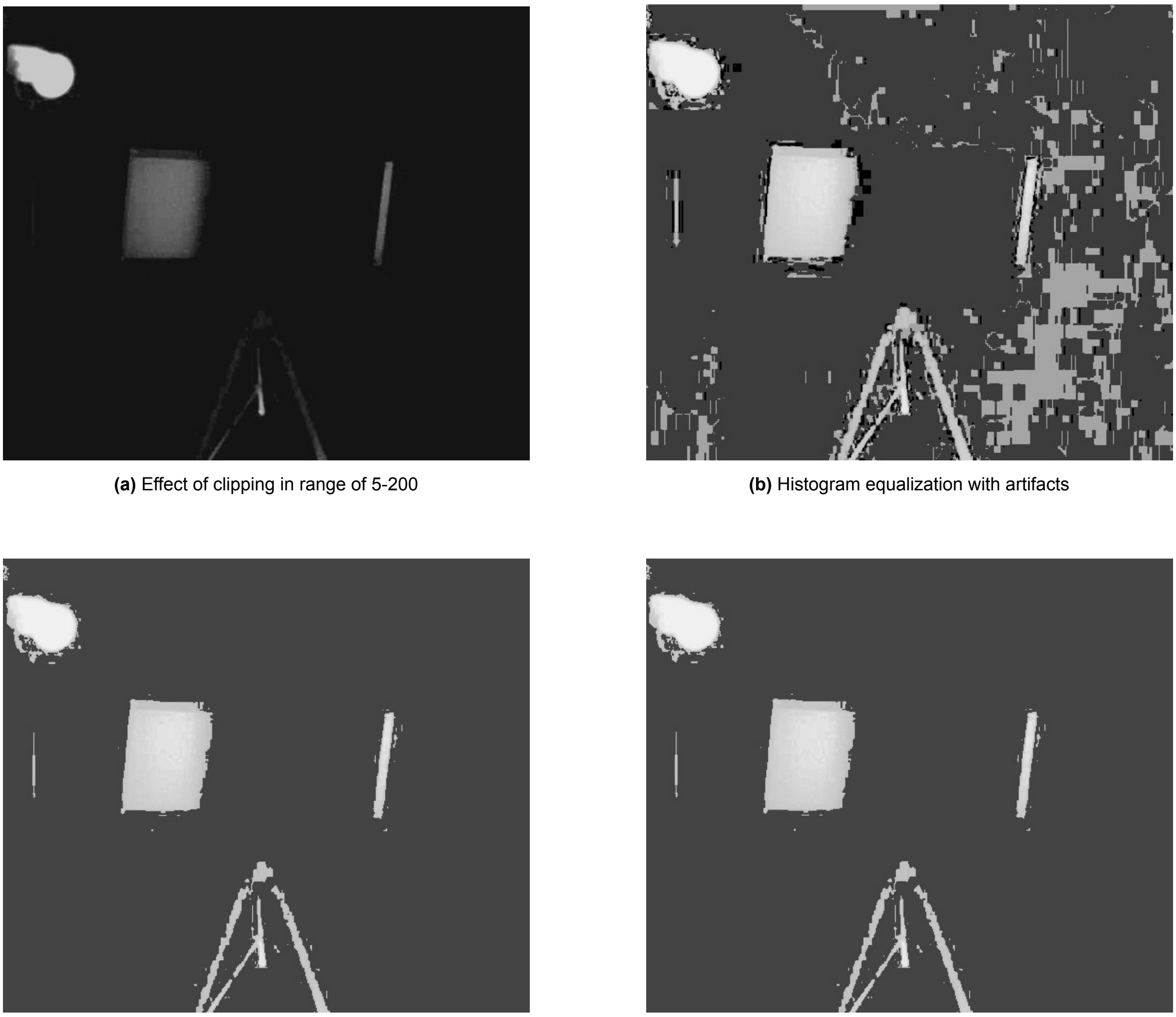
(a) Original grayscale image

(b) Brightened image

Figure 2.16: Effect of brightening function *imlocalbrightening* on the original dim image

As the target in this scenario is very faint, special considerations must be taken to be able to detect and match accurate features. This is done by implementing certain image preprocessing steps to accentuate the contrast between the target and ambient background while lessening the effect of the strong IR source in the frame.

The *imlocalbrighten* function is the first step to enhance the dim and dark target. The next image processing technique that is used is referred to as ”clipping”, which limits pixel values to a specified range and serves to further define the target from the background. The final step is known as ”histogram equalization”, which further enhances the contrast between pixels in the image. The histogram of an image refers to a count of grayscale values in an image, which for 8-bit images such as those from the IR sensor has a grayscale range of 0-255, with 0 being completely black while 255 is completely white. With images with a strong background IR source, the histogram will have a many more pixels in the lower part of the graph and then have a spike at the higher end representing the strong IR source. Histogram equalization attempts to equalize this distribution so that the dim and bright pixels are more evenly spread, effectively enhancing a very dark image. One effect of this equalization is called ”artifacting”, which is the appearance of apparent shapes in empty parts of the image. This is due to the subtle differences of the regions being equalized, leading to these phantom shapes. Luckily, this is easily fixed by adjusting the clipping parameters. If a dark void in an image displays artifacting after histogram equalization, upping the lower limit of the clipping process will erase it completely.



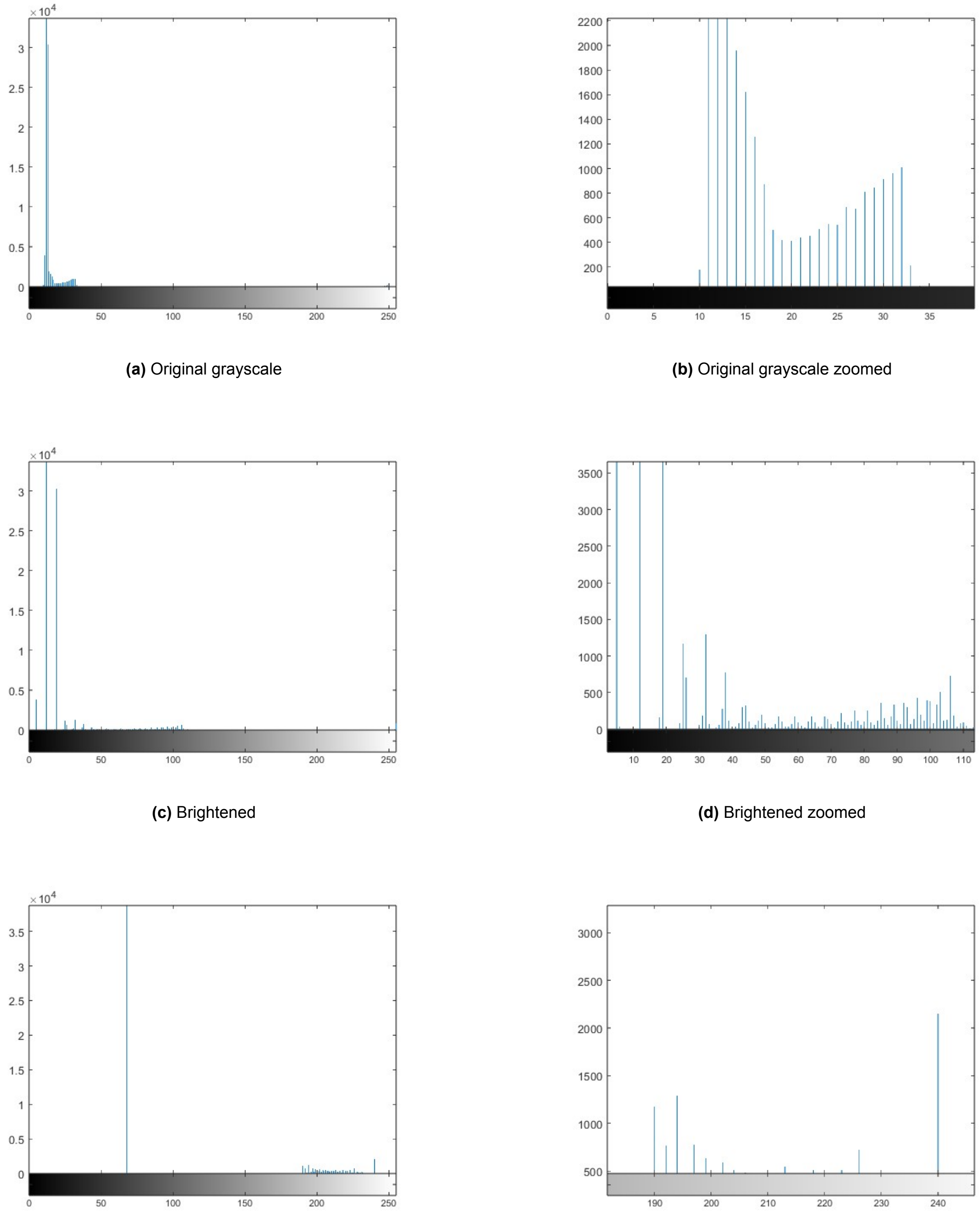
(a) Effect of clipping in range of 5-200

(b) Histogram equalization with artifacts

(c) Corrected clipping in range of 20-200

(d) Histogram equalization with the corrected clipping metric

Figure 2.17: Correct clipping can rid the histogram equalized image of undesirable artifacts



(a) Original grayscale

(b) Original grayscale zoomed

(c) Brightened

(d) Brightened zoomed

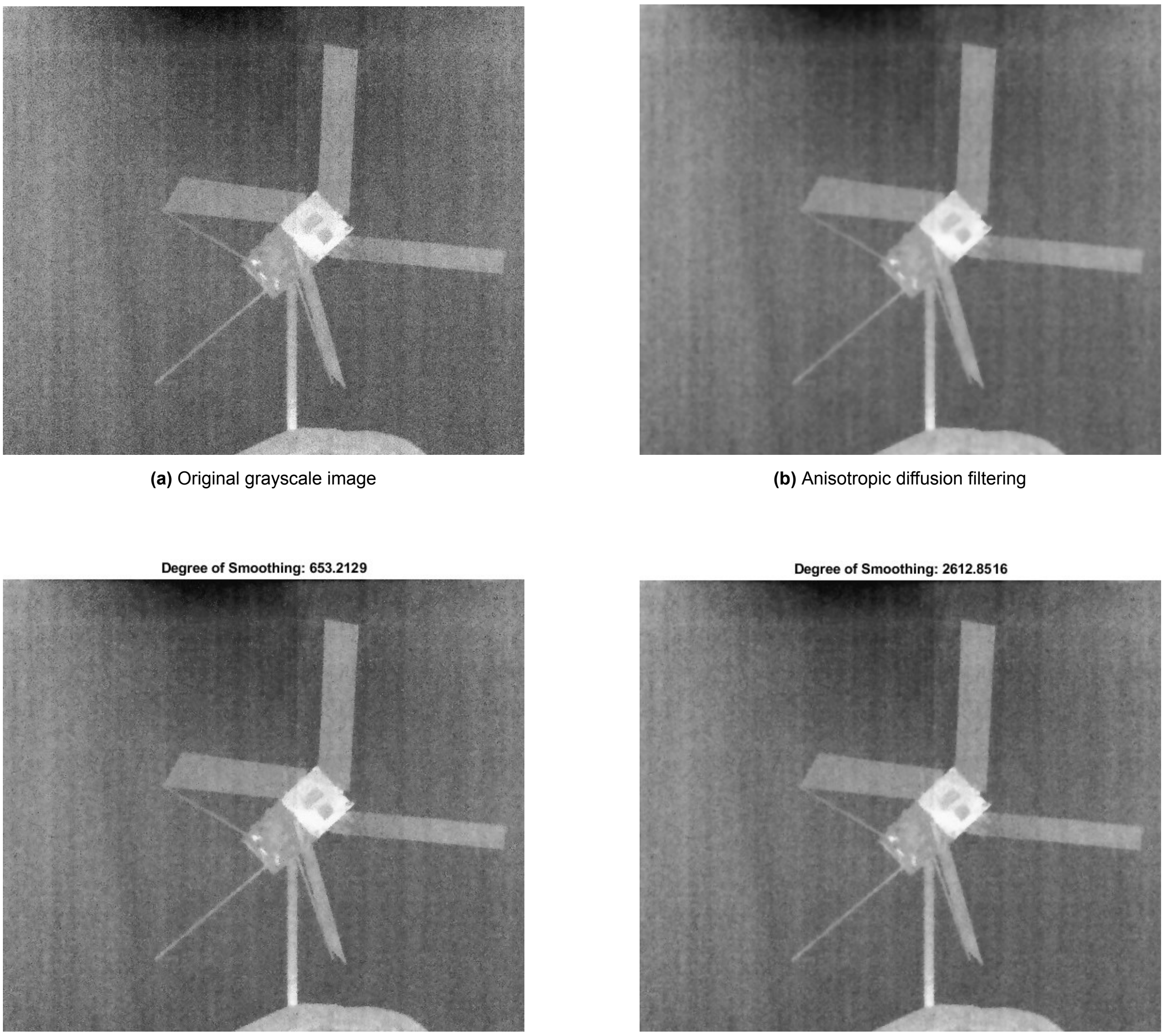
(e) Histogram equalization

(f) Histogram equalization zoomed

Figure 2.18: Histogram plots the original, brightened, and histogram-equalized images.

Other Preprocessing Methods Considered

There are other preprocessing methods that were considered and tested to help with the adverse effects of lighting conditions. For example, in the IR images of a target in occlusion, the images can display so-called ”striation artifacts” vertically across the image. During initial experimentation, this seems to only occur in the occlusion scenario, when the target and background are both near ambient temperature which leads to a very small scale of temperature values represented in RGB values. These artifacts could potentially be interpreted as edges and subsequently features by the feature detection methods, leading to an attempt to suppress this undesirable effect. A promising solution was the implementation of a Gaussian bilateral filter. By defining a region in the image containing the undesirable artifact and approximating the variance of the noise in this small region, the bilateral filter smooths the image while preserving strong edges. This works relatively well in the occlusion scenario, but turns non-edge features blurry which can have an adverse effect of the feature recognition phase, and ultimately did not rid the image of striation. Another technique tested was anisotropic diffusion filtering, which selectively smooths or preserves features based on their local characteristics. This worked well as it can be tuned specifically to preserve low-contrast edges and leads to less blur in the target, but once again did not totally eliminate the striation. However, when each were implemented to preprocess images before feature detection, neither method produced any distinguishable improvement in performance.



(a) Original grayscale image

(b) Anisotropic diffusion filtering

(c) Bilateral filter. Degree of smoothing is 2x patch variation

(d) Bilateral filter. Degree of smoothing is 8x patch variation

Figure 2.19: Anisotropic and Bilateral filtering in an attempt to mitigate vertical striation artifacts in the image of the target in occlusion.

2.2. Experiment Design

When considering a method of V&V for the pose estimation algorithms, there are numerous options regarding data sources to achieve this goal. While outlining the scope of this research, Figure 1.1 referenced four possible sources: experimental data, simulation data, data from literature, and real-world data from space. The most coveted of these sources is of course the space data option, however this data is extremely difficult to come by in detail, and is often safeguarded as the proprietary property of the entities collecting the data. Data from literature can be more accessible, however it is usually highly specific to the application presented in the literature, and assumptions and setup details can be difficult to obtain. Both of these two sources do not guarantee relevance to the selected research approach and are impossible to tune.

Self-designed simulations and laboratory experiments can both be specifically tuned to the scenario of the research. Simulations are reproducible with 100% accuracy, highly scalable and can be designed to precisely replicate a desired scenario, but there are also drawbacks to a simulation approach. Though not always the case, a simulation setting can reveal researcher bias, or a fundamental misunderstanding, in the assumptions made in aspects of the scenario and simulation parameters. Parameters such as sensor noise can easily be tuned, and the input of a model-based target is fundamentally at odds with the unknown uncooperative target focus of this research. In this line of reasoning, the laboratory setting offers the interesting opportunity to experiment with real-world objects put together in an imprecise manner to represent random debris found in orbit. The laboratory setting also lends the opportunity to use real sensors with real noise and imperfections, and is ultimately the best option to more deeply understand the limitations of the proposed method. Laboratory experimentation also lends a deeper knowledge of the sensors, with topics integral to this research such as sensor calibration greatly benefiting from experimentation. It is for these reasons that a laboratory experiment is designed as a data source for the purpose of verification and validation of the pose estimation algorithms.

In contemplating the experimental setup to simulate an RPO mission in orbit, it is clear that conditions will likely be drastically different from the micro-gravity vacuum in which the application for this research will be. Though a thermal-vacuum chamber and drop tower would be nice, the argument can be made that these luxuries aren't necessary for the scope of this research. It has already been shown in 1.2.3 that the debris targets will likely be at temperatures within the range of -80°C to 40°C emitting radiation in the LWIR spectrum. Rotation and translation rates can be mimicked, and the background in the frame of a thermal image can be made to be as uniform as possible. Figure 2.20 below shows the blackbody radiation emittance from objects at different temperatures, supporting the validity of an experiment at room temperature.

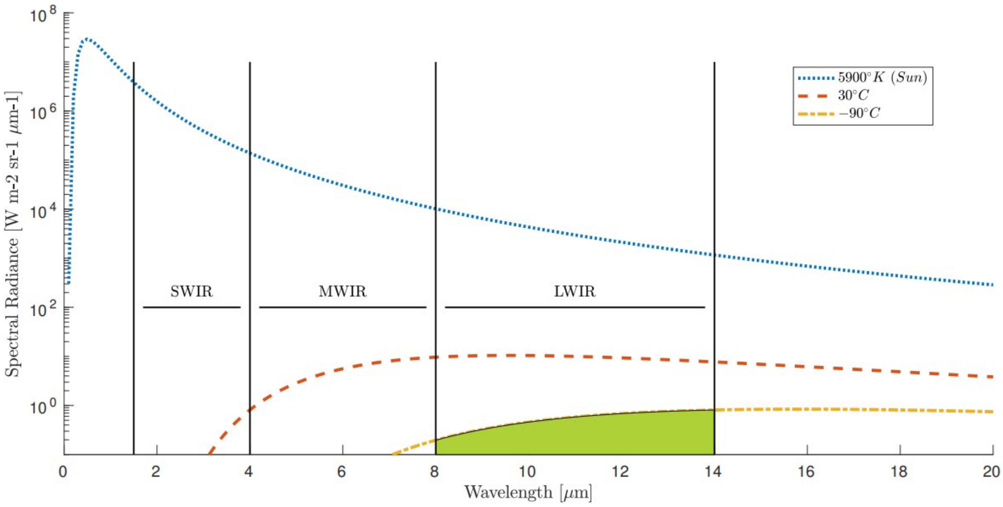


Figure 2.20: Blackbody radiation for 3 different temperatures, including that of the sun. The green region represents the total amount of LWIR spectrum radiation emitted from an object at -90°C. [60]

The design of the experiment to collect data for the pose estimation algorithm changed drastically during the course of the experimentation process. A successful experiment would lead to the accurate calibration of the sensors, and then reliably collect data from the LiDAR and infrared sensors through a range of environmental and target scenarios to fully analyse the approach. This required some practice and tuning before a reliable method was put into place for the final 3 days of testing (Eggman 3, Eggman 4 and Boxsat 1 in Table 2.2 below). However the other testing days, and indeed the many days learning the intricacies of the hardware preceding that, were not wasted. In fact, these days will be referenced in Chapter 3 as the most important days for learning the limitations of the system. Table 2.2 outlines the names that will be used in reference to each day of data collection. The design and reasoning behind each of these experiments will be detailed in subsequent sections.

Session Day	**Date**	**Target**	**Note**
Panda 1	4/8/2023	Delfi-PQ	Initial data gathering with Panda arm and a small target
Panda 2	14/8/2023	Delfi-PQ	Stop motion data gathering with Panda arm
Eggman 1	15/8/2023	Delfi-n3Xt	Initial experimentation with n3XT and Robotnik rover
Eggman 2	16/8/2023	Delfi-n3Xt	Data gathered on a wide variety of target motion
Eggman 3	8/9/2023	Delfi-n3Xt	New and improved calibration method
Eggman 4	11/9/2023	Delfi-n3Xt	Repeated experiments with new calibration
Boxsat 1	12/9/2023	Boxsat	Larger target capable of longitudinal rotation

Table 2.2: Session days in which experiments were preformed to estimate pose. Note that the experimentation process changed as the limitations of the system became clear.

2.2.1. Instrumentation

Hardware experiments are inevitably limited by the hardware available, however the TU Delft (TUD) Aerospace Engineering (AE) Faculty has a number of relevant instruments that are highly suitable for this research. The LWIR sensor provided and utilized throughout this research is a Teledyne FLIR A655sc uncooled microbolometer camera, while the LiDAR sensor utilized is a Livox Mid-70 solid-state scanning LiDAR. This Section will provide the specifications of these instruments while also discussing some of their important characteristics relevant to their use in this research.

The sensor setup can be seen in Figure 2.21 which shows how the LiDAR and the FLIR are positioned as close as possible to each other, minimizing the transform needed to align the point cloud to the IR image in the algorithm. A laser range finder was used to provide the ground-truth measurement of a specified reference point on the target to be checked against the LiDAR distance measurement and was updated and recorded periodically during experiments.

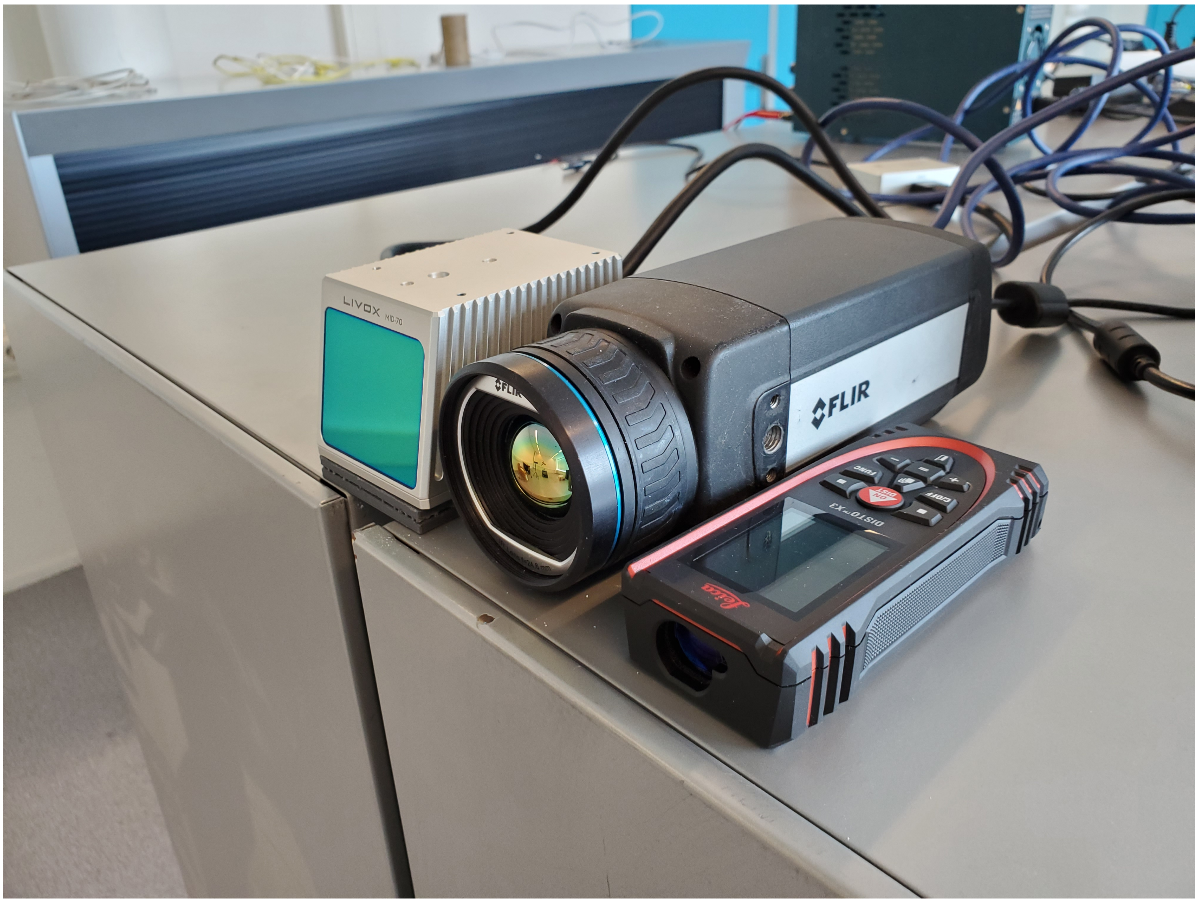
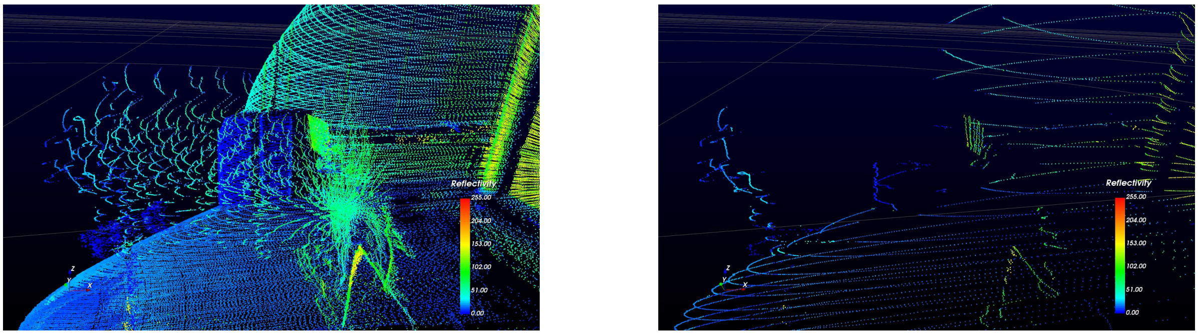


Figure 2.21: Sensor setup showing from left to right the Livox LiDAR, FLIR IR camera and a laser range-finder for the ground-truth distance

Livox Mid-70

Available for use at the AE faculty is the Livox Mid-70 high-performance LiDAR sensor. The Mid-70 records 100,000 points per second and can be set to various exposure times, as it is defined as a scanning LiDAR.[53] The lowest of these times is 100ms, a tenth of a second which would return 10,000 points. The longest exposure time is 3000ms and would return 300,000 points. Though a longer exposure time records more points, it is not suitable for motion. Figure 2.22 shows a scan of the author walking near a calibration target, viewed from the right side respective to the LiDAR's FOV. It demonstrates the problem with scanning LiDARs with regards to in-scene motion. In the left image, the exposure time is 3000ms and the motion is seen as a nondescript blur, while the right image has an exposure time of 200ms and shows the location of the moving author but has lost much of the detail of the scene, making it difficult to discern objects and features. This effect is known as motion blur.[18]

It is for this reason that the decision was made to record the motion of targets for the experiment in a **stop-motion** fashion. In practice, this means that in an experiment mimicking the scenario in which a target's relative motion is moving from left to right through the reference frame, the target will be scanned for a minimum of 3 seconds while it is still, be moved a small distance to the right of the reference frame, be scanned again, move to the right, etc. Though some experiments were done attempting to record the movement with a lower exposure in one continuous scan, it was decided that the stop-motion methodology was the only way do retrieve accurate data due to this inherent nature of scanning LiDARs as well as the lack of synchronization equipment available at the faculty for the Livox and infrared camera together.



(a) Movement near a target with an exposure time of 3000ms

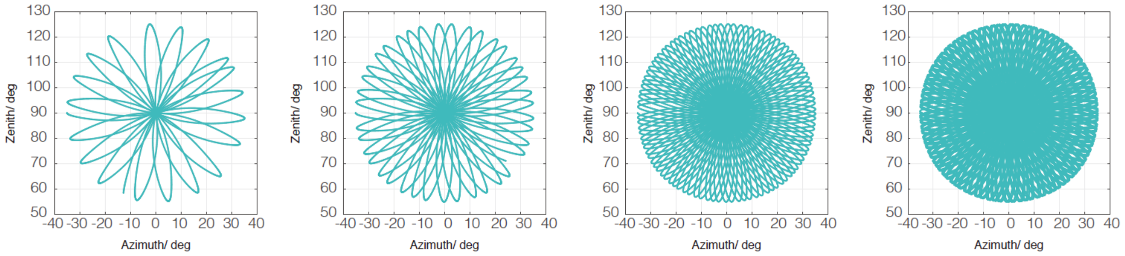
(b) The same scene with an exposure time of 200ms

Figure 2.22: Low exposure time eliminates motion blur but makes the scene harder to discern

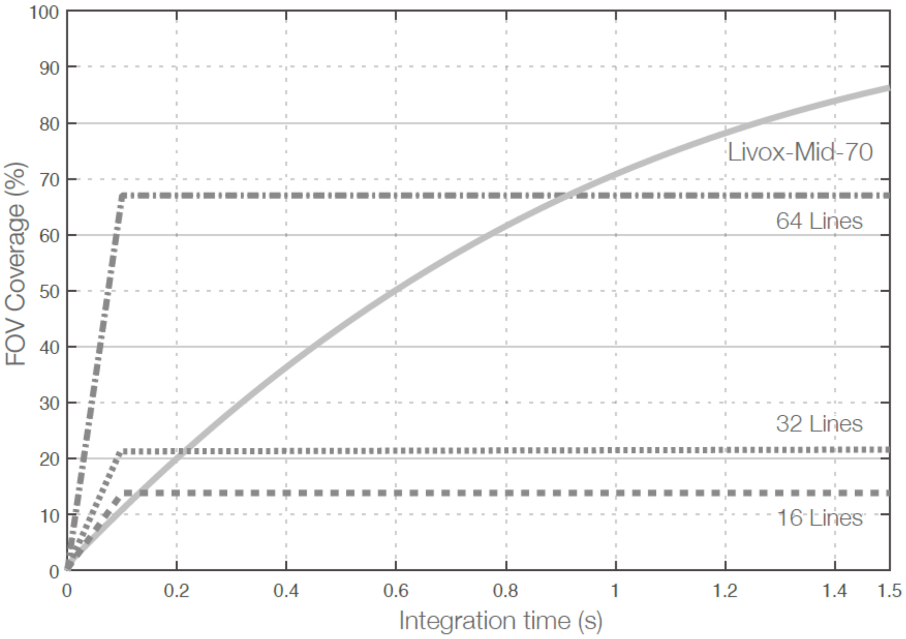
Laser Wavelength	905nm
FOV	70.4° (circular)
Close Proximity Blind Zone	0.05m
Distance Random Error	1σ (@20m) $\leq$ 2cm
Angular Random Error	1σ < 0.1°
Beam Divergence	0.28° (Vertical) x 0.03° (Horizontal)
Point Rate	100,000 points/s (first or strongest return)

Table 2.3: Livox Mid-70 specifications[53]

At this point, it becomes necessary to discuss the unique scanning pattern used by the Mid-70 and its relationship to point density. Point density refers to the amount of points per square meter at a certain distance. As Table 2.3 shows above, the Mid-70 has a 70.4° FOV projecting in a non-repeating pattern, and has a point rate of 100,000 points per second. As distance increases, the point density will lessen due to this FOV angle. For many LiDAR systems projecting a uniform pattern, the calculation of point density with respect to distance would involve simple geometry, multiplying the point rate with sample time and then dividing by the area in the FOV. However this is not the case for the Mid-70 due to its unique non-repeating, or quasi-random, scanning pattern shown in Figure 2.23a. Reminiscent of flower petals, the pattern precedes along the central axis and covers approximately 70% of the field of view at 1 second of integration time as shown in Figure 2.23.



(a) Mid-70 points following its non-repeating scanning pattern after 100, 500, 1000 and 3000 milliseconds of integration time respectively.



(b) FOV area covered with respect to integration time, as opposed to non-Livox mechanical scanning LiDARs on the market.

Figure 2.23: Livox-supplied depictions of its unique non-repeating (quasi-random) scanning technology.[53] Though interesting, it complicates the calculation of point density dramatically.

Due to the quasi-random nature and non-uniformity of the scanning pattern, the precise point density can only realistically be estimated and is a function of the converging center point's distance to the sensor and the distance of the point area to the center. Figure 2.24 shows a scene observing Delfi-n3Xt in which the point density is represented bu the color scale on the right side of the figure. As expected, the points are more dense at the center of the scan and progressively spread as the distance from the center increases. The wall behind is a flat uniform surface that more consistently shows this pattern than Delfi-n3Xt, whose more complex geometric shape causes some departures in the expected density.

The distribution of point density can be further analysed using histograms. Figure 2.25 shows this distribution for the Delfi-n3Xt target (excluding the points of its surrounding environment) as it moves farther away but remaining at the center of the scan, and as it moves from the left side of the scan to the right side. Initially it is logical to think that Figures 2.25d and 2.25f should show the same distribution, however the target's observed geometry from the point of view of the LiDAR changes as it moves preventing this. This demonstrates the difficulty in predicting point density with the Mid-70.

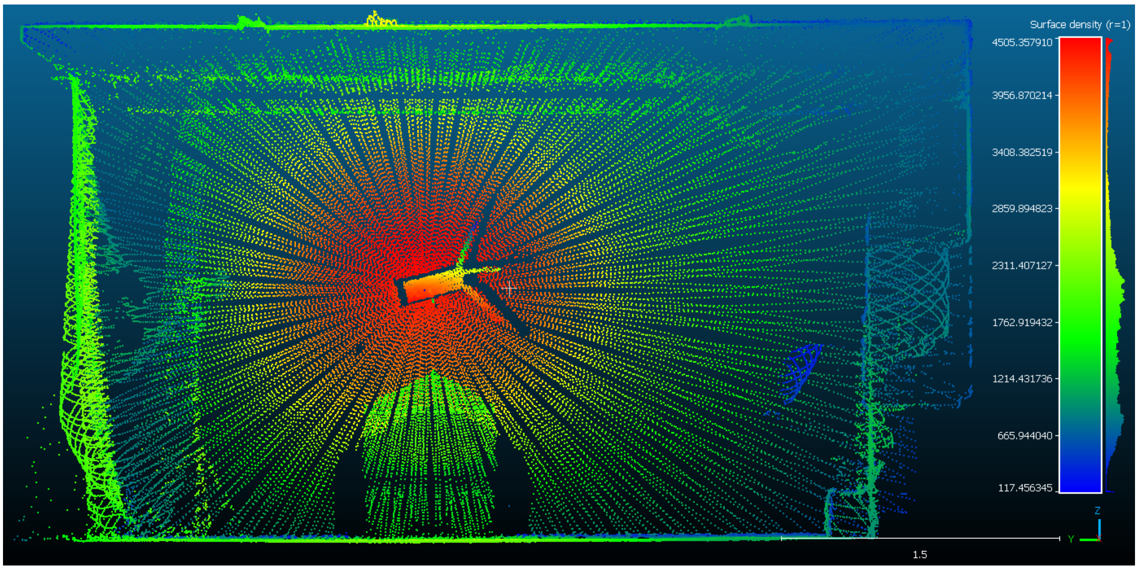
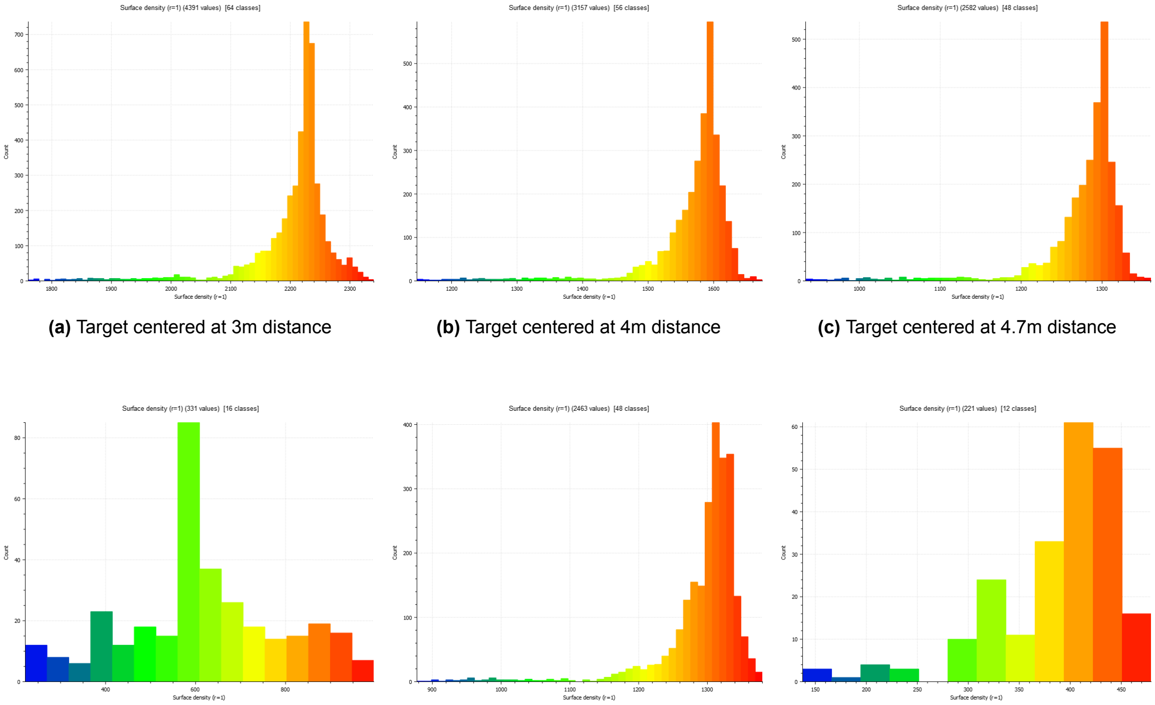


Figure 2.24: Delfi-n3Xt during the Series J experiment. The color scale represents the point cloud density in points per m^2.



(a) Target centered at 3m distance

(b) Target centered at 4m distance

(c) Target centered at 4.7m distance

(d) Target 1m left of center

(e) Target center

(f) Target 1m right of center

Figure 2.25: Histogram of point density on the segmented Delfi-n3Xt target for a 1 second integration time. The points on the surrounding environment are excluded. With the target in the center, (a-c) show reduced density with distance. (d-f) show the density at 5 meters distance drastically changing as the target moves 1 meter to either side of the center. Density is measured in points per m^2.

One way to approximately describe the distribution of the point cloud density with respect to distance from the center is as a reciprocal squared function, with the center as a high peak at x=0 that rapidly decreases along the positive and negative x-axis.

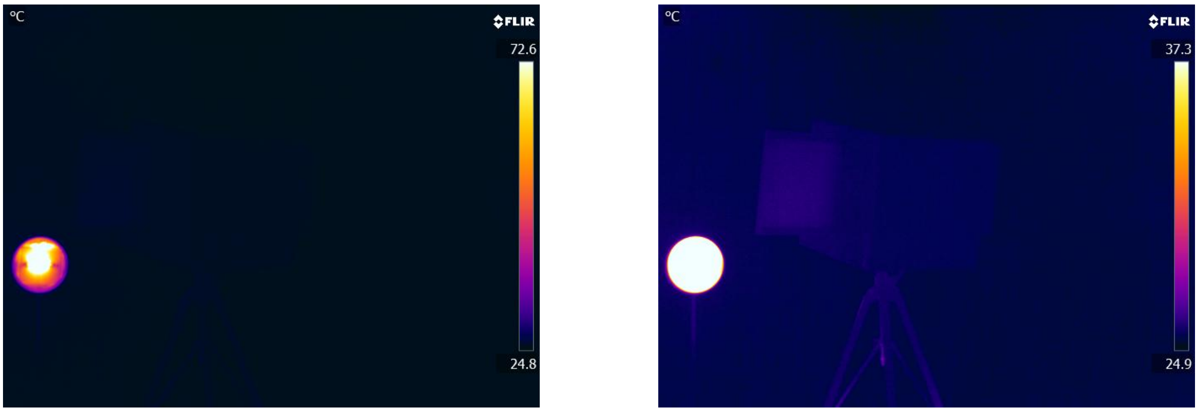
FLIR A655sc & Uncooled Microbolometers

The FLIR A655sc (referred to hereafter as simply FLIR) is a high-resolution LWIR camera with an uncooled 640 x 480 pixel microbolometer[54] that is available at the AE faculty. Bolometers produce an electrical signal using an element that changes resistance with temperature.[5] Though it has a high frame rate at 50 Hz, this is an unimportant metric due to the stop-motion nature of the experiments imposed by the scanning LiDAR. Much more relevant is its resolution, which allows for the images captured to be detailed enough for feature recognition and other computer vision tasks in the pose estimation process. The FLIR is able to be manually of automatically focused to ensure sharp images at various distances.

Resolution	640 x 480
Detector Pitch	17 μm
Spectral Range	7.5 - 14.0 μm
Detector Type	Uncooled microbolometer
Camera f-number	f/1.0
Dynamic Range	16-bit
NETD	<30mK
Frame Rate	50 Hz
Standard Temperature Range	-40° C to 150° C, 100° C to 650° C
Focus	Automatic or Manual (Motorized)

Table 2.4: FLIR A655sc specs[54]

Briefly discussed in Section 2.1.4 as it relates to the Feature Matching algorithm, an interesting aspect of LWIR images is the range of temperatures (wavelengths) represented in RGB (or grayscale). Figure 2.26 again shows a challenging scenario in which there is a strong infrared source, in this case a terrace heater, in the scene with the target. Figure 2.26a shows the automatic higher scale that the sensor imposes, as the FLIR tries to avoid saturating the upper bounds of the scale to be able to resolve the features of the hotter object. Due to this, the relatively cold target is completely hidden in darkness. Manually lowering the scale saturates the upper bounds of the color scale in the image, in other words something 60° C and 70° C will both be colored white which is the highest color representation on the scale. Though this turns the entire IR source white and hides any detail therein, the target is able to be seen in the image. This specific situation represents the worst-case scenario for the FLIR.

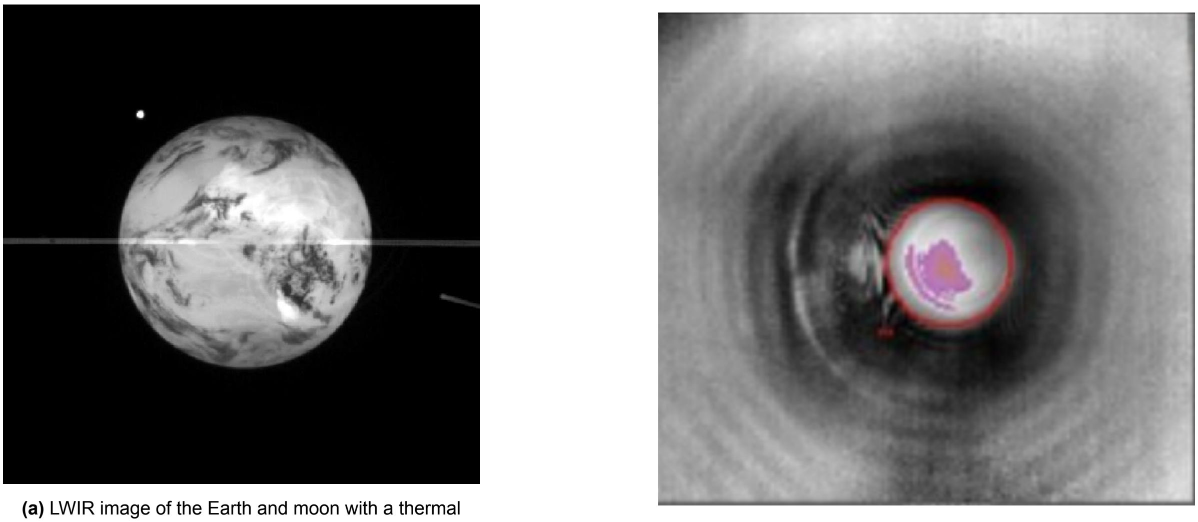


(a) Temperature scale obscures target

(b) Temperature scale allows target to be seen

Figure 2.26: Challenging lighting scenario in which the temperature scale is critical

The relevance of this sun in-view experiment seems intuitive, however there is more scientific nuance to its motivation to examine. At this point it becomes necessary to discuss the nature of uncooled microbolometer LWIR cameras and how they operate in the space environment, particularly in relation to the sun. The sun's radiation emission is dominated by the visible and NIR spectra, with <1% of its emission coming from the LWIR spectrum.[16] This is still a massive amount, considering the scale of the sun, however it is still <1%. It stands to reason, then, that the sun viewed in the LWIR spectrum would appear as a dim circle or dot since so little radiation is detected in this regime. However this is not the case for uncooled microbolometers, as exposure to the intense solar rays in vacuum leads to an effect known as thermal blooming, which can saturate (max out) affected pixels and leave an artifact in the image for an extended period of time.[27] Figure 2.27b shows the effect of intense thermal blooming on a 640x480 pixel uncooled microbolometer exposed directly to a laser over time to the point of permanent detector damage. A previously mentioned RPO mission, MEV-1 & MEV-2, used an uncooled microbolometer LWIR camera during their missions, but encountered thermal blooming during a calibration test viewing the Earth, moon and a transiting sun, as seen below in Figure 2.27a. The resulting artifact left by the sun's movement across the image is referred to as sun-streaking, a direct result of thermal blooming affecting the uncooled microbolometer.[44] Though this exact effect is difficult to reproduce in the atmosphere without a laser and a risk of equipment damage, the Boxsat D, E and F experiments exist to saturate a part of the frame as it would be during a sun in-view observation of the target.



(a) LWIR image of the Earth and moon with a thermal blooming artifact present from the sun's transition through the frame during MEV-2 [44]

(b) Bolometer thermal blooming induced by a laser in laboratory conditions [27]

Figure 2.27: Bolometers suffer from thermal blooming when an intense radiation source saturates a portion of the detector and can leave an artifact for a considerable amount of time.

Again, the representation of LWIR wavelengths as RGB or grayscale values is dependent on the range of wavelengths present in the image. When a portion of pixels are saturated, the range expands upwards and conceals the rest of the scene as dim and dark, as seen in Figure 2.26a. In addition to simulating thermal blooming from the sun, these experiments simulate a worst-case scenario and additionally serve to test the limits of the algorithms' capabilities, particularly the Feature Matching algorithm as it relies heavily on FLIR and computer vision to determine pose.

2.2.2. Calibration

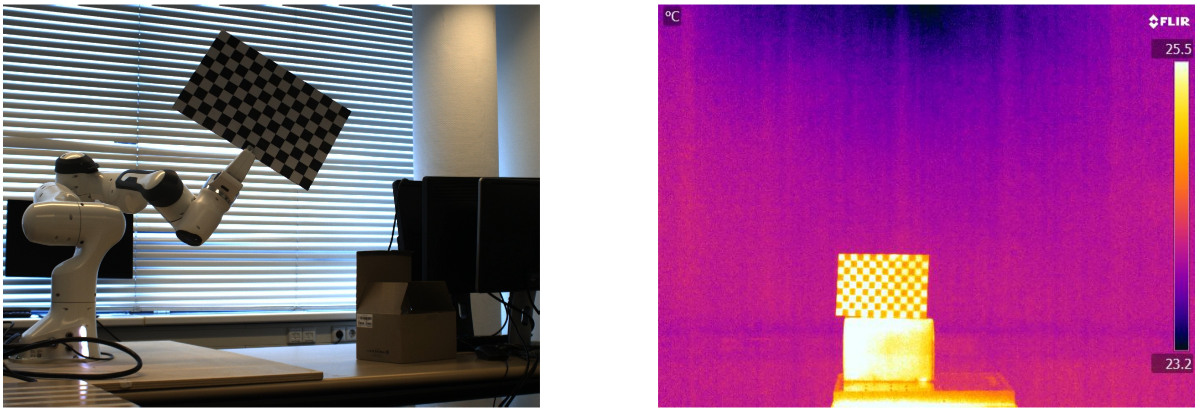
The importance of proper calibration for this research cannot be understated. As mentioned before, each experiment had value in the learning process and directly led to an improvement and refinement of the method of experimentation, however only the experiments of the Eggman 3, Eggman 4 and Boxsat 1 days were considered to be valid in delivering reliable data suitable for implementation into the algorithms. One of the main reasons for this, certainly for days Eggman 1 & 2 was due to incorrect calibration.

As stated in Section 2.1.1, calibration can be thought of as the sensor fusion process in mini. As the sensors are next to each other (see Figure 2.21) and not in the exact same point in space, calibration results in the extrinsic parameters, including the crucial transform function, and the intrinsic parameters of the FLIR camera. For more background and detail about these parameters, please refer to Section

1.2.4.

A chessboard (or checkerboard) is a common target for calibration of electro-optical sensors either individually or together for sensor fusion.[31] There was concern at the beginning of the experimentation process that the IR chessboard images would not be recognized in calibration algorithms, in which known distances of straight lines between the corners of squares are referenced, however the chessboards in the IR spectrum were just as likely to be recognized as in the visual spectrum with the proper care. The chessboards were able to be seen at ambient room temperatures with FLIR, however shining a spotlight onto the chessboard helped to sharpen the image and accentuate the contrast between squares, as the black squares absorbed more heat than the white squares. Though potentially unnecessary, the spotlight was used to ensure distinctive corners for calibration.

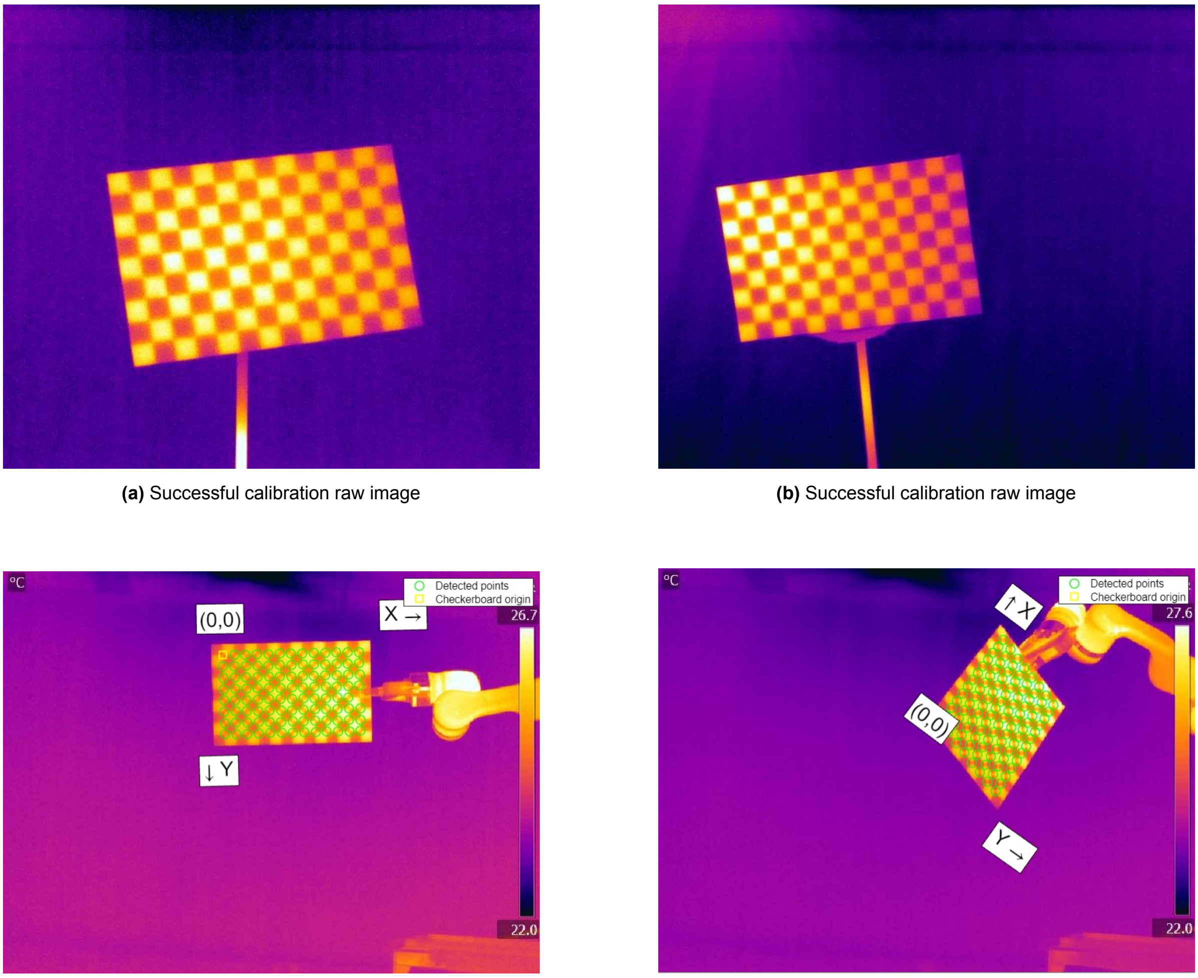
However, the calibration process is very sensitive to other aspects of the image. Figure 2.28 shows two imaging sequences that failed in the calibration process. Figure 2.28a was an early attempt at LiDAR-camera calibration in the visual spectrum with the Panda robotic arm holding the chessboard steady for a scan. This failed due to the blinds behind the chessboard. Figure 2.28b was one image in a series of calibration images that all failed due to the box on-top of which it was placed on, thus that day's data could not be used. From this, the lesson was learned to give the chessboard free space with a simple and uniform background. Figure 2.29 shows two raw IR images that were successfully used for calibration and two images with the resulting corner detection resultant of successful calibration.



(a) Visual spectrum calibration fail

(b) IR spectrum calibration fail

Figure 2.28: Calibration failures due to in-scene interference



(a) Successful calibration raw image

(b) Successful calibration raw image

(c) Calibration corner detection

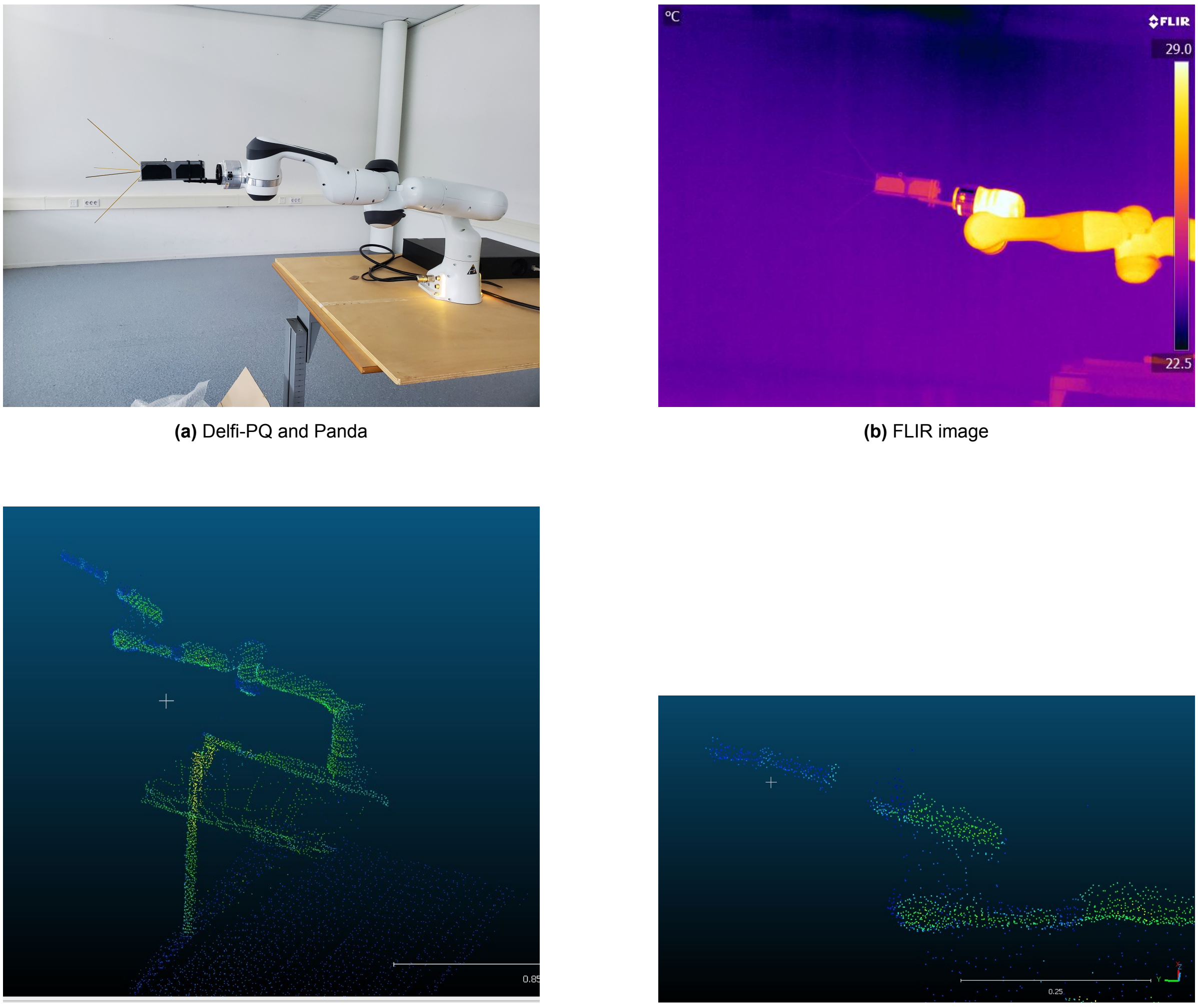
(d) Calibration corner detection

Figure 2.29: Successful calibration with chessboards in the IR spectrum

2.2.3. Panda Experiments

The Panda is a 7-DOF robotic arm made by Franka Emika capable of lifting and manipulating 3 kg.[12] The Panda was available for use at TU Delft's AE building and was determined to be a good way to rotate and translate a target in a precise manner to be viewed and recorded by the LiDAR and IR sensors. The Panda was an attractive option due to the ability to program precise movements and lock axes for accurate single-axis rotation, however in practice the Panda was prone to undesirable motion specifically during translation. This made it very difficult to achieve single axis translation over any distance more than a few centimeters.

Another issue was the Panda-imposed 3kg target weight limit. The initial target chosen was a 1:1 replica of the Delfi-PQ satellite, which is a 5x5x18 cm TUD-designed satellite launched in January 2022.[55] The Delfi-PQ was the target for the Panda 1 & 2 experiments in which it completed a series of translation and rotation maneuvers programmed for the Panda and was recorded by the LiDAR and FLIR sensors after a program of chessboard manipulation for Panda. Upon viewing the recorded data, a few issues were discovered and directly led to the improvement of the experiment.



(a) Delfi-PQ and Panda

(b) FLIR image

(c) Point cloud of PQ, Panda, and the desk

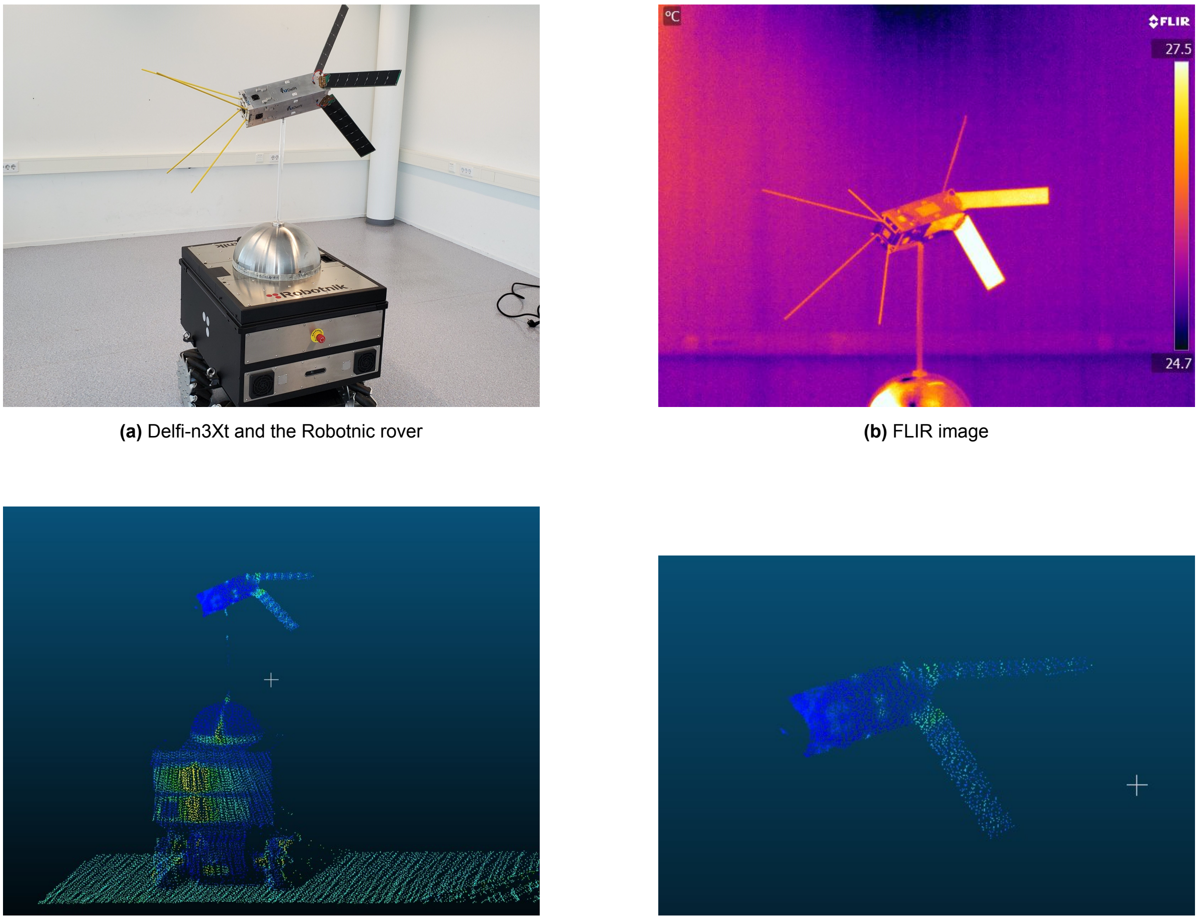
(d) Closer view of PQ

Figure 2.30: Delfi-PQ held by Panda in the visual, IR and as a point-cloud during Experiment: Panda 2.

Figure 2.30 shows data recorded during the Panda 2 experiment. Notice that the Panda gets rather warm during operation compared to the target, causing the overall temperature scale to slide and the RGB representations to shift. Even so, the FLIR is able to distinctly resolve the solar panels on the side of the target as well as its overall shape. However, the Panda dominates the scene. The point clouds show the main issue with the Panda 1 and 2 experiments; the point cloud of the target is very sparse. These points represent a scan of 3000ms (3 seconds) and returns less than 60 points of the PQ. Because of this, it is very difficult to detect features in the point cloud scene, even though the PQ's shape is rudimentary and has distinct planar features, thus making it difficult to register points to determine pose in either algorithm.

2.2.4. Eggman Experiments

The PQ-Panda issues were alleviated by introducing a new target as well as a new robot that was recently purchased by the TUD AE department. The robot is the Robotnik Kairos, a rover with a large flat top on which to place/attach objects. Robotnik is the surname of the villain ”Eggman” in the Sonic The Hedgehog series, thus these experiments were given the moniker Eggman.[48] The target was a 1:1 replica of the TUD-built Delfi-n3Xt, a 10x10x30cm satellite launched in 2013 that has since become space debris when transmission ceased after mission success.[13]



(a) Delfi-n3Xt and the Robotnic rover

(b) FLIR image

(c) Point cloud of n3Xt, Robotnik and the floor

(d) Closer view of n3Xt

Figure 2.31: Delfi-n3Xt ontop of the Robotnik rover in the visual, IR and as a point-cloud during Experiment: Eggman 1. The number of PQ points is 5,118.

Figure 2.31 shows how Delfi-n3Xt was mounted onto the rover, while also showing the data collected by FLIR and the Mid-70. The details in the FLIR image are very sharp and pronounced thanks to the target's larger size and minimal interference with the mount. The number of points above in Figure 2.31 (d.) is 5,118, also retrieved from a 3000ms scan. Recall that the PQ scan had less than 60 points. This is much better for feature detection and point-registration, as it will be shown in the results in Chapter 3.

Eggman 2 starts what are referred to as the ”Series” tests for the Delfi-n3Xt target. These are a series of recordings and measurements for a specific target encounter scenario. For example, Series J took place during Eggman 4 and records Delfi-n3Xt as the rover moves it 10 cm at a time from left to right in the frame for 1.60 meters total distance. Series K records the same motion, the rover travelling 10 cm at a time from left to right, however with n3Xt this time spinning 20 degrees every 10 cm. The Series tests are meant to enable the stop-motion recording of a range of translations, rotations, translation-rotations and under a number of different lighting conditions.

These scenarios are designed to represent the relative motion and conditions involving a chaser-target pair during an ADR mission. Specific mission profiles will likely differ, however the experiments designed represent translation and rotation over time during the approach. As Delfi-n3Xt is too small to be the target of an early ADR mission, it is possible to scale its translation to a prime candidate for removal: Envisat, as mentioned in Section 1.1. The body of Delfi-n3XT (not including antennae or solar panels) is 30 cm long[13] compared to the 10.02 m long body of Envisat (not including the solar panel)[47]. This length represents a 33.4:1 ratio in body length, which would represent Envisat translating 3.34 m between frames. Table 2.5 describes each Series experiment, with a more detailed description including start and end states available in Appendix B.

Experiment	Session Day	Lighting	Target Intrinsic Motion
Series A	Eggman 2	Direct	Translation left-right
Series B	Eggman 2	Direct	Translation left-right & rotation
Series C	Eggman 2	Direct	Translation far-near
Series D	Eggman 2	Direct	Translation far-near & rotation
Series E	Eggman 2	Direct	Rotation
Series F	Eggman 2	Direct	Rotation (video)
Series G	Eggman 3	Occlusion	Translation far-near
Series H	Eggman 4	Direct	Translation far-near
Series I	Eggman 4	Direct	Translation far-near & rotation
Series J	Eggman 4	Direct	Translation left-right
Series K	Eggman 4	Direct	Translation left-right & rotation
Series L	Eggman 4	Direct	Closer translation left-right & rotation
Series M	Eggman 4	Direct	Closer translation left-right
Series N	Eggman 4	Occlusion	Translation far-near & rotation
Series O	Eggman 4	Occlusion	Rotation
Series P	Eggman 4	Occlusion	Closer translation left-right & rotation
Series Q	Boxsat 1	Sun simulator	Translation left-right & rotation

Table 2.5: Delfi-n3Xt Series tests

Series G was the first test of the target in occlusion, continued in Series N, O, & P. In this scenario the target was left in a dark room for a minimum of 30 minutes but up to 2 hours (equilibrium temperature reached within 30 minutes, so no noticeable difference due to longer wait times) to allow it to reach ambient room temperatures, the result of which can be seen below in Figure 2.32. It should be noted that the instrument and recording suits run warm, and that ray reflection off of the target's metallic body into the FLIR can happen at the right angle.

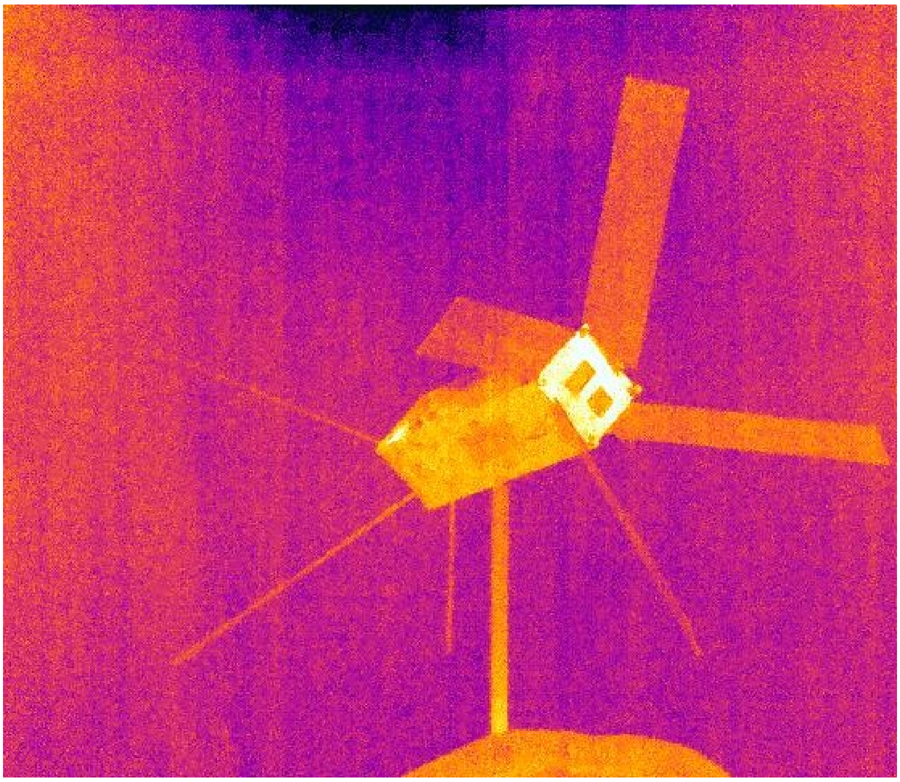


Figure 2.32: Delfi-n3Xt during Series O in occlusion

In an attempt to more comprehensively test the performance of FLIR, as well as to use relevant available resources, it was decided to use an available sun simulator to illuminate the Delfi-n3Xt target for Series Q. The simulator is small and was built to operate inside a reflective container while emitting wavelengths closely resembling that of the sun's visible spectrum. It heats up quickly and can only be operated for about 45 seconds at a time, and as all the targets were much bigger than the container designed for the sun simulator it had to be operated outside of the container, simply pointing at the target. The images (seen in Figure 2.33) closely resemble the images of occlusion, as the simulator isn't very strong (nowhere near the spotlight or terrace heater) and could only be operated for a short time, preventing the target from heating any meaningful amount. However, it was a worthwhile attempt.

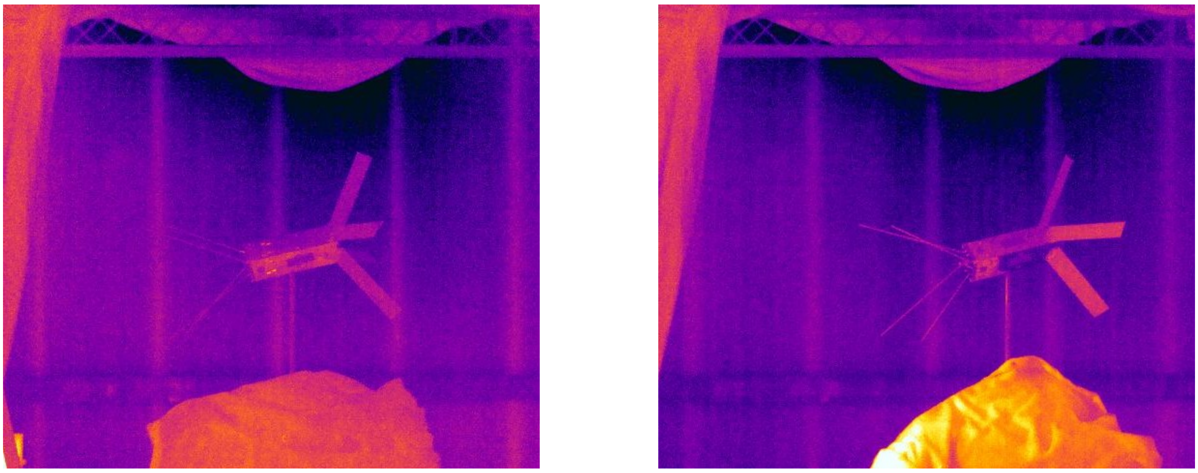


Figure 2.33: Series Q sun simulator experiment

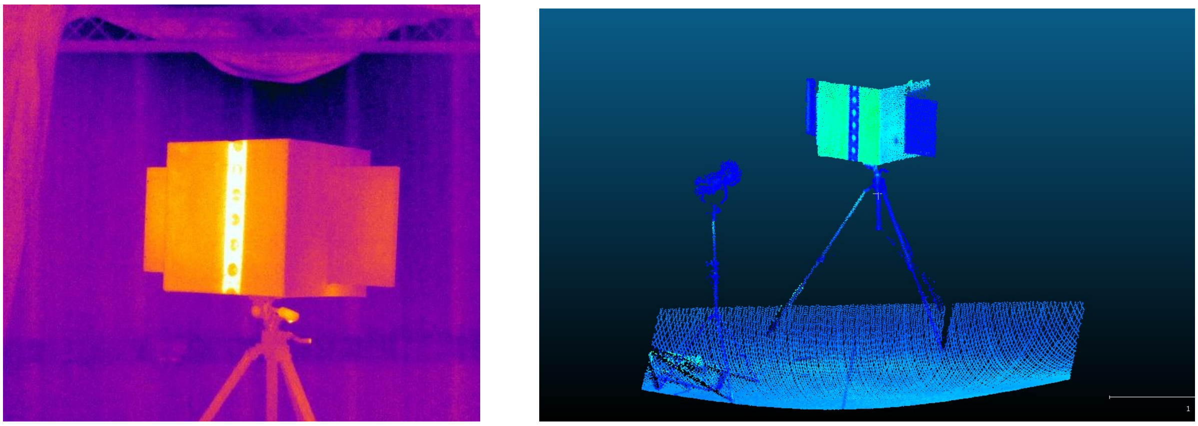
2.2.5. Boxsat Experiments

Boxsat was created to introduce variety and provide a target that was larger and could rotate around its longitudinal axis. Delfi-n3Xt was attached to a mount along an edge near its center. This means that it can spin around the axis of the mount through the center, but not longitudinally. The Panda's grip was not strong enough to hold neXt and would cover a part of the target if straps were used. A mounting system was contemplated that would attach to one or both sides, however it would be difficult to again avoid interference from the mounting as well as damaging the model. So Boxsat was created as a new target to provide a simple solution. Boxsat attached to a telescopic camera mount with factory-installed markings of degrees around two separate axes. Through this it was possible to record longitudinal rotation, two-axis rotation and to see how the system performed with a different target. The system is meant to determine the pose of *uncooperative unknown* targets after all.

Table 2.6 below lists each Boxsat experiment, with a more detailed description including start and end states available in Appendix B.

Experiment	Session Day	Lighting	Target Intrinsic Motion
Boxsat A	Boxsat 1	Direct	Rotation (Longitudinal Axis)
Boxsat B	Boxsat 1	Direct	Rotation (Off-axis)
Boxsat C	Boxsat 1	Occlusion	Rotation (Longitudinal Axis)
Boxsat D	Boxsat 1	Spotlight in-scene @ 50%	Translation left-right
Boxsat E	Boxsat 1	Spotlight in-scene @ 100%	Translation left-right
Boxsat F	Boxsat 1	IR Heater in-scene @100%	Translation left-right

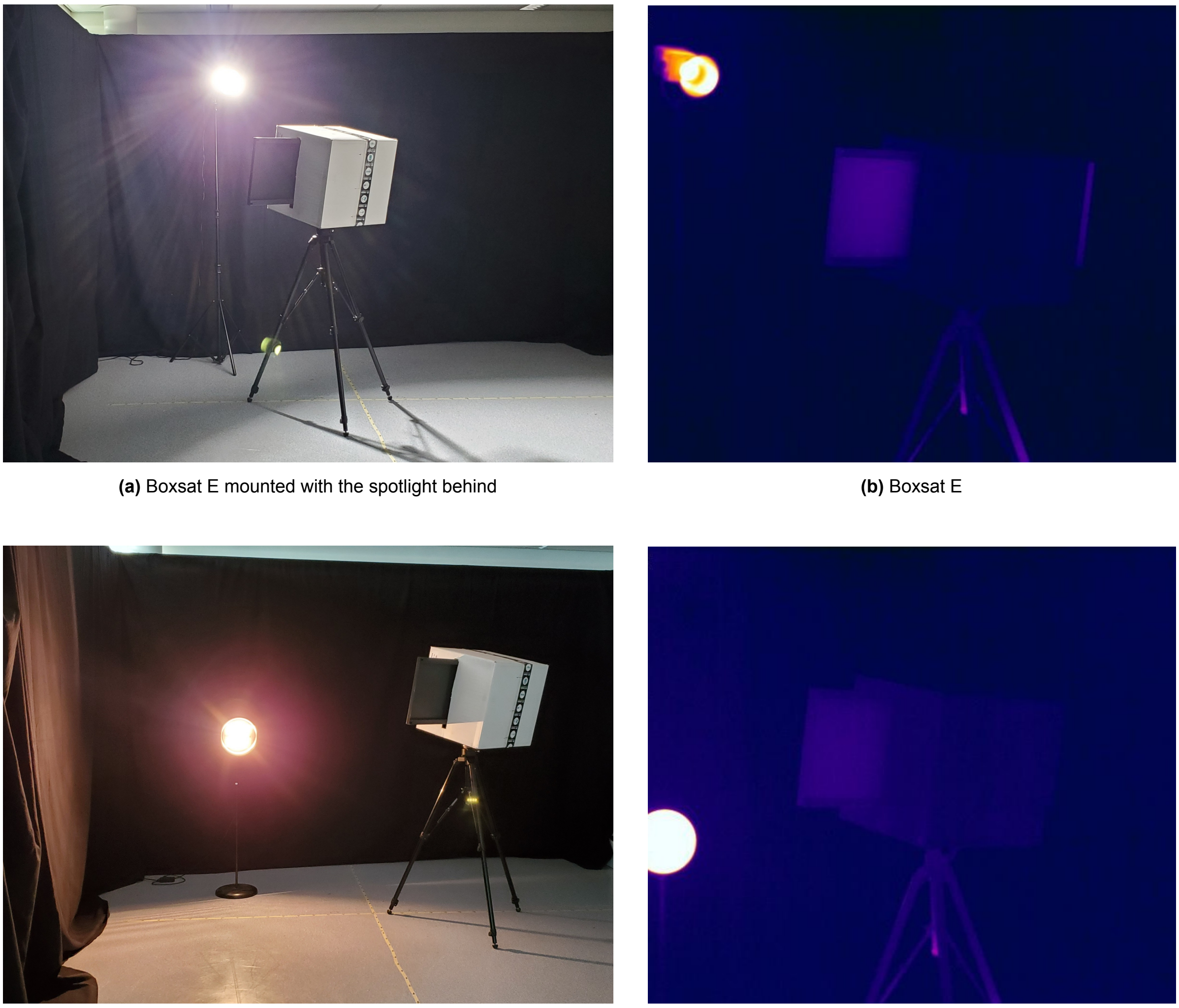
Table 2.6: Boxsat tests



(a) FLIR image of Boxsat in direct light

(b) Segmented LiDAR scan of Boxsat

Figure 2.34: Boxsat set on its two-axis tripod in the laboratory, and with the LiDAR-IR sensor view



(a) Boxsat E mounted with the spotlight behind

(b) Boxsat E

(c) Boxsat F with the terrace heater in-scene

(d) Boxsat F with the terrace heater in-scene

Figure 2.35: Boxsat with strong IR sources in-scene

Boxsat was also chosen to test a difficult scenario, already mentioned in Sections 2.1.4 and 2.2.1 during the discussions of the Feature Matching algorithm and FLIR camera respectively, in which a strong IR source is also in the scene with the target. Also in Section 2.2.1 it was already discussed how manually adjusting the temperature scale is essential to be able to view the target under these conditions. Boxsat D was the first of these tests with the spotlight previously used to illuminate the targets set at 50% power. Boxsat E increased the spotlight to 100% of its maximum power and was put on the other side of the scene. These three Boxsat experiments in particular are used to test limits of the algorithms while simultaneously simulating a space-environment scenario.

3

Results & Analysis

In this chapter, the results of both algorithms will be presented and discussed in depth. Before the results can be discussed, Sections 3.1 and 3.2 will first present the experimental results as the error (RMSE) in the average of the calculated transformation estimation compared to the ground truth for each experiment. With these results presented, Section 3.3 will analyse and discuss the results in depth and attempt to explain their sources before conclusions are drawn based on these results.

As a reminder of what each experiment entailed, Table 3.1 below defines each valid V&V experiment used in the algorithms based on the target's intrinsic motion and lighting conditions.

Experiment	Lighting	Target Intrinsic Motion
Series H	Direct	Translation far-near
Series I	Direct	Translation far-near & rotation
Series J	Direct	Translation left-right
Series K	Direct	Translation left-right & rotation
Series L	Direct	Closer translation left-right & rotation
Series M	Direct	Closer translation left-right
Series N	Occlusion	Translation far-near & rotation
Series O	Occlusion	Rotation
Series P	Occlusion	Closer translation left-right & rotation
Boxsat A	Direct	Rotation (Longitudinal Axis)
Boxsat B	Direct	Rotation (Off-axis)
Boxsat C	Occlusion	Rotation (Longitudinal Axis)
Boxsat D	Spotlight in-scene @ 50%	Translation left-right
Boxsat E	Spotlight in-scene @ 100%	Translation left-right
Boxsat F	IR Heater in-scene @100%	Translation left-right

Table 3.1: List of experiments with lighting conditions and target intrinsic motion defined

3.1. Color-ICP Algorithm Experimental Results

This section details the results obtained from the Color-ICP algorithm in each experiment. Recall that the RMSE is a measure of error between the calculated and ground-truth pose, with a higher RMSE indicating greater inaccuracy in the calculation.

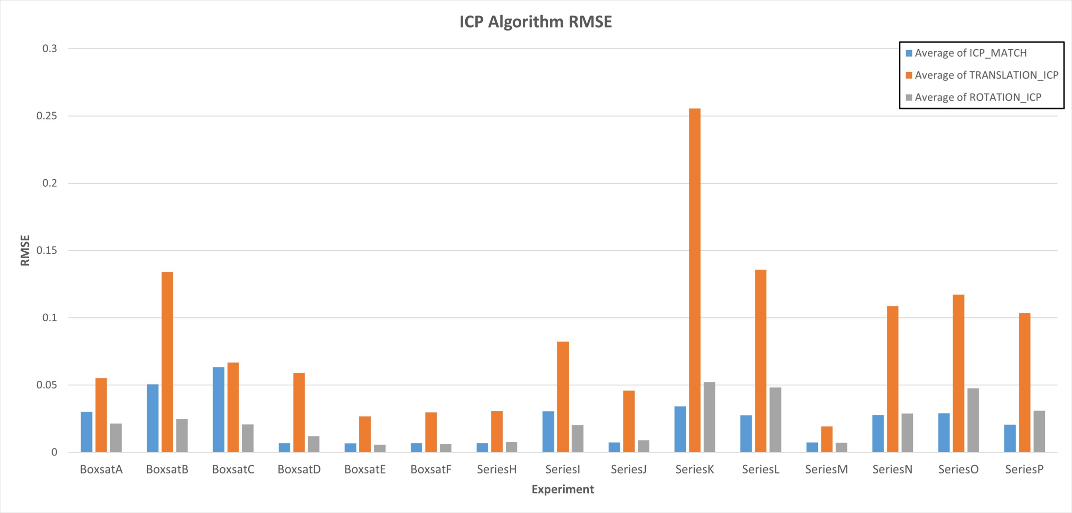


Figure 3.1: Average RMSE for the Color-ICP algorithm during all relevant experiments

Figure 3.1 shows the average RMSE of the Color-ICP algorithm's estimation of the rotation matrix (gray bar), translation matrix (orange bar), as well as the RMSE if the minimization metric (blue bar) of the transformed moving point cloud and the fixed one. The maximum and minimum values are listed below:

Metric	Value	Experiment	Lighting	Intrinsic Motion
Translation RMSE min	0.019	Series M	Direct	Closer translation left-right
Translation RMSE max	0.256	Series K	Direct	Translation left-right & rotation
Rotation RMSE min	0.006	Boxsat E	Spotlight in-scene	Translation left-right
Rotation RMSE max	0.052	Series K	Direct	Translation left-right & rotation
Minimization RMSE min	0.007	Boxsat D	Spotlight in-scene	Translation left-right
Minimization RMSE max	0.063	Boxsat C	Occlusion	Rotation

Table 3.2: Minimum and maximum average RMSE metrics for the Color-ICP algorithm for all relevant experiments

Figure 3.2 shows the same metrics (minimization in blue, translation in orange, and rotation in gray) but instead focuses only on the Eggman experiments, as does Table 3.3

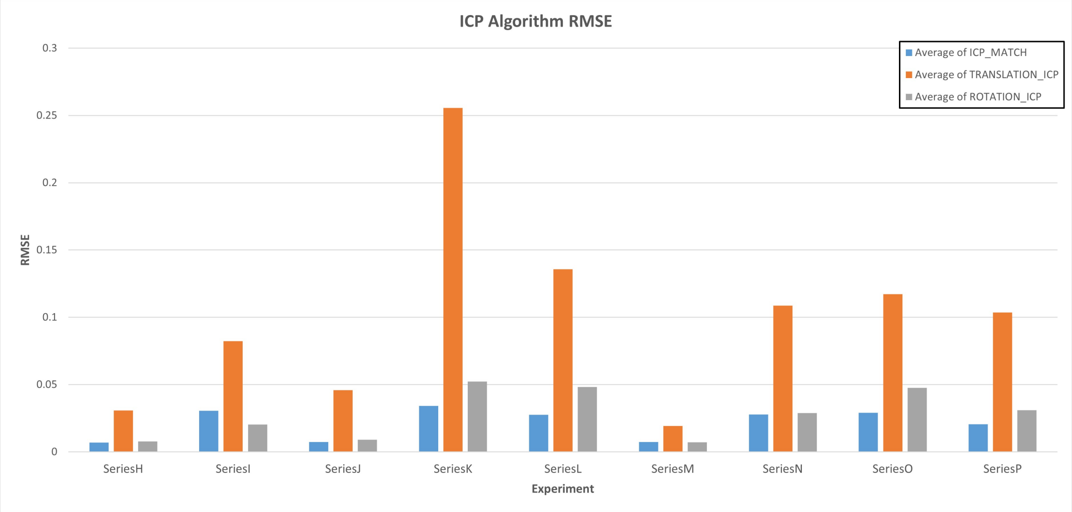


Figure 3.2: Average RMSE for the Color-ICP algorithm during the Eggman experiments

Metric	Value	Experiment	Lighting	Intrinsic Motion
Translation RMSE min	0.019	Series M	Direct	Closer translation left-right
Translation RMSE max	0.256	Series K	Direct	Translation left-right & rotation
Rotation RMSE min	0.007	Series M	Direct	Closer translation left-right
Rotation RMSE max	0.052	Series K	Direct	Translation left-right & rotation
Minimization min	0.007	Seires H	Direct	Translation far-near
Minimization max	0.034	Series K	Direct	Translation left-right & rotation

Table 3.3: Minimum and maximum average RMSE metrics for the Color-ICP algorithm during the Eggman experiments

Figure 3.3 shows the same metrics (minimization in blue, translation in orange, and rotation in gray) but instead focuses only on the Boxsat experiments, as does Table 3.4

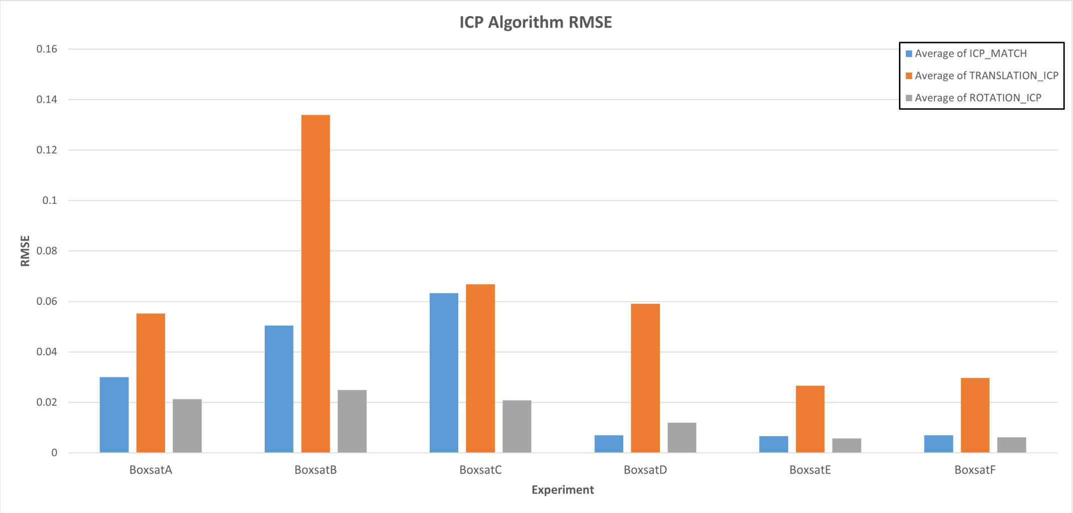


Figure 3.3: Average RMSE for the Color-ICP algorithm during the Boxsat experiments

Metric	Value	Experiment	Lighting	Intrinsic Motion
Translation RMSE min	0.027	Boxsat E	Spotlight in-scene	Translation left-right
Translation RMSE max	0.134	Boxsat B	Direct	Rotation (Off-axis)
Rotation RMSE min	0.006	Boxsat E	Spotlight in-scene	Translation left-right
Rotation RMSE max	0.025	Boxsat B	Direct	Rotation (Off-axis)
Minimization RMSE min	0.007	Boxsat D	Spotlight in-scene	Translation left-right
Minimization RMSE max	0.063	Boxsat C	Occlusion	Rotation

Table 3.4: Minimum and maximum average RMSE metrics for the Color-ICP algorithm during the Boxsat experiments

3.2. Feature Matching Algorithm Experimental Results

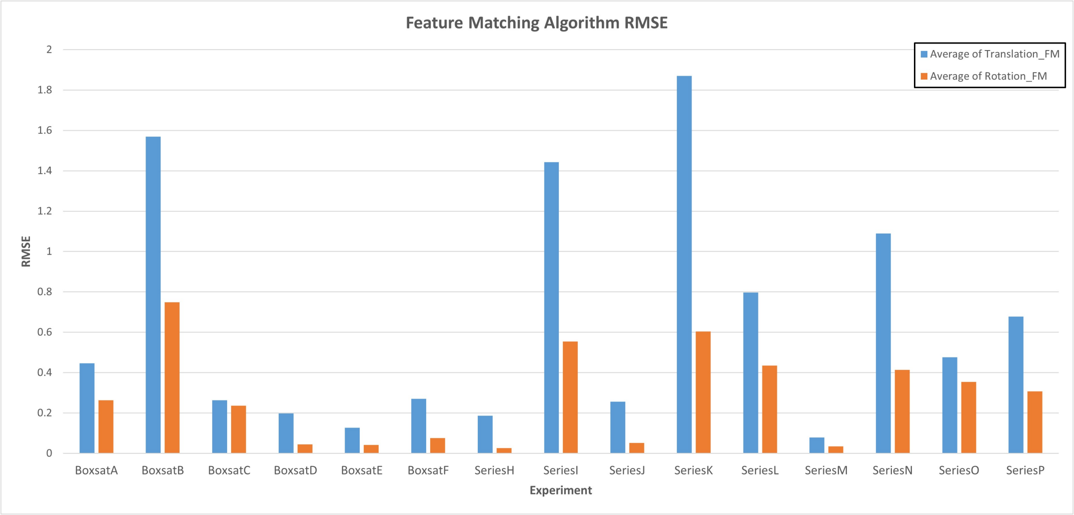


Figure 3.4: Average RMSE for the Feature Matching algorithm during all relevant experiments

Here, the average rotation matrix RMSE is shown in orange, while the Translation RMSE is shown in blue.

Metric	Value	Experiment	Lighting	Intrinsic Motion
Translation RMSE min	0.079	Series M	Direct	Closer translation left-right
Translation RMSE max	1.870	Series K	Direct	Translation left-right & rotation
Rotation RMSE min	0.0263	Series H	Direct	Translation far-near
Rotation RMSE max	0.749	Boxsat B	Direct	Rotation (Off-axis)

Table 3.5: Minimum and maximum average RMSE metrics for the Feature Matching algorithm during all relevant experiments

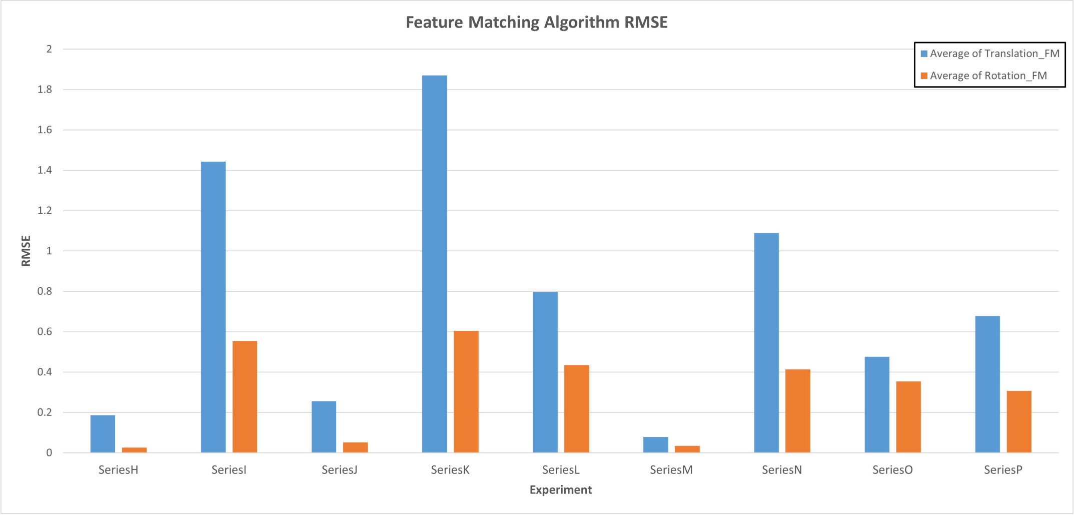


Figure 3.5: Average RMSE for the Feature Matching algorithm in the Eggman experiments

Once again, the average rotation matrix RMSE is shown in orange, while the Translation RMSE is shown in blue but only for the Eggman experiments.

Metric	Value	Experiment	Lighting	Intrinsic Motion
Translation RMSE min	0.079	Series M	Direct	Closer translation left-right
Translation RMSE max	1.870	Series K	Direct	Translation left-right & rotation
Rotation RMSE min	0.0263	Series H	Direct	Translation far-near
Rotation RMSE max	0.605	Series K	Direct	Translation left-right & rotation

Table 3.6: Minimum and maximum average RMSE metrics for the Feature Matching algorithm during the Eggman experiments

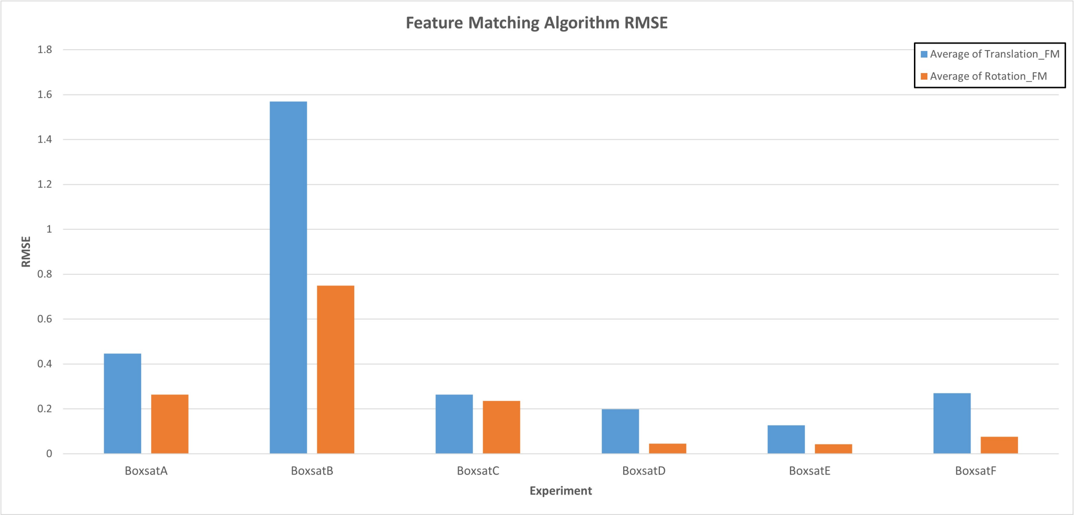


Figure 3.6: Average RMSE for the Feature Matching algorithm in the Boxat Experiments

And finally, the average rotation matrix RMSE is shown in orange, while the Translation RMSE is shown in blue but only for the Boxsat experiments.

Metric	Value	Experiment	Lighting	Intrinsic Motion
Translation RMSE min	0.127	Boxsat E	Spotlight in-scene	Translation left-right
Translation RMSE max	1.569	Boxsat B	Direct	Rotation (Off-axis)
Rotation RMSE min	0.042	Boxsat E	Spotlight in-scene	Translation left-right
Rotation RMSE max	0.749	Boxsat B	Direct	Rotation (Off-axis)

Table 3.7: Minimum and maximum average RMSE metrics for the Feature Matching algorithm during the Boxsat experiments

From this data it can immediately be seen that the Feature Matching algorithm did not perform as well as the Color-ICP algorithm. This is true for both metrics, but particularly for the error in translation. The next section will interpret these results and provide various insights into their root causes.

3.3. Discussion

Some of the results obtained defy expectation. In this section, the results of both algorithms developed will be discussed, and some conclusions will be drawn as to why these results came about as well as explanations of their possible sources. Section 3.3.1 discusses the expectations against the realities of the results obtained, after which the accuracy of the algorithms will be discussed in Section 3.3.2 before discussing the potential sources of error in Section 3.3.3. Continuing in the discussion, Section 3.3.4 will describe the limitations of each method before relating to, and learning from, a real-world RPO mission, finally closing the discussion with a postulated use case for the methods presented in this research.

3.3.1. Theory vs. Observations

Though this research was not approached with any bias or definite performance expectations, there were some preemptive ideas on which lighting and movement scenario would benefit each algorithmic approach. Many of the results were unexpected and proved to be very interesting.

Performance with Respect to Lighting & Motion

One of the main research questions postulated regarded the IR-LiDAR combination's independence of lighting conditions. Though this is logical theoretically, the reality of the experiment preformed indoors introduced some irregularities compared to the real-world use case in the vacuum of space. Figure 3.7 shows the resulting RMSE for both algorithms in direct lighting, while Figures 3.8 and 3.9 show the RMSE for both in scenarios of occlusion and IR background lighting respectively.

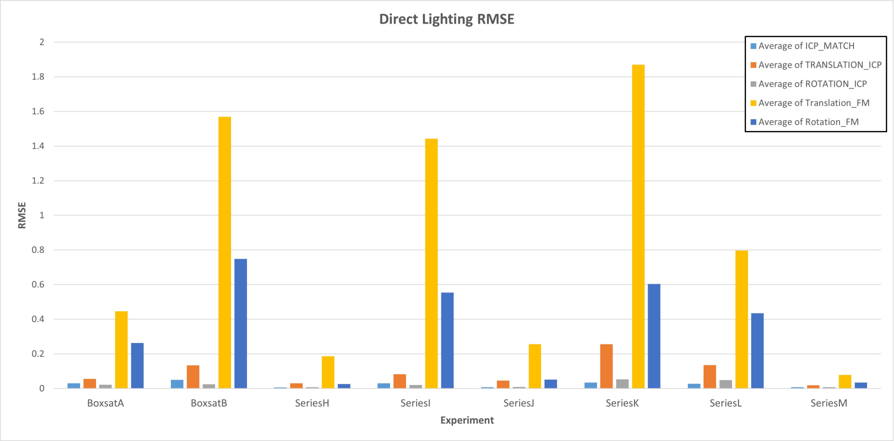


Figure 3.7: RMSE for Color-ICP algorithm and Feature Matching Algorithm for target in direct lighting

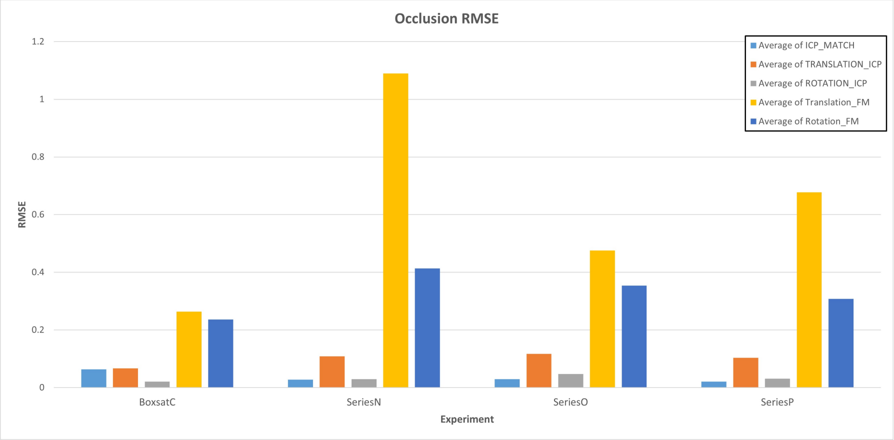


Figure 3.8: RMSE for Color-ICP algorithm and Feature Matching Algorithm for target in occlusion

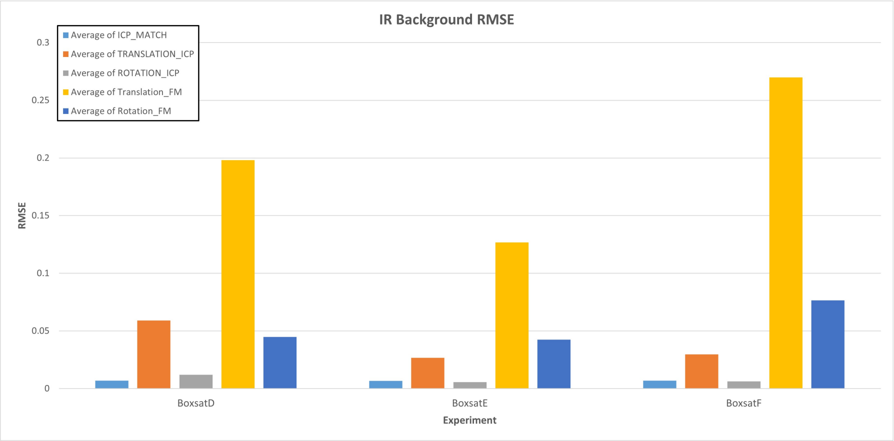


Figure 3.9: RMSE for Color-ICP algorithm and Feature Matching Algorithm for target with harsh IR background radiation

The Feature Matching algorithm proved to be rather temperamental. Though this is not entirely unexpected, as the IR sensor is noisier and more sensitive to background interference, some interesting and promising results were obtained from the use of this method.

It is interesting to see how the Feature Matching algorithm has such a wide range of results in direct lighting, as seen in Figure 3.7. Some of its most accurate results are present as well as its most erroneous. Series H & I were both in direct lighting at the same distance, however Series I included target intrinsic rotation. The same is true for Series J & K respectively. It seems that in both cases, the addition of target intrinsic rotation caused higher RMSE values. It can be seen in Series L that the performance of the Feature Matching algorithm was drastically improved for the scenario of target intrinsic rotation and translation when the target was closer to the sensor suit. Series I & N have the same intrinsic motion (far to near with rotation) but have different lighting conditions. Surprisingly, Series N performed better in the dark than Series I in direct light. The same can be said for Series L & P, though not as drastic.

As it was mentioned before, the Feature Matching algorithm is incredibly sensitive to a wide array of interdependent tuning parameters. From experience, it can be said that the tuning of both the feature detection and matching parameters is somewhat of a dark art, in that it takes a great amount of understanding and patience to tune the algorithm correctly to not retrieve ridiculous results, detecting and matching features that have no place in a legitimate result. Because of this, it can not be ruled out that the parameters were not optimized to the fullest for each scenario and measurement, even after so many attempts. This leads to an outstanding question regarding the potential performance enhancement offered by the implementation of an automated parameter optimization step. This is an active area of research in the field of computer vision, and techniques range from evolutionary algorithms to deep learning methods. This topic is diverse and complex, and an academic textbook on the matter can be found in the following source: [58]. The possibility is intriguing.

One of the most visually striking experiments is that of Series M during the Eggman experiments, shown below in Figure 3.10. In this scenario, the target (Delfi-n3xt) is close, in high resolution and feature-rich. This scenario is unsurprisingly one of the more ideal scenarios for the Feature Matching algorithm, though Series H and Series J are also quite accurate with a far-to-near detection scenario.

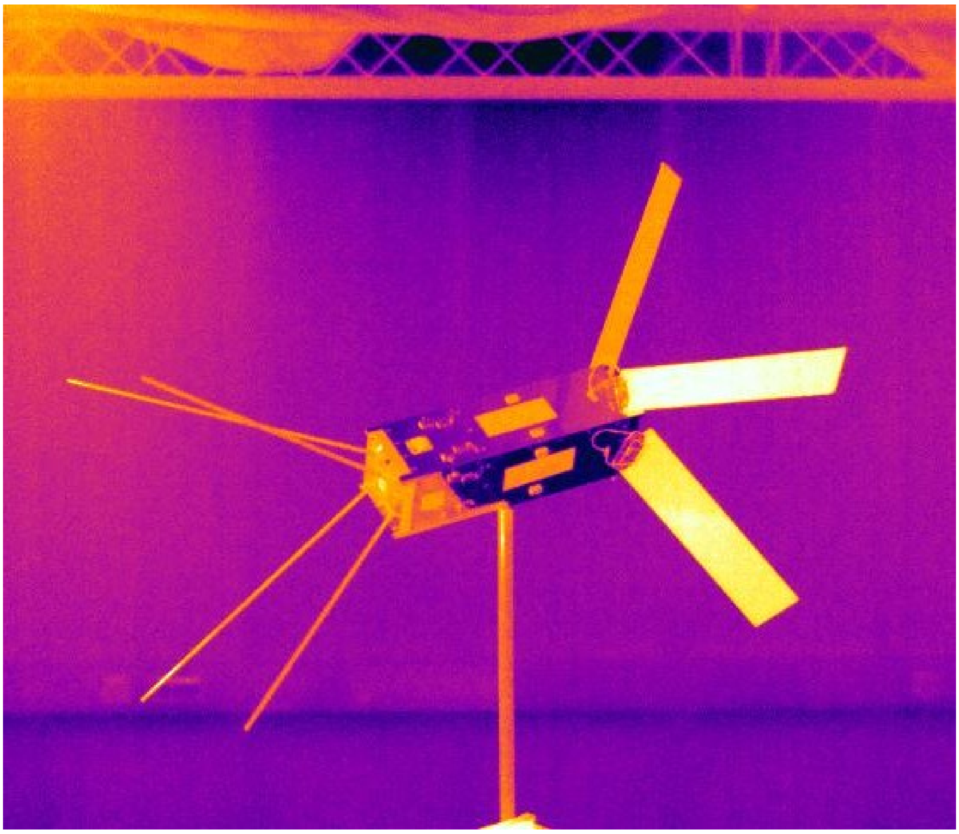


Figure 3.10: Ideal feature detection scenario in Series M

Computer vision is a vast and fast-moving river of possibilities and potential. The anisotropic filtering, bilateral filtering, and eventually implemented histogram equalization were researched as solely an application for the worst case scenario of Boxsat D, E, & F, in which the target is all but hidden due to the strong background IR source. Upon implementation of the cropping and histogram equalization detailed in Section 2.1.4, it seems that these techniques actually helped to turn these worst case scenario experiments into one of the more accurate estimations. These methods transformed the images into something very different visually, but yet the features were maintained and distinct. This distinct enhancement was consistent throughout the frames and led to a surprisingly low and consistent RMSE.

In ICP algorithms, the addition of color information into the equation is a relatively new concept without a wealth of prior knowledge indicating its utility in scenarios in which the color range is subtle. Regarding the Boxsat D, E & F experiments, which were designed to represent an extreme and difficult scenario, the Color-ICP algorithm made no use of image processing processing tools such as histogram equalization, as was the case in the Feature Matching algorithm, to accentuate the target from the background. However this appears to not be necessary. When looking at the Boxsat experiment results in Figure 3.3, Boxsat D, E & F all have better accuracy compared to Boxsat A, B &C. One idea is that the neutralization of the color variance experienced in the first three Boxsat experiments assisted in the Color-ICP registration. One can imagine that the motion of the target in direct light would create inconsistent heating and therefore color between observation frames. However, it is interesting that the strong IR source shines onto the target, albeit its backside, during the final three Boxsat experiments leading to an inconsistent color between frames. This is, of course, to a much lesser extent compared to direct lighting scenarios. An example of this subtle inconsistency can be seen below in Figure 3.11.

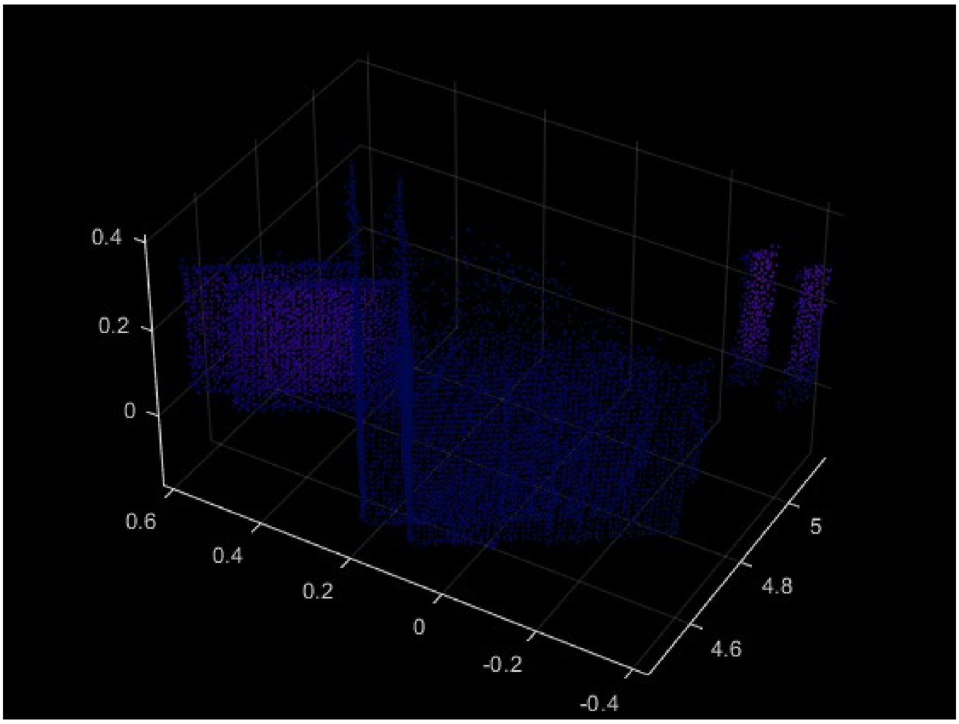


Figure 3.11: Minimal color variation between point clouds assisting in Color-ICP registration during Boxsat E

Other than the harsh IR background scenario, the occlusion scenario delivered the most consistent point color between frames. As the targets were left in the dark for a considerable time before the occlusion experiments were performed, the targets remained at ambient room temperature throughout each experiment, giving a consistent (or minimally variant) color value between frames. One addendum to this statement is that the sensor could occasionally change RGB temperature ranges between frames due to the FLIR camera's fidgety automatic scaling during instances of small temperature ranges in a scene. Figure 3.12 below shows an example of this. Please note that this was generally a very rare occurrence and would be immediately rectified when observed. Another exception, which happened more often, is in the case that IR radiation from elsewhere in the laboratory is reflected by the surface of Delfi-n3Xt. This is an unfortunate reality of environmental testing limits in experiments. As it was mentioned before, the Livox and computers for data gathering all run hot, with summer sunshine causing the dark laboratory curtains to be warm in addition to the author's own body heat.

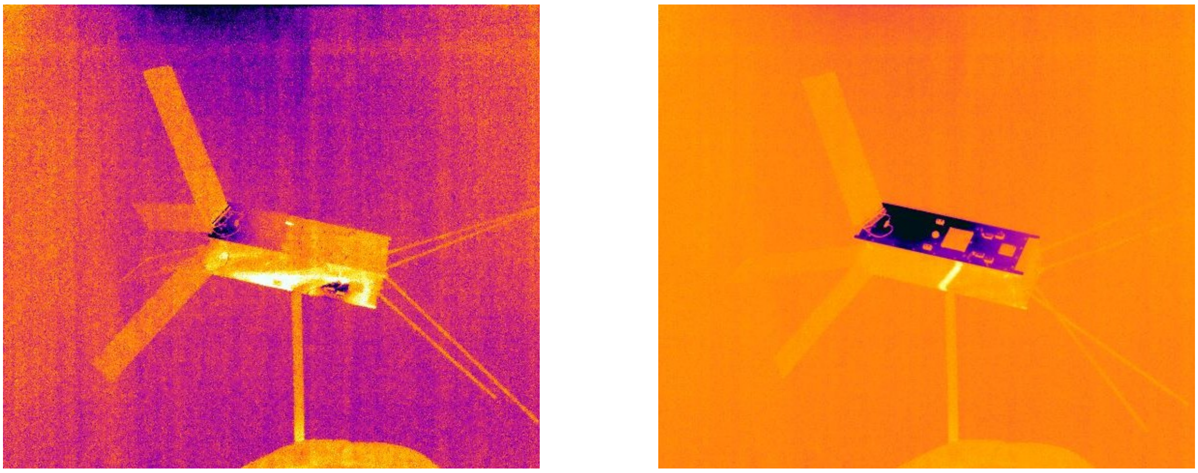


Figure 3.12: An example of the FLIR camera unexpectedly changing temperature scales between frames during an occlusion scenario. Reflected radiation can also be seen.

In terms of target intrinsic motion, the addition of rotation to translation resulted in a comparatively higher RMSE for the ICP algorithm. This is not unexpected as it results in a more complicated registration scenario. As it was discussed in Section 2.1.3, the inlier ratio becomes particularly when when intrinsic rotation is present, as surfaces visible in one frame disappear in the next. Comparing Series K & L and J & M shows that a smaller distance to the target also improved performance.

ICP algorithms work by iteratively minimizing the differences between points, with the Generalized Color-ICP presented here considering the location, surface normals and color of both point clouds involved in the registration. The addition of color, relating to the fused IR-LiDAR points, could initially be seen as a boon for the registration process in that it considers an extra variable to assist in the registration process. However, it is in fact the occlusion experiments (Series N, O, P, Boxsat C) that have a lower RMSE compared to the experiments exposed to, and being heated by, direct light. This confirmed an initial suspicion that a change in pose would lead to time-variant heating and cooling of the target that could be a detriment to the color registration ICP metric. One frame can contain a surface-point heated in direct light, while the second frame could see that same surface-point being cooled in the shadow. Therefore it seems that the experiments in occlusion and with harsh background IR radiation benefit the algorithm slightly due to their consistent or low color variance between frames. This can be examined more deeply through a review of the L*a*b* color space.

L*a*b* Color Space and the Visual Spectrum

If the Color-ICP algorithm instead used a visible-spectrum camera, would it also suffer from this change of light incidence? It can be reasoned that the color registration with these sensors would suffer from the same fate, as the RGB values of a target in the visible spectrum would also change due to changes in light incidence between frames. A surface in sunlight transitioning into shade will have a different RGB color value between the two observations. However to better understand how this actually affects the registration, a deeper look into MATLAB's Generalized Color-ICP method is required. Based on the original research that MATLAB's function is based on, this method does not register a point's RGB value. It instead converts pixels from the RGB color space to the L*a*b* color space, and uses these L*a*b* values as its color metric during registration.

The L*a*b* color space (also known as CIELAB) was introduced in 1976 by the International Commission on Illumination (CIE in French) with L* being Lightness, and a* and b* being red-green and yellow-blue color-opponent dimensions respectively. The RGB color space is not well suited for describing differences in colors as it is not perceptually uniform, meaning that the Euclidean distance between two RGB colors is not proportional to the perceived distance. The L*a*b* color space is more appropriate to describe color distances as it has more perceptual uniformity and can better describe high illumination variability.[25]

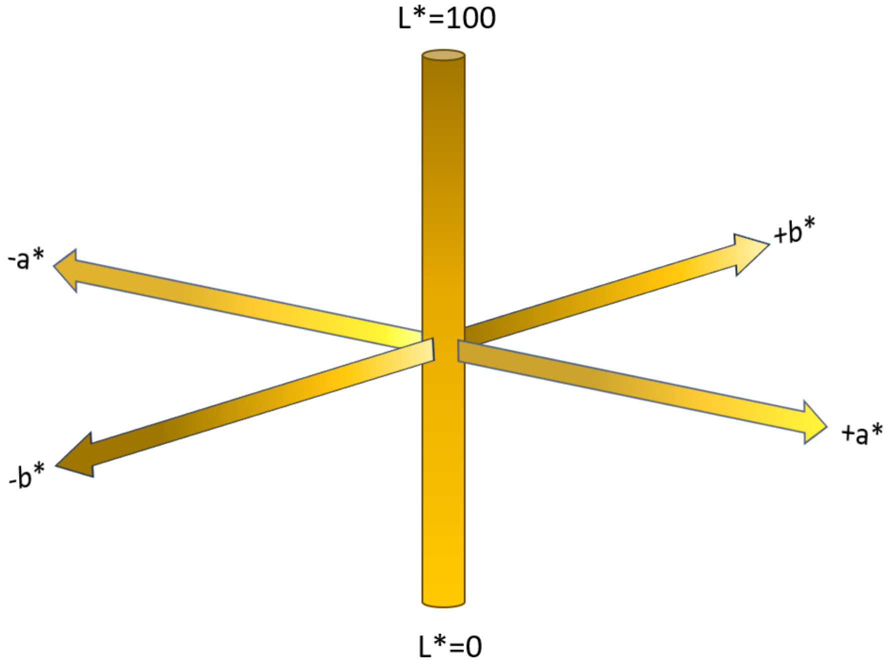
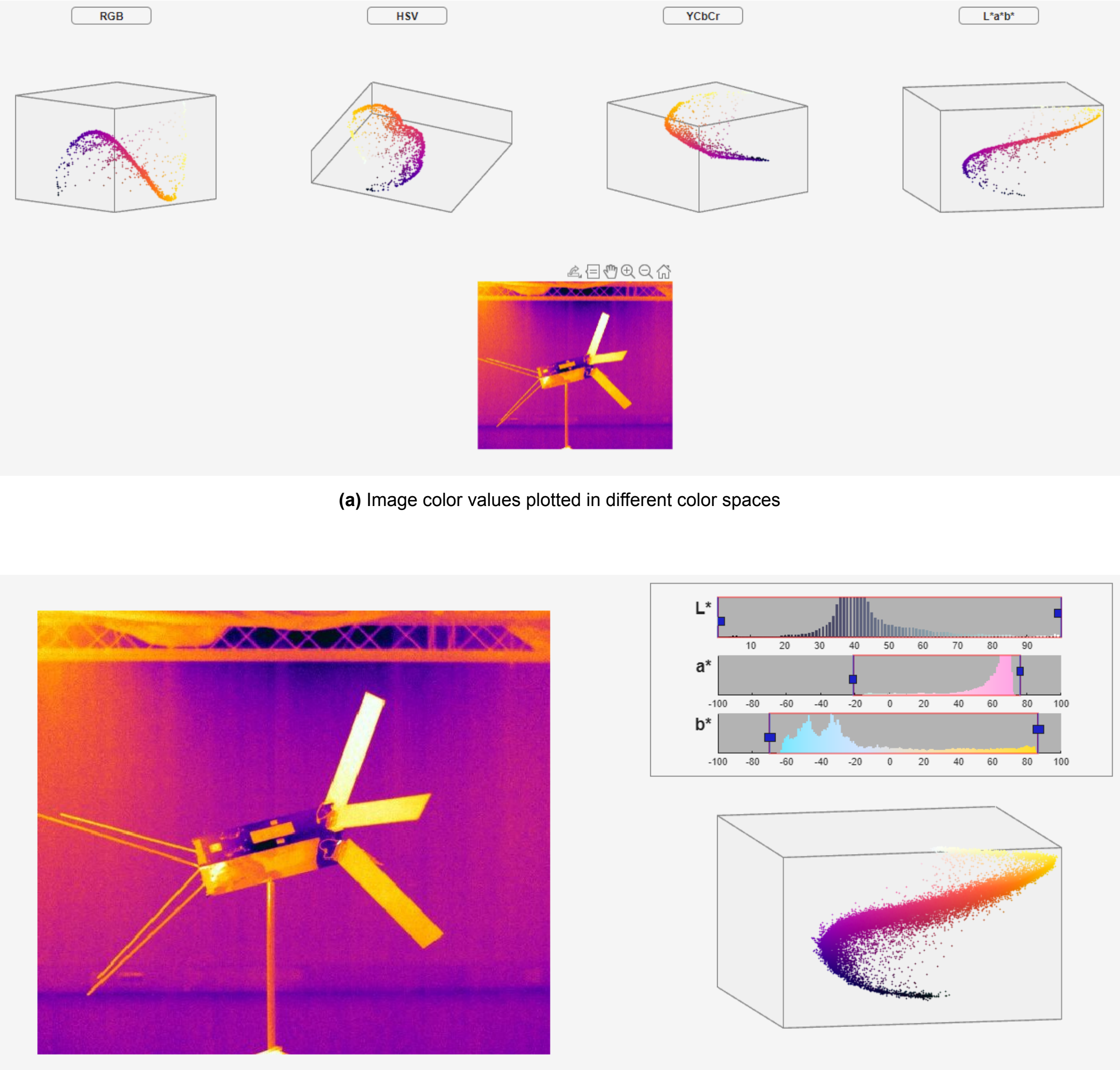


Figure 3.13: The L*a*b* color space. Recreated from [25]

As previously discussed, Generalized-ICP algorithms strive to minimize the Euclidean distance between point locations, normals and, with the Color-ICP method, color as well. The L*a*b* is well suited for the color component as it better represents the Euclidean distance to be minimized during the registration. As mentioned above, the L*a*b* strives to be perceptually uniform, meaning that it represents color in a way that corresponds more closely to human vision. Thinking again about the change in light incidence of the surface of the target as it rotates and translates, this change would affect the Lightness (L*) component in particular as the incidence and intensity of light on a target changes with time. With this rationalization, the L* component in the visual spectrum would be affected as well and the color registration would again be detrimental to the accuracy of the registration in direct lighting as opposed to occlusion.

The unique nature and differences between color spaces makes this an interesting aspect of the Color-ICP algorithm. For use in a real ADR mission, it would be useful to think about how the color of a target changes in different color spaces in both the visual and infrared spectra, and what each color space can do to ensure a light agnostic value. Based on the discussion of the L*a*b* space, it would be interesting to create a Color-ICP algorithm that uses only the a*b* elements, eliminating the Lightness component that varies so much with light incidence and intensity. Figure 3.14 below shows the same IR image of Delfi-n3Xt plotted in four different color spaces.



(a) Image color values plotted in different color spaces

(b) Image color values in the L*a*b* color space

Figure 3.14: Delfi-n3xt in direct light shown in different MATLAB color spaces

3.3.2. Accuracy & RMSE Comparison

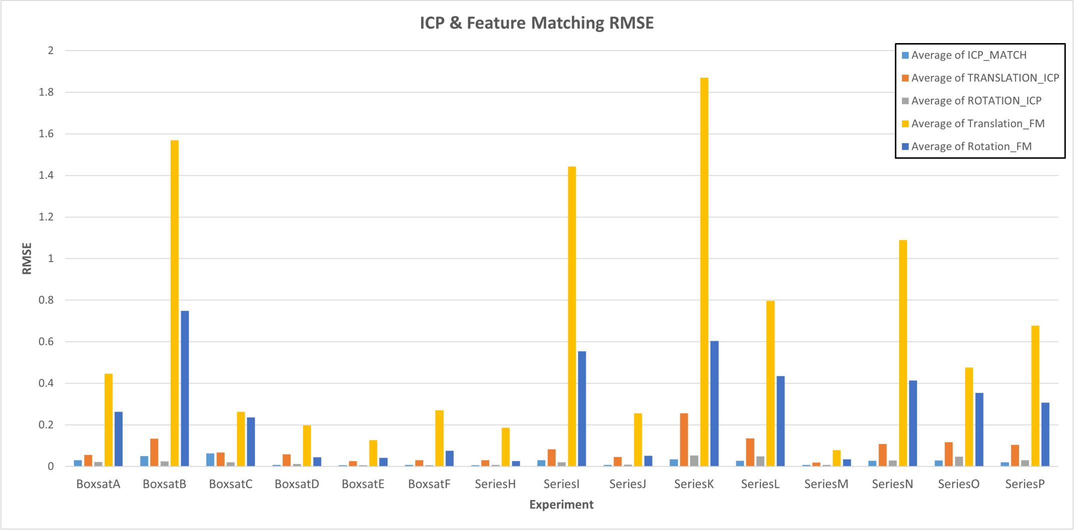


Figure 3.15: RMSE for Color-ICP algorithm and Feature Matching Algorithm together

A quick look at Figure 3.15 shows the drastic outlier that is the translation matrix RMSE of the Feature Matching algorithm. The rotation matrix of this algorithm is more erroneous compared to the Color-ICP algorithm as well, but not as drastic. The Color-ICP algorithm seems to be much more accurate by a large margin. The temperamental nature of the Feature Matching algorithm was discussed above, but so were the inaccuracies of the color registration.

The translation component of the results appears to be somewhat of an Achilles's heal for the Color-ICP algorithm as well, with error results that are higher than that of rotation, though much lower in comparison to the Feature Matching algorithm. Remember that this is in the world coordinate frame of reference. Rotation seems to be a metric that is universally more accurate compared to translation, with the Color-ICP algorithm performing incredibly well across the range of experiments.

Both algorithms saw an increased RMSE in scenarios of target intrinsic translation and rotation compared to their translation-only counterparts. As it was discussed above, this causes the ICP algorithm to rely more on an accurate definition of inliers, while it decreases the overall usable features in the Feature Algorithm and makes matching more difficult with this change in perspective.

For the Feature Matching algorithm, recall that it is dependent on having a high LiDAR point density on the target. The results of experiments reinforces this, with the RMSE of lateral translation experiments being higher than their far to near counterparts. While the target ventured to less point-dense areas during the lateral experiments, the far to near experiments saw the target remain in the point clouds dense center, which continued to get denser as it grew near.

These results are based on the estimated transforms from each algorithm compared with the control transformation based on the intrinsic motion of the target recorded during the experiments. However, due to imperfect experimental controls, small errors can creep in. More on this in Section 3.3.3 below.

3.3.3. Further Potential Sources of Error

As this was a hardware V&V experiment involving bespoke algorithms, there are many real-world and formulaic sources of error to be considered. Throughout the discussion of the results, various error sources have already been examined, however the possibilities have not yet been exhausted and include possibilities that were not anticipated prior to the conclusion of the development process.

The above sections showed that the Feature Matching algorithm was consistently less accurate compared to the Color-ICP algorithm. One reason for this can be explained in the way in which the 3D point clouds were ultimately used in each algorithm. The Color-ICP algorithm denoised and downsampled its point clouds, effectively minimizing the intrinsic error in the LiDAR sensor to create uniform and coherent shapes, and would register parameters including surface-normals and color.

The Feature Matching algorithm, on the other hand, needed as many LiDAR points as possible to blanket the image pixels in imPts so that the features detected in the image could have a 3D location in the world coordinates. In this algorithm, there is no denoising or voxelization, as any reduction in the point cloud count could result in the missed opportunity to register a strong and accurate feature. Therefore, the noise and inaccuracies of the LiDAR persist throughout the algorithm, leading to more inaccurate results compared to the Color-ICP algorithm.

One idea to fix this problem is to implement a shape-based interpolation filter. It is possible to detect various shapes in point clouds such as planes and (semi-)cylinders. It would be interesting to detect the planar faces in the point clouds of both Delfi-n3Xt and Boxsat, replace points within that plane with a dense and flat section of points based on the surface normals and colors of their neighbors, and have that new dense point cloud for use in the Feature Matching algorithm. This would guarantee that more features have an associated LiDAR point in addition to cleaning the point cloud as voxelization did for the Color-ICP algorithm.

A separate potential source of error has to do with errors in the experimental setup. Figure 3.16 and Figure 3.17 on the following page show examples of inaccurate and accurate control transformations respectively. An accurate control transfer is based on the intrinsic transformation recorded during the experiment. The experiment was done as accurately as possible, with measurements for target intrinsic rotation and translation being recorded for every frame. However, as it is the real world, errors can creep in, as shown in Figure 3.16. These errors were often small, but not insignificant. This is not a case of cascading error, as the transformation only relates the current fixed and moving point clouds in each frame without any further time history, so the error form this source should remain relatively constant as there was no point in the experiment when a drastic translation or or rotation error was recorded. An explanation for the Series experiments could be the wheels of the Robotnik rover. They are able to go forwards and backwards as well as laterally, however the wheel design that makes this possible seems to allow for periodic motion in an unintended direction.

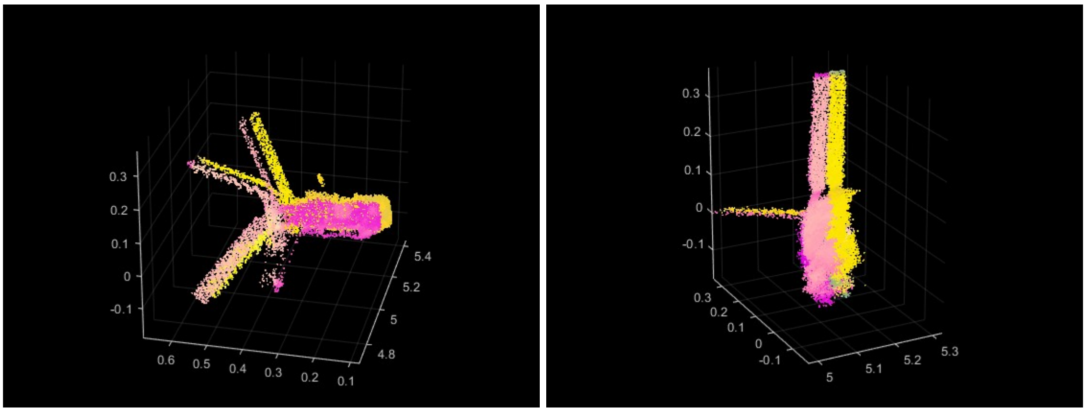


Figure 3.16: Two offset control transforms due to inaccuracies in the experiment

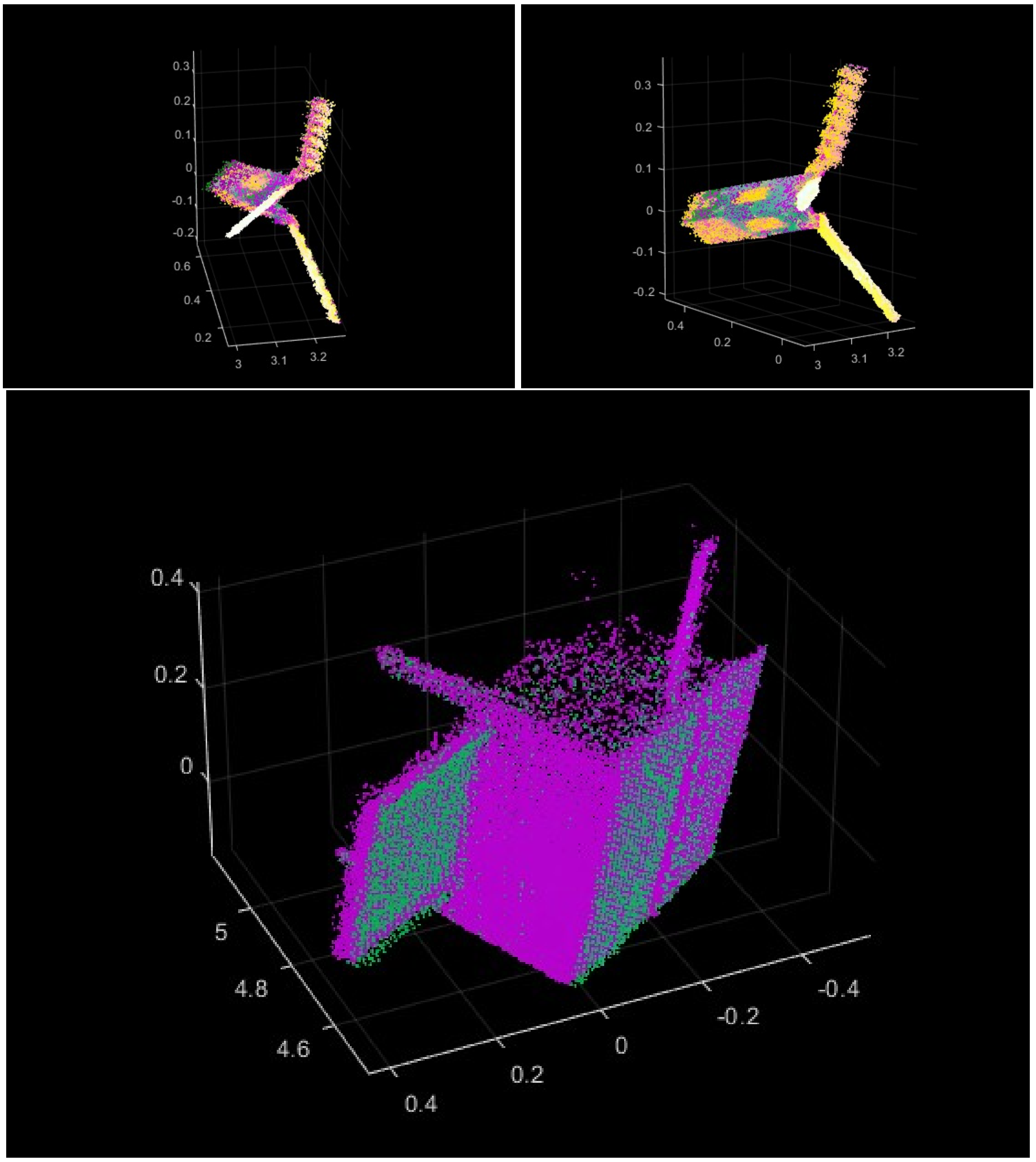


Figure 3.17: Examples of well estimated control transforms due to accurate experimental configurations

3.3.4. Limitations, Real-World Examples & Use Case

Now that the methods and results have been discussed, it is possible to draw some some conclusions on the limitations of each algorithm, while also discussing a relevant real-world mission before a use case for these methods is postulated.

Limitations

The Color-ICP has performed very well during the experiments. Its main struggles stem from background interference, such as close proximity to a non-reflective background, noise induced from the LiDAR sensor, and variance in color from a moving target with respect to a heat source. With all these things in consideration, the Color-ICP algorithm still drastically outperforms the Feature Matching algorithm. A question still remains about *this particular LiDAR sensor's* ability to sense small and distant objects. The Livox Mid-70 necessitated the use of stop-motion photography, which is a luxury that cannot be afforded to real-life orbital applications. It also has an unfortunate habit of curving it's photons around corners and edges, and registering points in empty space as well as interpolating points between planes with similar x-axis values. If this particular LiDAR sensor was to be considered for in-situ pose estimation, it would be inadvisable. A real time sensing, and hopefully more accurate, LiDAR with a consistent pattern and point density would be much more appropriate.

The FLIR IR sensor was high quality but was still susceptible to sensor noise in indoor occlusion including the adverse vertical striration effect in this particular lighting. The strong IR background radiation source was implemented to challenge the IR sensor, but CV and image enhancing techniques proved to be, in some cases, more accurate than with the utilization of raw data. Overall this sensor preformed well under a wide range of conditions, and it has proven that many of the negative assumptions about the IR sensor's suitability for CV and pose estimation tasks can be laid to rest.

Considering specifically the instruments used in this research, the Livox LiDAR's error caused few problems for the Color-ICP algorithm due to the denoise and downsampling steps. However the Feature Matching algorithm could not afford to incorporate these steps, as it is dependant on having as many imPts as possible. This intrinsic sensor error was compounded by the algorithm's extreme sensitivity to tuning parameters and led ultimately to a less accurate pose estimation method, albeit a promising one with intriguing possibilities for improvement.

It is also noticeable in every point cloud recorded of the Delfi-n3xt that the antennae are missing. The antennae are essentially small-diameter rods, too small for the LiDAR to accurately detect it seems, with the NIR photons registering the pattern of the antenna as lumps on the wall of the background, which was in some scenarios more than 4 meters away from the target. Based on the consistency of this effect, it can be reasoned that without the wall behind the target, the photons influenced by the antennae would continue on without reflecting back to the sensor.

This leads to an interesting thought experiment about the Livox's effective range. The product specifications state that it has a detection range of up to 260 meters, achievable only if the target has a reflectivity of at least 80%.[53] Recall the discussion in Section 2.2.1 of the Livox's unique scanning pattern and point density. The laser emits an ever expanding cone leading to less dense coverage per square meter as distance from the sensor and cone-center increases. Once again, this would be disastrous for the Feature Matching algorithm as it depends on having the densest LiDAR coverage possible.

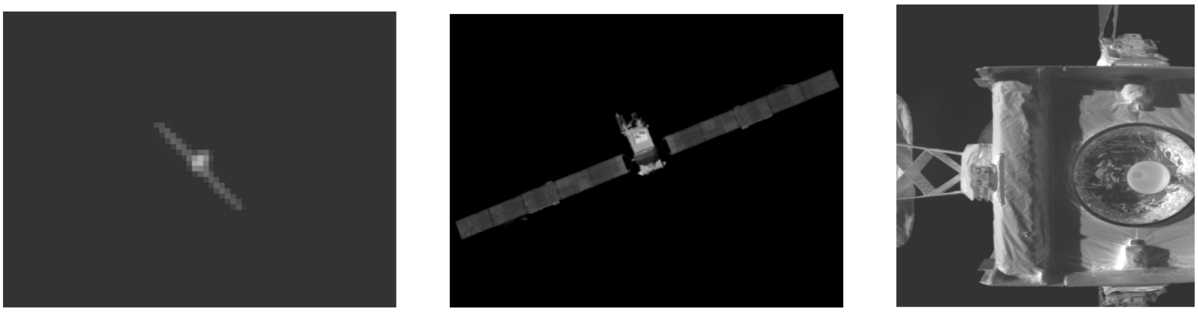
All of these things will be taken into consideration to formulate a potential use case for the LiDAR-IR combination after the following Section, in which an interesting RPO mission will be discussed.

Empirical Relevance Supporting These Findings

Laboratory size constraints and terrestrial testing limitations led to the experiments being downsized compared to the large distances and clear lines-of-sight found in a real ADR scenario. However, the results from these experiments can be extrapolated to larger distances, with additional data from two real RPO spacecraft providing empirical evidence to support these deductions. As mentioned before in

Section 1.2.2 discussing past missions implementing electro-optical sensors for non-Earth observation, the Northrop Grumman MEV-1 & MEV-2 missions successfully used visual cameras, LWIR cameras and a scanning LiDAR to approach and attach to separate known cooperative targets in GEO. The LWIR camera used by the MEV spacecraft is the Malin Space Science Systems ECAM-IR3A utilizing an uncooled microbolometer with a resolution of 640x480 pixels and a 17μm pitch.[44][51] The FLIR A655sc used in this research also utilizes an uncooled microbolometer with a resolution of 640x480 pixels and a 17μm pitch,[54] leading to a convenient comparison of the experimental data with real-world data. The main difference is that the MEV used four of these LWIR cameras, two narrow-FOV and two wide-FOV, to provide resolution at a variety of ranges, while the FLIR camera available had a fixed FOV.

Northrop Grumman states that the narrow-FOV LWIR cameras could detect and track their client target at a range in excess of 10km, with the target fully resolved (though quite pixelated) at a range of 3.2km, with the wide-FOV LWIR cameras transitioning for optimal performance within 15m.[44] Figure 3.18 shows LWIR images from the MEV-2 mission demonstrating the resolution of the target at progressively smaller distances.



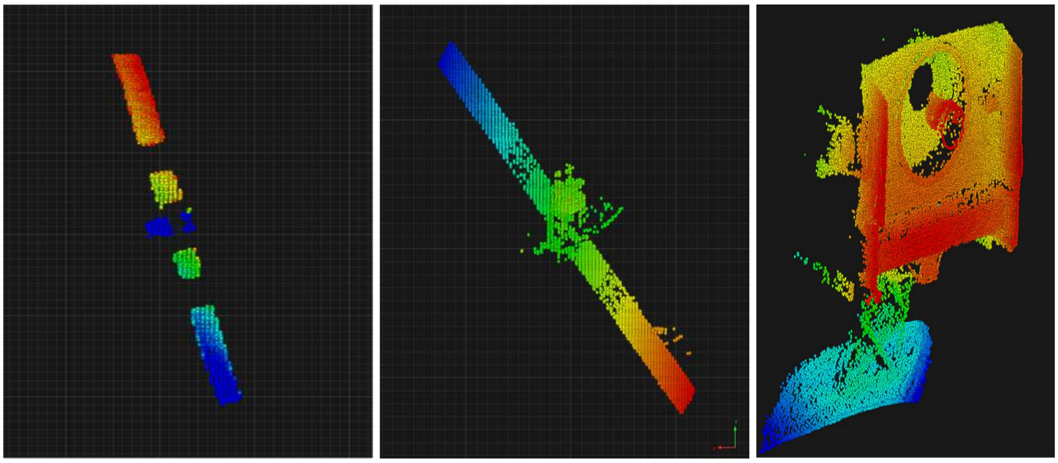
(a) Narrow-FOV image of target at 3.2 km distance

(b) Narrow-FOV image of target at 160 m distance

(c) Wide-FOV image of target at 4 m distance

Figure 3.18: Intelsat 10-02 GEO target imaged by MEV-2 with its ECAM-IR3A narrow and wide FOV LWIR cameras[44]

As in this research, the MEV also used a scanning LiDAR for remote sensing of its target, however the target was known and cooperative with a 3D model provided to the chaser beforehand as a reference with which to match points. The Jena-Optronik RVS3000-3D (pictured earlier in this document as Figure 1.5) was used for this purpose and was able to track the target at a distance in excess of 2km.[44] Figure 3.19 below shows data collected during the mission, and though it is not explicitly stated, the scanning pattern shows neatly ordered lines consistent with line-scanning technology. This is a key difference to the experiment as the Mid-70, as discussed in Section 2.2.1, used the more complicated and non-uniform non-repetitive ”flower petal” scanning pattern. As in the experiment, point density from the MEV data increases as distance to target decreases, and sensor noise and false returns occur in the raw data. Another aspect of note in this data is the gaps in points recorded mid-length in the solar panels. The author of the source document does not give an explanation for this omission, but it is consistent with the experience of this research’s experiments and is instantly recognizable.



(a) Scan at 2000m distance

(b) Scan at 220m distance

(c) Scan at 7m distance

Figure 3.19: Intelsat 10-02 GEO target scanned by MEV-2 at various distances[44]. The scanning pattern shown seems consistent with line-scanning.

This Section further demonstrates the relevance of the experiment developed by referencing real-world mission data from similar sensors, and allows for a more realistic postulation of a potential use case.

Theorized Use Case

The IR sensor worked well sensing up to 5 meters distance in the experiments due to laboratory size constraints, however it has been shown in the MEV missions that a LWIR camera with the same resolution is able to detect and track GEO communications satellites at a range in excess of 10km, and fully resolve the target at 3km. At these distances, the IR sensor's individual pixels would encompass more of the target's area than tested in the experiment. For the Color-ICP algorithm, this would mean that the color of the target's details would not be able to aid as much in the estimation of the transformation. For the Feature Matching algorithm, long distances would render the target as an nondescript blur, with features becoming more and more difficult to discern at greater distances. However even this blur has utility for object detection in the beginning of the pose estimation process.

It was noted early on that the Livox LiDAR does not preform well in distances closer than 2 meters. Conversely, the FLIR sensor seems to only improve the closer the target gets, though there is a limit based on the size of the target and the FLIR's FOV.

Occlusion in a space setting would be positive for the Color-ICP algorithm, and the harsh background IR source as positive for both algorithms through the use of CV and image processing techniques to create a more defined target separated from its background.

It is for these reasons that the following use case in an ADR mission targeting an unknown uncooperative target is postulated:

10km-2km: IR camera detects and tracks the blurry object, beginning to resolve strong features such as corners as distance decreases.

2km-300m: LiDAR begins to scan the target while the IR continues to image the target. At this point the fused Color-ICP algorithm can be used with the IR aiding in registration by providing large surface area pixel color values.

300m-100m: Begin gathering more accurate data as IR details become clearer and points are more densely populated. Color-ICP continues while the Feature Matching algorithm starts to provide accurate feature details.

100m-2m: Details are sharp and points are dense in this range allowing both sensors to retrieve the most accurate data. Both algorithm are working in unison.

<2m: Very close range Feature Matching with scale provided by LiDAR data time-history. Feature Matching with IR camera only, with the LiDAR previously providing transformations from pixels to world coordinate values without the need for new LiDAR points.

An advantage of this combination is that it works in a wide range of lighting scenarios, which cannot be said about visual camera methods. It can be used during occlusion, direct sunlight and in cases when the sun, Earth or moon are in the view-frame.

4 Conclusion

In this final main chapter of the research, the research questions postulated in Section 1.1.2 will be answered in Section 4.1 with the knowledge and experience gained during the duration of this work. After this, some resulting conclusions and reflections will be discussed in Section 4.2 before a few final recommendations for further research will be outlined in Section 4.3.

4.1. Research Questions Revisited

With the research nearing its end, the research questions formulated at the end of the preceding Literature Study phase can be reexamined and answered in a more succinct way as a result of the previous Chapters in this document. Sub-questions will reference the particular Section of relevance within this document to ease navigation.

RQ-1 What algorithm best suits the fusion and pose estimation goals?

- For the fusion of LiDAR and IR data, the ICP plane-to-plane with color retrieved accurate pose estimations consistently throughout each experiment. For the use case in which visual cameras would not be able to operate, the Color-ICP algorithm performs even better than in the direct lighting case, as the colors remain more consistent throughout the frames.
- The Feature Matching algorithm has some very promising aspects, but its use is limited to medium to close range to the target based on sensor resolution. There is a very-close range scenario in which the Feature Matching algorithm would work when the Color-ICP would not.

a. At what point should the data fusion take place?

The fusion should take place after some initial preprocessing of both the point clouds and IR images, followed by feature-based fine-tuning. IR images must first be undistorted, and in the case of the FLIR camera, cropped to remove the temperature scale and FLIR watermark. The point cloud was segmented beforehand, however this could also take place after the fusion. After this initial step, points and pixels can be related by detecting features in both datasets and estimating a transformation relating the two. Further fine tuning using Procrustes analysis leads to a more accurate relation, and therefore, fusion. It is with this final transformation that the datasets are fused. The fusion of raw data would lead to inaccuracies due to image distortion, and the feature alignment helps to insure that important reference points are properly aligned. As the ADR environment is spacious, there is no need to first identify or categorize objects in the scene before the fusion can take place. Section 1.2.4 introduced various fusion architectures while Section 2.1.1 detailed how this research question was answered.

b. How can the fusion be calibrated?

The MATLAB LiDAR-Camera calibration app is highly inaccurate and erratic, as are the fusion functions it provides. Instead, calibration was done by matching corner pairs in point clouds and images. Procrustes analysis and a geometric transform estimation provided the fine-tuned transformations to relate the image pixels to LiDAR points, and LiDAR points to image pixels. This process is described in Section 2.1.1.

c. What computer vision tasks can aid the process?

Histogram equalization and pixel value cropping were essential for detecting the target in harsh background IR lighting conditions. Anisotropic filtering and bilateral filtering showed promise in reducing vertical stiration in occlusion scenarios, but the effect on pose estimation was minimal. The most important CV tools for the research found use as an integral part of the second algorithm developed, which was feature detection and matching. Namely BRISK parameters were observed to be the most reliable for rotation. Section 2.1.4 describes the development of the Feature Matching algorithm and includes discussions on all of these CV tools.

d. How will the pose be estimated?

Pose was estimated by use of 3D transformation matrices. These matrices describe the initial orientation and position of the target in relation to the chaser, as well as its change in position and orientation over time. A firm understanding of the relationship between the world coordinates and target intrinsic coordinates is essential for estimating accurate parameters. The transformation found can be easily converted between coordinate systems to describe target pose as seen from the chaser and target intrinsic pose, hence relative pose. This method of observation over time retrieves the full 12 DOF relative pose. Essential to the undertaking of this research, the theory behind this pose representation was given in Section 1.2.5 while Sections 2.1.3 and 2.1.4 provided the transformations as they apply to the Color-ICP and Feature Matching algorithms respectively.

RQ-2 How does the method developed perform?

- The Color-ICP algorithm performed very well through a wide range of scenarios. The most accurate Color-ICP estimations were done in occlusion and harsh lighting conditions due to the consistent color of the target between frames. The Feature Matching algorithm was less accurate, though the accuracy was improved with the assistance of image processing techniques. For both algorithms, target intrinsic rotation led to less accurate results.

a. How accurate is the pose estimation?

Series K was the most difficult scenario for both algorithms. This involved both intrinsic lateral translation and rotation at a greater distance and included target observation angles in which the point clouds were very messy. The Color-ICP algorithm in general was very accurate, with its most accurate estimation being medium range targets and generally preferring large planar surfaces such as that of the Boxsat. The Feature Matching algorithm saw varying success in terms of accuracy. Its highest accuracy was in direct lighting scenarios with rich features and head-on translation-only motion present, though its accuracy was improved in the harsh lighting scenario with the assistance of image processing techniques. Sections 3.1 and 3.2 display the results for each of the two algorithms while Section 3.3.2 discusses their accuracy.

b. How can an experiment be designed to test the algorithm?

An experiment was designed with two target objects to be observed, Delfi-n3xt and Boxsat. These range in size and geometry and both are able to rotate by a precise amount, with the Boxsat also capable of multi-axis rotation. A calibration target, a chessboard, can be used for the calibration of the IR sensors intrinsic parameters as well as the extrinsic parameters between the IR and LiDAR sensors. The targets were translated a precise amount at varying distances with the assistance of the Robotnik rover. Varying scenarios were designed to capture instances of motion including rotation, translation, and a combination of the two. Lighting was provided by a powerful spotlight providing scenarios of direct sunlight on the target, and it was also used in combination with a terrace heater in the scenario of a strong IR source present in the field of view. The targets also preformed a range of motion in darkness at ambient temperature, the so-called occlusion scenario. The design, development and reasoning for this experiment setup can be found in Section 2.2.

c. What are the limitations of the LiDAR and thermal infrared sensors?

The LiDAR sensor can be noisy, especially at very close distances as well as instances of unfavorable target observation angles. As a scanning LiDAR was used, the experiment had to be designed around stop-motion movement, allowing the target to stay still while being scanned. This particular LiDAR also has a unique scanning pattern that provides a higher point density on a target at the center of its FOV, disadvantaging scenarios in which the target is on either edge of the FOV. The FLIR camera experienced striration in instances of occlusion due to the small range of temperatures present in the scene. Harsh IR lighting conditions were challenging for the sensor, though this was overcome with image processing. The FLIR also has relatively low resolution compared to a visual-spectrum camera, therefore it has a range limitation in which features will become less than the size of a pixel, however the MEV missions show that this resolution is more than acceptable for in-orbit use at kilometers distance. The Robotnik robot proved to be very useful, but its unique wheels were susceptible to unwanted translations affecting overall performance. A discussion on the inherent characteristics and limitations of the sensors can be found in Section 2.2.1, with their limitations in conjunction with the algorithms discussed in Section 3.3.4.

d. What scenario is this method best suited for?

The sensors and algorithms are best used in a scenario in which they can be used together. The advantage is their ability to see in complete darkness, so it would be invariant to periodic occlusion. In this use case, the IR camera would detect the target at 10km distance and begin tracking its motion. At ranges between 2km and 300m, the LiDAR begins to scan the target performs initial pose estimations with the Color-ICP algorithm aided by the color provided by the IR camera's large surface area pixels. The range between 300m and 100m would see both the algorithms being used as details become sharper and points denser, with the Color-ICP derived transformations being used as a starting point for the Feature Matching algorithm. As the target closes within 100m, both algorithms supplement each other and achieve their most accurate estimations. Closer than 2m, the LiDAR would no longer be reliable, and would necessitate an IR-only pose estimation in which target scale would have already been determined during the previous phases. This use case is formulated in greater detail at the end of Section 3.3.4.

e. What are the advantages and disadvantages of the method?

The Color-ICP algorithm is very robust against all lighting conditions and is very accurate due to the denoising and voxelization steps implemented to negate LiDAR sensor errors. However very small objects (such as the Delfi-n3Xt antennae) are not detected in this method. Fusion assists in estimation during occlusion and harsh lighting conditions, but can be a detriment to direct lighting scenarios involving target temperature instability. The Feature Matching algorithm can be assisted by a wide range of CV and image processing techniques and is highly tunable, however this tunability can be a double-edged sword. The user can also specifically choose what feature detection methods to use based on the specific use case. This algorithm requires as many imPts as possible which means that the LiDAR data cannot afford to be voxalized to negate sensor error. Overall the Feature Matching algorithm is less accurate compared to the Color-ICP algorithm and could benefit greatly from automated tuning. Section 3.3 discusses these aspects in great detail and is the result of the research presented herein in general.

4.2. Conclusions Summarized

In general, the Color-ICP algorithm was extremely accurate, more so than the Feature Matching algorithm. However the fusion in the Color-ICP algorithm served as a detriment to the estimation in scenarios of target rotation in direct lighting, as areas would heat up or cool down between frames, making the inclusion of color detrimental. As it was discussed, it can be reasoned that visual spectrum images would suffer the same fate due to its sensitivity to lighting in the L*a*b* color space. However the Color-ICP algorithm excelled in other cases, and some inaccuracy can be put down to experimental error due to inaccuracies in the experimental setup.

The Feature Matching algorithm makes much more use of the fusion and relies heavily on the performance of the IR sensor. It requires as many LiDAR points as possible, ideally covering each pixel, which would make the downsampling of the point cloud detrimental to imPt-pixel matching. It is highly tunable, which can also be a detriment as it is very sensitive to multiple interlinked detection and matching parameters, however this also allows for tuning in very specific use cases and makes it highly versatile for a wide range of targets, including non-rigid targets which the Color-ICP algorithm is not able to estimate. In general, the Feature Matching algorithm shows a lot of promise and deserves to be researched further for this application and others.

This sensor pairing is able to operate in total darkness as well as in direct sunlight and with background IR interference. It is robust to objects of no interest in the scene and has been shown to be a promising option for unknown uncooperative target pose estimation. The methods developed herein serve as an important stepping stone for further research. The pairing of LiDAR and LWIR sensors has too often been overlooked in pose estimation research, which this body of work proves to be a grave misjudgement. The advancement of this study has a clear path, with experimental and algorithmic improvement recommendations made clear throughout this document.

The methods developed herein show promise in applications with unknown uncooperative targets and should be considered as one potential option for implementation. As it has previously been postulated in this research, the problem of space debris requires a diverse set of solutions. No one single methodology will be appropriate for all targets. It is in this vein that this research can be applied, as one step towards one of many interesting ways to approach orbital debris.

4.3. Recommendations

From the results, conclusions and overall experience during this research, the following points are recommended for further study and consideration:

- Both the Color-ICP and the Feature Matching algorithms should be tested in a much larger environment.

  Great data and results were retrieved from the experiments presented in this document. However one burning question remains: at what distances do each algorithms fail with these sensors? Of course the minimum distance is known due to the Livox's very close range inaccuracies and the IR sensor's field of view, but the other extreme would be interesting to know. Due to laboratory space constraints, this could not be explored in this research, though the experience of the MEV missions made it possible to briefly explore and discuss with real mission data.

- TU Delft Aerospace faculty should consider creating an open-source RPO pose estimation dataset.

  Open-source tools and information were incredibly important resources for this research. Open-source datasets are used widely in academia and for general interest hobbyists, such as the renowned KITTI dataset for LiDAR-Camera fusion in autonomous cars[15], and help to advance the field without having to collect new data individually. A TU Delft RPO dataset would be a great resource to advance the field and to spread the name and reputation of the university.

- The Feature Matching algorithm would benefit from optimized and automated tuning parameters.

  The tuning of the feature detection and feature matching parameters is easily the most time-intensive process of the two algorithms developed. A real space-borne application of this research would need to be autonomous and have a method of tuning these parameters to detect strong features and reject bad matches. The development of autonomous tuning optimization is *highly* recommended to continue this research.

- A rail-based target translation system would be a more accurate experimental setup.

  The Robotnik rover allowed for great mobility, however a rail-based translation system would allow for more precision in a single axis of motion.

- Consider a custom IR stereovision, or IR-Visual stereovision system.

  This would be a very interesting prospect, as stereovision allows for depth estimation with only 2D images and a IR-Visual combination could provide higher image resolutions with the possibility of IR pixel interpolation based on the higher resolution visual image. It is strongly limited by its baseline distance, though, and would only be suitable for close range use.

- Consider developing a new custom Generalized Color-ICP algorithm in an other color space that is more robust to changes in light incidence.

  As it was discussed in the L*a*b* subsection of Section 3.3.1, the Color-ICP algorithm was affected by changes in light incidence corresponding to the L* (lightness) component in this color space, which is designed to be perceptually uniform. For the application to an orbital target-chaser scenario, it would be worthwhile to see how color registration can be made to be more robust to the changes in color expected of a target point in time-variant illumination.

References

[1] European Space Agency. *Envisat*. URL: https://www.esa.int/Enabling_Support/Operations/Envisat.

[2] Pablo Fernández Alcantarilla, Adrien Bartoli, and Andrew J Davison. "KAZE features". In: *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VI 12*. Springer. 2012, pp. 214–227.

[3] Brian J Anderson and Donald G Mitchell. "The Space Environment". In: *Fundamentals of Space Systems*. Oxford University Press, June 2005. ISBN: 9780195162059. DOI: 10.1093/oso/9780195162059.003.0002. eprint: https://academic.oup.com/book/0/chapter/422621207/chapter-pdf/52557355/isbn-9780195162059-book-part-2.pdf. URL: https://doi.org/10.1093/oso/9780195162059.003.0002.

[4] Erik Blasch. *Handbook of multisensor data fusion: theory and practice*. Feb. 2017. ISBN: 9781315219486. DOI: 10.1201/9781420053098.

[5] Robert W. Boyd and Robert C. Hilborn. "Radiometry and the Detection of Optical Radiation". In: *American Journal of Physics* 52.7 (July 1984), pp. 106–107. ISSN: 0002-9505. DOI: 10.1119/1.13578. eprint: https://pubs.aip.org/aapt/ajp/article-pdf/52/7/668/11597983/668_1_online.pdf. URL: https://doi.org/10.1119/1.13578.

[6] Samarth Brahmbhatt. *Practical OpenCV*. Apress, Nov. 2013. ISBN: 978-1-4302-6079-0.

[7] Ji Dong Choi and Min Young Kim. "A sensor fusion system with thermal infrared camera and LiDAR for autonomous vehicles and deep learning based object detection". In: *ICT Express* (2022). ISSN: 24059595. DOI: 10.1016/j.icte.2021.12.016.

[8] *Compact Space Camera MCAM*. Feb. 2022. URL: https://satsearch.co/products/mcse-compact-space-camera-mcam.

[9] Nikolaus Correll et al. *Introduction to Autonomous Robots: Mechanisms, Sensors, Actuators, and Algorithms*. 1st. MIT Press, Cambridge, MA, 2022.

[10] F. Cremer et al. "A comparison of decision-level sensor-fusion methods for anti-personnel landmine detection". In: *Information Fusion* 2.3 (2001), pp. 187–208. ISSN: 1566-2535. DOI: https://doi.org/10.1016/S1566-2535(01)00034-3. URL: https://www.sciencedirect.com/science/article/pii/S1566253501000343.

[11] James Dawson. "Space debris characterization using thermal imaging systems". In: (Sept. 2010), p. 10.

[12] Franka Emika. *PANDA - DATASHEET*. 2019. URL: https://www.wiredworkers.io/wp-content/uploads/2019/12/Panda_FrankaEmika_ENG.pdf.

[13] eoPortal. *Delfi-n3Xt (Delfi Triple CubeSat NeXt)*. 2012. URL: https://www.eoportal.org/satellite-missions/delfi-n3xt#mission-status.

[14] ESA. *Sending a Satellite Safely to Sleep*. URL: https://www.esa.int/Space_Safety/Clean_Space/Sending_a_satellite_safely_to_sleep.

[15] Andreas Geiger, Philip Lenz, and Raquel Urtasun. "Are we ready for autonomous driving? The KITTI vision benchmark suite". In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*. 2012, pp. 3354–3361. DOI: 10.1109/CVPR.2012.6248074.

[16] David G. Gilmore. *Spacecraft Thermal Control Handbook, Volume 1 - Fundamental Technologies (2nd Edition)*. American Institute of Aeronautics and Astronautics/Aerospace Press (AIAA), 2002. ISBN: 978-1-884989-11-7. URL: https://app.knovel.com/hotlink/toc/id:kpSTCHVFT2/spacecraft-thermal-control/spacecraft-thermal-control.

[17] Colin Goodall. "Procrustes Methods in the Statistical Analysis of Shape". In: *Journal of the Royal Statistical Society. Series B (Methodological)* 53.2 (1991), pp. 285–339. ISSN: 00359246. URL: http://www.jstor.org/stable/2345744 (visited on 11/09/2023).

[18] Lutz Groll and Andreas Kapp. "Effect of Fast Motion on Range Images Acquired by Lidar Scanners for Automotive Applications". In: *IEEE Transactions on Signal Processing* 55.6 (2007), pp. 2945–2953. DOI: 10.1109/TSP.2007.893945.

[19] Northrop Grumman. *Multimedia*. URL: https://news.northropgrumman.com/multimedia/photo/Space/space-systems/spacelogistics-mev-2.

[20] Jan Hartmann, Jan Helge Klüssendorff, and Erik Maehle. "A comparison of feature descriptors for visual SLAM". In: *2013 European Conference on Mobile Robots*. 2013, pp. 56–61. DOI: 10.1109/ECMR.2013.6698820.

[21] IBM. *What is Computer Vision?* URL: https://www.ibm.com/topics/computer-vision.

[22] Information Science Institute. *Rendezvous and Proximity Operations (RPO)*. URL: https://www.isi.edu/centers-serc/research/rendezvous-and-proximity-operations-rpo/.

[23] Kaitlyn Johnson. *Key Governance Issues in Space - Rendezvous and Proximity Operations*. Tech. rep. Center for Strategic and International Studies (CSIS), 2020.

[24] G M Koretsky, J F Nicoll, and M S Taylor. *A Tutorial on Electro-Optical/Infrared (EO/IR) Theory and Systems*. 2013.

[25] Michael Korn, Martin Holzkothen, and Josef Pauli. "Color Supported Generalized-ICP". In: *Proceedings of the 9th International Conference on Computer Vision Theory and Applications (VISIGRAPP 2014) - Volume 3: VISAPP*. INSTICC. SciTePress, 2014, pp. 592–599. ISBN: 978-989-758-009-3. DOI: 10.5220/0004692805920599.

[26] John V. Lambert, Thomas J. Osteen, and Butch Kraszewski. "Determination of debris albedo from visible and infrared brightnesses". In: *Space Debris Detection and Mitigation*. Ed. by Firooz A. Allahdadi. Vol. 1951. International Society for Optics and Photonics. SPIE, 1993, pp. 32–36. DOI: 10.1117/12.156557. URL: https://doi.org/10.1117/12.156557.

[27] G. D. Lewis et al. "In-band low-power laser dazzle and pixel damage of an uncooled LWIR thermal imager". In: *Technologies for Optical Countermeasures XV*. Ed. by David H. Titterton, Robert J. Grasso, and Mark A. Richardson. Vol. 10797. International Society for Optics and Photonics. SPIE, 2018, 107970F. DOI: 10.1117/12.2325261. URL: https://doi.org/10.1117/12.2325261.

[28] Timothy Luu, Stephane Ruel, and Martin Labrie. "TriDAR Test Results Onboard Final Shuttle Mission, Applications for Future of Non-Cooperative Autonomous Rendezvous and Docking". In: Sept. 2012.

[29] MathWorks. *cameraParameters*. URL: https://www.mathworks.com/help/vision/ref/cameraparameters.html.

[30] MathWorks. *detectBRISKFeatures*. URL: https://www.mathworks.com/help/vision/ref/detectbriskfeatures.html.

[31] MathWorks. *estimateCameraParameters*. URL: https://www.mathworks.com/help/vision/ref/estimatecameraparameters.html#responsive_offcanvas.

[32] MathWorks. *extractFeatures*. URL: https://www.mathworks.com/help/vision/ref/extractfeatures.html?s_tid=doc_ta.

[33] MathWorks. *Local Feature Detection and Extraction*. URL: https://www.mathworks.com/help/vision/ug/local-feature-detection-and-extraction.html.

[34] MathWorks. *matchFeaturesInRadius*. URL: https://www.mathworks.com/help/vision/ref/matchfeaturesinradius.html.

[35] MathWorks. *pcdownsample*. URL: https://www.mathworks.com/help/vision/ref/pcdownsample.html?s_tid=doc_ta.

[36] MathWorks. *pcregistericp*. URL: https://www.mathworks.com/help/vision/ref/pcregistericp.html?searchHighlight=register%20icpands_tid=srchtitle_support_results_2_register%20icp#bupmua2-1_sep_bupmua2-4.

[37] MathWorks. *rgb2gray*. URL: https://www.mathworks.com/help/matlab/ref/rgb2gray.html.

[38] CR McBryde. *SPACECRAFT VISUAL NAVIGATION USING APPEARANCE MATCHING AND MULTI-SPECTRAL SENSOR FUSION A Dissertation Presented to The Academic Faculty*. 2018.

[39] Dibyendu Mukherjee, QM Jonathan Wu, and Guanghui Wang. "A comparative experimental study of image feature detectors and descriptors". In: *Machine Vision and Applications* 26 (2015), pp. 443–466.

[40] NASA Space Flight. *Atlantis gifts TriDAR to Orbital's Cygnus for ISS rendezvous and docking*. URL: https://www.nasaspaceflight.com/2011/07/atlantis-gifts-tridar-orbitals-cygnus-iss-rendezvous-docking/.

[41] National Air and Space Museum. *Spinning Out of Control: Gemini VIII's Near-Disaster*. URL: https://airandspace.si.edu/stories/editorial/spinning-out-control-gemini-viii.

[42] Roberto Opromolla et al. *A review of cooperative and uncooperative spacecraft pose determination techniques for close-proximity operations*. Aug. 2017. DOI: 10.1016/j.paerosci.2017.07.001.

[43] Orion17. *Flowers in UV | (UV Photography)*. Sept. 2015. URL: https://www.youtube.com/watch?v=yVDi4GVdj0w.

[44] Matt Pyrak and Joseph Anderson. "Performance of Northrop Grumman's Mission Extension Vehicle (MEV) RPO imagers at GEO". In: June 2022, p. 28. DOI: 10.1117/12.2631524.

[45] Nola Redd. "Bringing satellites back from the dead: Mission extension vehicles give defunct spacecraft a new lease on life - [News]". In: *IEEE Spectrum* 57 (Aug. 2020), pp. 6–7. DOI: 10.1109/MSPEC.2020.9150540.

[46] *RVS 3000*. URL: https://satsearch.co/products/jena-optronik-rvs-3000.

[47] Simone Servadio. "High Order Filters for Relative Pose Estimation of an Uncooperative Target". PhD thesis. Oct. 2017.

[48] Siliconera. *Sega Explains How Dr. Robotnik Came To Be Called Eggman*. 2016. URL: https://www.siliconera.com/sega-explains-dr-robotnik-came-called-eggman/.

[49] *Stereo bench*. Feb. 2022. URL: https://satsearch.co/products/mcse-stereo-bench.

[50] Yu Su et al. "Accurate Pose Tracking for Uncooperative Targets via Data Fusion of Laser Scanner and Optical Camera". In: *The Journal of the Astronautical Sciences* 69 (Oct. 2022). DOI: 10.1007/s40295-022-00344-w.

[51] Malin Space Science Systems. *ECAM-IR3A*. 2022.

[52] Richard Szeliski. *Computer vision: algorithms and applications*. Springer Nature, 2022, p. 367.

[53] Livox Tech. *Livox Mid-70 User Manual*. v1.2. Feb. 2021.

[54] TELEDYNE FLIR. *FLIR A655sc*. URL: https://www.flir.eu/products/a655sc/?model=55001-0301andvertical=rd+scienceandsegment=solutions.

[55] *The birth of the Delfi-PQ minisatellite*. URL: https://www.tudelft.nl/en/stories/articles/the-birth-of-the-delfi-pq-mini-satellite.

[56] The European Space Agency. *The Kessler Effect and how to stop it*. URL: https://www.esa.int/Enabling_Support/Space_Engineering_Technology/The_Kessler_Effect_and_how_to_stop_it.

[57] Peter Todorov. "Multi-camera Calibration". PhD thesis. 2019, pp. 13–16.

[58] Marco Alexander Treiber. *Optimization for computer vision*. Springer, 2013.

[59] V.A. Weidlich. "Thermal Infrared Face Recognition". In: *Cureus* (Mar. 2021).

[60] Özgün Yılmaz et al. "Thermal analysis of space debris for infrared-based active debris removal". In: *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering* 233 (Nov. 2017), p. 095441001774091. DOI: 10.1177/0954410017740917.

[61] Cassie Zhang, Yuhan Zhang, and Zixin Huang. "CS231A Project Final Report: Comparing Multiple Methods for Camera Calibration". In: *Stanford* (2022).

[62] Zhengyou Zhang. *A Flexible New Technique for Camera Calibration*. Tech. rep. Microsoft Corperation, 1998.

A
Source Code

Though the initial research and forays into pose estimation were done in C++, Python and MATLAB, the final form of both the ICP and Feature Matching algorithms were solely coded in MATLAB. The development of both algorithms is detailed in Sections 2.1.3 & 2.1.4 respectively. The full scrips for both algorithms, as well as the calibration algorithm, is located in the following GitHub repository: `https://github.com/ConorCreagh/Pose_ICP-FeatureMatching`

MATLAB 2023b is the minimum required version to run this code (with the color-ICP metric being introduced in this version), and the following packages are required:

- LiDAR Toolbox
- Computer Vision Toolbox
- Image Processing Toolbox
- Statistics and Machine Learning Toolbox

B
V&V Experiments

Experiment	Target	Lighting	Target Intrinsic Motion	Start-End X Distance (m)	Start-End Y Distance (m)	Number of Frames Recorded
Series H	Delfi-n3Xt	Direct	Translation far-near	5.0-3.0	0-0	21
Series I	Delfi-n3Xt	Direct	Translation far-near & rotation	5.0-3.0	0-0	21
Series J	Delfi-n3Xt	Direct	Translation left-right	5.0-5.0	-0.8-0.8	17
Series K	Delfi-n3Xt	Direct	Translation left-right & rotation	5.0-5.0	-0.8-0.8	17
Series L	Delfi-n3Xt	Direct	Closer translation left-right & rotation	3.0-3.0	0.5-(-0.6)	12
Series M	Delfi-n3Xt	Direct	Closer translation left-right	3.0-3.0	0.5-(-0.6)	12
Series N	Delfi-n3Xt	Occlusion	Translation far-near & rotation	5.0-2.5	0-0	26
Series O	Delfi-n3Xt	Occlusion	Rotation	2.5-2.5	0-0	18
Series P	Delfi-n3Xt	Occlusion	Closer translation left-right & rotation	3.0-3.0	0.5-(-0.6)	12
Boxsat A	Boxsat	Direct	Rotation (Longitudinal Axis)	4.5-4.5	0-0	18
Boxsat B	Boxsat	Direct	Rotation (Off-axis)	4.5-4.5	0-0	12
Boxsat C	Boxsat	Occlusion	Rotation (Longitudinal Axis)	4.5-4.5	0-0	18
Boxsat D	Boxsat	Spotlight in-scene @ 50%	Translation left-right	4.5-4.5	-0.3-0.5	9
Boxsat E	Boxsat	Spotlight in-scene @ 100%	Translation left-right	4.5-4.5	-0.3-0.5	9
Boxsat F	Boxsat	IR Heater in-scene @100%	Translation left-right	4.5-4.5	-0.3-0.5	9