



Lehrstuhl für Raumfahrttechnik
Prof. Prof. h.c. Dr. Dr. h.c.
Ulrich Walter



Technische Universität
München

Bachelorthesis
**Execution and Evaluation of an Experimental Study for the
Quantification of Influencing Factors on the Quality of
Optical Pose Estimation in Earth's Orbit**
RT-BA 2018/12

Autor:
Niklas Hab



Supervisor:

Dipl. Ing. (Univ.) Martin Dziura
Lehrstuhl für Raumfahrttechnik / Institute of Astronautics
Technische Universität München



Acknowledgments

At this point I would like to thank everyone who supported me writing this thesis.

A special thank you goes to my supervisor Dipl. Ing. Martin Dziura, who always took his time for any questions and gave me very helpful advice when needed, despite having a very busy schedule.

I also want to express my gratitude to my roommate Thomas Wegele, who also always had an open ear for problem discussions regarding the study. Furthermore, I am very grateful for Ina Conrad and Patrick Klotz for enthusiastically supporting me in the correction of this work.



Zusammenfassung

Autonome Nahbereichsoperationen im Weltall wie On-Orbit Servicing oder aktive Weltraummüllbeseitigung benötigen fortgeschrittene Regelungsalgorithmen auf Seiten des Verfolgersatelliten. Im Fall eines unkooperativen und unbekanntes Zielobjekts ist es dabei hilfreich, sowohl dessen relative Lage als auch Position mit einer gewissen Frequenz und ohne Hilfe optischer Marker in Echtzeit zu bestimmen und letztendlich dreidimensional zu rekonstruieren.

In jüngster Zeit wurden bereits verschiedene Arten von Sensoren und Algorithmen für dieses Problem vorgeschlagen und getestet, unter anderem auch von dem Forschungsteam des Real-Time Attitude Control And On-Orbit Navigation Laboratory (RACOON-Lab) der Technischen Universität München. Dabei konnte in Erfahrung gebracht werden, dass neben den verwendeten Sensoren und Algorithmen auch vorherrschende Umgebungsbedingungen im Orbit wie der Einfallwinkel des Sonnenlichts relativ zum Kamerasystem eine entscheidende Rolle hinsichtlich der Rekonstruktionsgüte des Zielobjekts spielen.

Diese Arbeit beschäftigt sich mit der Auswahl und Quantifizierung dieser externen Einflussparameter. Dazu wurde am lehrstuhleigenen Hardware-in-the-Loop (HIL) Simulator eine umfangreiche Versuchsreihe durchgeführt. Der Ansatz beruht dabei auf einem visuell basierten System. Mit Hilfe in einer vorangegangenen Arbeit entwickelten Metrik konnten die dabei von zwei verschiedenen Tracking-Algorithmen generierten Trajektorien auf ihre Qualität bewertet werden.

Die Ergebnisse zeigen eine beachtliche Abhängigkeit der Lagebestimmung von den gegebenen Lichtverhältnissen, welche jedoch auch sehr Algorithmus-spezifisch ist. Die durch die experimentelle Studie erzeugte Datenbank liefert zudem eine Grundlage für weitere Untersuchungen, wodurch in Zukunft keine neuen Aufnahmen für eine Integration und einen Vergleich mit zusätzlichen Algorithmen generiert werden müssen.



Abstract

Autonomous proximity operations such as active space debris removal and On-Orbit Servicing missions require advanced control algorithms on the part of the tracking satellite. In the case of an uncooperative and unknown target, it is helpful to determine both its relative attitude and position with a certain frequency and without the aid of optical markers in real time and ultimately to reconstruct it three-dimensionally.

Recently, different types of sensors and algorithms have been proposed and tested for this problem, among others by the research team of the Real-Time Attitude Control and On-Orbit Navigation Laboratory (RACOON-Lab) of the Technical University of Munich (TUM). In addition to the sensors and algorithms used, it was found that prevailing environmental conditions such as the angle of incidence of the sunlight relative to the camera play a decisive role in the reconstruction quality of the target object.

This thesis deals with the selection and quantification of these external influencing factors. For this purpose, a detailed parameter study was conducted at the RACOON-hardware-in-the-loop (HIL) simulator. In doing so, yet, only inspection scenarios were considered. The quality of the trajectories generated by two different tracking algorithms could be evaluated with the help of a metric developed in a previous work.

The results show a considerable dependence of the pose determination quality on the given illumination conditions, which is, however, also conditional on the used algorithm. The during this work generated database can now also function as a basis for further studies, which will allow other algorithms being integrated and compared in the future without the effort of new data generation.



Table of Contents

1	INTRODUCTION	1
2	STATE OF THE ART	3
3	TASK DEFINITION	5
3.1	Scenario description	5
3.2	Objectives of the work	5
3.3	Methodology	6
3.4	Differentiation	6
4	INTRODUCTION IN USED ALGORITHMS AND METRIC	8
4.1	DIFODO	8
4.1.1	General information	8
4.1.2	Mathematical fundamentals	8
4.2	ZED Tracker	10
4.3	Used Metric	11
4.3.1	Main Struct	11
4.3.2	Mathematical fundamentals	13
4.3.3	Adaption	16
5	EXECUTION OF THE STUDY	17
5.1	RACOON-Lab introduction	17
5.1.1	General information	17
5.1.2	Satellite-Mockup	19
5.1.3	Cameras	20
5.1.4	Lightning conditions	22
5.1.5	Optitrack Reference System	23
5.2	Selection of parameters	23
5.2.1	Possible parameter variations	24
5.2.2	Main settings DIFODO	26
5.2.3	Main settings ZED Tracker	27
5.2.4	Software settings	29
5.2.5	Hardware settings	30
5.2.6	Image Pre-Processing	31
5.3	Design of Experiments	33
5.3.1	General Information	33
5.3.2	Adaption to the scenario	35
5.4	Main study	36
5.4.1	Recording software	36



5.4.2	Experiment setup and execution	37
6	EVALUATION	43
6.1	Angle of incidence of the sunlight	44
6.2	Inclination angle of the target satellite	49
6.3	Correlations between the parameters	53
6.4	Angular velocity	57
6.5	Comparison of used tracking algorithms	57
7	DISCUSSION	60
7.1	Interpretation of the results	60
7.2	Parameter impact on the pose estimation	61
7.3	Critical reflection of the study	62
8	CONCLUSION	64
A	REFERENCES	66
B	APPENDIX	69
B.1	Detailed parameter of the experiments	69
B.2	Results	74

List of Figures

Fig. 1—1: Logo: RACOON riding an on-orbit servicer	1
Fig. 2—1: Taxonomy of EO sensors for spacecraft applications.	3
Fig. 3—1: Coordinate systems of the RACOON-Lab.....	5
Fig. 4—1: Pin-hole camera model [22]	9
Fig. 4—2: By DIFODO estimated trajectory for a single fly-around of 360 deg.....	11
Fig. 4—3: drift angle δ ; a lead (left) or a delay (right) of the estimated trajectory (blue) referring to the real trajectory (red).....	13
Fig. 4—4: Bestfit-circle for a point cloud in 2D.....	14
Fig. 4—5: Bestfit-circle for a point cloud in 3D.....	14
Fig. 4—6: Required values for calculating δi	15
Fig. 5—1: Illustration of the RACCON-Lab concept.....	17
Fig. 5—2: 11 degrees of freedom of the target (right) and the chaser satellite (left) provided by the Hardware Facility	18
Fig. 5—3: LabVIEW GUI.....	19
Fig. 5—4: target satellite mockup	20
Fig. 5—5: chaser satellite	20
Fig. 5—6: ZED stereo camera from StereoLabs [33].....	21
Fig. 5—7: Sun and earth albedo simulator [17]	22
Fig. 5—8: Spectral distribution over wavelength of the sun simulator (blue) compared to ETR ASTM E-490 standard spectrum (red) [35].....	22
Fig. 5—9: Qualitative comparison of 3D reconstruction results using two different illumination angles, taken with the ZED.....	24
Fig. 5—10: confidence map with an applied value of 85 during a simulated fly- around from top left to bottom right	28
Fig. 5—11: Simple illustration of the applied boxfilter [4]	32
Fig. 5—12: comparison between the unfiltered and filtered depth map of the ZED as seen by the camera during the experiment and its resulting estimated trajectories calculated by the DIFODO algorithm	33
Fig. 5—13: range of experiments; the colored lines represent the detailed, isolated variation of parameters, whereas the blue points show the coherent analyzation	36
Fig. 5—14: variation of the targets inclination angle as seen by the ZED.....	39
Fig. 5—15: CATIA model of the HIL facility including a visualization of the variation of the lightning incidence for the isolated analysis.	40
Fig. 5—16: Defined “point zero” of the sun simulator unit.....	41
Fig. 6—1: Course of the five metric parameters during the variation of the incidence angle of the sun (angles higher than 90 deg excluded), displaying results of the DIFODO algorithm on the left and of the ZED Tracker on the right.....	46



Fig. 6—2: Development of the drift course with an increasing incidence angle of the sun (DIFODO)	47
Fig. 6—3: Deviations in the course of the Residuum (DIFODO).....	47
Fig. 6—4: Development of the drift course with an increasing incidence angle of the sun (ZED Tracker).....	48
Fig. 6—5: Comparison of estimated trajectories for different settings	48
Fig. 6—6: Course of the five metric parameters during the variation of the inclination angle of the target, displaying results of the DIFODO algorithm on the left and of the ZED Tracker on the right.....	51
Fig. 6—7: Development of the courses of drift and Residuum for an increasing inclination angle of the target (DIFODO)	52
Fig. 6—8: Development of the courses of drift and Residuum for an increasing inclination angle of the target (ZED Tracker).....	53
Fig. 6—9: Development of the maximum drift angle as well as of the Mean Residuum for a coherent variation of the parameters for DIFODO (left) and ZED Tracker (right)	54
Fig. 6—10: Progression of the Drift within the experimental range (DIFODO).....	55
Fig. 6—11: Progression of the drift for an increasing sun angle (ZED).....	56
Fig. 6—12: Intensity of the drift flares over the experimental range (DIF)	58
Fig. 6—13: Comparison of good results and their counterparts for each algorithm	59
Fig. 7—1: Comparison of the drift flare for the same setting: 15_0 (DIF).....	62



List of Tables

Tab. 4—1: Definition of the five metric parameters.....	12
Tab. 5—1: technical data of the on-board computer.....	20
Tab. 5—2: technical specifications of the ZED [6], [15]	21
Tab. 5—3: Overview of important variable parameters (algorithm setting excluded).....	25
Tab. 5—4: DIFODO parameters.....	26
Tab. 5—5: results in the pose estimation with different pyramid levels	27
Tab. 5—6: ZED Tracker parameters	27
Tab. 5—7: calibrated intrinsics of the ZED	30
Tab. 5—8: applied boxfilter values for the study.....	32
Tab. 5—9: numeric results on pose estimation for raw and filtered data	33
Tab. B.1—1: Incidence angle sun.....	69
Tab. B.1—2: Inclination angle target	71
Tab. B.1—3: Correlating Experiments	72
Tab. B.2—1: Incidence angle sun (DIFODO)	74
Tab. B.2—2: Inclination angle target (DIFODO)	75
Tab. B.2—3: Correlating experiments (DIFODO)	76
Tab. B.2—4: Incidence angle sun (ZED Tracker)	78
Tab. B.2—5: Inclination angle target (ZED Tracker).....	79
Tab. B.2—6: Correlating Experiments (Zed Tracker)	80



Abbreviations

API	Application Programming Interface
CCW	counter-clock-wise
CPU	Central Processing Unit
CS	Coordinate System
DoE	Design of Experiments
DOF	Degrees of Freedom
EO	Electro-Optical
EKF	extended Kalman Filter
GPU	Graphics Processing Unit
HIL	Hardware-in-the-loop
IEKF	iterated extended Kalman Filter
LEO	Low Earth Orbit
fps	frames per second
LIDAR	Light Detection and Ranging
LRT	Institute of Astronautics (ger.: Lehrstuhl für Raumfahrttechnik)
MSI	Multi-Layer Insulation
OOS	On-Orbit Servicing
OpenCV	Open Source Computer Vision
RACOON-Lab	Robotic Actuation and On-Orbit Navigation Laboratory
RAM	Random-Access Memory
RANSAC	random sample consensus
RGB-D	Red Green Blue-Depth
RMSE	Root Mean Square Error
RSM	Response Surface Design
SDK	Software Development Kit
SLAM	Simultaneous Localization and Mapping
TUM	Technical University of Munich
VO	Visual Odometry

1 Introduction

Proximity operations and docking between spacecraft are key maneuvers in spaceflight. So far, these missions have been mainly performed by crewed space vehicles and involved cooperative target spacecraft. In regard to the new generation of space robotics missions such as active space debris removal and On-Orbit servicing (OOS), the main challenge is to perform these operations autonomously for an uncooperative target. The capability to do so was considered by NASA a high-priority cross-cutting technology required for the decade 2011-2021 and beyond, since autonomous systems would be able to service the majority of space debris and end-of-life satellites using the same algorithms and methods. [1] Thereby, the developed technologies must be applicable to a wide array of target objects (defunct satellites, spent upper stages, fragmented debris objects) in a wide gamut of the current status (partially cooperative, rotating, tumbling).

Meanwhile, there are over 21000 known and catalogued orbiting objects, that are classified by the US Space Surveillance Network as uncooperative (as of Jan. 2018), which covers objects larger than about 5–10 cm in Low-Earth Orbit (LEO) and 30 cm to 1 m at geostationary altitudes (GEO). In addition to the ones catalogued, there are also a significant number of other orbiting objects too small to be tracked by ground stations whose geometric appearance is unknown. [2]

Amongst others, this challenge is dealt with at the Institute of Astronautics (ger.: Lehrstuhl für Raumfahrttechnik, LRT) of the Technical University of Munich (TUM), which established a research laboratory for Real-Time Attitude Control and On-Orbit Navigation (RACOON-Lab). The facility has an extensively equipped hardware-in-the-loop (HIL) simulator for rendezvous scenarios in earth's orbit, where different constellations between a target satellite and a tracking satellite, remotely controlled by a teleoperator, can be mapped and orbit-mechanically correctly modeled. [3]



Fig. 1—1: Logo: RACOON riding an on-orbit servicer

In missions of OOS, such as manipulation, relocation, active debris removal, or simply observation of the target objects, it is necessary to operate in close proximity and perform relative navigation maneuvers. Therefore, it is helpful to digitally capture and display the attitude and position (tracking) as well as the shape (mapping) of the target.

Currently, Light Detection and Ranging (LIDAR) sensors are mostly used in this research area, which is also called GNC (Guidance, Navigation, Control). As a cost-effective, compact and energy-efficient alternative, camera-based systems, such as the ZED, however, are gaining importance and are now to be examined for their usefulness for inspection scenarios in space.

In advance of this work, a modularly designed software framework was developed for the RACOON-Lab, which implements a flexible real-time capable toolchain to perform all necessary tasks for 3D reconstruction. In [4], first hardware tests results regarding the quality of reconstruction indicated a significant impact of different illumination conditions. At the same time, errors in the trajectory estimated by a tracking algorithm have been identified as a probable reason for a non-sufficient volumetric reconstruction. In order to determine the impact of those external parameters quantitatively, a metric based on the idea to evaluate the quality of an estimated fly-around trajectory by applying a circular fit function was developed. This thesis now deals with the quantization of those influences and is thereby structured as follows.

Chapter 2 provides an overview about the state of the art SLAM approaches for a pose estimation of an uncooperative and unknown target. Precise functionalities of corresponding algorithms are not further presented, however, the tracking algorithms used for this study are described in chapter 4. Besides that, the fundamental principle of said metric is explained, including its modules, which are the basis of the evaluation.

Chapter 5 provides all necessary steps for the execution of the study, by first introducing the hardware facility, which is mandatory for an understanding of the work. Thereafter, a detailed list of all possible influencing factors is given, also explaining the selection of the parameters used for the study as well as the settings for the remaining. Finally, the experimental scope and setup is presented. To ensure reproducibility, a clean documentation is provided.

Chapter 6 contains a complete and careful preparation of the generated data, before the results are being discussed in chapter 7. This does not only include a critical reflection on the test series, but also the deliberation of the systematic and statistic measurement errors.

Lastly, chapter 8 summarizes the whole work and gives an outlook for possible future projects on the way to a comprehensive solution for the correct position estimation and finally 3D reconstruction of an unknown object in orbit is provided.

2 State of the Art

OOS scenarios have been gaining importance in the last decade, particularly due to the ever-increasing problem of space debris. [5] In order to support such proximity operations, several vision-based techniques exist to determine the relative pose of an uncooperative orbiting object (also referred to as target) with respect to the spacecraft (also chaser/servicer).

When operating in close-proximity, the need for autonomy is motivated by the fact that communication delays or lack of coverage make relative navigation maneuvers based on ground control station commands unfeasible. [6] This is the main reason why pose determination is a complex task requiring innovative algorithmic and technological solutions. Specifically, some of the key factors driving the choice of pose determination algorithms and sensors are the mission scenario and the type of target in terms of cooperativeness. Thereby, uncooperative targets can be divided into two categories, depending on whether geometrical information about their shape and size is available, or is fully unknown.

For spacecraft pose determination, typically either passive monocular/stereo camera or active LIDAR (Light Detecting and Ranging) systems are used, which are also summarized under the term of an electro-optical (EO) sensor. An overview about the potential of EO sensors for spacecraft applications is shown in Fig. 2—1. [7]

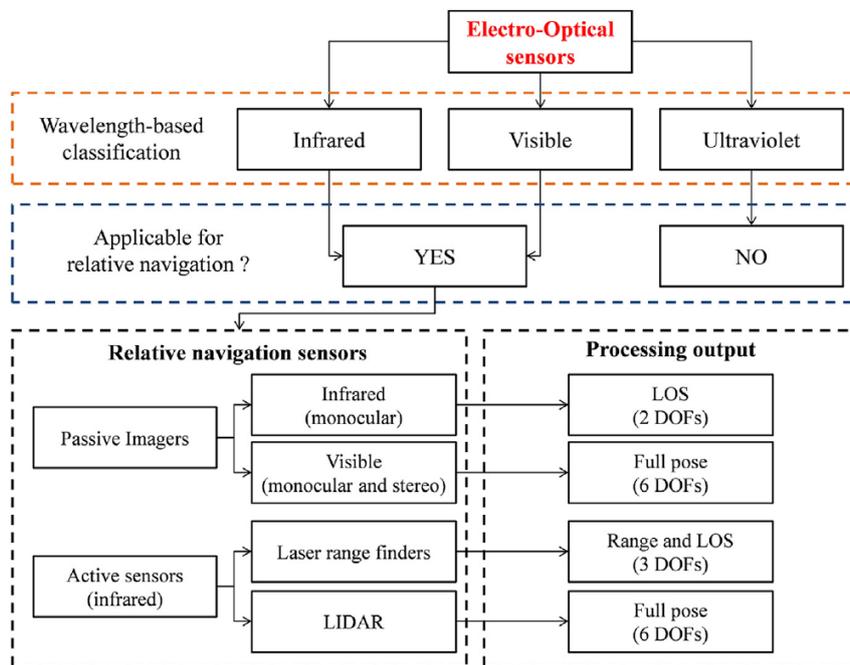


Fig. 2—1: Taxonomy of EO sensors for spacecraft applications.

In the case of an uncooperative as well as unknown target, besides determining its absolute pose, it is also necessary to build a model on board using the available measurements. This task can be accomplished by adapting the well-known concept of Simultaneous Localization and Mapping (SLAM) [8], inherited from the fast-growing robotic community. In most applications, the environment is assumed to be static. Nevertheless, observability issues can still be tackled if the non-static scene is composed of rigid objects (while the problem of performing SLAM within deformable

largescale environments is yet largely unexplored [9]), which makes the latter adoptable for space applications.

So far, several SLAM approaches exist, dealing with unknown, noncooperative space targets, while most recent ones are making use of either stereo or an RGB-D camera, which basically is a hybrid of a monocular camera and a passive LIDAR.

A feature-based SLAM algorithm relying on an extended Kalman Filter (EKF) filter is proposed in [10]. As this work focuses exclusively on the filtering part of the relative navigation architecture, a set of 3D features, detected and tracked by a stereovision system, is assumed to be given as an input to the EKF. [11, 12] combine this EKF-based SLAM filter with a random sample consensus (RANSAC)-based surface reconstruction algorithm to generate a model of the unknown target which can be used for a further refinement of the estimated pose. However, the real-time applicability of this technique is to be further analyzed. A solution to solve the SLAM problem for an unknown tumbling target by using a smoothing approach is proposed in [13], based on the work of [14]. Thereby, the algorithm exploits 3D measurements from a stereo camera and allows estimating the entire relative state, including linear and angular velocities of the object as well as its center of mass, principal axes and diagonal inertia matrix. On the other hand, [15] demonstrates a method relying on multiple iterated extended Kalman Filters (IEKFs) and a maximum a posteriori (MAP) inertia tensor identification algorithm. Another EKF-SLAM based approach for real-time relative state estimation on unknown spinning target using stereovision is presented in [16].

However, it can be stated that the problem of estimating the relative state while simultaneously building a model of an unknown target using spaceborne processing recourses still present open issues. For instance, despite the availability of several filtering/smoothing architectures providing various levels of performance, the capabilities and limitation of the distinct types of EO sensors suitable for relative navigation are not always considered in a consistent way. As well, to be able to achieve safe and reliable proximity operations in the given scenario, [7] also suggests to include realistic models for the target/chaser rotational and translational relative dynamics within the developed architecture, to analyze the stability of presented algorithms and sensors in regard to orbital environment influences.

Currently, this matter is amongst other things tackled in [4, 17]. With the help of a HIL simulator, the application of 3D object reconstruction in orbit can be tested, aiming to improve not only GNC algorithms in terms of situation awareness, efficiency and recourse consumption. First experimental results performing SLAM with the help of a stereo camera stated a crucial impact on the reconstruction quality due to different illumination conditions. Therefore, [17] suggests a quantization of all influence parameters regarding the quality 3D reconstruction and therefore pose estimation. This statement is also corroborated by [6].

3 Task definition

3.1 Scenario description

The general scenario for the application of the algorithms is the approach of a chaser satellite to an uncooperative as well as unknown target object in orbit, e.g. a disused satellite or a space debris part, during a servicing or deorbiting mission. Uncooperative means that the target is not controllable at any point of the mission whereas unknown refers to the shape of the object. The chaser is located at such a distance and position relative to the target that it is completely and extensively within the field of view of a passive camera system. Chaser and target move relatively to each other and relatively to the background. The target rotates uniformly around an axis.

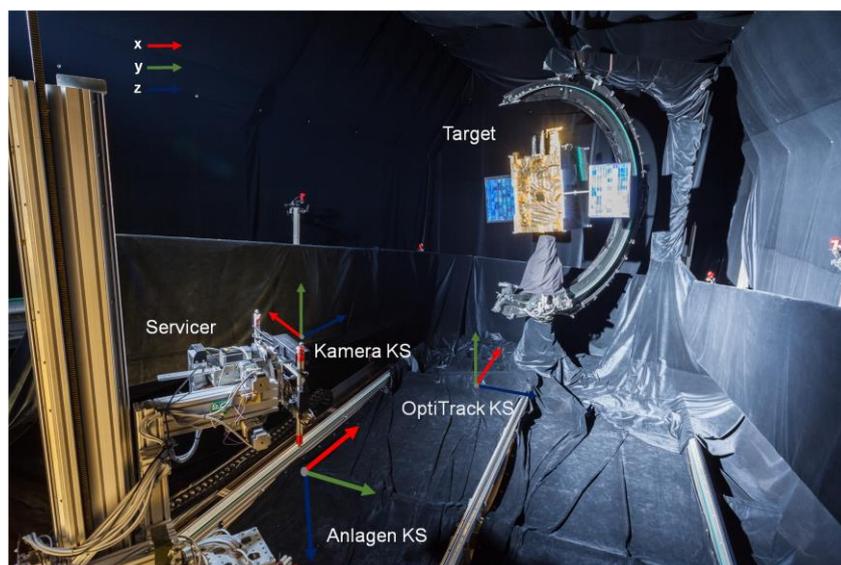


Fig. 3—1: Coordinate systems of the RACOON-Lab

This exact scenario could be simulated within the RACOON hardware facility, illustrated in Fig. 3—1. With the chaser being at a static position, a relative movement was realized by rotating the target around its yaw-axis. The generated trajectory is therefore always a circular path of the servicer around the target with a constant distance as well as angular velocity. A detailed description of the facility and its particular characteristics can be found in chapter 5.1.

3.2 Objectives of the work

The actual goal of this work is the execution and evaluation of a series of experiments in the RACOON laboratory. In this series, a few crucial parameters such as the angle of incidence of artificial sunlight are to be changed systematically and their influence on the quality on an optical pose estimation is to be shown. A reasonable selection of these parameters, but also a suitable setting of the used tracking algorithms plays a decisive role. In addition to a literature search, a pre-test series should also be helpful in the selection and, at the same time, enable familiarization with the test bench. The test results are to be evaluated quantitatively and linked with the corresponding influencing factors.

Furthermore, a metric, which was developed by Flavio Rehn in a previous bachelor thesis [18], is to be further improved and adapted for a quantitative evaluation of the test series. The focus is also on a clean scientific methodology, thorough preparation and proper documentation. Care should also be taken during the evaluation to ensure that the measurement data is carefully processed.

Thus, this thesis should show first in-depth approaches to the influence of certain factors on the 3D reconstruction quality, which can be used as reference values for further projects. Hence, possible difficulties of the teleoperation tasks in the context of the proximity navigation of the OOS can be highlighted and potential for improvement can be identified.

3.3 Methodology

Since the goal of this work was strictly defined in the beginning, so was the approach. This thesis is based on previous work by Tim Wiese, who implemented a software framework, able to evaluate SLAM algorithms and integrated cameras [19], and Flavio Rehn's developed metric. Therefore, it was essential to get an overview of the work which was done by them for understanding functionalities of the software and metric.

With the knowledge gained, the experiments could be designed. In order to carry out a scientifically correct and quantitative test series, the experiments had to be properly prepared. Thus, the study could be carried out with as little effort as possible and at the same time maximum understanding of the system. The corresponding procedure is presented in chapter 5.3. However, it was also essential to select suitable parameters, which have a large influence on the reconstruction quality and can be reliably varied in the RACOON Lab at the same time. In the course of the work, a list of all possible influencing factors was compiled and finally filtered for essential ones. In addition, all other influencing factors had to be set to fixed values to guarantee the reproducibility of the tests. With the help of a preliminary test series, these parameters were set to the best possible values regarding the results of the position estimation to not falsify the results in the main test series. Furthermore, first experience with the hardware environment of the RACOON Lab could be made, which is indispensable for a trouble-free and accurate test execution.

For evaluating the data generated by the test series, the metric had to be modified for processing a high amount of data. Not only had it be adapted for the specific scenario of the detailed study but also a possibility to concretely show the correlating results between single parameters had to be developed.

3.4 Differentiation

To be able to prove the influence of different factors on a pose estimation of a satellite experimentally and quantitatively, the software for data generation and processing is equally necessary as the hardware simulator of the corresponding scenario.

Fast and accurate 6 degrees of freedom (DOF) Visual Odometry (VO) is a relatively new field of research but is gaining importance in current robotics. Normally it is focused on terrestrial scenarios and most VO algorithms can be found in interior applications and ground-based systems. [20, 21] Yet, the requirements of a space environment are far more demanding. Nevertheless, by now, a variety of different



approaches regarding this matter are existing, using diverse algorithms as well as sensors. [7, 21]

However, this thesis does not discuss and compare any algorithms or sensors other than those already integrated. Neither are new algorithms implemented or evaluated for the use in the RACOON-environment.

Under the assumption that the biggest error in 3D-reconstruction lies in the trajectory, which is estimated by the tracking algorithm, the metric and therefore the evaluation of the study is limited to that. Results of mapping algorithms in specific will not be considered in this work.

4 Introduction in used algorithms and metric

4.1 DIFODO

4.1.1 General information

Developed by Jaimez and González-Jiménez and published in 2015 [22], Differential Odometry (DIFODO) is a simple tracking algorithm for visual odometry. The method applies the range flow constraint equation to sensed points in the temporal flow to derive the linear and angular velocity of the sensor in a rigid environment. For that, only the depth map of a of a RGB-D frame is required. It is also promised to be very fast, running with up to with the highest camera framerate of 60 Hz which provides tracking in real time on a single desktop CPU core. It is even able to further increase performance by sampling down input parameters. However, this algorithm has no advanced functionality to minimize the drift of position estimation over time, such as graph optimization or loop closure.

4.1.2 Mathematical fundamentals

The following derivations aim to explain the range flow constraint and therefore the functionality of the algorithm itself and are mostly extracted from [22] and [23].

The point velocities are formulated in relation to the camera motion and if a sufficient number of points are considered, the latter can be calculated.

Let $Z: (\Omega \subset \mathbb{N}^2) \rightarrow \mathbb{R}$ describe a depth image taken by a 3D depth camera, i.e. an image between pixel coordinates $w = (u, v) \in \Omega$ and a depth value Z . Then the range flow constraint is by development of the Taylor series from Z to u and v and derivation after the time t :

$$\dot{Z} = \frac{\partial Z}{\partial t} + \frac{\partial Z}{\partial u} \dot{u} + \frac{\partial Z}{\partial v} \dot{v} + O(\Delta t, \dot{u}, \dot{v}) \quad (4-1)$$

and therefore

$$\dot{Z} \simeq Z_t + Z_u \dot{u} + Z_v \dot{v} \quad (4-2)$$

if the terms of higher order are neglected. This condition is permissible under the assumption that either the movement between consecutive images is small and can therefore be assumed to be linear, or that the observed points belong to local planar patches.

Now the camera velocity can be calculated. Since \dot{Z} , \dot{u} and \dot{v} are unknowns, they should be expressed in terms of the camera translation and rotation velocity. The three partial derivatives of Z can be directly calculated from the depth images. Fig. 4—1 illustrates the “pin-hole model” [24], which defines the relationship between the motion of an arbitrary point $P = (x, y, z)^T$ in the reference system of the camera and its optical

flow $\dot{w} = (\dot{u}, \dot{v})$ in the image plane assuming that the pixel and spatial coordinates of P are time-varying.

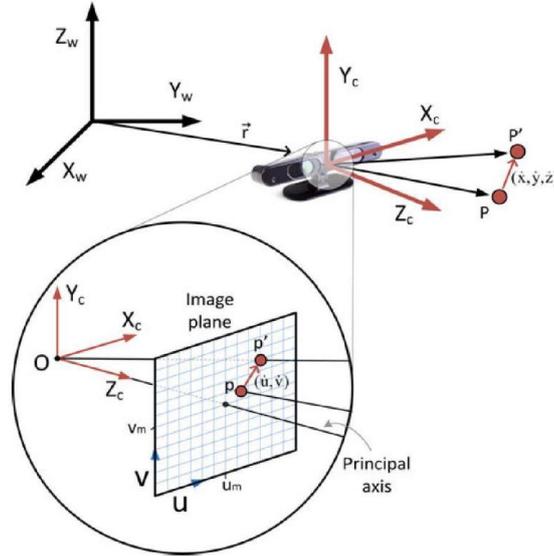


Fig. 4—1: Pin-hole camera model [22]

In the following, (v_m, u_m) is the image center (principal point), whereas f_x and f_y are the focal length values expressed in pixels:

$$v = f_y \frac{y}{z} + v_m \quad (4-3)$$

$$u = f_x \frac{x}{z} + u_m \quad (4-4)$$

This results in the following derivations:

$$\dot{v} = f_y \left(\frac{y\dot{z} - z\dot{y}}{z^2} \right) \quad (4-5)$$

$$\dot{u} = f_x \left(\frac{x\dot{z} - z\dot{x}}{z^2} \right) \quad (4-6)$$

With the camera velocity $\xi = (v_x, v_y, v_z, \omega_x, \omega_y, \omega_z)^T$ (v : translation velocity, ω : rotation velocity), the point velocities $\dot{x}, \dot{y}, \dot{z}$ of any 3-D point can be calculated. Thus, with respect to the camera reference frame, and under the assumption of a rigid and static scene, the velocity of a considered point in the global coordinate system is zero with the help of the Euler derivations can be expressed as

$$\dot{P} = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} = \begin{pmatrix} -v_x - z\omega_y + y\omega_z \\ -v_y + z\omega_x - x\omega_z \\ -v_z - y\omega_x + x\omega_y \end{pmatrix} \quad (4-7)$$

In the following (4-7) can be inserted in (4-5) as well as in (4-6). Z is hereby equivalent to z .

$$-Z_t = -\dot{z} + Z_u f_x \left(\frac{x\dot{z} - z\dot{x}}{z^2} \right) + Z_v f_y \left(\frac{y\dot{z} - z\dot{y}}{z^2} \right) \quad (4-8)$$

Finally, (4-8) is inserted in (4-2) resulting in

$$\begin{aligned}
-Z_t = & \left(1 + \frac{xf_x}{z^2} Z_u + \frac{yf_y}{z^2} Z_v\right) (v_z + y\omega_x - x\omega_y) \\
& + \frac{f_x}{z} Z_u (-v_x + y\omega_z - z\omega_y) \\
& + \frac{f_y}{z} Z_v (-v_y - x\omega_z + z\omega_y)
\end{aligned} \tag{4-9}$$

Equation (4-9) is linear with respect to the six camera velocity terms. This allows a solvable algebraic system to be built if at least six linearly independent equations for six points are present in the image. The linearization applied to derive the depth flow equation assumes differentiability of the depth images, and either small displacement of the scene or constant depth gradients.

In order to solve the “small displacement” restriction, which aims that movements between successive images must be as small as possible, the algorithm uses a pyramid calculation with Gaussian filtering of the input images on several levels. On higher planes, i.e. a more strongly filtered image of lower resolution, the movements may be larger with constant approximation quality. The camera speed is estimated on each plane, from coarse to fine, and is used to transform the image on the underlying plane in 3D space with the currently estimated values and to project it back. This results in intermediate images that show significantly less movement than the original images.

In practice, however, a much higher number of points (N) as the previously mentioned six points are considered to make the solution robust to noise and approximation errors. This leads to an overdetermined system and to solve that camera motion, the method of weighted least squares can be applied:

$$WA\xi = WB \rightarrow \xi \approx (A^TWA)^{-1}A^TWB \tag{4-10}$$

where $A \in \mathbb{R}^{N \times 6}$ contains the coefficients that multiply ξ in (4-9), $B \in \mathbb{R}^{N \times 1}$ contains the temporal derivative of depth for each pixel and $W \in \mathbb{R}^{N \times N}$ is a diagonal matrix containing the weights associated with the uncertainty of each equation.

From the solutions for the translation and rotation speed values of the camera in each time step, the overall estimation of the camera position is finally obtained by numerical integration and transformation into the global coordinate system.

For more detailed mathematical derivations, implementation details and benchmark results of the algorithm, refer to the corresponding publication [22].

4.2 ZED Tracker

Information about the exact functionality of the intern algorithm of the ZED stereo camera are publicly not available. Yet, it is assumed that it falls into the category of a SLAM algorithm, which mainly relies on OpenCV (Open Source Computer Vision) libraries. Meanwhile, the algorithm should have been mostly developed by the manufacturer of the camera, namely Stereolabs, itself, which is the reason for the poor information to be found in corresponding literature. Yet the Application Programming Interface (API) of the camera revealed information about drift reducing functions like spacial mapping (also known as loop closure) as well as pose smoothing for graph optimization.

Since the study was conducted with the ZED intern recording software in order to guarantee the latest software updates of the ZED Software Development Kit (SDK), the algorithm could also be used for the study. With the given functionalities, it was considered to be a valuable reference to the DIFODO tracker to identify the biggest differences in performance. (see chapter 5.4.1)

4.3 Used Metric

The trajectory, which is calculated by a tracking algorithm just presented, describes the position of the camera relative to the target in a camera fixed coordinate system (CS). In the given scenario of a simulated flight of the chaser around the target, the estimated trajectory has a circular form, as seen in Fig. 4—2.

Thereby, an error can only be recognized qualitatively. To be able to make an absolute statement regarding the quality of the pose estimation, various evaluation procedures must be taken into account. Two of the most common methods calculate the absolute trajectory error (ATE) or the relative pose error (RPE) of the trajectory. [25]

The metric, which is presented in the following, is making use of the latter method.

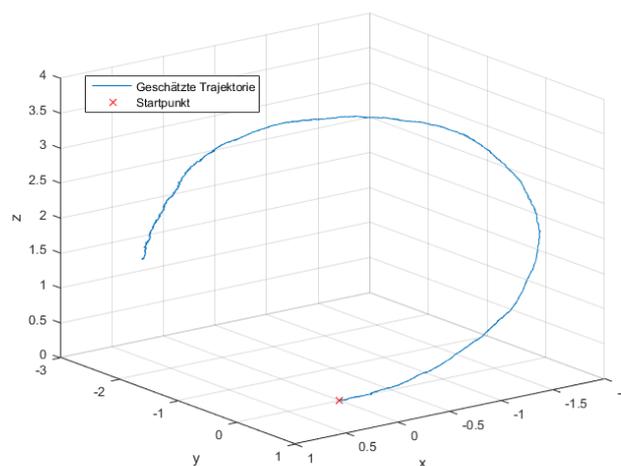


Fig. 4—2: By DIFODO estimated trajectory for a single fly-around of 360 deg

The main struct of the metric, which was used to evaluate the experimental study quantitatively, was developed in a student thesis by Flavio Rehn [18]. Following is mostly extracted therefrom.

First, a quick overview of the basic concept and functionality is given, and the mathematical fundamentals are presented. However, this will be limited to the very essential parts. In the end, it is explained how said metric was adapted to the specific scenario of experiments, which were conducted during the work.

4.3.1 Main Struct

The main use of the metric is to analyze the influence of different tracking algorithms as well as of software or hardware parameters on the quality the pose estimation of an object. Until now it was not possible to generate a ground-truth reference trajectory in the RACOON hardware facility due to difficulties in the exact pose determination and

orientation of corresponding sensors as well as rotation velocity of the target. Therefore, a geometrical figure was calculated with the help of fitting-algorithms to substitute the reference trajectory. This, in the following called “bestfit-circle”, is a circle $K \in \mathbb{R}^3$ made of n points, which is the closest geometrical figure to the point cloud of the real trajectory generated by an algorithm.

With the help of this bestfit-circle, a possibility to compare certain characteristics of the real point cloud with a reference is existing. This results in five modules or metric parameters, which allow a quantitative evaluation of the estimated trajectory, shown in Tab. 4—1.

Tab. 4—1: Definition of the five metric parameters

Metric parameters	Definition
C_F	Distance between the center points C_K and C_G
r_F	Difference between the radii r_K and r_G
ε	Elevation between K and the reference plane
R	Mean Residuum of the estimated trajectory relating to K
D	Drift of the estimated trajectory in relation to K

The reference plane is defined as the plane in which the target is rotating. All reference points were determined with a motion tracking system called OptiTrack (further information can be found in section 5.1.5) and are marked with a G in the index. C_G is therefore the center point of the target and r_G is the radius of a possible reference trajectory. It can also be referred to as the distance between the camera lens and C_G . However, C_K is the center point of K and r_K the radius of K .

Hence, if all metric parameters have a value of zero, the point cloud of the real trajectory would be identical with the bestfit-circle K as well as the reference trajectory. Only the Residuum R and Drift D therefore refer to K , whereas C_F , r_F and ε are depending on the reference plane. This makes latter more passive modules. Even though it is mandatory for a comprehensive evaluation of the estimated point cloud to compare its pose, diameter and inclination with a possible real trajectory, these parameters are very susceptible for errors in the pose detection of said characteristic points captured with the OptiTrack system. A more comprehensive discussion about this can be found in chapter 7.

The drift D on the other hand was considered a very important module of the metric with a significantly high impact on reconstruction quality [4], even before this thesis was written. Since the estimated trajectory has not been able to display the same circle progress as the circle part which was recorded in the study (about 270 degrees), the pose of the servicer is not estimated correctly relative to the target. The individual positions of the trajectory thus lag the positions of the true trajectory, either before or behind. This effect even increases over time.

D results from the drift angle δ , which displays the angle of the delay or lead of a single point compared to the corresponding reference point, as shown in Fig. 4—3. Thus, D describes a vector showing the angles δ_i of every timestamp i .

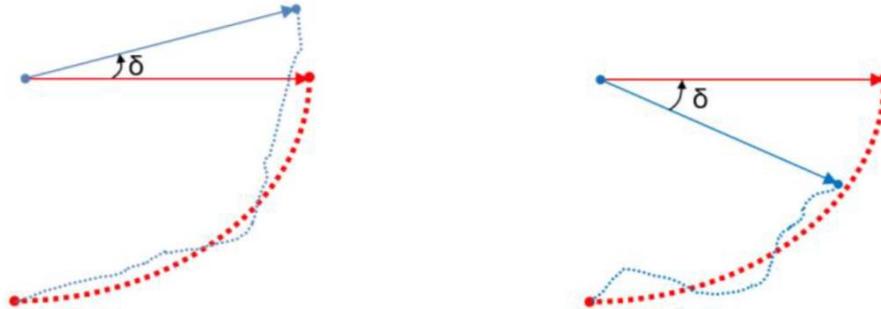


Fig. 4—3: drift angle δ ; a lead (left) or a delay (right) of the estimated trajectory (blue) referring to the real trajectory (red)

Furthermore, in the context of this work the elevation ε describes the angle between the plane of the bestfit-circle (the best adapted plane of the point cloud; in the following referred as bestfit-plane) and the reference plane of the rotating target. In this study, the reference plane will always be parallel to the XZ-plane of the camera CS (see Fig. 1—1) The Residuum R , on the other hand, characterizes the deviation of all points of the estimated trajectory to the respectively next point on the circle K .

4.3.2 Mathematical fundamentals

Besides the determination of the five metric parameters an essential part of the presented metric is the calculation of the bestfit-circle for the generated point cloud, as it is the basis for the evaluation by functioning as a reference trajectory. Nevertheless, this chapter only explains the approach for this calculation, not the exact mathematical derivations. For more details regarding this matter refer to [18] as well as [26], [27] and [28].

The final approach chosen for this metric essentially comprises three steps:

1. Reduction of the 3D problem to a 2D one by projecting the point cloud on the bestfit-plane
2. Calculation of an adapted circle for the 2D dataset of the estimated trajectory by applying the Least Square Fit Method (LSF)
3. Reverse transformation of the 2D circle into 3D coordinates.

Fig. 4—4 illustrates the bestfit-circle K calculated in the 2D plane, whereas Fig. 4—5 shows the final result in 3D. It is to be noted though, that this is not the only possibility to get a solution for described problem.

For further calculations of the five metric modules like the elevation ε however, it is indispensable to know the orientation of the bestfit-plane in space, which is defined through the unit normal vector n_K . Its determination takes place in the first of the mentioned steps but will not be explained in more detail.

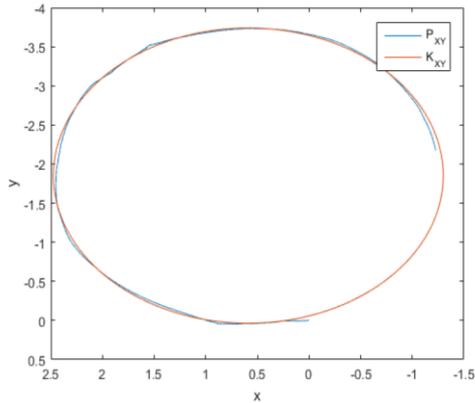


Fig. 4—4: Bestfit-circle for a point cloud in 2D

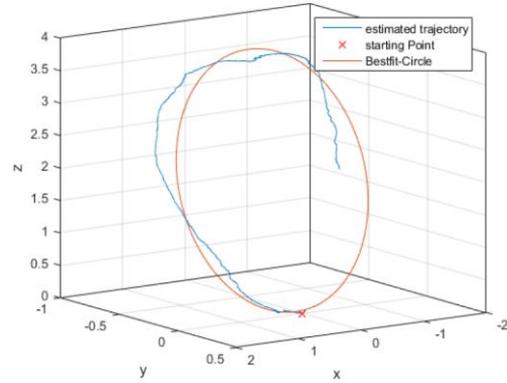


Fig. 4—5: Bestfit-circle for a point cloud in 3D

Thus, for getting the angle between two intersecting planes, their two normal vectors must be considered. As already mentioned, since the scenario of the study was a “straight flight around the target” and therefore the rotating plane of the target is not changing, the reference plane is always the same. It is defined through its normal vector n_G . Consequently, the elevation ε can be calculated with the following common formula, which can also be found in [29]:

$$\varepsilon = \arccos\left(\frac{|\langle n_K, n_G \rangle|}{\|n_K\| \|n_G\|}\right) \quad (4-11)$$

Other modules which have to be checked for errors are the center point C_K and the radius r_K of the bestfit-circle K . Since the reference points C_G and r_G are given through the OptiTrack system, the divergence result in:

$$C_F = \left| \overline{C_G C_K} \right| \quad (4-12)$$

$$r_F = r_G - r_K \quad (4-13)$$

The mean residuum R as well as the drift D on the contrary are not as self-explanatory as previous parameters. For the Residuum, the Euclidian distance between all points P_i on the estimated trajectory and the in each time closest points K_i on the bestfit-circle $d_{i,min}$ are determined. The vector d_{min} represents all n minimal distances between those single points and is defined as:

$$d_{min} = (d_{0,min}, d_{1,min}, \dots, d_{i,min}, \dots, d_{n-1,min})^T \in \mathbb{R}^n \quad (4-14)$$

The Residuum R can be described as the average value of the smallest distances between the estimated trajectory and the bestfit-circle in 3D and is therefore the mean value of d_{min} :

$$R = \frac{1}{n} \sum_{i=0}^{n-1} d_{i,min} \quad (4-15)$$

Thus, R expresses a form of geometric similarity of the estimated trajectory to a circle. The larger the value R is, the less the course of the estimated trajectory resembles that of a circle. To inspect this factor even in more detail, the distances $d_{i,min}$ were

separated into its XY-component $d_{i,minXY}$ as well as its Z-component $d_{i,minZ}$, where the XY-component corresponds to the distance value in the bestfit-plane.

The drift \mathbf{D} on the other hand is completely determined in 2D within the bestfit-plane, since only the rotational error is of importance. The scenario is illustrated by Fig. 4—6.

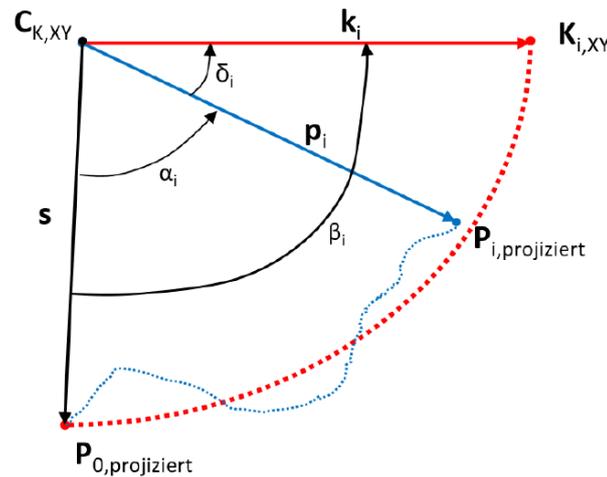


Fig. 4—6: Required values for calculating δ_i

To obtain the drift \mathbf{D} , the drift angle δ_i must be calculated first. Thereby, δ_i is simply the difference between the two auxiliary angles β_i and α_i , which can be determined with help of the normal vectors k_i , p_i and s :

$$\delta_i = \beta_i - \alpha_i \quad (4-16)$$

Consequently, \mathbf{D} results in

$$\mathbf{D} = (\delta_0, \delta_1 \dots \delta_i \dots \delta_{n-1})^T \in \mathbb{R}^n \quad (4-17)$$

However, it turned out that for the quantitatively evaluation of influencing factors like the sunlight, the impact of the drift is hardly to be illustrated vividly. Since the distribution of the drift during the simulated fly-around is the actual interesting and important factor, it cannot be described as a single number. A possibility to remedy that problem is to take the absolute maximum value reached by δ_i , since this is a reference value for the error in the pose estimation caused by the drift \mathbf{D} . This value is called $\delta_{i,max}$:

$$\delta_{i,max} = \max(|\mathbf{D}|) \quad (4-18)$$

The absolute value of \mathbf{D} can be taken because a possible lead of the estimated trajectory (negative δ_i) has the same impact on the reconstruction quality as a delay (positive δ_i).

For further details as well as for information about the implementation in Matlab please refer to [18] again.

4.3.3 Adaption

The metric was developed as an analyzing tool and evaluation criterium for a flexible scenario for used sensor systems, tracking algorithms and influencing factors. Its implementation was done in Matlab.

To analyze a huge amount of data generated by a vast number of experiments, an option to automatically feed that data into the metric had to be implemented. Furthermore, the design of the simulated scenario changed. When before a slanted fly-around was replicated by tilting the rotation axis of the target, this was adjusted to a straight fly-around by leaning and rotating the target with help of another axis in the HIL facility. Thus, it was possible to see the reference plane as a constant, which was a big advantage for the quantitative evaluation.

Also, since two different tracking algorithms were used, they had to be matched in units. Due to complications in generating the trajectory with the DIFODO algorithm, it was not yet possible to solve the unit mismatch in the software framework itself. Thus, a possibility to select the used algorithm was implemented.

5 Execution of the study

Before the experimental study itself is presented, all the main elements of the RACOON-Lab are explained, which are mandatory for understanding the work done. Hence, there is an overview about the technical data of the HIL facility as well as the settings of hardware and software parameter as well as algorithm settings. Additionally, it is demonstrated how the series of tests was planned by using a statistically validated method called Design of experiments (DoE).

5.1 RACOON-Lab introduction

5.1.1 General information

5.1.1.1 Background

The RACOON-Lab is, as mentioned in the beginning, a simulator for the maintenance of satellites and removal of space debris, by incorporating the relevant key aspects of those missions into an end-to-end simulation. It is operated by the LRT at the TUM. The design is based on the architecture of a tele-operated space mission, as can be seen in Fig. 5—1. [30]

The work in the Racocon-Lab concentrates mainly on two main topics: Human Spacecraft Interaction (HSI) and Autonomous Guidance, Navigation and Control (GNC). In the first priority area, developments are carried out to ensure the optimal integration of humans into the control of a robotic spacecraft, where the main focus is the consideration of semi-autonomous controllers and human intervention possibilities. In the second area, which also is the focus on this thesis, improved algorithms for partially or fully autonomous final approaches to uncooperative targets are developed and tested.

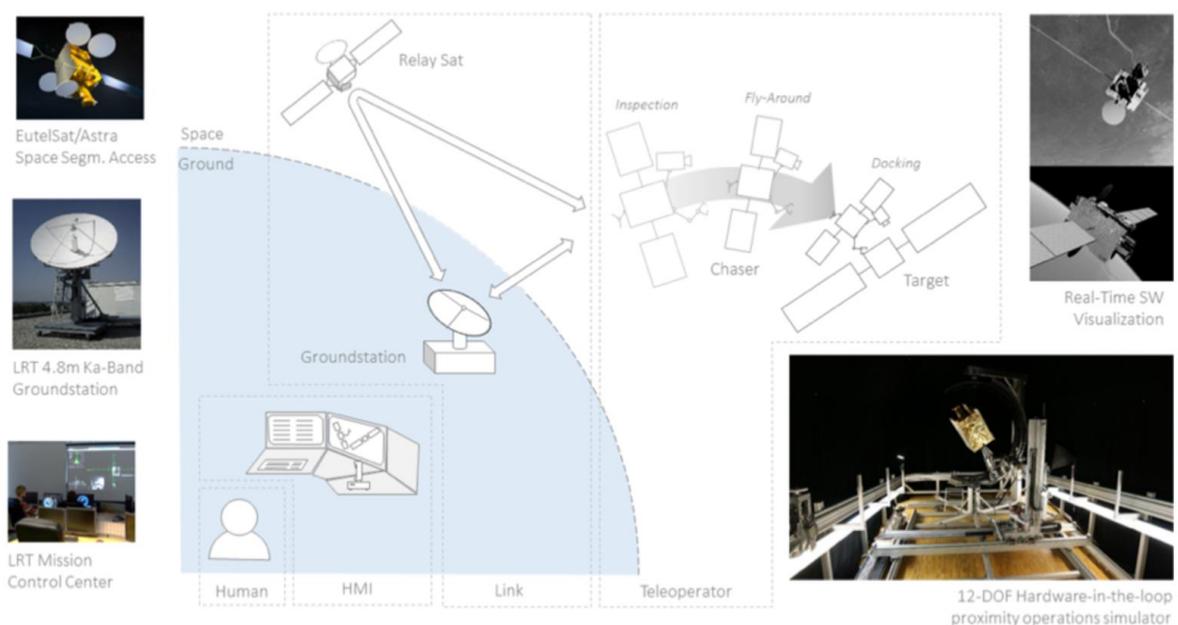


Fig. 5—1: Illustration of the RACOON-Lab concept

Furthermore, the purpose of the facility is the verification of sensor technologies using real hardware sensors, especially proximity sensors and optical cameras, under realistic conditions.

5.1.1.2 Range of Motion

At the moment, the HIL facility has 13 DoF, which are measurable with a proportionate accuracy. However, two of them are belonging to the sunlight simulator, which leaves the main components with the remaining 11 DoF. The target satellite is mounted on a 3-axis rotational head in the middle of a C-shape (also referred as C-sled), located at the assumed satellite's center of gravity, but has 5 rotational axes in total and thus is mechanically overdetermined. Additional to its 3 basic DoF it is possible to tilt the target over the C-sled up to 180 degrees around the horizontal axis and let the whole setup rotate around a stable vertical axis with help to the C-sled unlimited, as well. This enables the facility to simulate endless rotation and tumbling scenarios around any rotational axis. The chaser satellite on the other hand has 3 axes for rotation and 3 linear axes for translation. Fig. 5—2 shows the facility in operation and given annotations visualize the robotic 11 DOF excluding sun/earth position.

The distance from chaser to target for proximity operations ranges from zero to 5m, which is scaled down by a factor of 4, therefore correlating to a real distance of 20m.

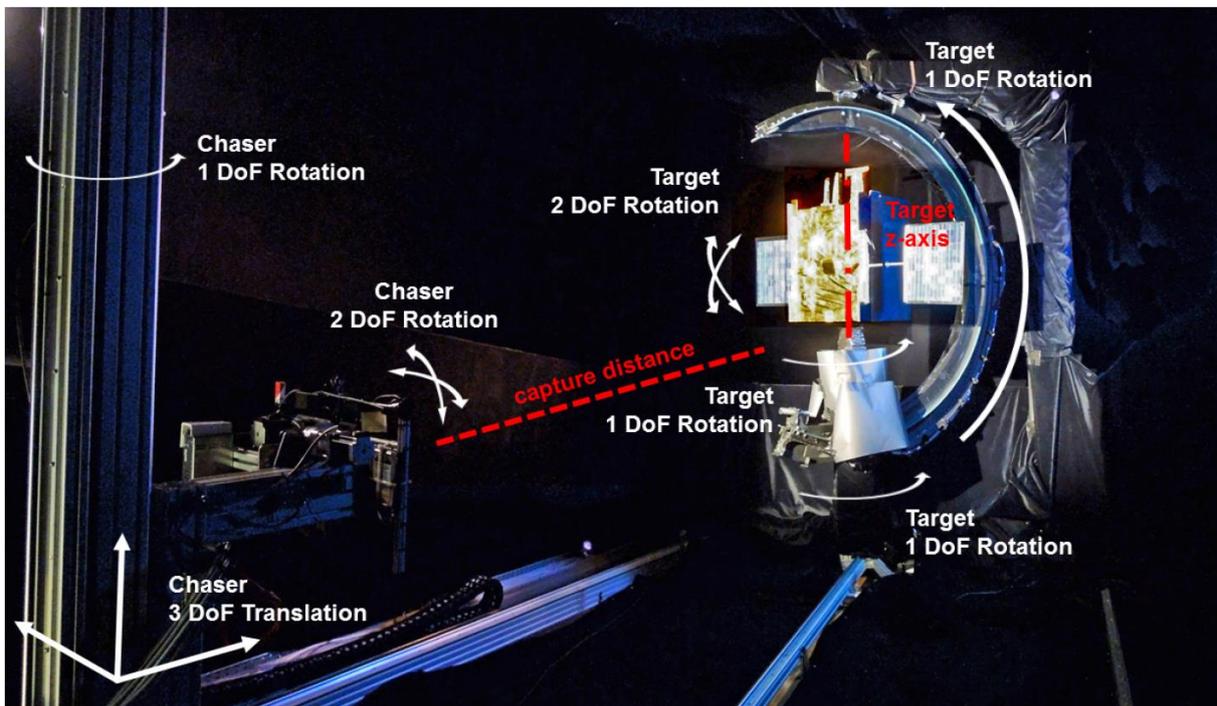


Fig. 5—2: 11 degrees of freedom of the target (right) and the chaser satellite (left) provided by the Hardware Facility

5.1.1.3 Hardware Facility Control

The HIL facility is driven by a Graphical User Interface (GUI), which is implemented in LabVIEW. The control software allows it to operate every single axis of the facility individually, supporting following modes:

- Network: moves the hardware simulator based on commands received though a User Datagram Protocol. Required target positions of facility components can also be directly entered.
- RefSearch (Reference Search): executes an automated sequence to bring the corresponding axis back to initial reference position (position zero) achieved through reference switches.
- Stopping: stops the axis immediately to prevent possible damage. When this mode is engaged, it also prevents any further movement of the axis.
- Docked/PathDemo: both modes are not used yet.

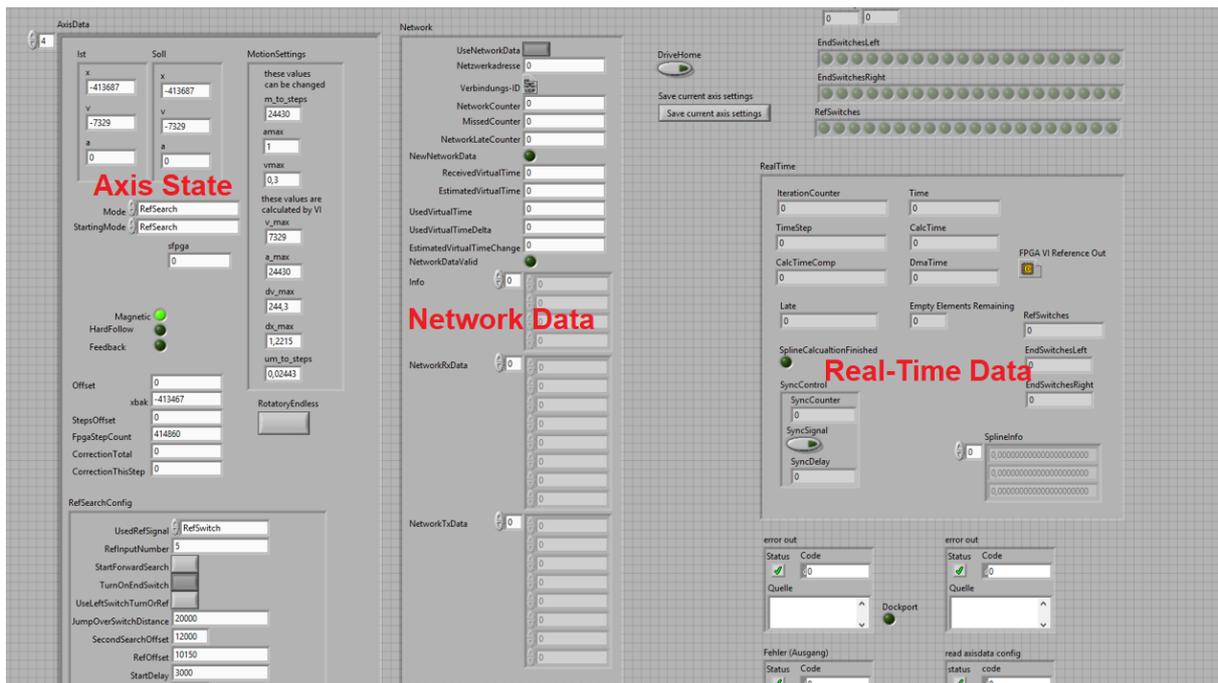


Fig. 5—3: LabVIEW GUI

Furthermore, the Axis State is containing all information that differs of each axis, including current and targeted state vector for position, velocity and acceleration, operating mode, limits for the range of motion as well as conversion factors. Any data related to translational and rotational movements is displayed in motor steps.

The relative value of each step is different from axis to axis, since partly different mechanics are built into the facility components. Conversion factors from steps to meter or steps to degree are already existing, however the user is still responsible for applying them to get usable units. Additionally, the accuracy of said factors is not documented, which makes it complicated to use them in a detailed study.

A verification of the conversion factors, alongside with a revision of the current LabVIEW GUI, is in work in a parallel student thesis. Fig. 5—3 shows a cut out the current version of the interface.

5.1.2 Satellite-Mockup

The target satellite used for this study is a mockup representing the body of a generic geostationary satellite with its characteristic surfaces and structures, such as blank

metal parts, solar arrays and several small boxes for the satellite's instruments. Large areas are covered with reflective insulation foil, also known as Multi-Layer Insulation (MSI). Especially the shiny silvery lateral surfaces as seen in Fig. 5—4 reflect most of the incident light, making them look like a flat mirror. This high reflectivity of the lateral surfaces turned out to be a special challenge for the tracking algorithms in all experiments. The used solar arrays are only mounted on one side on the plate.

The dimensions of the main body are approximately 600 mm x 700 mm x 1000 mm, which is also scaled down by a factor of 4 compared to a real satellite. With the solar panel added, the mockup has a width of even 1600 mm which is slightly within the potential diameter of 1800 mm restricted though the dimensions of the C-sled.

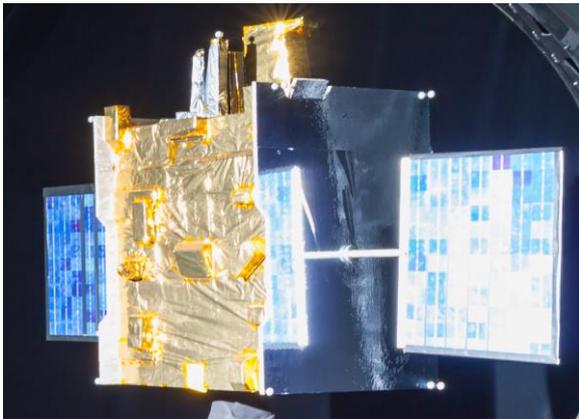


Fig. 5—4: target satellite mockup

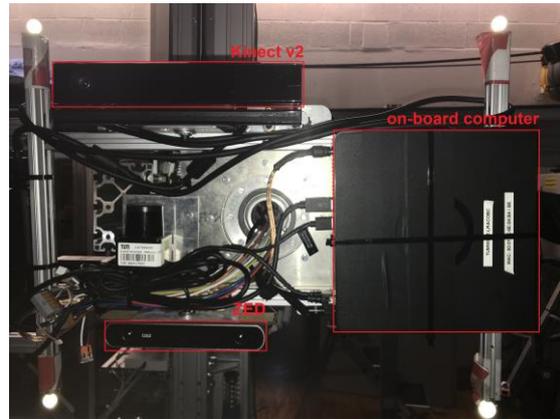


Fig. 5—5: chaser satellite

The chaser, on the other hand, is not represented by a geometry mockup, but by an open sensor platform showed in Fig. 5—5. Besides the two sensor systems Kinect v2 and ZED it also carries a ZOTAC MAGNUS EN979 mini PC as an on-board computer with relevant technical specifications presented in Tab. 5—1. [17]

Since the simulation is based on CAD models imported from CATIA the geometry of the chaser and target can easily be replaced and adapted to specific scenarios.

Tab. 5—1: technical data of the on-board computer

CPU	Intel Core i5-5200U (dual core, 2.2 GHz – 2.7 GHz)
RAM	16 GiB DDR3
GPU	GeForce GTX 960 with 3 GiB GDDR5, 192 bit
Hard drive	256 GiB SSD

5.1.3 Cameras

As already mentioned, two different sensors are currently integrated in the facility: StereoLabs' ZED and Microsoft's Kinect v2. Both sensors fulfill the requirements of presented algorithms and provide visual image as well as depth information by even

using two different technologies: computation-based depth sensing using stereo vision in case of the ZED and a monocular camera combined with a separate depth-sensor also known as RGB-D technology for the Kinect v2.

However, for the experimental study only the ZED was considered and thus it is the only one being introduced in the following. More information regarding the Microsoft Kinect v2 can be found in [31].

Titled as “the world’s first 3D camera for Depth Sensing and Motion Tracking” by the manufacturer StereoLabs [32] and shown in Fig. 5—6, the stereo camera calculates depth information using two high-resolution RGB sensors. The principle is, that two images of the same object are displayed from slightly offset positions and with the help of Epipolar geometry, a disparity image in grayscale and finally the depth image with one depth value per pixel is calculated.

Due to the high frame rate by the corresponding image resolution it was expected to get good results in tracking accuracy. Though, the maximum frame rate of 100 frames per second are only available at an image resolution of 672 x 376 pixels. An overview of the technical data relevant for this camera is given in Tab. 5—2.



Fig. 5—6: ZED stereo camera from StereoLabs [33]

Tab. 5—2: technical specifications of the ZED [4], [34]

Manufacturer	StereoLabs
Sensors	Stereo RGB
Interface	USB 3.0
Resolution [px]	672 x 376 1280 x 720 1920 x 1080 2208 x 1242
Frame rate [Hz]	100, 60, 30, 15
Depth range	0.7m – 20m
Accuracy	0.4mm – 1mm
Field of view	90° (H) x 60° (V) x 110° (D) max
Dimensions	175 mm x 30 mm x 33 mm
Weight	159 g

Released in 2015, this stereo camera additionally comes with an SDK including software samples and a documented API.

5.1.4 Lightning conditions

Realistic orbital illumination in the HIL facility is generated by a dedicated sun simulator unit as well as an earth's albedo unit, which were both designed in two previous student theses [35], [36] and are illustrated in Fig. 5—7. To guarantee a spectral distribution close to the sunlight, both units contain a metal-halide lamp, as seen in Fig. 5—8 in example of the sun simulator, whereas the simulated sun has a radiation intensity of 575 W and the albedo lamp of equivalent 250 W even adjustable through an optical filter.

In theory, both units are also able to move along a sled around the facility representing the orbital position, also called mean anomaly, of the satellites. Therefore, different illumination incidence angles can be simulated, even dynamically changing lightning conditions during an operation are possible. However, during this work, it was not yet possible to vary the position of the albedo simulator with a sufficient accuracy because a proportionate control was not implemented yet. But since the albedo unit would have been a constant in the parameter study anyways as correlating angle interrelationships between sun and earth's albedo could not be simulated yet properly either, it was not further considered to assure a reproducibility of the test series.



Fig. 5—7: Sun and earth albedo simulator [17]

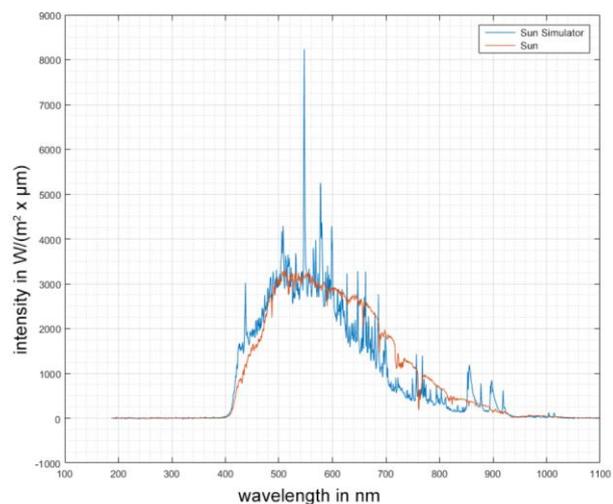


Fig. 5—8: Spectral distribution over wavelength of the sun simulator (blue) compared to ETR ASTM E-490 standard spectrum (red) [35]

Furthermore, to prevent the sensors to register any reflections of the room, over 90% of the installation is covered with a black Molton cloth, which reflects less than 1% of the incident light [37]. Therefore, the same situation as in orbit can be created, simulating the space environment, where the satellite is in front of an infinitely distant background. Nevertheless, it is not entirely possible to filter every single light reflection. There still are places, mostly components of the facility itself e.g. the C-sled of the target satellite, which cannot be covered and thus are tremendous perturbations to the algorithms. A possibility to limit this error is presented in chapter 5.2.6.

5.1.5 Optitrack Reference System

Additionally, a Motion Capture System called Optitrack is integrated into the hardware facility, which was also implemented as part of a student thesis [38]. Developed by NaturalPoint [39] it allows to determine the exact attitude of nearly each component and is even able to indicate the velocity of moving objects. With the help of 8 cameras, distributed on specific locations, a systematic measurement accuracy in millimeter range is achieved for moving objects in praxis, for static modules even $<1\text{mm}$ [38], even though the system is said to be capable of sub- $20\ \mu\text{m}$ accuracy in optimal conditions [40].

In this work, the system was mostly used to validate all the measurements, which were done during the preparation for the test series, as far as possible. Important factors were the absolute pose in case of the chaser satellite and the attitude for the target. To get the pose information of a single component with help of the Optitrack, said objects, besides being within the field of view of the installed cameras, needed to have at least 3 markers attached to get successfully determined. The sun simulator however, since it moves on a sled around the whole facility, does not have those points attached nor is even located in the corresponding area to be located by the cameras. Verifying its position was therefore not possible with the Optitrack System.

So far, an exact verification of the accuracy of the Optitrack System is yet to be done. Thus, for this study, an overall systematic accuracy of 1mm was assumed, as it was determined in [38]. A statistic error, however, was defined for every needed axis individually. The from the Motion Capture System registered “error-per-marker” values were thereby not considered for the calculation due to occasionally very high deviations.

Every measurement conducted with help of Optitrack was additionally secured by a manual measurement, to avoid large errors due to e.g. an imprecise calibration of the cameras. As already mentioned, Optitrack does not cover the whole facility. Measurements needed, which were out of range of the system, therefore had to be conducted by hand. Due to a very precise execution, in all conscience, as well a systematic accuracy of 1mm could be achieved.

5.2 Selection of parameters

After the introduction of the environmental conditions, this chapter gives an overview of all presettings, which had to be determined before the actual detailed parameter study could be started. For this purpose, a preliminary test series had to be conducted. On one hand, this series allowed it to analyze various input factors regarding their influence on the pose estimation qualitatively and thus supported the selection of critical ones for the study. Furthermore, all other variable factors could be set to approximately “good values”. Even though this term is relative and thus it could not be guaranteed that the constant parameters are at its best value, this still was a very important step to assure a reproductivity. On the other hand, it was possible to get familiar with the HIL facility and to learn how it is controlled, since a trouble free main series only can be executed with experience in the hardware.

However, factors used for the detailed study were found quickly. Since the scope of the thesis was clarified from the beginning only a very few critical parameters could be changed, guaranteeing the clarity of the work and limiting testing effort.

Several publications named the main disadvantage of mono and stereo cameras during rendezvous and docking maneuvers in the high sensitivity to sun illumination. [41–43] Since the study was planned to be conducted with the ZED, therefore a stereo camera, the impact of lightning could not be neglected. This was also proven in previous work in the RACOON-Lab. Qualitative test campaigns showed a significant influence of the incidence angle of the sunlight on the quality of the results of reconstruction [4, 18], illustrated in Fig. 5—9.

However, this thesis still had to be quantified. Hence, it was decided to analyze the impact of the sunlight further since all necessary conditions for a detailed study of said parameter were given in the HIL facility as well.

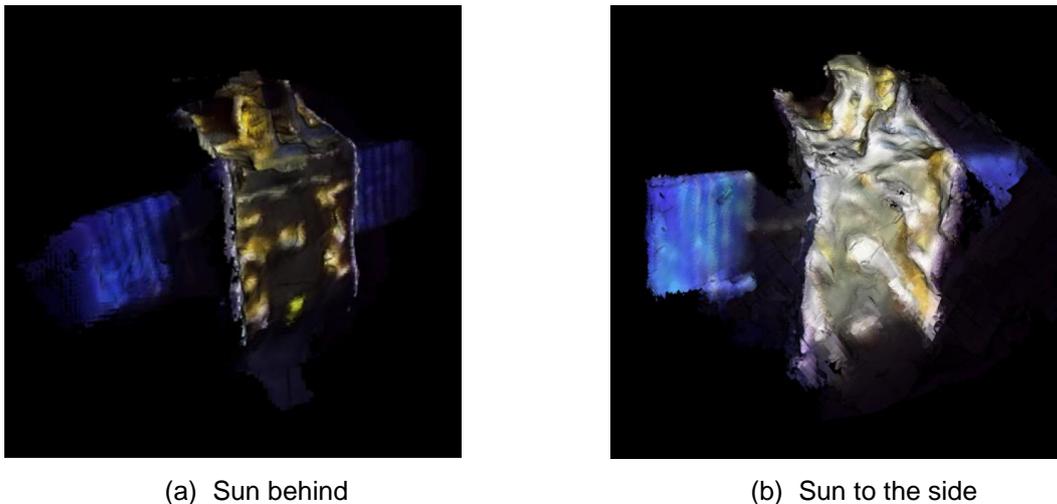


Fig. 5—9: Qualitative comparison of 3D reconstruction results using two different illumination angles, taken with the ZED.

5.2.1 Possible parameter variations

Yet, as a first step for the parameter study, to get an overview of the system and to find out which other influencing factors have a significant impact on the pose estimation, a list of all possible variables had to be compiled. In Tab. 5—3 all elaborated factors are listed. In order to keep the table clear, please note that not all possible variables are mentioned but only those which were considered to have a substantial impact on the test results, identified through the pretest series and previous test campaigns.

For an example, any settings of the tracking algorithms cannot be found in table. However, those are explained in the following chapters since their influence on the reconstruction must not be neglected either. Besides that, it is presented how parameters not chosen for the study were set to constant values.

Tab. 5—3: Overview of important variable parameters (algorithm setting excluded)

	Software	Hardware
Influencing Factor	frame rate	rotation velocity
	resolution	inclination target
	depth mode	illumination incidence sun
	brightness	distance chaser-target
	contrast	direction of rotation
	hue	starting orientation target
	saturation	type of sensor
	gain	type of lightning
	exposure	details mockup
	whitebalance	intensity of illumination
	boxfilter ¹ settings	

As already mentioned, the incidence of the sunlight was considered to have a crucial impact on the pose estimation and thus reconstruction quality. However, during the pretest series, another factor seemed promising in its optimizing potential as well. Proposed to further investigation in [18, 19], the rotational velocity of the target satellite appeared to also have an impact on the pose estimation.

Relating to Hardware Facility of the RACOON-Lab, this leads to following parameters that were decided to be varied:

- Illumination incidence of the sun simulation unit
- Inclination of the target mockup
- Rotation velocity of the target mockup / frame rate

Thereby, the first two factors both are influencing the angle of incidence of the sunlight. The illumination incidence of the sun simulator unit represents the angle regarding the position of the camera, since the unit can be moved along an axis around the facility, whereas the inclination of the target mockup mostly influences the direct angles of the reflection. Both scenarios are further explained as illustrated in section 5.4.2.

In case of the angular velocity, it was assumed, that the slowest possible velocity provides the highest information density, therefore the best results in tracking. This thought is also supported by the manufacturer. [33] However, a qualitative study in [18] proved exactly the opposite. After first experiments could not verify any of those claims, it was thus decided to be fully evaluated.

¹ definition given in section 5.2.6

Yet, the rotation velocity of the target mockup is also connected to another factor, namely the frame rate of the sensor camera. In theory, both factors can be handled equally. For example, the amount of information given to the algorithms through the camera for a frame rate of 30 fps and a rotation velocity of 4 deg/s is the same as for a frame rate of 15 fps and a rotation velocity of 2 deg/s. Consequently, in the following, these two parameters are merged. As a result, it was possible to add another variable to the study without increasing the testing effort. To analyze the impact of the rotational velocity, single frames of the dataset were cut out after the recording to simulate a lower framerate and thus a higher velocity. Nevertheless, since all recordings were conducted using a stable frame rate and rotational velocity, for a better understanding both factors are summarized under the term of the rotational velocity with the unit degrees per second in this work. In the future, under the assumption of recordings with variable fps or rotational velocity, it makes sense to define a unit called frames per degree or degrees per frame.

The following chapters present, how all constants were set, including the settings of used tracking algorithms. However, it is to be noted that chosen constant parameter settings were not quantitatively determined and possibly are therefore not the optimal values but resulted in a sufficient tracking accuracy and a subjectively satisfying output.

5.2.2 Main settings DIFODO

The DIFODO algorithm basically has 3 different parameters, that can be changed and that have an impact on the reconstruction, which are listed in Tab. 5—4. These had to be identified in the pretest series and set to proper values to not interfere with the main study.

Tab. 5—4: DIFODO parameters

parameter	set value	definition
downsample	1	factor by which the side length of the input image is divided to speed up the calculation.
ctf_levels	4	number of levels of the Gaussian filter pyramid (see section 4.1)
max_depth	5.0 [m]	maximum distance of an input data point

By sampling down the recorded data, an increased value leads to faster results to the detriment of image quality and therefore pose estimation accuracy. But since the difference in calculation time was not significant for the evaluation, downsampling was not considered.

A variation of the number of levels of the Gaussian filter pyramid used by the DIFODO algorithm does not seem to have a huge impact on the pose estimation around its standard value of 4, however, variations from said value slightly worsened the results except for the elevation, as seen in Tab. 5—5.

Any changing in the `max_depth`, thus the maximum depth detection, did not have any influence on the results due to the applied boxfilter (section 5.2.6) and thus was left at its default value.

Tab. 5—5: results in the pose estimation with different pyramid levels

ctf_levels	C_F [m]	r_F [m]	ε [°]	R [m]	$\delta_{i,max}$ [°]
2	1.2552	0.1909	25.27	0.1908	88.89
4	0.6894	0.0372	40.29	0.1663	86.95
8	1.4001	0.1329	50.12	0.2218	89.20

5.2.3 Main settings ZED Tracker

Even though one cannot find much information regarding the ZED intern tracking algorithm, the cameras API is documented very well. With its help it was possible to identify several parameters shown in Tab. 5—6, which were considered to have a direct input for the tracking accuracy of the camera for the given scenario of a simulated flight around a target satellite.

Tab. 5—6: ZED Tracker parameters

parameter	set value	definition
enable_spacial_memory	true	enables the camera to learn and remember its surroundings
enable_pose_smoothing	false	enables smooth pose correction for a small drift
DepthMaxRangeValue	5000 [mm]	maximum distance of an input data point
setConfidenceThreshold	85	sets a threshold to reject depth values based on their confidence

Listed parameters were tested for improvements on tracking accuracy and drift reduction on several test datasets.

Both `spacial_memory` as well as `pose_smoothing` is said to help correct positional tracking drift by the API. The former function is also known as loop closure. Thereby, while moving the camera in its environment, key images and contextual information will be saved. When the tracking detects an already-visited zone, it uses its memory to compute a robust position estimation that cancels eventual drifts. However, as this behavior can generate a sudden jump in the returned positions, `pose_smoothing` can be used to spread the drift correction across the upcoming estimations.

By enabling the former parameter little improvements in drift could be achieved. Since this mode also does only need a few resources to run it was decided to enable it during

the main test series. On the other hand, the mode `pose_smoothing` did not seem to have notable impact. It was therefore not considered and left on default. Also, it was desirable to link certain factor with characteristics in the drift course. Smoothing them out would have been contra productive for the evaluation.

Any change in the `DepthMaxRangeValue`, equally to `max_depth` from the DIFODO tracker, does not have any impact on tracking.

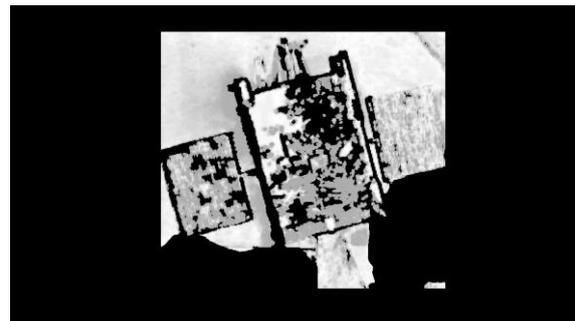
Other tracking parameters like `set_floor_as_origin` or `enable_imu_fusion` were considered to not have any impact on the tracking quality in the given scenario, neither.

The depth sensing function `setConfidenceThreshold` however has another role. By directly changing the depth map of the camera, this parameter has the same impact on both tracking algorithms.

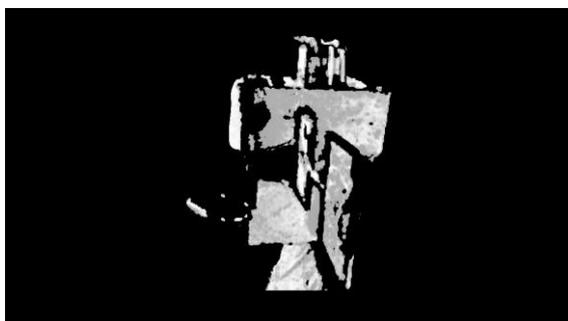
Thereby each depth pixel is assigned a corresponding confidence. By setting a threshold value between 0 and 100, certain depth values based on their confidence can be rejected. A lower value means more confidence and precision but less density, whereas an upper value reduces filtering which means more density, but less certainty. Hence, a value of 85 proved its best practice for the overall scenario by filtering the lowest confidence values and at the same time maintaining a mandatory high density level.



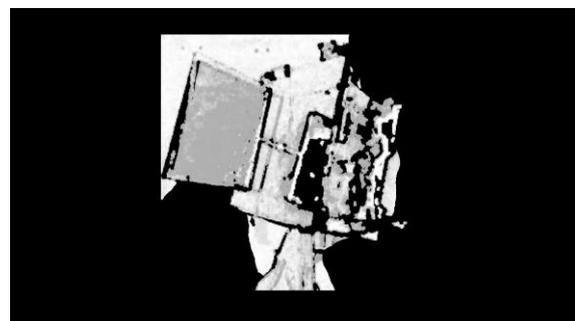
(a)



(b)



(c)



(d)

Fig. 5—10: confidence map with an applied value of 85 during a simulated fly-around from top left to bottom right

Mainly, this was an attempt to lower wrong depth detection of the sensor on different stages during a simulated fly-around caused by critical reflections on the satellites surface, as seen in Fig. 5—10. Even though overall improved results and a smaller

drift could be monitored, in some settings the confidence level of the camera on “wrong depths” was even higher than the confidence on actual structures and thus could not be filtered.

5.2.4 Software settings

Besides any settings of the algorithms or image processing possibilities, which could be set after the required data was recorded, there are still lots of parameters, most of them belonging to the camera, which had to be determined and set to constant values before the study started.

The cameras resolution is one of them. Both, maximum image information as well as fast data processing or storing had to be guaranteed. A resolution of 1280 x 720 [px] proofed itself as a good capturing mode, by at the same time providing a high frame rate. As already mentioned, correlating changes in frame rate and rotation velocity of the target are considered to have the same impact on the quality of pose estimation. With the same requirements regarding information density and data size, a frame rate of 30 fps was chosen.

Also, several depth modes are available to improve performance in certain applications: ULTRA, QUALITY, MEDIUM and PERFORMANCE. These settings adjust the level of accuracy, range and computational performance of the depth sensing module.

- The Ultra mode offers the highest depth range and better preserves accuracy along the sensing range.
- The Quality and Medium modes offer a compromise between quality and speed.
- The Performance mode disables some filtering operations and operates at a lower resolution to improve performance and frame rate.

Considering the given scenario, the Quality mode was used for the study, even though no notable difference to the Medium mode could be worked out during the pretest series.

All other camera settings like brightness, contrast, hue, saturation, etc. were set to auto. Due to the strongly changing lighting conditions during the experiments, constant best practice values could not be determined. In order to not falsify the results of the study due to poorly set camera settings for certain illumination scenarios, the settings were not set manually. Although mentioned parameters were considered to have an influence on the reconstruction quality, determining every single one of them for certain lightning conditions would have been beyond the scope of this work.

The necessary stereo camera calibration is done automatically by the SDK, the calculated intrinsics can be accessed through the API afterwards. For the study used parameters are displayed in Tab. 5—7: calibrated intrinsics of the ZEDTab. 5—7.

Tab. 5—7: calibrated intrinsics of the ZED

Focal length	f_x [px]	681.948
Focal length	f_y [px]	681.948
Optical center	c_x [px]	655.158
Optical center	c_y [px]	337.851

5.2.5 Hardware settings

As already mentioned, the rotation velocity of the target satellite as well as its inclination angle, and additionally the angle of incidence of the sunlight were chosen to be the variables in this study. Nevertheless, corresponding general conditions had to be set by limiting the variation appropriately.

In case of the angular velocity, a maximal as well as minimal value had to be determined. It turned out, that a rotation velocity below 1 deg/s at a frame rate of 30 fps could not be achieved with the current sensor computer, as the Random-Access Memory (RAM) would have to buffer over about 10000 images for a single fly-around. However, the 16 GB RAM available is not sufficient for this purpose. To verify the maximal velocity, at which it would be possible for the algorithms to still detect a pose, the same procedure as in the main series was conducted. A fly-around was simulated with the lower limit of 1 deg/s. Afterwards, single frames were cut out so simulate a higher velocity. Skipping one frame for example would therefore lead to a simulated velocity twice as high as the original, skipping three to a velocity four times as high etc. The highest skippable number of frames and therefore the highest velocity, at which algorithms were still able to track the target, was four frames or 5 deg/s. As a result of this, velocities of the driven 1 deg/s, 2 deg/s as well as 4 deg/s were analyzed in the main series, covering nearly the whole range. The scope of changes in the inclination of the target as well as the illumination incidence of the sun can be found in section 5.3.2.

Another important hardware parameter is the distance between chaser and therefore camera and target. It was assumed, that the smaller the distance is, the more information of the image could be gathered and thus the better the pose estimation would be. This statement was also yet verified in previous work on this topic. [4, 18] Furthermore, the ZED uses triangulation to estimate depth from the disparity image. The depth resolution changes thereby described by the following formula:

$$Dr = Z^2 \times \alpha \quad (5-1)$$

where Z is the distance and α a constant. Consequently, depth accuracy decreases quadratically over the distance. According to the manufacturer, the configuration that offers the best depth accuracy is to place the camera at a working distance of 1 meter of the scene. [44] Hence, the corresponding distance was minimized as far as possible, resulting in a value of 1,90 meters (camera lens to target rotational center), which was the shortest distance to the target satellite still being completely within the field of view of the sensor in any scenario.

Furthermore, the starting orientation of the target determined over which axis the mockup was tilted (see Fig. 5—14). This turned out to be a deciding factor on how strong reflections would be during the experiments due to the different surface coating. Additionally, since the rotation was realized with help of the C-sled, a basic approach to rotate the target satellite minimum 360 degrees to simulate a complete flight around and have fully information of color and depth images was not possible. Only an angle of about 270 degrees could be covered without the target being obscured by said C-sled. Thus, the side of the satellite facing the suspension could never be completely recorded by the camera. Both factors considered, it was decided to align the target in a way, that one of the two silvery lateral sides being in front of the sled. Thereby, most details of the target could be covered. Besides that, it was assumed that with a growing inclination angle the front side of the silver insulation foil facing more and more to ground, resulting in less strong reflections.

Any impact on the pose estimation through the direction of the rotation however was not discovered during the preliminary test phase. Any experiments in the main series were thus conducted with a counter-clock-wise (CCW) rotation.

Additionally, for a more authentic model there was an option to either test with or without a mounted dish. To simulate a scenario realistically as possible, the dish was attached for the study.

5.2.6 Image Pre-Processing

For the reference scenario given for this work, which is simulated in a hardware environment, a simple three-dimensional box filter was implemented to suppress any background structures of the simulator and thus reduce the considered information of the raw input data to a cuboid region. This area is defined in horizontal and vertical direction by minimum and maximum pixel coordinates u_{min} , u_{max} , v_{min} , v_{max} , whereas in object direction it is defined by minimum and maximum depth values z_{min} , z_{max} in millimeters (visualized in Fig. 5—11).

In practice, the satellite model in the simulator is not located in front of a distant background and thus outside the sensitivity range of the depth camera but is anchored to a carrier frame and surrounded by the laboratory walls. Even though those are completely covered in black Molton cloths, it cannot be guaranteed that all the satellite's reflected light is completely absorbed. These remaining visible structural elements would not appear in an on-orbit scenario and would prevent an accurate tracking of the satellite's rotation, since the algorithm does not expect objects moving in relative motion to the surrounding. Therefore, the filter was applied to cut out and ignore the remaining visible structural elements of the HIL laboratory during the experiments.

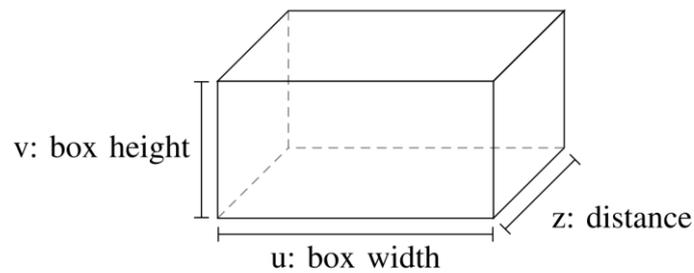


Fig. 5—11: Simple illustration of the applied boxfilter [4]

Tab. 5—8 shows the best practice dimensions of the boxfilter used for this study. Since the boxfilter had to be set to a constant value through the whole series of experiments to make sure it does not interfere with the results, due to the variation of the orientation of the target satellite a compromise had to be made. On one hand, a good filter would represent the dimensions of the image recorded by the camera by showing as much details of the satellite as possible and at the same time cut out any other unwelcomed elements. On the other hand, however, the change in the inclination of the target up to 45 degrees (see section 5.4.2) changes displayed dimensions as well. Consequently, two options were possible: firstly, extending the vertical as well as the horizontal values of the filter resulting in little parts of some facility elements like the target mount being within the filter in specific scenarios or, secondly, leaving the filter dimensions rather tight and thus cutting out parts of the target and therefore important information. But since the tracking algorithms used should be robust against smaller image parts that move relative to the main object, the former was chosen.

Yet, the depth range of the boxfilter was restricted as far as possible. As already mentioned, the depth detection of the sensor was challenged due to very little lightning and strong reflections on the satellite surface resulting the camera in sometimes not being able to display depth information correctly. An approach to solve this problem was to cut the satellite shortly behind its rotational axis, only leaving its front side within the filter. Therefore, it could be achieved to eliminate any structure and reflections taking place behind that plane, guaranteeing much better results as seen in Fig. 5—12. Tab. 5—9 also shows that all metric parameters could be improved, being able to even nearly halve the maximum drift during a fly-around.

Tab. 5—8: applied boxfilter values for the study

parameter	value
u_{min}	340 [px]
u_{max}	980 [px]
v_{min}	60 [px]
v_{max}	640 [px]
z_{min}	1100 [mm]
z_{max}	2100 [mm]

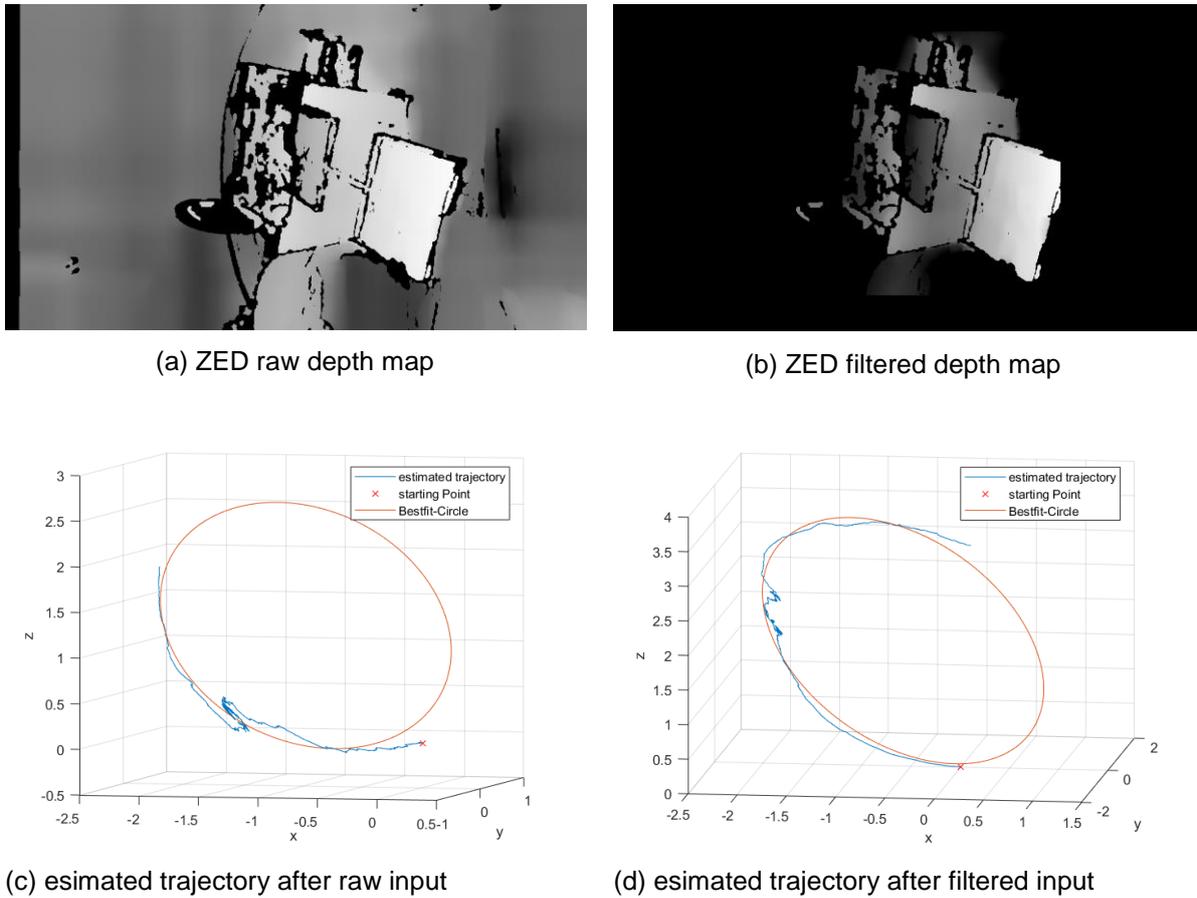


Fig. 5—12: comparison between the unfiltered and filtered depth map of the ZED as seen by the camera during the experiment and its resulting estimated trajectories calculated by the DIFODO algorithm

Tab. 5—9: numeric results on pose estimation for raw and filtered data

	C_F [m]	r_F [m]	ε [°]	R [m]	$\delta_{i,max}$ [°]
raw image	1.1706	0.4892	20.00	0.1321	154.41
filtered image	0.9832	0.0050	18.07	0.1263	84.74

5.3 Design of Experiments

5.3.1 General Information

Every quantitative study in any work field is based on statistics. Hence, DoE has also become a very important tool in engineering sciences. To plan, evaluate and therefore secure a test series statistically, various approaches to the system are possible. However, this chapter only introduces some of them very shortly and presents the

concept chosen for this work. More detailed information regarding DoE can for instance be found in [45].

Defined as “method for efficient planning and evaluation of test series” by [45], the goal is to achieve the greatest possible understanding of the system and therefore learn as much as possible about the relationships between influence variables (in the following also referred as inputs) and results (also referred as outputs) with as little experimental effort as possible. Therefore, information obtained with DoE on the relationships between input and output is statistically verified and the effects of input variables and their interactions on output are quantifiable. In addition, DoE serves as a generic term for methods for evaluating experimental data, such as determining the significance of various input parameters.

Results from experiments are always subject to a stochastically fluctuating measurement error. In order to minimize this influence, some experimental plans for a physical study propose to carry out tests several times with the same stage combinations. Thereby an average value can be calculated, and the influence of measurement errors can be reduced. As an alternative, if a lot of experiments are executed in the same environment, a single randomly chosen iterative can be conducted several times, presuming the error for the holy system. To limit the testing time in this thesis, the latter approach as chosen for this work.

Main technical terms in DoE are:

- Factors: parameters varied in the experimental design
- Levels: discrete values of a factor specified in the experimental design
- Effects: Indicate the effect of a factor on the system.

The most common and at the same time probably most simple methods for DoE are the Full-Factorial Design and Partial-Factorial Design or, to name only one of many more detailed methods, e.g. the Response Surface Design (also called Response Surface Method, RSM).

The former represents an experimental plan, in which every possible level-combinations of every single factor is tested. Thus, the whole system can be analyzed and any correlations between input factors discovered. The test effort n_r results from the number of factors n_f and the number of levels n_s :

$$n_r = n_s^{n_f} \quad (5-2)$$

The Partial-Factorial Design, as indicated by the name, only test previously defined levels and therefore not all possible settings. Knowledge gained is afterwards transferred to the whole system, which is particularly advantageous by a high number of input factors.

The main idea of RSM is to use a sequence of designed experiments to obtain an optimal response or to determine very significant inputs. Basically, the term RSM can be understood as a subtopic of DoE including a collection of statistical and mathematical methods by itself.

5.3.2 Adaption to the scenario

Since the number of variable parameters for the main series was determined to two, namely the inclination angle of the target satellite and the angle of incidence of the sun simulator unit, the scope of the study still had to be specified. The rotation velocity is not considered for the test planning, since its variation does not result in an increased testing effort, as already mentioned, but can be changed afterwards.

Therefore, the initial situation is having 2 factors, that had to be analyzed on various levels. Since it was not the main goal to find a single optimum for both parameters, but to point out interrelations between the illumination condition and the accuracy of the pose estimation, a Partial-Factorial Design was no option. Due to the many levels of each factor, a Full-Factorial Design had to be combined with a rather classic approach of One-factor-at-a-time (OFAT) analysis. Thus, it was possible to fully examine the system and at the same time to create a data bank with a detailed overview of different lightning conditions in the RACOON-Lab.

Considering all of that, the study was basically divided into two parts:

- detailed, isolated variation of the inclination angle of the target as well as of the angle of incidence of the sun simulator unit
- coherent analysis of both target inclination and angle of the sun together

Thereby, the main goal of the first stage was to get a deep understanding on how each of those factors interfere individually with the attempt of a pose estimation performed with the help of a stereo vision camera. For being able to do so, iterations had to be selected as little as possible, resulting in a change of 3 degrees per experiment for both sunlight indication angle as well as inclination angle of the target for each experiment. Due to the environmental conditions of the hardware facility, even finer iterations were not considered to mandatorily result in a better understanding of the system and clearer results.

In the second stage, due to varying both factors simultaneously, any possible interrelations between each other could be identified, since both parameters have an impact on the intensity of surface reflections. For this, 10 levels for the sun and 7 levels in case of the target were considered to provide sufficient results, resulting in an iteration of 15 degrees per experiment for each factor.

The scope of the variations however was limited by the facility. Aiming to cover an experimental range as wide as possible, in both study parts a span of 0-90 degrees was defined in case of the inclination of the target and even 0-135 degrees relative to bot chaser and target for the angle of the sun simulator unit. The idea behind this was to also have a backlight scenario, with the sun positioned not behind the sensor but vis-à-vis, generating even more challenging conditions.

With (5—2), this resulted in 147 experiments that had to be conducted, several reference tests excluded.

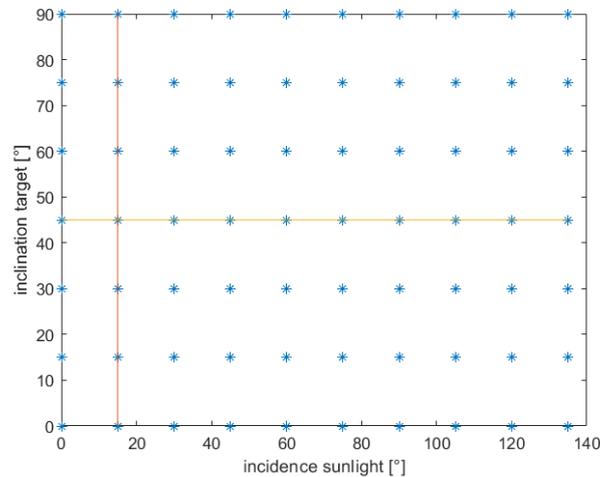


Fig. 5—13: range of experiments; the colored lines represent the detailed, isolated variation of parameters, whereas the blue points show the coherent analysis

As seen in Fig. 5—13, which illustrates the range of the experiments conducted, for the isolated observation of both inclination of the target and the position of the sun simulator unit, the particular other parameter had to be set to a constant value as well.

While analyzing the incidence of the sunlight, it was decided to tilt the target satellite by 45 degrees, since experiences made in the preliminary series led to the conclusion, that surface reflections on that setup were balanced, therefore promising good results. At the same time, an angle of 45 degrees left a bigger latitude for evaluating. The same goes for the angle of incidence of the sun. The pretest series indicated good results for a rather small angle of incidence relative to the camera. With the camera being directly behind the chaser satellite, the hardware casted a little shadow on the target satellite. By rotating the sun by 15 degrees, shadowing could be avoided.

5.4 Main study

After all constant settings for the main series of tests were defined, the detailed parameter study is now presented. Its goal is to analyze the impact of different illumination conditions on the quality of an optical pose estimation of a target satellite in LEO. By using a stereo vision camera, at the same time an estimation on the applicability of the camera in that specific scenario can be made. Furthermore, the challenges, which came up during both pretest and main test series are explained. These empirical values are intended to provide a basis for further work within the HIL facility.

5.4.1 Recording software

The original plan was to record required images with a modular software framework called RVD SLAM developed in another student thesis before this work, as already mentioned. This framework implements a real time capable toolchain to perform all necessary parts of 3D object reconstruction, including tracking the targets relative attitude and position and generating a trajectory out of the data input provided from a sensor. Shortly before this study was conducted, Stereolabs ZED SDK 2.4 was released. Said software framework however only was implemented for SDK 2.0 at that

time. Integrating the latest update into the framework turned out to not being possible in the given time. Since this update came with improved depth sensing as well as performance, it was decided to manually record the data with help of the ZED intern recording tool. Thereby, videos are stored as a svo file, which is the proprietary data format provided by Stereolabs for this purpose.

For evaluation purpose, the recorded.svo files had to be converted into a RGB-D dataset, which then can be used by the software framework again. The converting script however is possible to generate the pose trajectories estimated both by the ZED API as well as the DIFODO algorithm, too. The latter still had to be implemented into the software though. Since this work only deals with the tracking and not the mapping of objects, produced trajectories are very sufficient for this study. Therefore, the RVD SLAM software framework did not have to be used, as even the described boxfilter for the important image pre-processing could be integrated into the converting tool. In the time between the execution of the study and evaluation, even ZED SDK 2.6 was released, which then also was used for the Evaluation.

5.4.2 Experiment setup and execution

The previously described scenario now was carried out in a detailed test series. As already mentioned in 5.1.1.3, the control of the hardware facility through the LabVIEW GUI for each axis is displayed in motor steps.

Tab. 5—10: Constant steps set for the detailed parameter study additionally to its conversion factors (CF)

Axis	Definition	Set steps	Existing CF	Used CF
0	C trans. X	195000	85600	
1	C trans. Y	30000	85600	
2	C rot. Z	-450	24430	
3	C trans. Z	-160000	226250	
4	C rot. Y	0	24430	
5	C rot. X	-77350	24430	
6	T C-sled rot. Z	-60000 to 60000	24430	24890
7	T C-sled rot. Y	*	161000	190986
8	T rot. X	-5500	144000	
9	T rot. Y	300	24430	
10	T rot. Z	77000	24430	
11	S trans.	*	106666	160500
12	S rot.	*	32000	32000

Tab. 5—10 shows the basic settings of chaser (C), target (T) as well as the sun simulator unit (S) adjusted for the study. Since the target satellite does not have any translational DOF, the relative distance between the latter and the chaser (and thus the sensor) is defined through the axis 0. Axes 1 and 3 are therefore responsible for the horizontal and vertical alignment of the camera, whereas the set steps of axis 2 guarantee parallelism of the camera-CS to the Optitrack-CS. This facilitates any measurements conducted with help of the Optitrack System. Since both sensors are attached to the servicer in a opposite matter (see Fig. 5—5), Axis 5 enables the use of the ZED stereo camera by rotating the whole unit by 180 degrees around its X-axis. Axes 8-9 were used to orient the target, setting the normal vectors of the mockups surface planes congruent to the camera-CS. This orientation was thus the initial position for the variation of the targets inclination.

The simulated fly-around was accomplished by letting the target rotate around axis 6, ensuring the rotation axis to be constant in every experiment, even after tilting the mockup. However, this resulted in not being able to drive a full turn of 360 degrees, since the sled would cover the target at some point of the rotation, as mentioned before. An angle of about 270 degrees proved to be the biggest angle possible being covered, without the sled interfering with the recording. Additionally, to guarantee a constant rotation velocity over the same angle of rotation, the distance, which was needed by the sled to accelerate, was added to the rotation.

Marked axes in the table above were the variables and therefore changed after every single experiment, namely axis 7 in case of the target inclination and axes 11 and 12 in case of the position of the sun simulator unit and thus the angle of the illumination incidence.

Summarized, to be able to execute the study as planned, conversion factors of the steps to common units like meters or, in case of a rotational movement, degrees/radian are mandatory for axes 6,7 as well as 11 and even 12. Furthermore, the relative distance between chaser and target had to be determined with a sufficient accuracy. Since reliable conversion factors were not integrated into the software yet, existing values had to be verified by ancillary measurements, that were conducted both manually as well as with help of the Optitrack System. The determined and used conversion factors *steps_to_m* / *steps_to_rad* are shown in Tab. 5—10 alongside the existing ones.

As described in 5.3.2, the study basically was divided into two parts, differentiating between an isolated analysis of both inclination of the target as well as the angle of incidence of the sunlight and a coherent view of both parameters. Considering this, two different scenarios had to be looked at.

First, the orientation of the target had to be constantly varied. During that time, the sun simulator was at a fix position, focusing the rotational axis of the mockup. To iterate through the inclination angle of the mockup, simply the determined *steps_to_rad* factor had to applied on corresponding axis. Fig. 5—14 illustrates the progress in the orientation.

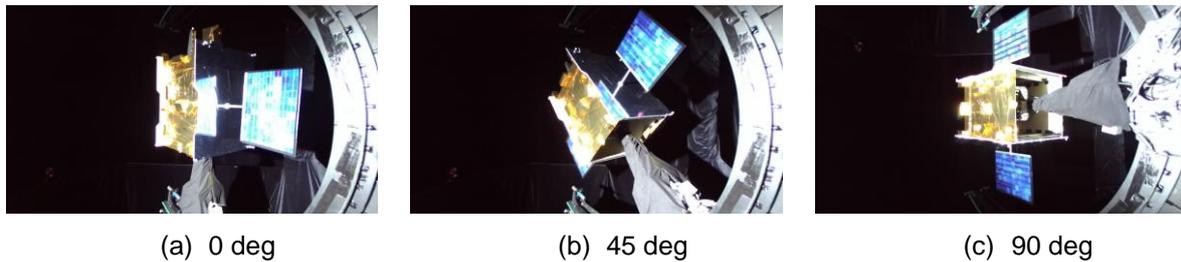
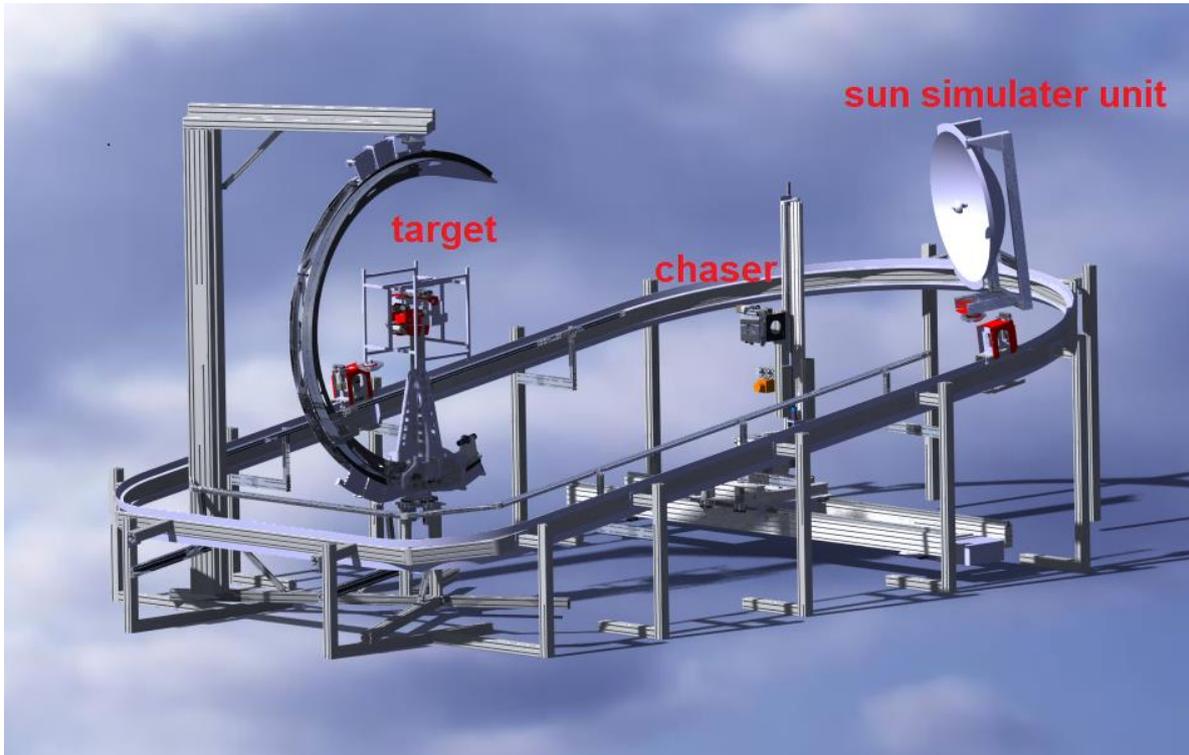


Fig. 5—14: variation of the targets inclination angle as seen by the ZED

However, to be able to variate the position of the sun simulator unit in a reproducible manner, first, a detailed look at the positioning of the single components of the HIL facility had to be taken, especially on the geometry of the path leading around it. For this purpose, amongst other things the CATIA model of the laboratory was used. Said model is also illustrated in Fig. 5—15a. Thus, the exact form of the rail and its pose relative to the target could be determined. Measurements taken from the CATIA model, particularly, the relative positioning of individual components to each other, were verified as well, since an exact transfer of said model in actual hardware could not be guaranteed. In case of greater deviations from the theoretical value, the self-measured value was used. In the following, corresponding values are mentioned explicitly, whereas others were taken from CATIA.

Furthermore, for both axes belonging to the sun simulator, no step conversion factors in meter/rad were existent. Since the unit can only move on the rail around the main facility, it is not within the field of view of the Optitrack System. As a result, its exact position could only be determined manually. Also, no reference point is implemented for those axes yet, meaning errors during the drive of the suns translational or rotational axis can only be spotted very hard and its correction only possible with “eyesight” accuracy.

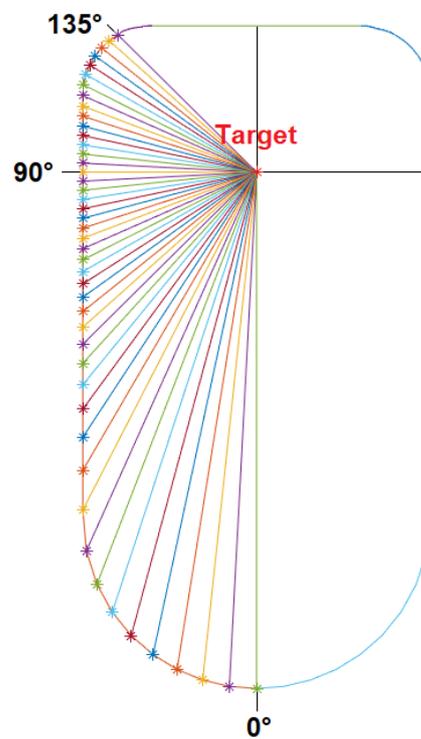
With this information, the position of the sun simulator unit on the rail as well as its angle relative to the camera and target satellite could be determined. A top view of the facility, highlighting its surrounding path is seen in Fig. 5—15b (the positioning of single components illustrated do not represent the experimental setup). An angle of 0 deg was defined at the very end of the rail with the sun being congruent with the X-axis of the target, also being directly placed behind the sensor, whereas an angle of 90 deg was set on targets Y-axis, therefore the sun being besides the mockup. The rotation on the illustrated geometry around the thus decentered body was possible with simple vector addition. Thereby the geometry was segmented into its different parts, at which the iteration process was applied. Afterwards the from was set together again. The results of the calculations are shown in Fig. 5—15c, illustrating every single setting of the sun positioning on the rail during its isolated analysis. However, these calculations first were made for points on the rail.



(a) isometric view



(b) top view



(c) exact model of the surrounding rail, displaying the positions of the sun

Fig. 5—15: CATIA model of the HIL facility including a visualization of the variation of the lightning incidence for the isolated analysis.

Since the rail around the facility also is elevated in relation to the horizontal plane and the actual lamp of the sun is mounted on a motor, leaving it high above the rail (see Fig. 5—7a, Fig. 5—15a), the error in the position caused by that had to be considered as well. On the lateral straight path, this was achieved by simple trigonometry, whereas the error on the rounded geometry parts was linearly approximated.

With the sun being rotated by 135 degrees while moving on the rail during the analyzation of different lightning conditions, in order to get usable results, the horizontal orientation of the light beam had always been oriented onto the rotational axis of the target satellite. Due to the missing linking of the translational and rotational axes of the simulator, corresponding adjustments had to be conducted by eyesight between every experiment. To be able to detect the exact orientation of the light beam, a laser pointer was attached to the simulator unit right beneath the lamp, which made it possible to align the ray reproducibly. By additionally applying a switch to the laser, the laser did not interfere with the recordings. However, this was only possible by switching the laser pointer on and off after each iteration, making the study more effortful. Thus, besides a not referable accuracy, this is only thought as a temporary solution. An improvement for described problem, amongst other things, is further presented in chapter 8.

Due to the missing *RefSearch* mode for both axis 11 and 12, the initial point at 0 deg also functioned as a reference point since the hardware environment made it possible to verify the position on the trail as well as the orientation of the simulator unit at this point very well. Therefore, after adjusting corresponding axes, the step count of both axes was manually set to zero, starting to count the motor steps during the study from this point on the trail. Fig. 5—16 shows that specific position on the trail of the hardware facility. The congruence of the rotation axis of the middle wheel and the connecting line of the trail construction must be taken into account.



Fig. 5—16: Defined “point zero” of the sun simulator unit

Adapting the vertical orientation of the sun simulator however, since the elevation of the rail caused height differences between the lamp and the target during the test series, was not considered. Due a missing motor for that axis, it could only be

controlled by hand, which would have made a sufficient accuracy and therefore reproducibility impossible.

The same described procedure was done in the second part of the study, only varying sun and target simultaneously.

Furthermore, to be able to indicate the static measurement uncertainty, 10 reference experiments were conducted, all with the same randomly selected setting for both sun and target. Conveyed to the hole system, it was possible to determine the statistical deviation, that the facility is subject to.

A detailed summary of all the parameters of the performed experiments can be found in the appendix, alongside with the settings for each axis for every experiment conducted.

NOTE: While conducting the isolated variation of the sun position, a measurement error was detected, causing the systematically changed angles to be slightly off accuracy. Due to measurement deviations between the CATIA model and the real hardware facility, the variation angles were calculated with a wrong distance between the target satellites rotational axis to the surrounding trail in facilities X-direction, therefore changing the rotational center point. As a result, the rotational angle of 3 deg is subject to an error of +0.19 deg per iteration. However, this error is gaining importance until the 90 deg mark, before falling off in the same pace again, leading to a maximum error of +5.69 deg after 30 iterations. For evaluation purposes though, said deviation was not considered to have a relevant effect on the quantification of analyzed parameters, since it is justifiable with regard to the possible accuracy of the whole system.

Nevertheless, the measurement error was resolved for the further course of the test series, thus, reference recordings could be conducted, allowing to still make a qualitative statement on the quality of pose estimation for very crucial angle settings like 90 deg.

The exact impact of the error on the isolated analysis of the incidence angle of the sun for each angle can be also found in the appendix.

6 Evaluation

After the successfully conducted main test series, this chapter provides an evaluation of the results. As already mentioned, the uncertainty of the hardware facility was determined by recording ten reference trajectories with the exact hardware as well as software settings. The incidence angle of the sun and the inclination angle of the target were randomly selected within the range of experiments. The resultant relative standard deviation could then be transferred to the whole system and, at the same time, being linked with the metric modules. Thus, the testing effort could be limited enormously. The results can be seen in Tab. 6—1.

Tab. 6—1: Standard deviation for each metric module

Metric module	Relative standard deviation				
	C_F	r_F	ε	R	$\delta_{i,max}$
DIFODO	±28.17%	±18.30%	±16.93%	±05.91%	±14.54%
ZED Tracker	±10.44%	±12.52 %	±05.29%	±06.79%	±10.28%

Now, both parameters analyzed are, with help of the presented metric, individually linked to the quality of an optical pose estimation and therefore 3D object reconstruction. Furthermore, any correlations between the angle of incidence of the sunlight and the orientation of the target are worked out. By post-processing the image data, different angular velocities could be simulated. Therefore, the impact of a varying information density handed to the algorithms could be analyzed. Even more, the tracking algorithms used in this work are compared, thus providing an outlook for improvements.

However, for the sake of clarity, both hardware parameters are fully evaluated only for an angular velocity of 1 deg/s. Besides the general impact of the rotational velocity on the quality of pose estimation, any correlations between the information density and both the incidence of the sun as well as the inclination of the target are thereupon analyzed separately.

Thereby, at first, respectively for both tracking algorithms, an analysis of the numeric result output of the metric is presented, before single modules are evaluated in more detail, focusing on the error in the estimation of the linear deviation (Residuum) and on the drift in the estimation of the angular position, as these two factors proved themselves to be most likely responsible for the errors in the object reconstruction in previous work. [4, 17–19] A comparison of both algorithms is summarized afterwards. During the evaluation, it also turned out that reliable statements of influencing factors regarding the drift could only be made by analyzing its course, especially for DIFODO. Reasons for that as well as a discussion about the importance of the other metric parameters can be found in chapter 7.

An overview of all the generated data for each part of the study is also appended. Furthermore, for clarity reasons, in the following a specific incidence angle of the sun of e.g. 15 deg is also referred as S_15, whereas an inclination angle of the target of e.g. 60 deg is displayed as T_60. While varying both parameters simultaneously, a

specific setting is expressed as 15_60, where the first angle is referring to the sun and the second to the target.

6.1 Angle of incidence of the sunlight

Previous studies within the RACOON-Lab regarding the pose estimation of a satellite, by also using DIFODO as vision-based tracking algorithm, determined a significant effect on the quality of object reconstruction due to a different incidence angle of the sunlight. A positioning of the sun simulator behind the recording sensor led to a qualitative much better result than a lateral positioning, assumed to be caused by more critical reflections on the satellites surface (see section 5.2).

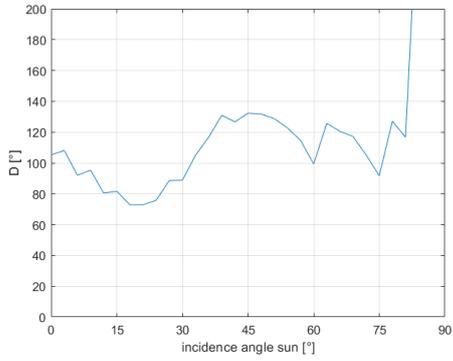
For that to be verified, the incidence angle of the sun was constantly varied, by letting the target a stable angle of 45 deg. However a sun angle of 135 deg was experientially covered, results showed that especially from an incidence angle higher than 90 deg, the quality of the generated trajectories is subject to an uncertainty. Besides all metric modules, especially the drift is occasionally deflecting strongly. In order to present reliable data concerning the error in the pose estimation, backlight recordings are not considered for now. Also, the measurement error previously explained must not be neglected.

Fig. 6—1 gives an overview of the trend of all metric modules, comparing both used tracking algorithms at the same time. However, it should be noted that for the drift D the always maximum drift angle $\delta_{i,max}$ during a fly-around is illustrated, since the "damage" that was done at a certain point to the trajectory cannot be repaired, whereas the Residuum R is a mean value. C_F , r_F and ε are just plain flaw sizes. (see chapter 4.3.2)

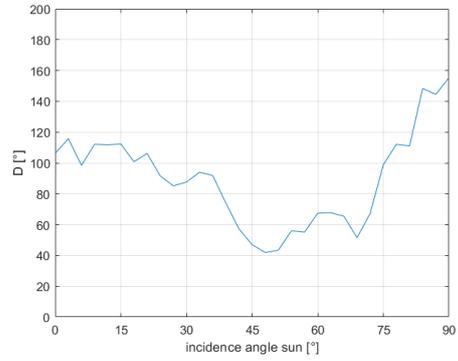
When using the DIFODO algorithm for trajectory estimation, with an increasing incidence angle of the sun nearly all metric parameters are growing as well, indicating a higher deviation of the estimated trajectory to its ground truth trajectory. Only the Mean Residuum does not allow to make a well-founded statement regarding its trend, since its value oscillates around its total mean of 0.17 m. Also, the maximum drift angle $\delta_{i,max}$ is starting with an averagely value an angle of 0 deg, before falling off and rising again from about 30 [+1.90] deg to almost constant value with little fluctuations.

DIFODO

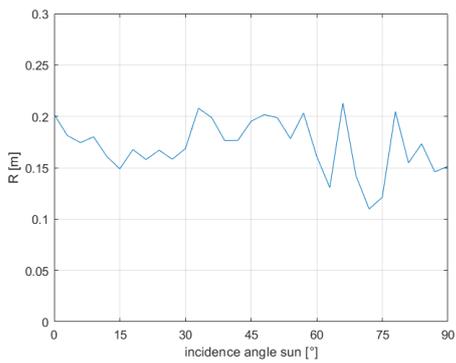
ZED Tracker



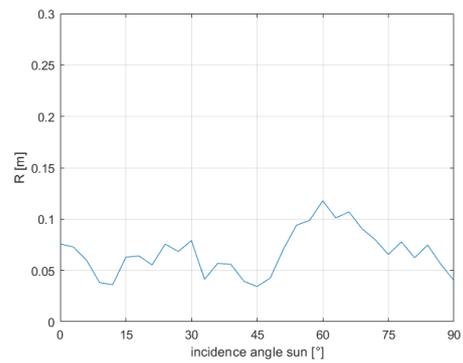
(a)



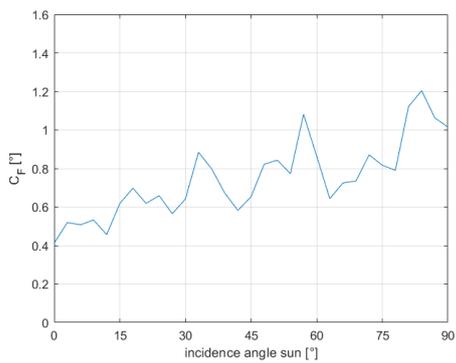
(b)



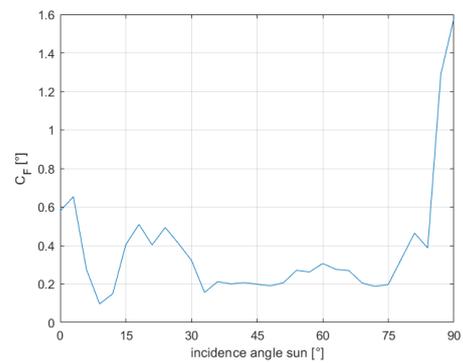
(c)



(d)



(e)



(f)

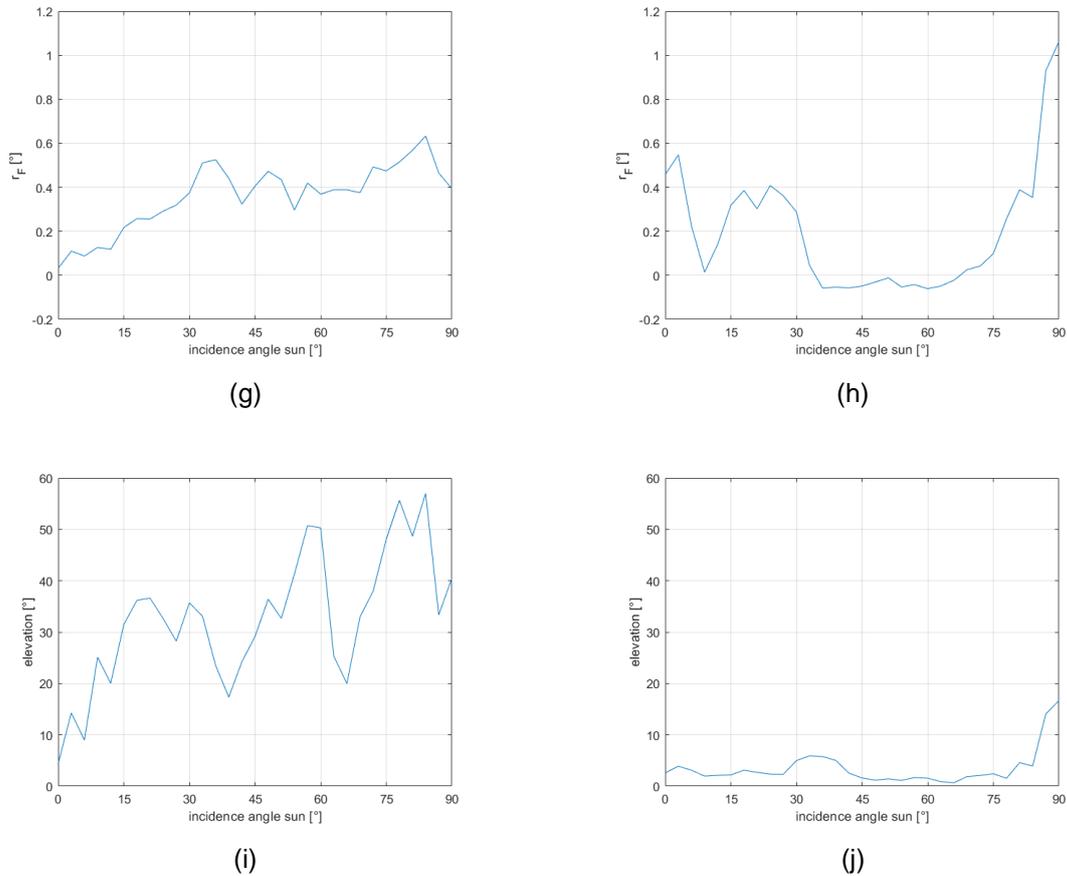


Fig. 6—1: Course of the five metric parameters during the variation of the incidence angle of the sun (angles higher than 90 deg excluded), displaying results of the DIFODO algorithm on the left and of the ZED Tracker on the right

By taking a qualitative look at the course of the drift for each fly-around, it stands out, that that not only $\delta_{i,max}$ is growing nearly constantly with a higher light incidence angle, but also the “smoothness” of the drift course this decreasing, as seen in Fig. 6—2. The slightly higher values for $\delta_{i,max}$ for angles up to 15 [+0.95] deg are interpreted in section 7.1. Furthermore, in all illustrations at some point a rapid increase of the drift compared to rest of the course (in the following also referred as “drift-flares”) can be detected, in some settings more pronounced as in others. This characteristic finds its peak at Fig. 6—2b, with even two such flares noticeable. For angles from 84 [+5.32] deg, $\delta_{i,max}$ increases to not usable values.

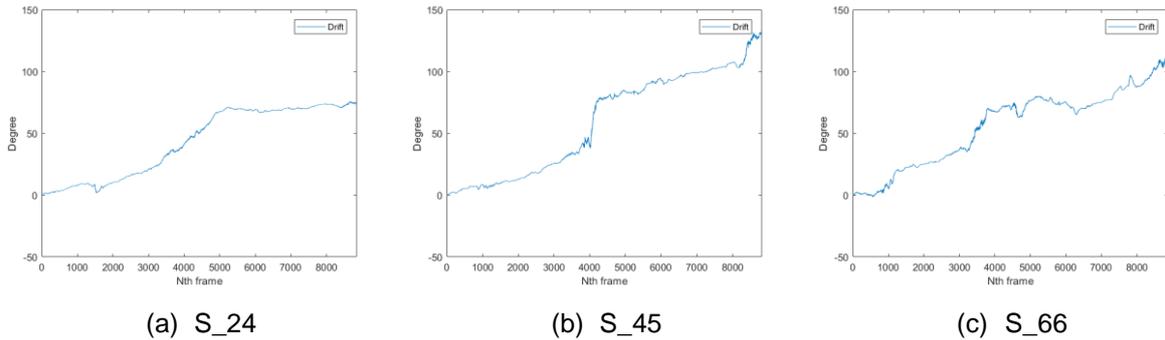


Fig. 6—2: Development of the drift course with an increasing incidence angle of the sun (DIFODO)

The course of the Residuum for each experiment conducted does not indicate a specific trend, however, at angles between 51 [+3.23] deg and 63 [+3.99] deg the estimated trajectory starts with a rather big translational deviation to its bestfit-circle, as seen in Fig. 6—3a. This error drops very quickly at all settings though.

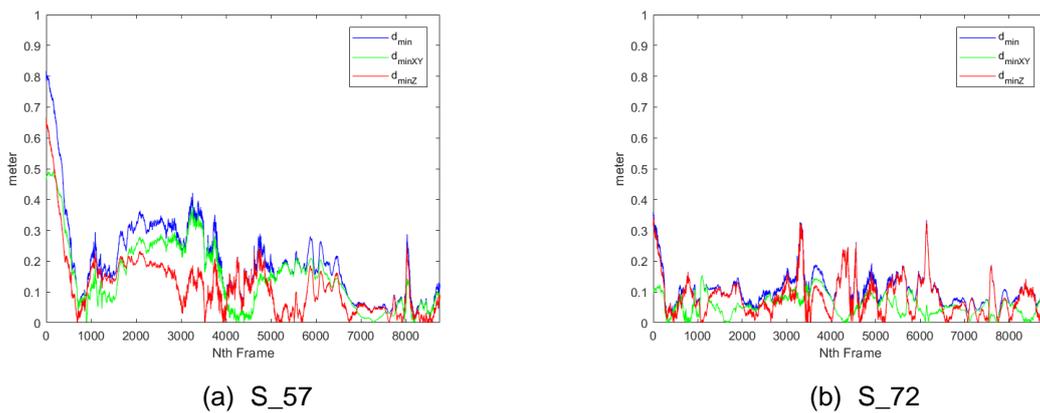


Fig. 6—3: Deviations in the course of the Residuum (DIFODO)

On the other hand, while analyzing results generated with help of ZED intern tracking algorithm, a comparable good outcome for every metric module was detected for angles between approximately 30 [+1.90] deg to 75 [+4.75] deg, with the exception of R .

Nevertheless, in case of the drift course, characteristic phases for the range of the varying angle of the sun can be found again, shown in Fig. 6—4. At comparatively small angles, the drift develops first in negative direction before bending sharply into a nearly linear line with positive slope at around 3000 frames. For angles even smaller (0 to 21 [+1.33] deg) the orientation of the drift course starts with an positive angle before breaking into the in Fig. 6—4a illustrated course, in which described phenomenon can already be estimated. Afterwards, the drift drops constantly, delivering the best generated results for an angle of 48 [+3.04] deg, before slowly increasing again. With a growing angle, the course is also getting more uneven.

The course of the Residuum, similar to results of the DIFODO algorithm, is not further demonstrative. However, between 54 [+3.42] deg and 72 [+4.56] deg, a spike of d_{min} can be detected at around 6000 frames, being responsible for the higher R values.

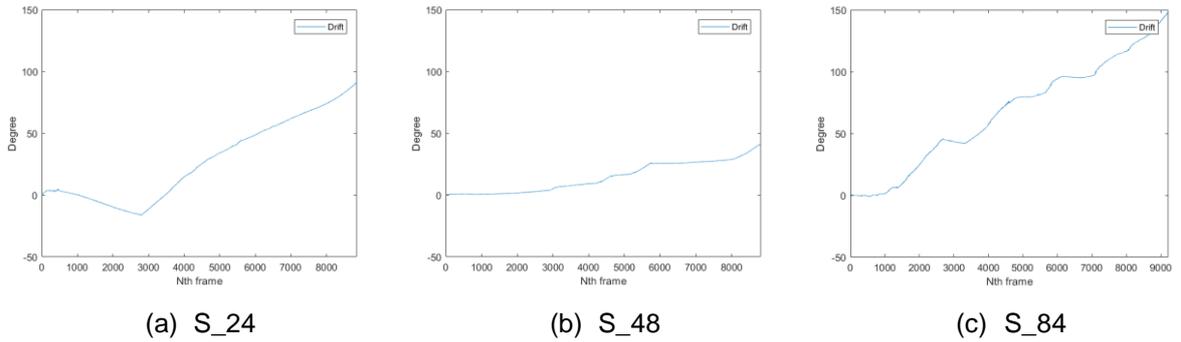


Fig. 6—4: Development of the drift course with an increasing incidence angle of the sun (ZED Tracker)

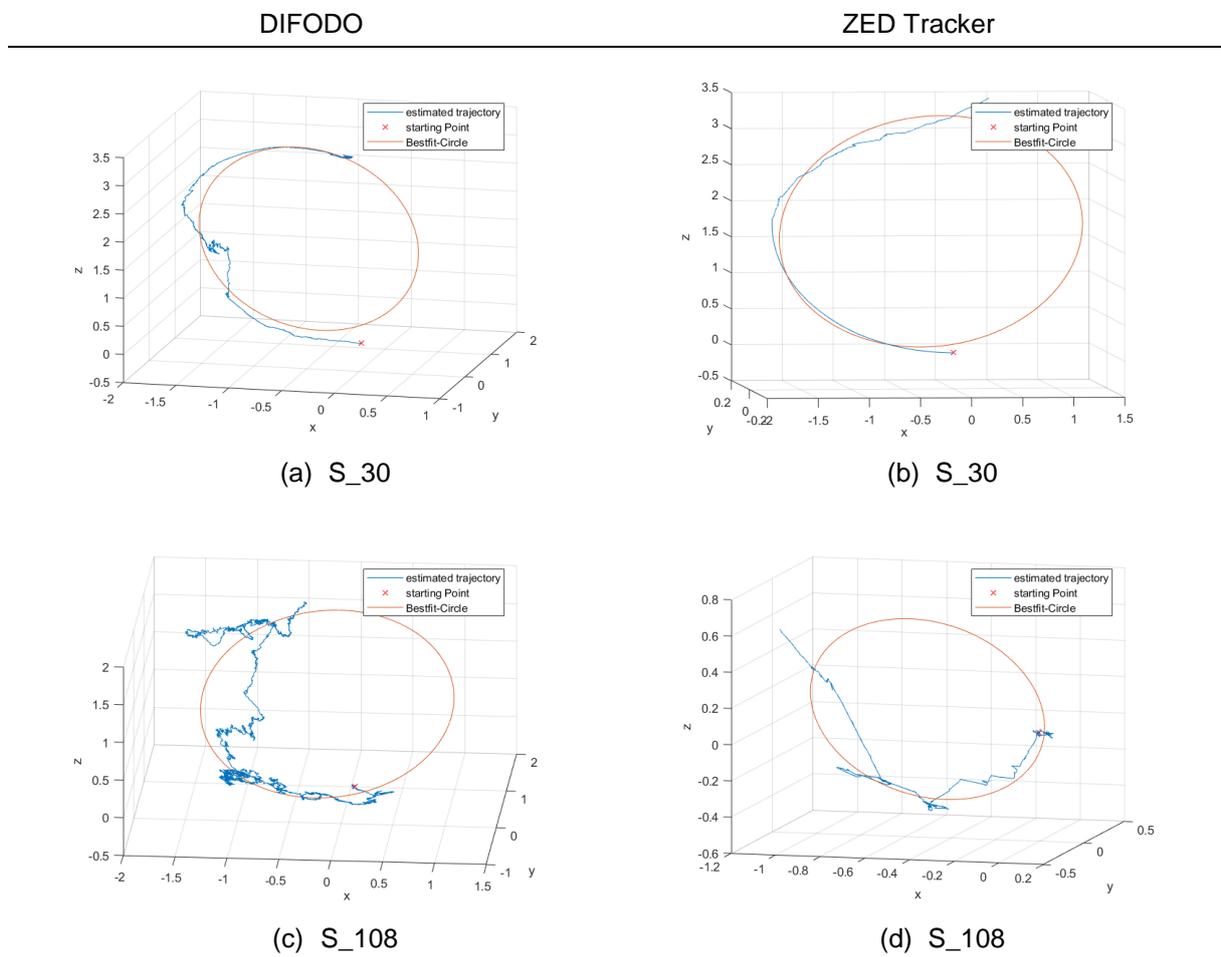


Fig. 6—5: Comparison of estimated trajectories for different settings

Now, considering sun angles higher than 90 deg, for both algorithms, it is noticeable that all metric modules are increasing. Especially the maximum drift angle $\delta_{i,max}$ occasionally deflects strongly, starting for DIFODO at even 84 [+5.32] deg, whereas the ZED Tracker delivers stable drift values up to 105 [+3.80] deg. However, these high values are inconsistent, and no pattern could be noticed yet. Yet, regardless the outbursts, reasonable trajectories still can be generated by both algorithms, even though enormously lacking in quality compared to compared to lower incidence angles, as seen in Fig. 6—5. Nevertheless, angles between 111 [+3.42] deg and 120 [+1.90] deg seem to be the limit for a practical pose estimation. An explanation for describes phenomena can be found in chapter 7.1.

6.2 Inclination angle of the target satellite

To make a reliable statement regarding the impact different illumination conditions on the quality of pose estimation, the attitude of the satellite mockup had to be considered as well since different tilt angles of the target also indicate a different incidence angle of the light.

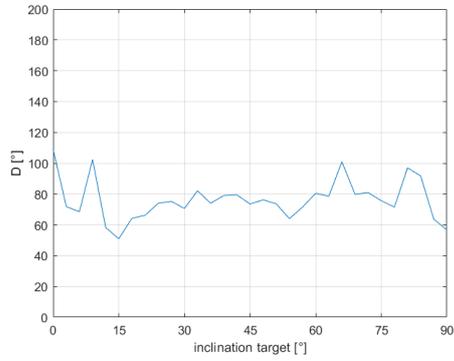
Previous work on this topic as well as first results of the preliminary study led to the assumption, that a tilted target would result in a better reconstruction of the target due to minor reflections on the silvery satellite surface. For that to be verified, the targets inclination was constantly varied, while the sun simulator unit was at a steady position of 15 deg.

Fig. 6—6 allows to make a prediction regarding the trend of the quality of pose estimation for both algorithms again, by plotting the results of the metric modules for each experiment over the whole range.

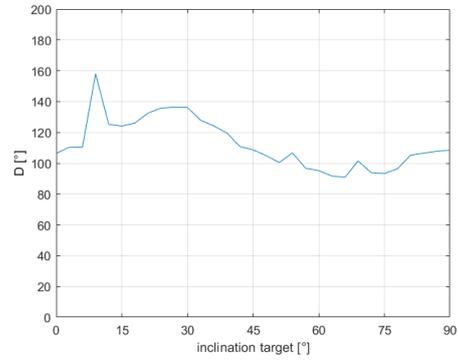
While looking at the results generated by the DIFODO algorithm, Fig. 6—6a/c are showing that both the maximum drift angle $\delta_{i,max}$ as well as the Mean Residuum R are at their lowest values from about 12 deg to 57 deg, with having much higher values at very little of very high inclination angles, although R has a nearly constant value until 60 deg, with the exception of an outlier at 3 deg. In Fig. 6—6c with an increasing angle lightly decreasing values for C_F be seen, whereas a trend for r_F and the elevation ε is difficult to spot, however, the lowest values for r_F can be achieved for angles between 48 deg and 66 deg.

DIFODO

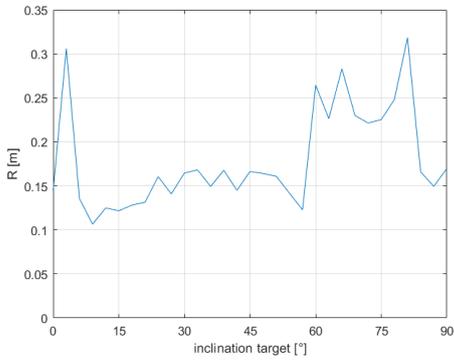
ZED Tracker



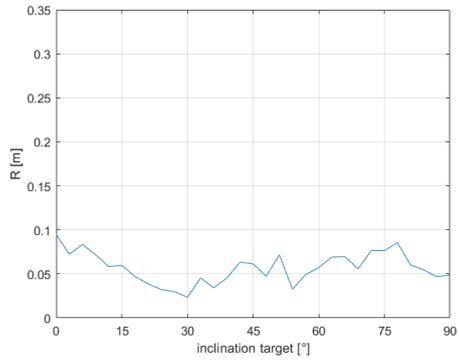
(a)



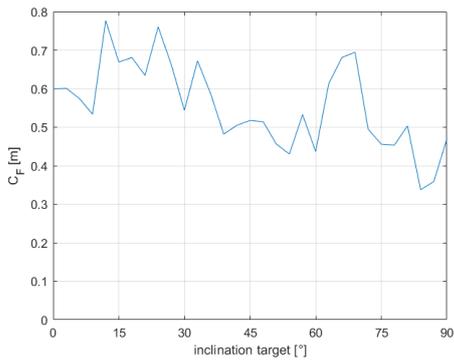
(b)



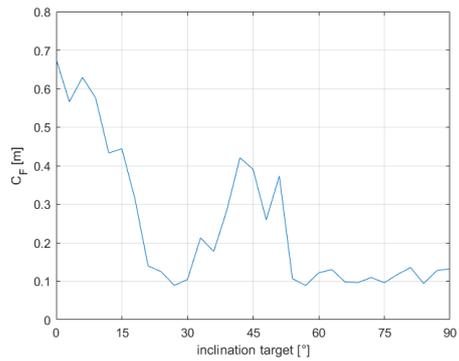
(c)



(d)



(e)



(f)

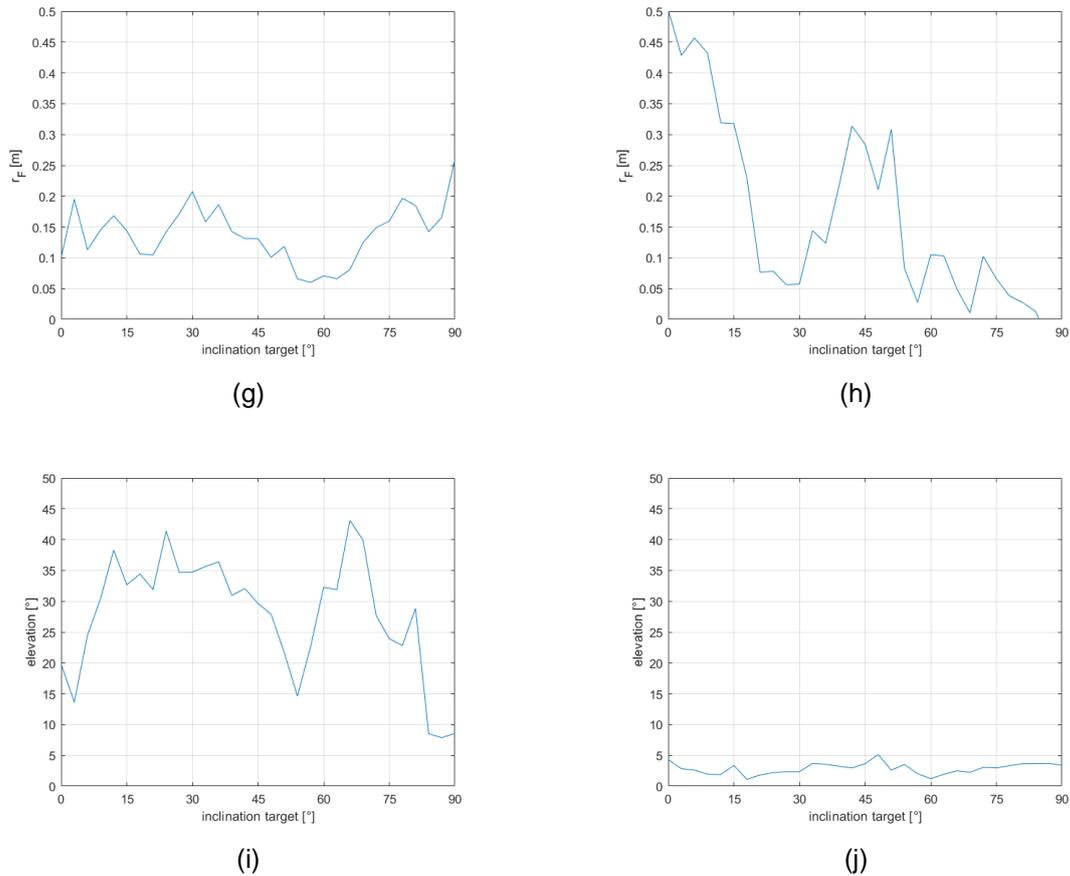


Fig. 6—6: Course of the five metric parameters during the variation of the inclination angle of the target, displaying results of the DIFODO algorithm on the left and of the ZED Tracker on the right

Again, more information is provided by the analysis of the courses of both D and R , illustrated in Fig. 6—7. Thereby, especially the huge drift flare in Fig. 6—7a at about 4000 frames is striking. Yet, this sharp cut in the drift course is only seen for very small inclination angles of the target. However, it is nearly linearly decreasing in the further course of the study, until being almost not visible anymore at tilt angles between 30 deg and 54 deg. Beginning at 60 deg, the drift starts again to grow, peaking at an angle of 81 deg. The graph differs from the first due to a more even gradient. This “dome” in the course of the drift, which can be seen in Fig. 6—7c, however, becomes less noticeable after for further tilting.

The course of the linear deviation for each experiment behaves like expected from the mean results gathered. No big differences can be determined up to 60 deg, before R is increasing with nearly the same pace as the drift. These high values even remain unchanged up to 84 deg. Yet, it is noteworthy, that the linear deviation in Fig. 6—7d grows simultaneously with the angular deviation caused by the drift flare at a tilt angle of 0 deg.

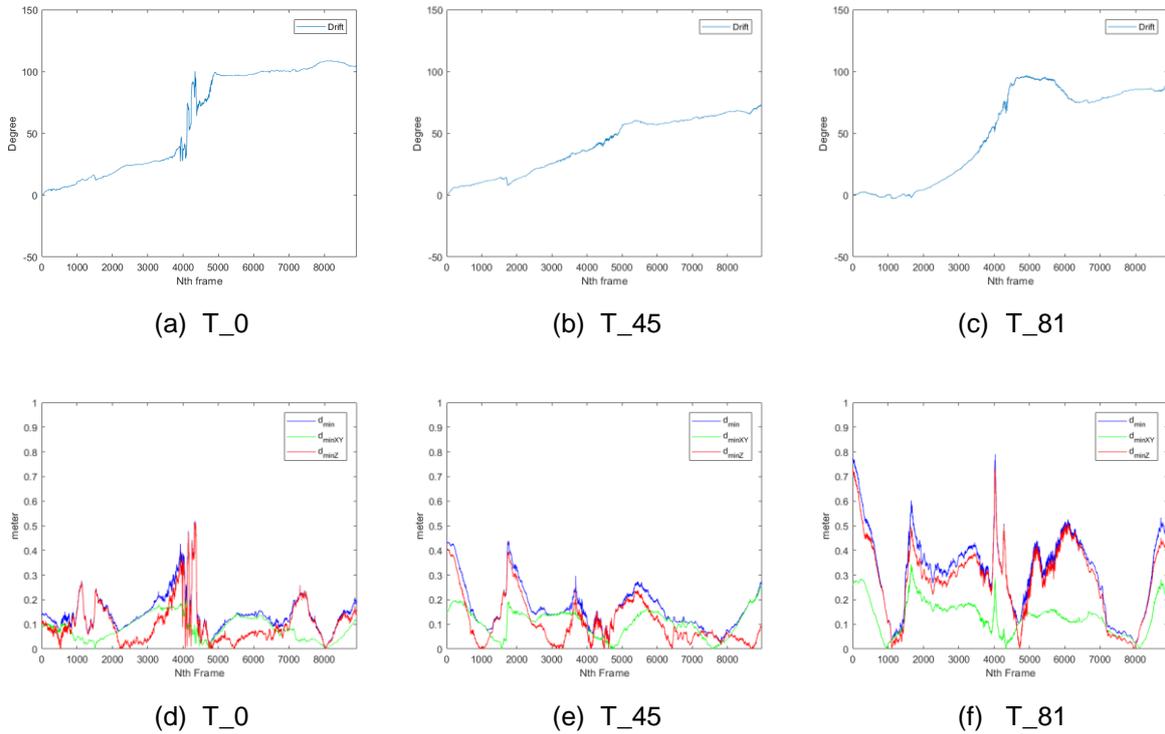


Fig. 6—7: Development of the courses of drift and Residuum for an increasing inclination angle of the target (DIFODO)

On the other hand, again there are fundamental differences between the results of both tracking algorithms. For the ZED Tracker, lowest values for $\delta_{i,max}$ can be found from 45 deg, while a clear statement to the trend of the Residuum cannot be made. This also applies to the elevation, yet the average angle between the calculated bestfit-plane of the estimated trajectory and the reference plane is comparably little. The courses for C_F and r_F are both looking similar, with values being really high for very little tilt angles as well as for angles around 45 deg, but at the same time really low for angles around 30 deg and angles higher than 54 deg (Fig. 6—6f/h).

However, with a stable inclination angle of the sun simulator, during the process of varying the tilt angle of the target, the course of the drift generated with the ZED tracker of a single experiment is only barely changing. As seen in Fig. 6—8c, for comparably small inclination angles, three major breaks in the otherwise almost linearly running drift course can be detected. Yet, as the tilt angle grows, the first irregularity is constantly smoothing out, while the further development of the drift stays equal. Solely the slope of the graph is decreasing slightly, resulting in an overall smaller $\delta_{i,max}$ for higher tilt angles. (Fig. 6—8f)

Despite the Mean Residuum is staying on a nearly constant low value during this part of the study, a change in the development of the linear deviation can still be determined. With the minimal distance d_{min} between estimated trajectory and its bestfit-circle being at an averagely stable value, for angles from 48 deg the error distribution changes to nearly no linear deviation up to 4000 frames before it is spiking up and peaking at 6000-7000 frames with d_{min} being about five times higher than the mean. (Fig. 6—8b/e)

All this considered, this development can also be seen in the generated trajectories, with Fig. 6—8d showing smaller deviation in the angular position but a higher linear deviation for the last third of a simulated fly-around than Fig. 6—8a.

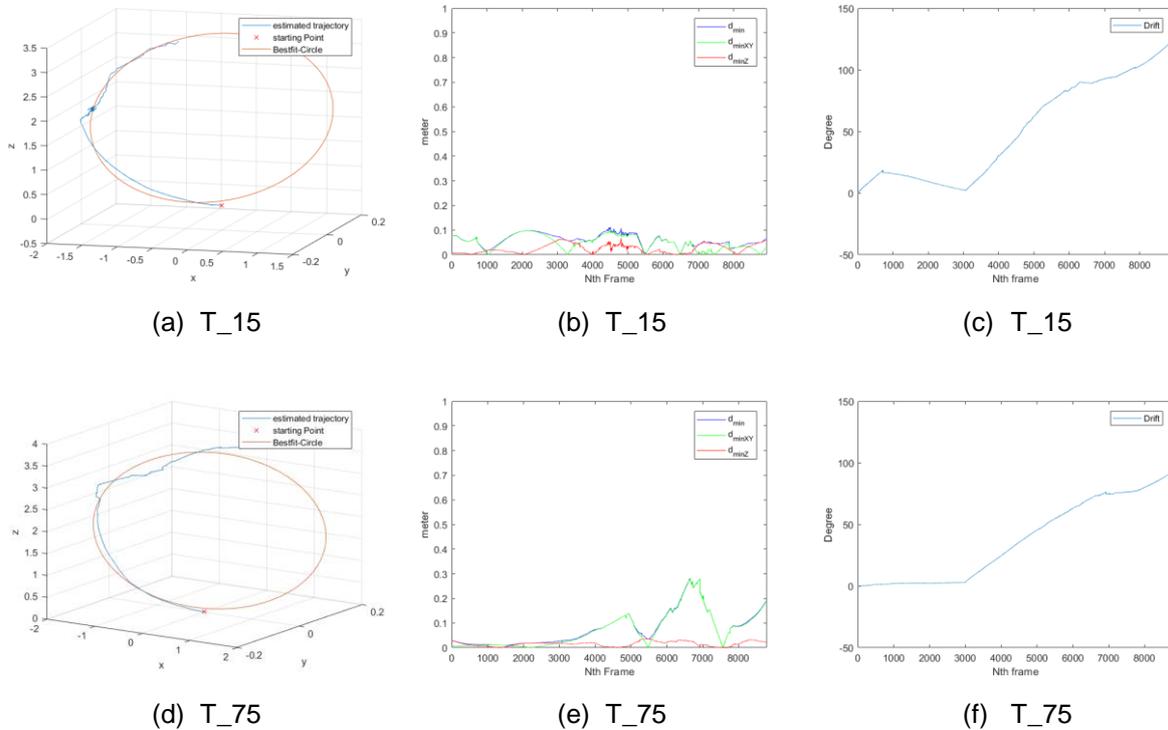


Fig. 6—8: Development of the courses of drift and Residuum for an increasing inclination angle of the target (ZED Tracker)

6.3 Correlations between the parameters

So far, the impact of different lightning conditions on the pose estimation of the target satellite was respectively analyzed by an isolated consideration of each parameter, namely the incidence angle if the sun as well as the tilt angle of the target. However, possible interrelations between those factors were not reflected upon, so far.

Thus, to be able to make a coherent and complete statement regarding the influence of different illumination conditions, it was necessary to variate both factors simultaneously over the whole range. For this purpose, the variation angle for each parameter was increased from 3 deg to 15 deg.

Although in the test series angles from 0 deg to 135 deg in case of the sun variation were covered again, for now, the same measure is taken as before, excluding angles higher than 90 deg for the general evaluation to guarantee a reliable presentation of the results.

As for the isolated parts of the study, firstly, Fig. 6—9 illustrates the correlations in the maximum drift angle as well as for the Mean Residuum for both tracking algorithms used.

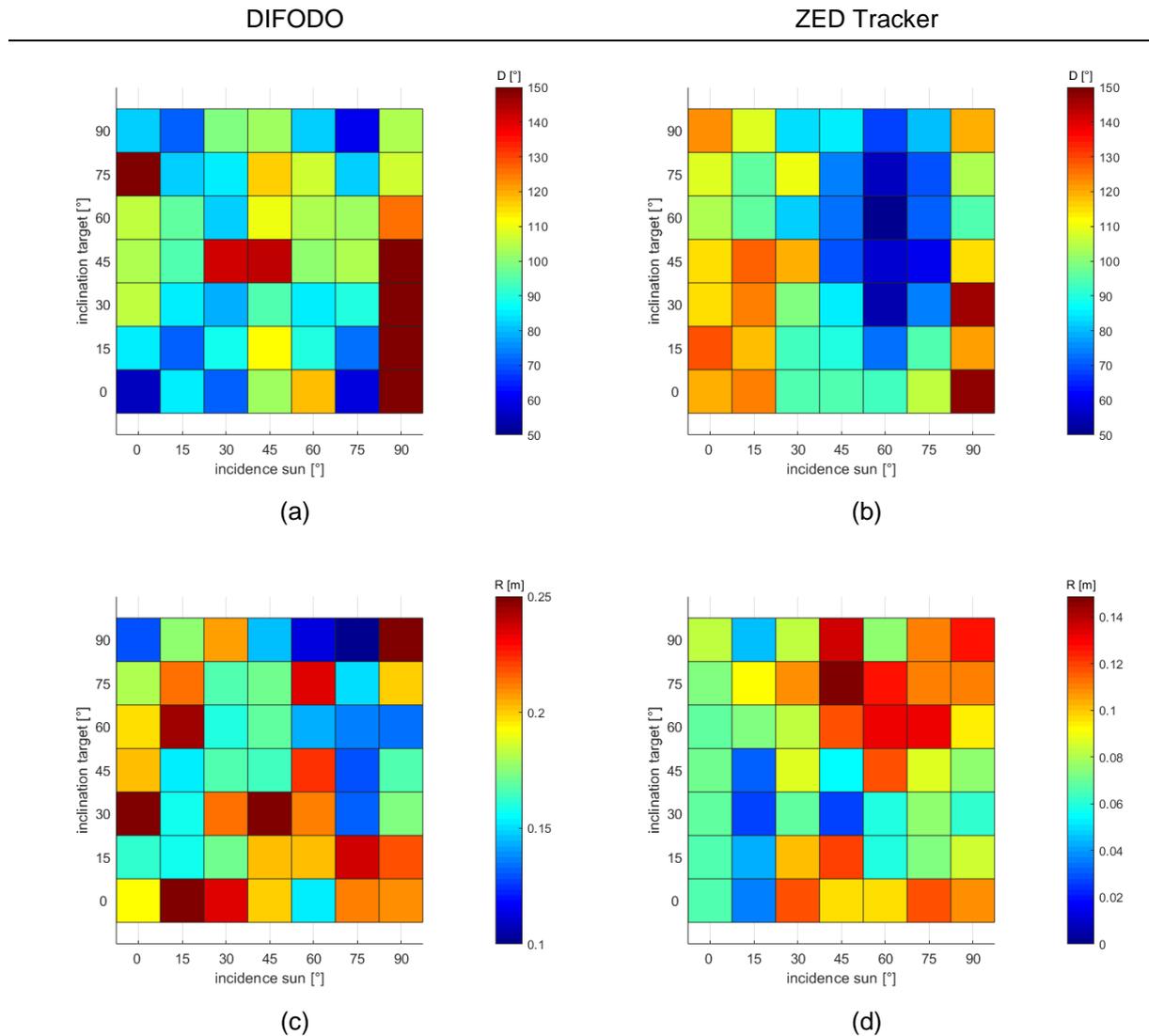
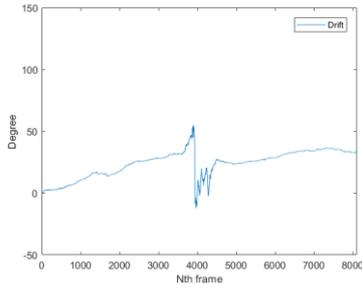


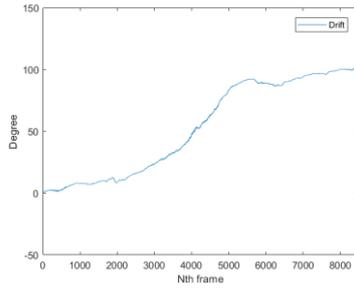
Fig. 6—9: Development of the maximum drift angle as well as of the Mean Residuum for a coherent variation of the parameters for DIFODO (left) and ZED Tracker (right)

By looking at the data generated by the DIFODO algorithm again, thus only taking the numerical values of $\delta_{i,max}$ and R into account, Fig. 6—9a fundamentally indicates comparably good results for the maximum drift angle of an estimated trajectory for low inclination angles of the target over nearly the whole range of the varied sun angle. Only at an angle of 90 deg for the sun, multiple values are far beyond a drift angle of 250 deg. An explanation for this phenomenon is given in chapter 7.1. Also, occasionally low drift angles can be found for a strong target inclination as well.

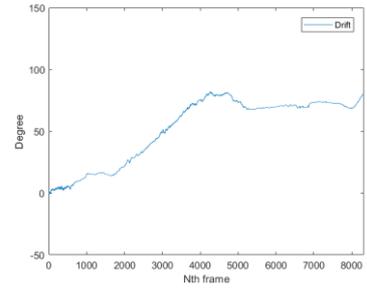
For the Mean Residuum, Fig. 6—9c shows the averagely best values for large angles at both parameters. Most striking are the over the range of the sunlight incidence almost constantly high errors in the linear deviation for tilt angles up to 30 deg, as well as good values for a tilt angle of 90 deg.



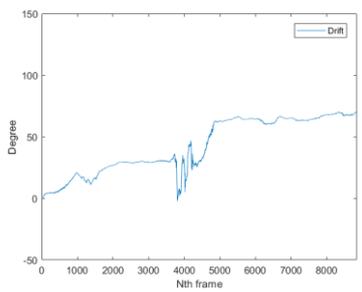
(a) 0_0



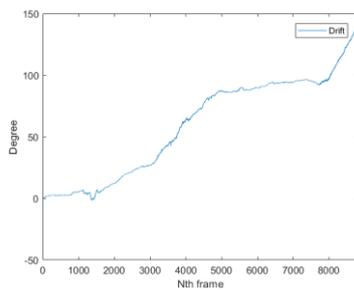
(b) 0_45



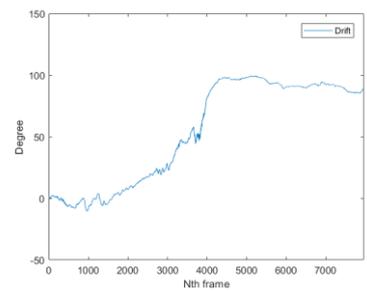
(c) 0_90



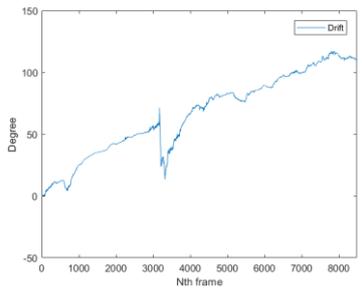
(d) 30_0



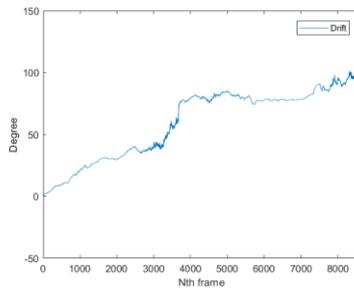
(e) 30_45



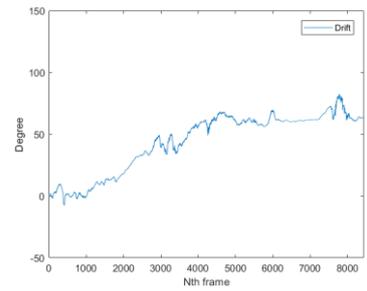
(f) 30_90



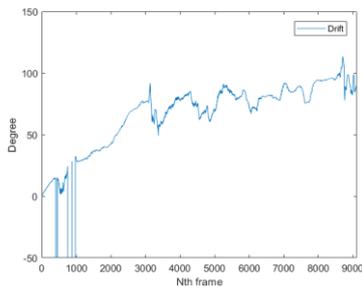
(g) 60_0



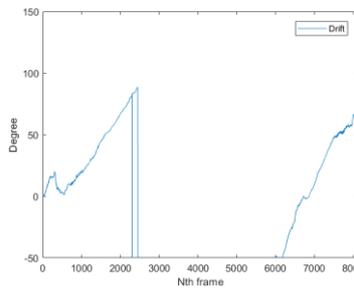
(h) 60_45



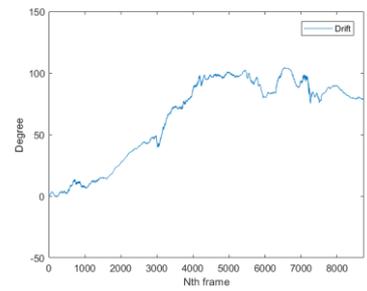
(i) 60_90



(j) 90_0



(k) 90_45



(l) 90_90

Fig. 6—10: Progression of the Drift within the experimental range (DIFODO)

However, to be able to make a valid statement on the exact impact of both varied hardware parameters on D and R , again the courses of both metric modules during a single simulated fly-around must be analyzed.

Fig. 6—10 thereby completes the results already shown in Fig. 6—2 as well as Fig. 6—7. Furthermore, it can be detected, that incidence angle of the sun has an enormous impact on the drift course, despite the tilt angle of the target. On the other hand, varying the target does not seem to change the fundamental course of the drift for each setting qualitatively, but determine the size and the form of the firstly in 6.1 described drift flares.

Also, a linear dependency of the timestamp of said drift flares and the adjustment of the sun can be determined. During the increase of the sun angle, abnormalities in the drift course, particularly well seen for a target inclination of 0 deg, are moving forward in time. While in Fig. 6—10a the flare starts at about 4000 frames, therefore after about have a turn, Fig. 6—10g shows it beginning at about 3000 frames yet. Also, this phenomenon can be detected for any inclination setting of the target, differently pronounced. Additionally, the clarity of the drift flares is decreasing. With the sun being placed at 90 deg, any flares are barely noticeable anymore. At the same time, at this setting the DIFODO algorithm starts struggling with generating a definite drift course, seen in Fig. 6—10k, resulting in in very high errors.

The courses of the linear deviation again do not reveal more insight on the influence of sun position or target attitude than the numerical value of the Mean Residuum. The single anomalies detected match with the results from section 6.1 and 6.2.

On the other hand, for the ZED Tracking algorithm Fig. 6—9b shows a nearly even distribution of equal values over the range of the target, with constantly small drift angles for an incidence angle of the sun from 45 deg to 75 deg. The variation of the target however does not seem to have a crucial impact on the results for the drift. The three characteristic phases for the course of the drift illustrated in Fig. 6—4 remain the same within the whole experimental range, with a chance of the tilt angle of the target only having very little influence on the slope of the drift graph. This even applies for crucial settings regarding reflections like a target angle of 0 deg, as seen in

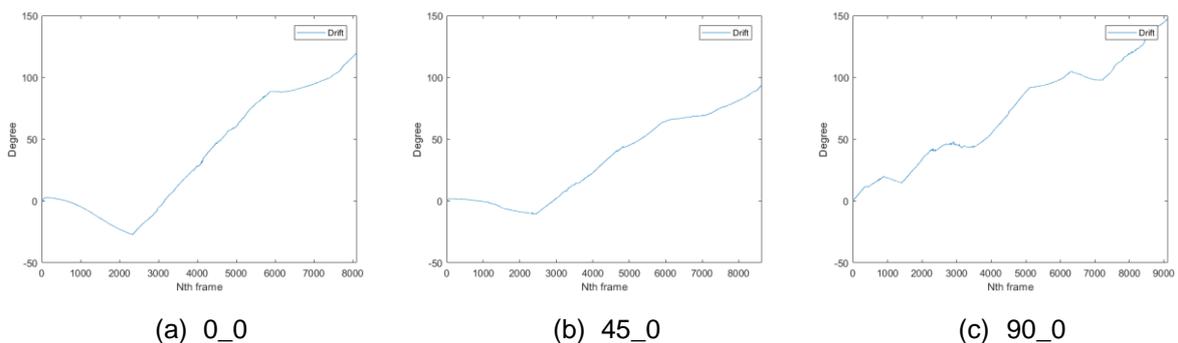


Fig. 6—11: Progression of the drift for an increasing sun angle (ZED)

A clear statement regarding the progression of R in Fig. 6—9d cannot be made, although comparatively good values can be found at little sun angles. Overall, the linear deviations of the estimated trajectory generated by the ZED Tracker regarding its bestfit-circle are comparably low.

Again, considering sun incidence angles higher than 90 deg for both algorithms, the observations presented in chapter 6.1 are underlined over the whole range of the target inclination. Thereby, a sun angle of 90 deg seems to be a critical point for drift values, also seen in Fig. 6—10j/k. Yet, a pattern for the drift reflection cannot be noticed, either. Furthermore, also 120 deg is the limit for a partly reasonable trajectory generation, regardless the target inclination angle, supporting the results presented before.

6.4 Angular velocity

Additionally, the impact on the angular velocity on the quality of the estimated trajectory was fully evaluated. As already mentioned, it was assumed that a little velocity as possible would lead to the best results in tracking, due to providing the maximum information density to the algorithms. However, in first qualitative experiments, this thesis could not be proved.

Finally, to be able to make a quantitative statement, results of different angular velocities were considered for the whole experimental range. By evaluating the numeric output of the metric, it turned out, that in case of DIFODO the mean maximum drift angle is increasing with an also growing velocity, whereas the linear deviations are slightly decreasing, as listed in Tab. 6—2. For other metric modules, no trend could be detected. Yet, by comparing single data sets and graphs, no crucial qualitative differences in the courses of drift as well as Residuum appeared.

For the ZED algorithm on the other hand, differences in the angular velocity do not seem to have any impact on the quality of pose tracking at all.

Tab. 6—2: Mean values of the maximum drift angle and Residuum for different angular velocities (values of a sun angle higher than 90 deg excluded)

Angular Velocity	DIFODO		ZED Tracker	
	$\delta_{i,max}$ [°]	R [m]	$\delta_{i,max}$ [°]	R [m]
1°/s	88,56	0,1792	98,90	0,0728
2°/s	109,82	0,1502	99,80	0,0735
4°/s	123,39	0,1446	100,13	0,0734

6.5 Comparison of used tracking algorithms

To summarize the results of the study, the most important phenomena for both algorithms are presented respectively. Furthermore, similarities as well as differences are elaborated.

With the DIFODO algorithm, overall best results were generated for little incidence angles of the sun up to 40 deg. While increasing the sun angle, the “smoothness” of the drift course drops drastically for high angles. This as well results in an uneven course of the estimated trajectory, no matter the maximum drift angle. Also, at certain points in the simulated fly-around, drift flares can be detected, differently strong pronounced for respectively varying angles in sun and target. Regardless the inclination of the target, the highest flares can be seen at sun angles from 30 deg to 75 deg, whereas over the range of the target inclination the intensity of the flares seems to be lowest at also medium angles. This correspondence is also seen in Fig. 6—12. Thereby, ΔD_{100} is the maximum slope within a time of 100 frames.

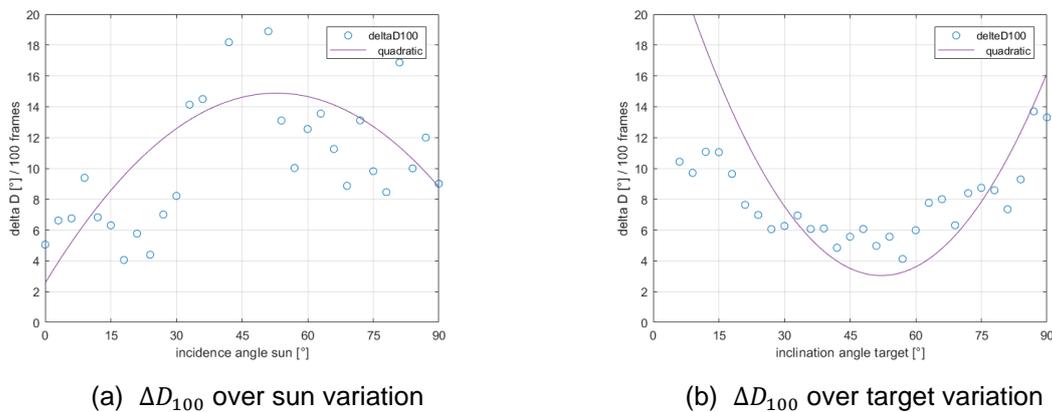


Fig. 6—12: Intensity of the drift flares over the experimental range (DIF)

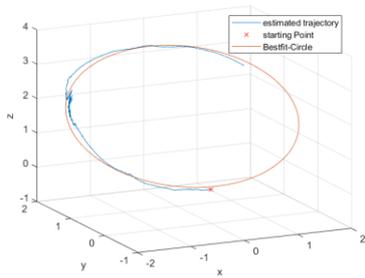
Yet, varying the target angle does not have a huge impact on the maximum drift angle specially, but on the characteristics on the flares. This can be seen especially for a target angle of 0 deg, illustrated in Fig. 6—10.

On the other hand, these characteristics cannot be seen for the ZED Tracker. On the contrary to DIFODO, overall best results were achieved at higher sun angles, namely from 45 deg to 75 deg. The characteristic courses of the drift over the range of the sun seen in Fig. 6—11 are constant for the whole system. Varying the target angle has comparable very little impact on the metric modules. As well, drift flares like seen for the DIFODO are nonexistent. However, at little sun angles a break can be seen in the drift course, no matter the target angle. Especially the average Mean Residuum is much better for the Zed Tracker. Together with an also overall smoother drift course, generated trajectories are more stable.

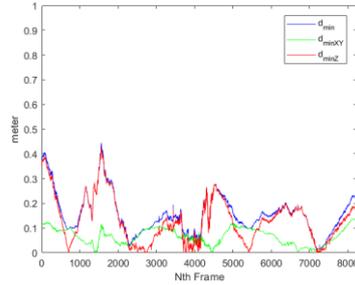
Equally for both algorithms, a sun incidence angle of 90 ± 5 deg is a decisive setting, since drift values are occasionally deflecting and other metric modules are increasing as well. Yet, reasonable trajectories can be generated for angles up to 115 ± 10 deg.

A variation in the angular velocity did not have a crucial impact on the pose estimation. Even though an averagely lower drift could be achieved at a velocity of 1 %/s for DIFODO, other metric modules did not indicate any improvements. For trajectories generated with the ZED Tracker, no impact at all could be identified. Fig. 6—13 compares overall good results for each algorithm with their respectively counterpart.

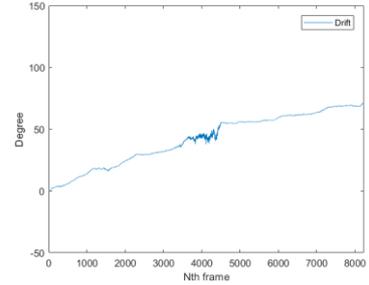
DIFODO



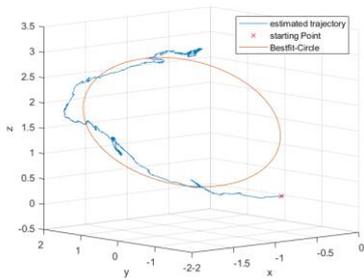
(a) 15_15



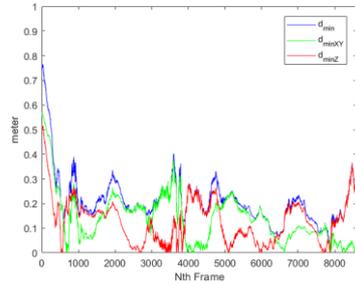
(b) 15_15



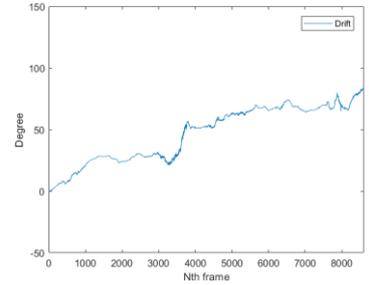
(c) 15_15



(d) 60_30

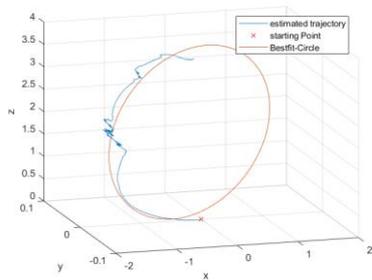


(e) 60_30

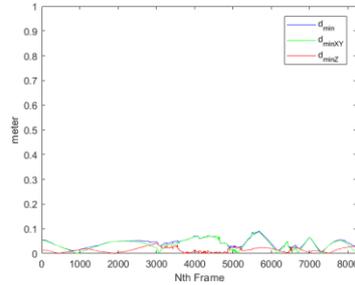


(f) 60_30

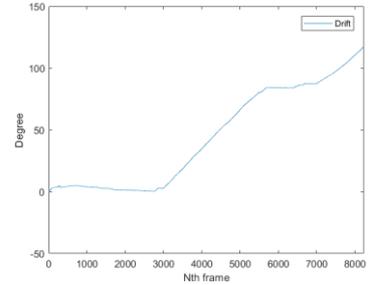
ZED Tracker



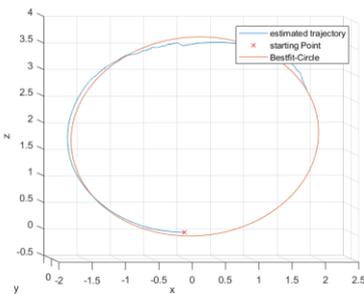
(g) 15_15



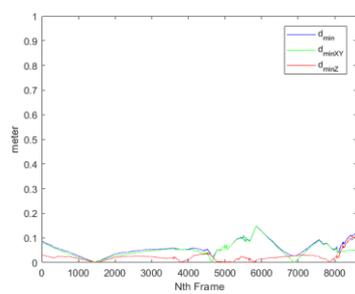
(h) 15_15



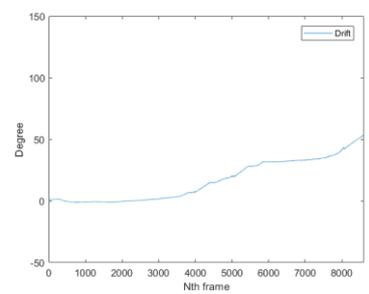
(i) 15_15



(j) 60_30



(k) 60_30



(l) 60_30

Fig. 6—13: Comparison of good results and their counterparts for each algorithm

7 Discussion

After evaluating the study, in this chapter certain monitored characteristics are interpreted for each algorithm. Furthermore, the test series and the resulting outcomes are reflected critically, focusing on the metric as an analyzing tool as well as used presets, and the achieved accuracy.

7.1 Interpretation of the results

As in previous works described, trajectories generated with DIFODO are very susceptible to different lightning conditions. Particularly the drift flares in the trajectory are noteworthy. As priorly assumed, they can be linked to critical reflections on the satellites surface, mainly due to the high reflective aluminum-coated foil. Since every experiment simulated a fly-around of only 270(+10 for accelerating the axis) deg and, besides that, one of said lateral surfaces faced the c-sled, the strong reflections occurred always once during an experiment, slightly differing in the time of appearance caused by varying light incidence angle of the sun. For sun angles between 30 deg and 90 deg, a second flare in the drift can be detected right before the end of the fly-around. Yet, its characteristic was determined by the inclination angle of the target. For an angle of 0 deg, the silvery surface functions like a mirror, causing a very strong but very short reflection in the direct direction of the camera lenses. At the same time, a mirrored image of passing structures appears on the surface. As a result, a definite direction of rotation is no longer recognizable for the camera, causing the drift course illustrated in Fig. 6—10. With an increasing tilt angle of the target, the angle of reflection for that surface gets less critical, resulting in a more even drift course. However, for angles near 90 deg, reflections are getting stronger again due to more direct light incidence angles to the gold-coated topside of the satellite.

On the contrary, when using the ZED Tracker, the inclination of the target and thus the intensity of the reflections does not have any notable impact on the drift course. At the corresponding frames, no characteristics in the trajectory can be found. This can be explained by drift reducing functions in the algorithm. This can also explain the jumps in the drift course for the ZED for little light incidence angles (see e.g. Fig. 6—13i).

Since Spatial Memory was enabled, it theoretically allows the ZED camera to learn and recognize its surroundings during use, causing the device to correct its position in space and eliminate the positional drift that may have accumulated when recognizing an area. However, drift correction can sometimes cause jumps in the absolute pose. [44] In corresponding frames at about 3000 frames for the first jump and 5500 frames for the second, within the boxfilter little parts of the c-sled can be seen by the camera, thus encouraging the algorithm to correct its pose. With higher incidence angles of the sun, disrupting factors of the c-sled are not getting sufficiently illuminated, therefore leading to a smoother drift course.

Yet, this assumption could not be verified. Neither disabling the corresponding algorithm parameter, nor limiting the dimensions of the boxfilter as far as possible had a qualitative impact on the course of the drift.

For both algorithms, estimated trajectories significantly decrease in accuracy for high angles near 90 deg. An explanation can also be found in the limitations of the used camera. Since the sensor does not use infrared but color images for depth perception,

low-light environments should be avoided. With a high angle of the sun in relation to the camera, the target does not get illuminated sufficiently, complicating its recognition by the camera. Angles higher than 90 deg are illuminating the target only from behind. Due to a high intensity of the sun simulator in close range, the camera is only able to detect the sketch of the target. As a result, a clear direction of rotation cannot be detected anymore, and, once wrongfully identified, can only be hardly corrected. This explains the high $\delta_{i,max}$ values for certain trajectories, which lead in extreme cases entirely in the wrong direction. When the sun angle is increasing even further, and the simulator appears within the dimensions of the boxfilter, a sufficient information density for the camera is not given to be able to generate a trajectory.

7.2 Parameter impact on the pose estimation

The metric enables a through a tracking algorithm estimated trajectory to be evaluated quantitatively. With the help of a circlefit-algorithm, a groundtruth reference trajectory is not needed. So far, the quality of a through a tracking algorithm estimated trajectory is evaluated by five isolated error-modules, which allows to make a selective statement regarding the quality of the pose estimation. Thus, important factors like the rotatory or translational deviation can be linked individually to certain influencing factors. However, an absolute error-value, which describes the total error in the whole trajectory is not existing yet, which complicates the determination of the absolute impact of certain factors on the accuracy of a pose estimation.

This problem could be solved by conducting a sensitivity analyses of the metric modules, with the metric parameters as an input and e.g. the Root Mean Square Error (RMSE) between the estimated trajectory and a reference trajectory as an output. By varying single parameters individually, a statement regarding their impact on the absolute error can be made. Yet, an exact result can only be achieved by having the real, error-free trajectory as a reference. However, during this work generating a groundtruth trajectory was not possible, and thus an exact rating of the importance of a single parameter could not be conducted.

Nevertheless, due to previous works as well as subjective experience, primary the drift D and the partly connected Residuum R seem to be responsible for the quality of the pose estimation and therefore reconstruction. Furthermore, those are the only two modules directly referring to the reference trajectory in form of the bestfit-circle, whereas the remaining parameters (centerpoint deviation C_F , radius deviation r_F and elevation ε) are comparing this circle with an imaginary groundtruth trajectory (plane, centerpoint, radius) determined by Optitrack. (see chapter 4.3). Consequently, additional leeway for measurement errors is given.

Finally, the expressiveness of the numeric metric output must be discussed. Especially in case of the DIFODO algorithm, certain influencing factors like crucial reflections can only hardly be linked to a single maximum drift value. To be able to make a valuable statement, the course of the drift must be analyzed. Furthermore, the intensity of explained flares turned out to be subject to an uncertainty, as seen in Fig. 7—1.

Yet, the numeric outputs of the other metric parameters turned out to be more representative.

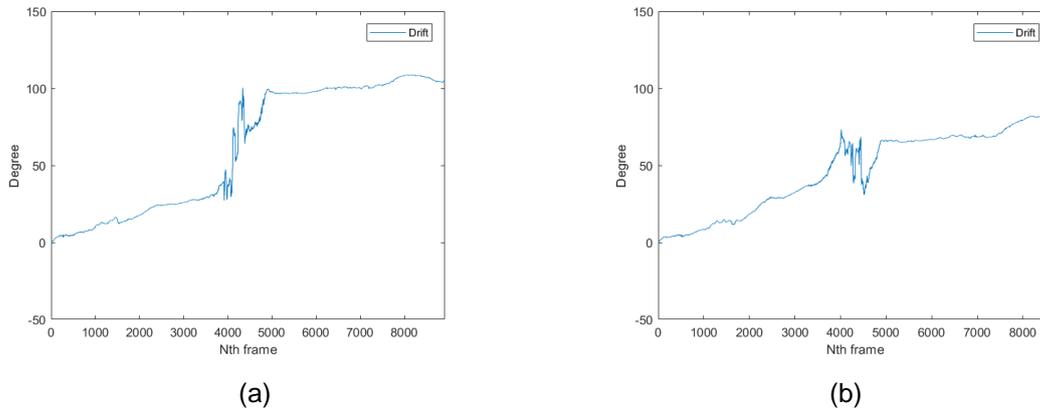


Fig. 7—1: Comparison of the drift flare for the same setting: 15_0 (DIF)

7.3 Critical reflection of the study

Besides the mentioned limitations of the metric, also possible influences due to the main presettings must be discussed. Since constant hardware and software parameter could only be tested qualitatively in advance of the study, a selection of the optimum value for the given scenario could most likely not be assured for all parameters. Especially the applied boxfilter generally allows an enormous improvement in the quality of the estimated trajectory. A best possible result is thus achieved adjusting its dimensions to the outline of the satellite, thus cutting disrupting factors like e.g. the c-sled out.

To be able to compare the generated trajectories without the filter interfering with the results, it had to be set to constant values. Yet, due to the wide range of variation in the inclination of the target, an ideal filter could not be guaranteed for each setting. Although the algorithms used are robust against smaller image parts that move relative to the main object, the exact impact of the used filter dimensions could not be determined. An algorithm, which optimizes the boxfilter in respect to the metric modules could be a possibility to standardize the mandatory process of image pre-processing.

Furthermore, software settings like brightness, contrast, exposure etc. of the used camera were not considered for the study, yet it is assumed that manually adjusted values for certain environmental conditions yield a possibility to improvements. By evaluating the impact of said parameters, predefined camera settings for corresponding scenarios could lead to an enhancement in the pose estimation. However, for an exact quantification extensive test series are necessary.

As already mentioned, a sufficient accuracy in the execution of the test series was only possible to a limited extent, both systematically as well as statistically. Measurements conducted with the Optitrack system as well as done manually yield errors. An absolute error estimation and accumulation of corresponding hardware settings could therefore not be determined, since a valid specification about the accuracy as well as precision of Optitrack is so far not existent. Especially axes 11 and 12, the translational and horizontal rotational DOF of the sun simulator unit yield improvement potential. Since no reference points were attached to either axis and they additionally were not within the range of view of Optitrack, derivation due to e.g. mechanical translation errors could

only be spotted by eye. Moreover, even though reproducibility in the horizontal orientation of the sun could be ensured sufficiently by attaching a laser beneath the simulator, an indication in the achieved accuracy is unfeasible. Yet, by considering the uncertainty of the whole facility in a reliable execution of a single experiment in form of the standard deviation given in chapter 6, it can be used as a good guidance value for the accuracy of the system.

8 Conclusion

In this work, a comprehensive parameter study regarding the quality of an optical pose estimation in earth's orbit was fully planned, executed and evaluated. Based on a previously developed metric, the influence of different illumination conditions could be quantified.

The study was conducted in a HIL facility belonging to the RACOON-Laboratory, an end-to-end simulator for proximity operations. In the course of the work, a list of important parameters could be elaborated for system design considerations. With the help of a pretest series, the impact of these factors was qualitatively evaluated, ensuring them to not interfere with the main study.

By applying statistically valid methods, influences of both the inclination angle of the sunlight relative to the stereo vision-based ZED camera as well as the inclination angle of a target satellite were quantified. Two different algorithms were used, representing a simple tracking algorithm (DIFODO) as well as a function-extended SLAM-algorithm (ZED intern).

For both algorithms, increasing the sun angle to high values deteriorates the accuracy of the estimated trajectory. For angles higher than 90 deg, the rotation direction of the satellite can no longer be reliably determined, resulting in an enormously increased drift. Since the used sensor does not use infrared but color images for depth perception, a sufficient information density cannot be provided at those angles. In case of DIFODO, a variation in the inclination angle of the sun impacts the drift course during a simulated fly-around. Reflections on the satellites surface, especially very small, but also for very big angles, cause a rapidly increase in the drift, respectively. This phenomenon cannot be detected for the ZED Tracker. An analyzation of different angular velocities of the target did not lead to meaningful results.

The during this work generated database can now also function as a basis for further studies, which will allow other algorithms being tested and compared in the future, without the effort of new data generation.

So far, the metric enables an isolated evaluation of different error-parameters, like the linear and angular deviation in the pose estimation. To be able to make an absolute statement regarding the error in the pose, a sensitivity analysis of said metric modules is considered to be highly valuable. However, for that a groundtruth trajectory as reference would be necessary.

Thus, as a next step it would be helpful to verify and improve the accuracy of the integrated motion capture system Optitrack. By doing that, a said groundtruth trajectory could be generated.

For further studies using the sun simulator unit of the facility, a reference point for the corresponding axes 11 and 12 must be installed. Furthermore, to ensure reproducibility, a linking of said axes is mandatory. Both the horizontal as well as vertical rotational need to be automatically oriented onto the rotational axis of the satellite mockup, while moving the sun in translational direction.

To be able to compare different sensor types, the software parameter of the camera also must be quantified. Also, switching though predefined camera settings for corresponding environmental conditions could yield a considerable improvement for the accuracy in the pose estimation.



Finally, an integration of the metric in the software framework of the RACOON-Lab is useful. Additionally, the metric could be extended in its application range by either adding the possibility to evaluate docking scenarios and/or the results of mapping algorithms.

A References

- [1] J. Ventura, "Autonomous Proximity Operations for Noncooperative Space Target," Dissertation, München, 2016_03.
- [2] *esa.int: space debris by the numbers*. [Online] Available: https://www.esa.int/Our_Activities/Operations/Space_Debris/Space_debris_by_the_numbers. Accessed on: Nov. 25 2018.
- [3] A. Fleischner, M. Wilde, and U. Walter, "Racoon - a hardware-in-the-loop simulation environment for teleoperated proximity operations," in *I-SAIRAS (Hg.) 2010 – Proceedings of the International Symposium on Artificial Intelligence, Robotics and Automation in Space*.
- [4] Martin Dziura, Tim Wiese, Jan Harder, "3D Reconstruction in Orbital Proximity Operations," in *IEEE 2017 – 2017 IEEE Aerospace Conference*.
- [5] J. N. Pelton, "Current Space Debris Remediation and On-Orbit Servicing Initiatives," in *SpringerBriefs in Space Development, New Solutions for the Space Debris Problem*, J. N. Pelton, Ed., Cham: Springer International Publishing, 2015, pp. 11–29.
- [6] V. Capuano, K. Kim, and Hu Juliette, "Monocular-Based Pose Determination of Uncooperative Known and Unknown," *69th International Astronautical Congress (IAC)*, 2018.
- [7] R. Opromolla, G. Fasano, G. Rufino, and M. Grassi, "A review of cooperative and uncooperative spacecraft pose determination techniques for close-proximity operations," *Progress in Aerospace Sciences*, vol. 93, pp. 53–72, 2017.
- [8] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part I," *IEEE Robot. Automat. Mag.*, vol. 13, no. 2, pp. 99–110, 2006.
- [9] C. Cadena *et al.*, "Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age," *IEEE Trans. Robot.*, vol. 32, no. 6, pp. 1309–1332, 2016.
- [10] A. Sonnenburg, M. Tkocz, and K. Janschek, "EKF-SLAM based Approach for Spacecraft Rendezvous Navigation with Unknown Target Spacecraft," *IFAC Proceedings Volumes*, vol. 43, no. 15, pp. 339–344, 2010.
- [11] F. Schnitzer, K. Janschek, and G. Willich, "Experimental results for image-based geometrical reconstruction for spacecraft Rendezvous navigation with unknown and uncooperative target spacecraft," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vilamoura-Algarve, Portugal, Oct. 2012 - Oct. 2012, pp. 5040–5045.
- [12] Schnitzer F., Sonnenburg A., and et al., "Bildbasierte SLAM-Relativnavigation und 3D-Rekonstruktion für das On-Orbit-Servicing von unbekanntem und unkooperativen Raumflugkörpern,"
- [13] B.E. Tweddle, Computer Vision-based Localization and Mapping of an Unknown, Uncooperative and Spinning Target for Spacecraft Proximity Operations, PhD, Massachusetts Institute of Technology, Cambridge, MA, 2013.

- [14] M. Kaess, A. Ranganathan, and F. Dellaert, "iSAM: Incremental Smoothing and Mapping," *IEEE Trans. Robot.*, vol. 24, no. 6, pp. 1365–1378, 2008.
- [15] S. Segal, A. Carmi, and P. Gurfil, "Stereovision-Based Estimation of Relative Dynamics Between Noncooperative Satellites: Theory and Experiments," *IEEE Trans. Contr. Syst. Technol.*, vol. 22, no. 2, pp. 568–584, 2014.
- [16] Y. Li and Y. Xie, "Relative state estimation of model-unknown spinning noncooperative target using stereo EKF-SLAM," in *2017 36th Chinese Control Conference (CCC)*, Dalian, China, Jul. 2017 - Jul. 2017, pp. 5893–5898.
- [17] J. Harder, M. Dziura, and S. Haberl, "Future Technologies for Operating Robots in Space,"
- [18] F. Rehn, "Experimentelle Studie mit RACOON und EPOS zur quantitativen Bewertung von Einflussfaktoren auf die Qualität einer 3D-Objektrekonstruktion im Erdorbit," Bachelorarbeit, Lehrstuhl für Raumfahrttechnik, Technische Universität München, München, 2017.
- [19] T. Wiese, "Integration einer RGB-D-Kamera mit Visual-SLAM-Algorithmen in eine Simulationsumgebung zur Verbesserung der Nahbereichsnavigation im Orbit," Semesterarbeit, Lehrstuhl für Raumfahrttechnik, Technische Universität München, München, 2015.
- [20] H. Hirschlüller, K. Schmidt, and M. Suppa, "Computer Vision for Mobile Robot Navigation," *Photogrammetric Week '15*, S. 143–154, 2015.
- [21] F. Endres, J. Hess, J. Sturm, D. Cremers, and W. Burgard, "3-D Mapping With an RGB-D Camera," *IEEE Trans. Robot.*, vol. 30, no. 1, pp. 177–187, 2014.
- [22] M. Jaimez and J. Gonzalez-Jimenez, "Fast Visual Odometry for 3-D Range Sensors," *IEEE Trans. Robot.*, vol. 31, no. 4, pp. 809–822, 2015.
- [23] H. Spies, B. Jähne, and J. L. Barron, "Range Flow Estimation," *Computer Vision and Image Understanding*, vol. 85, no. 3, pp. 209–231, 2002.
- [24] Forsyth - Ponce, "Computer Vision - A Modern Approach,"
- [25] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vilamoura-Algarve, Portugal, Oct. 2012 - Oct. 2012, pp. 573–580.
- [26] C. Karpfinger, *Höhere Mathematik in Rezepten: Begriffe, Sätze und zahlreiche Beispiele in kurzen Lerneinheiten*, 3rd ed. Berlin, Heidelberg: Springer Spektrum, 2017.
- [27] *Fitting a Circle to Cluster of 3D Points*. [Online] Available: <https://meshlogic.github.io/posts/jupyter/curve-fitting/fitting-a-circle-to-cluster-of-3d-points/>. Accessed on: Oct. 30 2018.
- [28] Randy Bullock, *Least-Squares Circle Fit*. [Online] Available: http://www.dtcenter.org/met/users/docs/write_ups/circle_fit.pdf. Accessed on: Oct. 30 2018.
- [29] H.-J. Bartsch, *Taschenbuch mathematischer Formeln*, 20th ed. München: Fachbuchverl. Leipzig im Carl-Hanser-Verl., 2004.

- [30] *Chair of Astronautics: "Racoon-Lab"*. [Online] Available: <https://www.lrt.mw.tum.de/index.php?id=54&L=0>. Accessed on: Nov. 02 2018.
- [31] *Microsoft: "Kinect hardware"*. [Online] Available: <https://developer.microsoft.com/de-de/windows/kinect>. Accessed on: Sep. 15 2018.
- [32] *StereoLabs: "ZED"*. [Online] Available: <https://www.stereolabs.com/>. Accessed on: Nov. 04 2018.
- [33] *StereoLabs: "ZED - Depth Sensing and Camera Tracking"*. [Online] Available: <https://www.stereolabs.com/zed/>. Accessed on: Nov. 02 2018.
- [34] C. Schmid, S. Soatto, and C. Tomasi, Eds., *A Comparison and Evaluation of Multi-View Stereo Reconstruction Algorithms: CVPR workshops ; New York, New York, USA, 17 - 22 June 2006*. Piscataway, NJ: IEEE, 2006.
- [35] A. Ciadamidaro, "Design and implementation of a sun simulator for a simulation environment for close-range navigation of satellites," Bachelorarbeit, Lehrstuhl für Raumfahrttechnik, Technische Universität München, München.
- [36] N. Reichenbach, "Development and integration of an earth's albedo simulator to evaluate sensors for closerange navigation of satellites in the earth orbit," Lehrstuhl für Raumfahrttechnik, Technische Universität München, München.
- [37] J. Harder, Ed., *RACOON - Infrastruktur: Specification Sheet*.
- [38] J. Otto, "Development of an Absolute Reference Measurement System in the Racoon Hardware Lab,"
- [39] *OptiTrack*. [Online] Available: <http://optitrack.com>. Accessed on: Oct. 12 2017.
- [40] *Optitrack: Motion Capture Robotics*. [Online] Available: <https://optitrack.com/motion-capture-robotics/>. Accessed on: Nov. 05 2018.
- [41] F. Aghili, M. Kuryllo, G. Okouneva, and C. English, "Fault-Tolerant Position/Attitude Estimation of Free-Floating Space Objects Using a Laser Range Sensor," *IEEE Sensors J.*, vol. 11, no. 1, pp. 176–185, 2011.
- [42] R. Opromolla, G. Fasano, G. Rufino, and M. Grassi, "Uncooperative pose estimation with a LIDAR-based system," *Acta Astronautica*, vol. 110, pp. 287–297, 2015.
- [43] T. W. Lim, M. ODowd, and P. Ramos, "Feature Detection for Pose Estimation," in *AIAA Guidance, Navigation, and Control Conference*, Kissimmee, Florida, 01052015, p. 25.
- [44] *StereoLabs: "ZED API"*. [Online] Available: https://www.stereolabs.com/docs/api/classsl_1_1Camera.html. Accessed on: Nov. 10 2018.
- [45] K. Siebertz, D. van Bebber, and T. Hochkirchen, *Statistische Versuchsplanung: Design of Experiments (DoE)*, 2nd ed. Berlin, Heidelberg: Springer Vieweg, 2017.

B Appendix

B.1 Detailed parameter of the experiments

Tab. B.1—1: **Incidence angle sun**; Inclination angle target: 45°/150000 steps; Vertical orientation sun simulator: 7° faced down; Shadowing through chaser: S_0 to S-12; Shadowing through C-sled: s_15 to S_135

Name	Incidence sunlight [°]	Steps axis 11	Steps axis 12
S_0	0	0	0
S_3	3 [+0.19]	-49153	-500
S_6	6 [+0.38]	-98646	-1100
S_9	9 [+0.57]	-148854	-1700
S_12	12 [+0.76]	-200246	-2300
S_15	15 [+0.95]	-271465	-3000
S_18	18 [+1.14]	-309499	-3500
S_21	21 [+1.33]	-370097	-4200
S_24	24 [+1.52]	-439034	-5000
S_27	27 [+1.71]	-521283	-5200
S_30	30 [+1.90]	-595290	-5100
S_33	33 [+2.09]	-656981	-4800
S_36	36 [+2.28]	-709459	-4400
S_39	39 [+2.47]	-754875	-4100
S_42	42 [+2.66]	-794770	-3900
S_45	45 [+2.85]	-830277	-3600
S_48	48 [+3.04]	-862247	-3300
S_51	51 [+3.23]	-891336	-2900
S_54	54 [+3.42]	-918057	-2600
S_57	57 [+3.61]	-942817	-2300
S_60	60 [+3.80]	-965947	-2100
S_63	63 [+3.99]	-987719	-1800
S_66	66 [+4.18]	-1008359	-1600
S_69	69 [+4.37]	-1028057	-1400
S_72	72 [+4.56]	-1046978	-1100
S_75	75 [+4.75]	-1065265	-700
S_78	78 [+4.94]	-1083046	-400
S_81	81 [+5.13]	-1100435	-200
S_84	84 [+5.32]	-1117538	-50

S_87	87 [+5.51]	-1134454	100
S_90	90 [+5.69]	-1151277	200
S_93	93 [+5.32]	-1168100	500
S_96	96 [+4.94]	-1185015	700
S_99	99 [+4.56]	-1201838	900
S_102	102 [+4.18]	-1219508	1100
S_105	105 [+3.80]	-1237289	1400
S_108	108 [+3.42]	-1255577	1600
S_111	111 [+3.04]	-1274337	1800
S_114	114 [+2.66]	-1292964	1900
S_117	117 [+2.28]	-1318491	1400
S_120	120 [+1.90]	-1336901	800
S_123	123 [+1.52]	-1355311	500
S_126	126 [+1.14]	-1373771	100
S_129	129 [+0.76]	-1392371	-100
S_132	132 [+0.38]	-1411181	-500
S_135	135	-1420566	-1000

Tab. B.1—2: **Inclination angle target**; Inclination angle sun: 15°/-271465 steps; Vertical orientation sun simulator: 7° faced down; No shadowing

Name	inclination target [°]	Steps axis 7
T_0	0	0
T_3	3	10000
T_6	6	20000
T_9	9	30000
T_12	12	40000
T_15	15	50000
T_18	18	60000
T_21	21	70000
T_24	24	80000
T_27	27	90000
T_30	30	100000
T_33	33	110000
T_36	36	120000
T_39	39	130000
T_42	42	140000
T_45	45	150000
T_48	48	160000
T_51	51	170000
T_54	54	180000
T_57	57	190000
T_60	60	200000
T_63	63	210000
T_66	66	220000
T_69	69	230000
T_72	72	240000
T_75	75	250000
T_78	78	260000
T_81	81	270000
T_84	84	280000
T_87	87	290000
T_90	90	300000

Tab. B.1—1: **Correlating Experiments**; Vertical orientation sun simulator: 7° faced down; Shadowing through chaser: S_0; Shadowing through C-sled: s_30 to S_135

Name	Incidence sunlight [°]	Steps axis 11	Steps axis 12	inclination target [°]	Steps axis 7
0_0	0	0	0	0	0
0_15	0	0	0	15	50000
0_30	0	0	0	30	100000
0_45	0	0	0	45	150000
0_60	0	0	0	60	200000
0_75	0	0	0	75	250000
0_90	0	0	0	90	300000
15_0	15	-245167	-2600	0	0
15_15	15	-245167	-2600	15	50000
15_30	15	-245167	-2600	30	100000
15_45	15	-245167	-2600	45	150000
15_60	15	-245167	-2600	60	200000
15_75	15	-245167	-2600	75	250000
15_90	15	-245167	-2600	90	300000
30_0	30	-563193	-5100	0	0
30_15	30	-563193	-5100	15	50000
30_30	30	-563193	-5100	30	100000
30_45	30	-563193	-5100	45	150000
30_60	30	-563193	-5100	60	200000
30_75	30	-563193	-5100	75	250000
30_90	30	-563193	-5100	90	300000
45_0	45	-798181	-3400	0	0
45_15	45	-798181	-3400	15	50000
45_30	45	-798181	-3400	30	100000
45_45	45	-798181	-3400	45	150000
45_60	45	-798181	-3400	60	200000
45_75	45	-798181	-3400	75	250000
45_90	45	-798181	-3400	90	300000
60_0	60	-939852	-1900	0	0
60_15	60	-939852	-1900	15	50000
60_30	60	-939852	-1900	30	100000
60_45	60	-939852	-1900	45	150000
60_60	60	-939852	-1900	60	200000

60_75	60	-939852	-1900	75	250000
60_90	60	-939852	-1900	90	300000
75_0	75	-1033167	-700	0	0
75_15	75	-1033167	-700	15	50000
75_30	75	-1033167	-700	30	100000
75_45	75	-1033167	-700	45	150000
75_60	75	-1033167	-700	60	200000
75_75	75	-1033167	-700	75	250000
75_90	75	-1033167	-700	90	300000
90_0	90	-1119181	0	0	0
90_15	90	-1119181	0	15	50000
90_30	90	-1119181	0	30	100000
90_45	90	-1119181	0	45	150000
90_60	90	-1119181	0	60	200000
90_75	90	-1119181	0	75	250000
90_90	90	-1119181	0	90	300000
105_0	105	-1205168	1900	0	0
105_15	105	-1205168	1900	15	50000
105_30	105	-1205168	1900	30	100000
105_45	105	-1205168	1900	45	150000
105_60	105	-1205168	1900	60	200000
105_75	105	-1205168	1900	75	250000
105_90	105	-1205168	1900	90	300000
120_0	120	-1303840	1100	0	0
120_15	120	-1303840	1100	15	50000
120_30	120	-1303840	1100	30	100000
120_45	120	-1303840	1100	45	150000
120_60	120	-1303840	1100	60	200000
120_75	120	-1303840	1100	75	250000
120_90	120	-1303840	1100	90	300000
135_0	135	-1420566	-800	0	0
135_15	135	-1420566	-800	15	50000
135_30	135	-1420566	-800	30	100000
135_45	135	-1420566	-800	45	150000
135_60	135	-1420566	-800	60	200000
135_75	135	-1420566	-800	75	250000
135_90	135	-1420566	-800	90	300000

B.2 Results

The results for and angular velocity of 1°/s are presented. Yellow marked values are displaying a false rotation velocity. Green marked values were initially wrongfully calculated, and thus corrected.

Tab. B.2—1: Incidence angle sun (DIFODO)

	C_F [m]	r_F [m]	ε [°]	R [m]	$\delta_{i,max}$ [°]
"trajectoryDIFODO_S_0.txt"	0,4124	0,0319	4,42	0,2019	105,26
"trajectoryDIFODO_S_3.txt"	0,5194	0,1102	14,28	0,1814	108,11
"trajectoryDIFODO_S_6.txt"	0,5074	0,0880	8,96	0,1744	91,97
"trajectoryDIFODO_S_9.txt"	0,5326	0,1274	25,11	0,1801	95,32
"trajectoryDIFODO_S_12.txt"	0,4563	0,1183	20,01	0,1612	80,45
"trajectoryDIFODO_S_15.txt"	0,6180	0,2175	31,48	0,1489	81,61
"trajectoryDIFODO_S_18.txt"	0,6977	0,2582	36,19	0,1677	72,87
"trajectoryDIFODO_S_21.txt"	0,6187	0,2564	36,62	0,1580	72,88
"trajectoryDIFODO_S_24.txt"	0,6587	0,2921	32,62	0,1671	75,78
"trajectoryDIFODO_S_27.txt"	0,5649	0,3194	28,23	0,1583	88,67
"trajectoryDIFODO_S_30.txt"	0,6413	0,3750	35,70	0,1686	88,80
"trajectoryDIFODO_S_33.txt"	0,8840	0,5113	33,11	0,2080	105,00
"trajectoryDIFODO_S_36.txt"	0,7982	0,5258	23,49	0,1988	116,72
"trajectoryDIFODO_S_39.txt"	0,6719	0,4412	17,33	0,1763	130,97
"trajectoryDIFODO_S_42.txt"	0,5821	0,3237	24,23	0,1766	126,71
"trajectoryDIFODO_S_45.txt"	0,6530	0,4058	29,16	0,1952	132,32
"trajectoryDIFODO_S_48.txt"	0,8219	0,4729	36,41	0,2017	131,78
"trajectoryDIFODO_S_51.txt"	0,8430	0,4349	32,68	0,1987	128,72
"trajectoryDIFODO_S_54.txt"	0,7728	0,2971	41,29	0,1783	122,78
"trajectoryDIFODO_S_57.txt"	1,0809	0,4197	50,74	0,2032	114,66
"trajectoryDIFODO_S_60.txt"	0,8652	0,3691	50,29	0,1613	99,37
"trajectoryDIFODO_S_63.txt"	0,6430	0,3891	25,31	0,1306	125,72
"trajectoryDIFODO_S_66.txt"	0,7245	0,3891	19,97	0,2129	120,57
"trajectoryDIFODO_S_69.txt"	0,7354	0,3755	32,97	0,1421	117,19
"trajectoryDIFODO_S_72.txt"	0,8711	0,4928	37,98	0,1098	105,18
"trajectoryDIFODO_S_75.txt"	0,8167	0,4749	48,09	0,1212	91,69
"trajectoryDIFODO_S_78.txt"	0,7901	0,5150	55,68	0,2047	127,11
"trajectoryDIFODO_S_81.txt"	1,1210	0,5688	48,68	0,1548	116,75
"trajectoryDIFODO_S_84.txt"	1,2043	0,6331	56,99	0,1733	283,97
"trajectoryDIFODO_S_87.txt"	1,0628	0,4652	33,35	0,1460	345,03
"trajectoryDIFODO_S_90.txt"	1,0148	0,3942	40,50	0,1513	273,05
"trajectoryDIFODO_S_93.txt"	1,2325	0,4331	26,22	0,1557	337,26
"trajectoryDIFODO_S_96.txt"	1,5952	0,8025	54,45	0,1538	77,86
"trajectoryDIFODO_S_99.txt"	1,6772	0,7130	70,96	0,1606	109,36
"trajectoryDIFODO_S_102.txt"	1,7701	0,8111	76,48	0,1593	334,12

"trajectoryDIFODO_S_105.txt"	1,6767	0,8842	75,25	0,1724	95,41
"trajectoryDIFODO_S_108.txt"	1,4852	0,6747	50,18	0,1802	139,13
"trajectoryDIFODO_S_111.txt"	0,9680	0,3126	34,82	0,1757	170,47
"trajectoryDIFODO_S_114.txt"	1,5796	0,6320	49,50	0,2662	92,56
"trajectoryDIFODO_S_117.txt"	1,4099	0,5466	44,55	0,2514	90,36
"trajectoryDIFODO_S_120.txt"	1,9153	0,6086	72,84	0,2547	82,31
"trajectoryDIFODO_S_123.txt"	2,3094	0,5329	80,44	0,3362	81,60
"trajectoryDIFODO_S_126.txt"	2,2164	0,1296	31,31	0,3618	283,75
"trajectoryDIFODO_S_129.txt"	2,9346	0,4111	87,66	0,6494	233,52
"trajectoryDIFODO_S_132.txt"	2,0565	-1,1299	75,95	0,5907	209,76
"trajectoryDIFODO_S_135.txt"	32,6599	-29,3338	76,26	0,8292	209,43

Tab. B.2—2: Inclination angle target (DIFODO)

	C_F [m]	r_F [m]	ε [°]	R [m]	$\delta_{i,max}$ [°]
trajectoryDIFODO_T_0.txt	0,5987	0,1004	19,82	0,1448	108,85
trajectoryDIFODO_T_3.txt	0,6011	0,1952	13,61	0,3060	71,83
trajectoryDIFODO_T_6.txt	0,5743	0,1130	24,46	0,1356	68,49
trajectoryDIFODO_T_9.txt	0,5338	0,1451	30,46	0,1066	102,45
trajectoryDIFODO_T_12.txt	0,7769	0,1684	38,30	0,1250	58,31
trajectoryDIFODO_T_15.txt	0,6692	0,1437	32,66	0,1218	50,97
trajectoryDIFODO_T_18.txt	0,6808	0,1058	34,45	0,1283	64,26
trajectoryDIFODO_T_21.txt	0,6344	0,1048	31,90	0,1316	66,18
trajectoryDIFODO_T_24.txt	0,7606	0,1419	41,41	0,1606	73,98
trajectoryDIFODO_T_27.txt	0,6625	0,1714	34,70	0,1410	75,24
trajectoryDIFODO_T_30.txt	0,5439	0,2076	34,73	0,1647	70,63
trajectoryDIFODO_T_33.txt	0,6725	0,1582	35,65	0,1682	82,09
trajectoryDIFODO_T_36.txt	0,5868	0,1864	36,41	0,1494	74,05
trajectoryDIFODO_T_39.txt	0,4822	0,1425	30,96	0,1679	79,01
trajectoryDIFODO_T_42.txt	0,5049	0,1312	32,07	0,1451	79,44
trajectoryDIFODO_T_45.txt	0,5176	0,1310	29,65	0,1664	73,50
trajectoryDIFODO_T_48.txt	0,5142	0,1008	27,88	0,1642	76,23
trajectoryDIFODO_T_51.txt	0,4563	0,1185	21,66	0,1610	73,63
trajectoryDIFODO_T_54.txt	0,4303	0,0659	14,61	0,1419	64,00
trajectoryDIFODO_T_57.txt	0,5329	0,0601	22,62	0,1229	71,50
trajectoryDIFODO_T_60.txt	0,4368	0,0707	32,30	0,2645	80,47
trajectoryDIFODO_T_63.txt	0,6132	0,0660	31,87	0,2265	78,52
trajectoryDIFODO_T_66.txt	0,6809	0,0807	43,12	0,2831	100,92
trajectoryDIFODO_T_69.txt	0,6946	0,1247	39,93	0,2299	79,84
trajectoryDIFODO_T_72.txt	0,4949	0,1490	27,73	0,2215	80,89
trajectoryDIFODO_T_75.txt	0,4559	0,1595	23,93	0,2254	75,69
trajectoryDIFODO_T_78.txt	0,4535	0,1967	22,83	0,2482	71,50
trajectoryDIFODO_T_81.txt	0,5035	0,1849	28,85	0,3184	96,99
trajectoryDIFODO_T_84.txt	0,3377	0,1420	8,50	0,1661	91,79

trajectoryDIFODO_T_87.txt	0,3587	0,1657	7,88	0,1495	63,75
trajectoryDIFODO_T_90.txt	0,4431	0,1302	8,54	0,1699	56,64

Tab. B.2 3: Correlating experiments (DIFODO)

	C_F [m]	r_F [m]	ε [°]	R [m]	$\delta_{i,max}$ [°]
trajectoryDIFODO_0_0.txt	0,5087	0,0773	27,22	0,1924	54,80
trajectoryDIFODO_0_15.txt	0,3775	0,1053	27,28	0,1612	85,29
trajectoryDIFODO_0_30.txt	0,4482	0,1456	12,66	0,2858	105,32
trajectoryDIFODO_0_45.txt	0,4625	-0,0390	0,89	0,2012	104,15
trajectoryDIFODO_0_60.txt	0,6979	-0,0346	27,71	0,1974	104,98
trajectoryDIFODO_0_75.txt	0,8722	0,1147	43,09	0,1810	123,92
trajectoryDIFODO_0_90.txt	0,4737	0,2006	24,55	0,1287	82,05
trajectoryDIFODO_15_0.txt	0,2326	0,1111	7,63	0,3676	84,98
trajectoryDIFODO_15_15.txt	0,5399	0,0495	40,71	0,1571	71,82
trajectoryDIFODO_15_30.txt	0,5936	0,1869	26,75	0,1573	84,79
trajectoryDIFODO_15_45.txt	0,5905	0,1350	24,24	0,1537	94,11
trajectoryDIFODO_15_60.txt	0,4296	0,1170	24,45	0,2451	96,75
trajectoryDIFODO_15_75.txt	0,3964	0,1658	23,07	0,2136	82,25
trajectoryDIFODO_15_90.txt	0,6437	0,3451	31,93	0,1762	71,28
trajectoryDIFODO_30_0.txt	0,4634	0,2988	26,03	0,2346	70,67
trajectoryDIFODO_30_15.txt	0,4989	0,1628	35,59	0,1721	88,97
trajectoryDIFODO_30_30.txt	0,7757	0,4351	40,35	0,2138	78,19
trajectoryDIFODO_30_45.txt	0,5415	0,2644	23,60	0,1671	141,24
trajectoryDIFODO_30_60.txt	0,5596	0,3932	19,02	0,1604	81,58
trajectoryDIFODO_30_75.txt	0,4872	0,3076	24,72	0,1672	85,08
trajectoryDIFODO_30_90.txt	0,5327	0,2385	15,27	0,2070	99,56
trajectoryDIFODO_45_0.txt	0,5017	0,1897	9,89	0,1991	101,94
trajectoryDIFODO_45_15.txt	0,7640	0,3200	45,79	0,2023	112,24
trajectoryDIFODO_45_30.txt	1,2111	0,6165	59,41	0,2543	94,47
trajectoryDIFODO_45_45.txt	0,7534	0,4342	35,37	0,1648	142,34
trajectoryDIFODO_45_60.txt	0,7919	0,4470	15,52	0,1701	110,37
trajectoryDIFODO_45_75.txt	0,7901	0,4330	14,20	0,1714	116,23
trajectoryDIFODO_45_90.txt	0,5984	0,4022	18,48	0,1460	101,97
trajectoryDIFODO_60_0.txt	0,6239	0,0879	26,86	0,1523	117,25
trajectoryDIFODO_60_15.txt	1,2021	0,4093	49,47	0,2015	89,71
trajectoryDIFODO_60_30.txt	1,2032	0,5095	65,07	0,2121	84,92
trajectoryDIFODO_60_45.txt	1,1645	0,4718	57,46	0,2228	101,35
trajectoryDIFODO_60_60.txt	0,7347	0,3201	31,02	0,1429	103,96
trajectoryDIFODO_60_75.txt	0,9463	0,4519	25,00	0,2337	107,15
trajectoryDIFODO_60_90.txt	0,4808	0,2643	20,12	0,1123	82,49
trajectoryDIFODO_75_0.txt	0,7216	0,1735	27,92	0,2104	58,32
trajectoryDIFODO_75_15.txt	0,9839	0,2722	61,61	0,2366	72,68
trajectoryDIFODO_75_30.txt	0,8468	0,3931	49,72	0,1311	90,12

trajectoryDIFODO_75_45.txt	0,7547	0,4213	40,64	0,1287	103,15
trajectoryDIFODO_75_60.txt	0,8160	0,3367	39,65	0,1363	101,75
trajectoryDIFODO_75_75.txt	0,7521	0,2452	15,80	0,1499	81,35
trajectoryDIFODO_75_90.txt	0,5376	0,3304	32,29	0,1022	59,96
trajectoryDIFODO_90_0.txt	0,8227	0,2019	33,17	0,2093	118,99
trajectoryDIFODO_90_15.txt	0,6788	0,1425	49,03	0,2185	262,04
trajectoryDIFODO_90_30.txt	1,1391	0,2779	63,48	0,1743	270,64
trajectoryDIFODO_90_45.txt	1,0002	0,6026	65,13	0,1679	276,30
trajectoryDIFODO_90_60.txt	0,8081	0,4969	32,51	0,1328	126,43
trajectoryDIFODO_90_75.txt	0,7207	0,3910	16,52	0,2000	107,43
trajectoryDIFODO_90_90.txt	0,7424	0,4906	37,50	0,2673	104,55
trajectoryDIFODO_105_0.txt	0,8872	0,2732	13,66	0,2875	56,40
trajectoryDIFODO_105_15.txt	1,3599	0,2353	52,10	0,3285	70,75
trajectoryDIFODO_105_30.txt	1,7953	0,6710	60,08	0,1996	43,49
trajectoryDIFODO_105_45.txt	1,2060	0,5774	48,00	0,1508	337,10
trajectoryDIFODO_105_60.txt	1,3086	0,4984	37,06	0,1627	102,92
trajectoryDIFODO_105_75.txt	1,0604	0,2824	30,74	0,1926	111,08
trajectoryDIFODO_105_90.txt	0,8295	0,2829	26,95	0,1317	110,94
trajectoryDIFODO_120_0.txt	1,0702	0,1872	9,33	0,2172	103,39
trajectoryDIFODO_120_15.txt	2,2796	0,4246	74,71	0,3624	333,26
trajectoryDIFODO_120_30.txt	0,3066	0,1252	27,59	0,3805	87,13
trajectoryDIFODO_120_45.txt	1,5067	0,5434	59,09	0,2289	76,21
trajectoryDIFODO_120_60.txt	1,0763	0,1697	51,18	0,2645	96,69
trajectoryDIFODO_120_75.txt	1,4459	0,4738	37,86	0,3374	139,31
trajectoryDIFODO_120_90.txt	1,6657	0,7538	47,96	0,1892	345,47
trajectoryDIFODO_135_0.txt	2,4024	-2,4547	10,77	0,8202	284,78
trajectoryDIFODO_135_15.txt	0,6039	-0,4820	45,21	0,4316	344,80
trajectoryDIFODO_135_30.txt	5,4869	-1,2096	50,85	0,5547	251,64
trajectoryDIFODO_135_45.txt	9,4468	-6,5320	72,48	0,5270	216,97
trajectoryDIFODO_135_60.txt	1,6157	0,4060	39,66	0,6118	228,92
trajectoryDIFODO_135_75.txt	1,8918	-0,3490	36,07	0,9277	186,30
trajectoryDIFODO_135_90.txt	28,6534	-27,3108	86,39	0,6035	247,70

Tab. B.2—4: Incidence angle sun (ZED Tracker)

	C_F [m]	r_F [m]	ε [°]	R [m]	$\delta_{i,max}$ [°]
trajectoryZed_S_0.txt	0,5797	0,4583	2,59	0,0758	106,27
trajectoryZed_S_3.txt	0,6536	0,5476	3,89	0,0728	115,78
trajectoryZed_S_6.txt	0,2756	0,2245	3,11	0,0601	98,40
trajectoryZed_S_9.txt	0,0965	0,0145	1,95	0,0381	112,19
trajectoryZed_S_12.txt	0,1496	0,1416	2,11	0,0362	111,88
trajectoryZed_S_15.txt	0,4049	0,3183	2,19	0,0629	112,31
trajectoryZed_S_18.txt	0,5095	0,3860	3,11	0,0641	100,80
trajectoryZed_S_21.txt	0,4031	0,3024	2,70	0,0554	106,20
trajectoryZed_S_24.txt	0,4934	0,4085	2,34	0,0757	91,58
trajectoryZed_S_27.txt	0,4116	0,3617	2,32	0,0684	85,19
trajectoryZed_S_30.txt	0,3231	0,2886	4,98	0,0791	87,57
trajectoryZed_S_33.txt	0,1561	0,0452	5,91	0,0413	94,02
trajectoryZed_S_36.txt	0,2126	-0,0583	5,78	0,0569	91,85
trajectoryZed_S_39.txt	0,2006	-0,0531	5,01	0,0559	74,03
trajectoryZed_S_42.txt	0,2071	-0,0569	2,55	0,0394	57,24
trajectoryZed_S_45.txt	0,1991	-0,0483	1,59	0,0344	46,99
trajectoryZed_S_48.txt	0,1906	-0,0303	1,17	0,0425	41,86
trajectoryZed_S_51.txt	0,2061	-0,0111	1,42	0,0704	43,41
trajectoryZed_S_54.txt	0,2709	-0,0523	1,12	0,0940	55,98
trajectoryZed_S_57.txt	0,2628	-0,0419	1,67	0,0988	55,08
trajectoryZed_S_60.txt	0,3068	-0,0605	1,56	0,1178	67,39
trajectoryZed_S_63.txt	0,2762	-0,0482	0,88	0,1011	67,74
trajectoryZed_S_66.txt	0,2695	-0,0220	0,67	0,1070	65,39
trajectoryZed_S_69.txt	0,2052	0,0261	1,87	0,0905	51,46
trajectoryZed_S_72.txt	0,1876	0,0424	2,08	0,0799	67,29
trajectoryZed_S_75.txt	0,1976	0,0989	2,43	0,0656	98,78
trajectoryZed_S_78.txt	0,3314	0,2555	1,53	0,0777	112,19
trajectoryZed_S_81.txt	0,4643	0,3893	4,58	0,0623	110,97
trajectoryZed_S_84.txt	0,3870	0,3539	3,91	0,0747	148,36
trajectoryZed_S_87.txt	1,2882	0,9298	14,06	0,0559	144,49
trajectoryZed_S_90.txt	1,5804	1,0624	16,67	0,0402	155,37
trajectoryZed_S_93.txt	1,4850	0,9422	13,13	0,0411	147,47
trajectoryZed_S_96.txt	1,7225	1,1168	20,22	0,0402	129,31
trajectoryZed_S_99.txt	1,8671	1,2185	21,56	0,0569	119,48
trajectoryZed_S_102.txt	1,8761	1,1945	26,12	0,0562	104,97
trajectoryZed_S_105.txt	1,9657	1,3160	29,14	0,0796	344,25
trajectoryZed_S_108.txt	1,9054	1,3540	15,48	0,0538	123,56
trajectoryZed_S_111.txt	2,0987	1,4640	5,86	0,0780	145,76
trajectoryZed_S_114.txt	2,0762	1,6565	14,42	0,0417	125,50
trajectoryZed_S_117.txt	1,9492	1,5517	8,21	0,0582	343,08
trajectoryZed_S_120.txt	2,0164	1,6073	20,49	0,0297	345,61
trajectoryZed_S_123.txt	1,9863	1,6594	4,11	0,0260	314,25
trajectoryZed_S_126.txt	1,9613	1,5947	15,53	0,0173	342,29

trajectoryZed_S_129.txt	1,7704	1,6760	17,94	0,0116	345,58
trajectoryZed_S_132.txt	1,9907	1,7671	9,95	0,0133	345,86
trajectoryZed_S_135.txt	1,9480	1,7323	13,77	0,0211	345,90

Tab. B.2—5: Inclination angle target (ZED Tracker)

	C_F [m]	r_F [m]	ε [°]	R [m]	$\delta_{i,max}$ [°]
trajectoryZed_T_0.txt	0,6760	0,5022	4,31	0,0948	106,37
trajectoryZed_T_3.txt	0,5661	0,4283	2,85	0,0725	110,36
trajectoryZed_T_6.txt	0,6294	0,4569	2,60	0,0836	110,52
trajectoryZed_T_9.txt	0,5758	0,4318	1,95	0,0720	158,08
trajectoryZed_T_12.txt	0,4331	0,3191	1,89	0,0584	125,20
trajectoryZed_T_15.txt	0,4439	0,3178	3,36	0,0598	124,11
trajectoryZed_T_18.txt	0,3137	0,2290	1,11	0,0472	126,09
trajectoryZed_T_21.txt	0,1400	0,0765	1,82	0,0389	132,54
trajectoryZed_T_24.txt	0,1247	0,0783	2,20	0,0324	135,81
trajectoryZed_T_27.txt	0,0893	0,0563	2,35	0,0301	136,30
trajectoryZed_T_30.txt	0,1044	0,0575	2,35	0,0236	136,19
trajectoryZed_T_33.txt	0,2127	0,1441	3,71	0,0455	127,91
trajectoryZed_T_36.txt	0,1773	0,1237	3,55	0,0343	124,25
trajectoryZed_T_39.txt	0,2839	0,2149	3,24	0,0454	119,60
trajectoryZed_T_42.txt	0,4207	0,3137	2,96	0,0634	110,82
trajectoryZed_T_45.txt	0,3904	0,2843	3,63	0,0614	108,68
trajectoryZed_T_48.txt	0,2595	0,2107	5,15	0,0475	104,90
trajectoryZed_T_51.txt	0,3730	0,3084	2,61	0,0717	100,44
trajectoryZed_T_54.txt	0,1068	0,0830	3,53	0,0328	106,69
trajectoryZed_T_57.txt	0,0889	0,0276	2,03	0,0494	96,73
trajectoryZed_T_60.txt	0,1219	0,1049	1,21	0,0571	95,18
trajectoryZed_T_63.txt	0,1304	0,1033	1,93	0,0693	91,75
trajectoryZed_T_66.txt	0,0983	0,0491	2,50	0,0698	90,83
trajectoryZed_T_69.txt	0,0965	0,0106	2,26	0,0558	101,45
trajectoryZed_T_72.txt	0,1098	0,1024	3,04	0,0764	93,88
trajectoryZed_T_75.txt	0,0962	0,0661	2,96	0,0764	93,32
trajectoryZed_T_78.txt	0,1175	0,0382	3,33	0,0857	96,37
trajectoryZed_T_81.txt	0,1356	0,0276	3,66	0,0602	105,21
trajectoryZed_T_84.txt	0,0943	0,0128	3,67	0,0547	106,52
trajectoryZed_T_87.txt	0,1274	-0,0359	3,70	0,0468	107,74
trajectoryZed_T_90.txt	0,1327	-0,0335	3,39	0,0489	108,49

Tab. B.2—6: Correlating Experiments (Zed Tracker)

	C_F [m]	r_F [m]	ε [°]	R [m]	$\delta_{i,max}$ [°]
trajectoryZed_0_0.txt	0,6732	0,5481	4,64	0,0669	120,09
trajectoryZed_0_15.txt	0,6508	0,5161	1,94	0,0668	128,69
trajectoryZed_0_30.txt	0,5014	0,3977	1,82	0,0675	115,52
trajectoryZed_0_45.txt	0,5171	0,4131	1,95	0,0707	114,47
trajectoryZed_0_60.txt	0,4363	0,3637	1,69	0,0683	103,23
trajectoryZed_0_75.txt	0,3859	0,3281	4,20	0,0742	109,26
trajectoryZed_0_90.txt	0,5153	0,4289	6,55	0,0822	123,24
trajectoryZed_15_0.txt	0,1019	0,0924	0,79	0,0367	124,65
trajectoryZed_15_15.txt	0,1165	0,1054	2,52	0,0427	117,75
trajectoryZed_15_30.txt	0,0673	0,0154	1,98	0,0256	124,77
trajectoryZed_15_45.txt	0,1722	0,1229	2,71	0,0310	127,59
trajectoryZed_15_60.txt	0,1851	-0,0472	2,37	0,0734	96,82
trajectoryZed_15_75.txt	0,1525	-0,0021	3,50	0,0912	96,19
trajectoryZed_15_90.txt	0,1805	-0,0785	3,94	0,0442	108,09
trajectoryZed_30_0.txt	0,7071	0,5187	5,05	0,1182	94,06
trajectoryZed_30_15.txt	0,5485	0,4030	6,58	0,1010	93,11
trajectoryZed_30_30.txt	0,3854	0,3301	4,71	0,0680	98,69
trajectoryZed_30_45.txt	0,3132	0,2782	3,16	0,0868	118,85
trajectoryZed_30_60.txt	0,2538	0,2234	2,60	0,0819	82,00
trajectoryZed_30_75.txt	0,2360	0,2095	5,15	0,1074	109,57
trajectoryZed_30_90.txt	0,2840	0,2515	6,09	0,0820	84,20
trajectoryZed_45_0.txt	0,2974	0,2369	8,39	0,0968	93,99
trajectoryZed_45_15.txt	0,2007	0,1673	4,13	0,1206	89,11
trajectoryZed_45_30.txt	0,1379	-0,0412	1,50	0,0278	85,10
trajectoryZed_45_45.txt	0,2389	-0,0630	4,50	0,0556	69,91
trajectoryZed_45_60.txt	0,4116	-0,1313	2,57	0,1175	71,97
trajectoryZed_45_75.txt	0,3369	-0,0116	4,92	0,1487	73,81
trajectoryZed_45_90.txt	0,3280	-0,0112	2,95	0,1348	84,39
trajectoryZed_60_0.txt	0,2128	-0,0415	3,98	0,0965	93,20
trajectoryZed_60_15.txt	0,0614	0,0680	0,66	0,0582	72,62
trajectoryZed_60_30.txt	0,1692	-0,0024	3,35	0,0584	54,03
trajectoryZed_60_45.txt	0,2934	-0,0437	0,48	0,1175	56,96
trajectoryZed_60_60.txt	0,3086	-0,0271	1,20	0,1310	42,80
trajectoryZed_60_75.txt	0,2913	-0,0448	1,56	0,1263	56,11
trajectoryZed_60_90.txt	0,3091	-0,0906	2,24	0,0746	67,51
trajectoryZed_75_0.txt	0,2485	0,2016	9,70	0,1176	105,96
trajectoryZed_75_15.txt	0,2218	0,2087	2,51	0,0734	94,11
trajectoryZed_75_30.txt	0,1438	0,0790	3,79	0,0751	74,69
trajectoryZed_75_45.txt	0,1789	0,1085	1,78	0,0875	60,56
trajectoryZed_75_60.txt	0,2455	0,0825	1,27	0,1322	70,60
trajectoryZed_75_75.txt	0,1588	0,1210	2,67	0,1107	70,16
trajectoryZed_75_90.txt	0,1675	0,0972	2,33	0,1113	80,73
trajectoryZed_90_0.txt	0,6384	0,5443	15,81	0,1071	147,82

trajectoryZed_90_15.txt	0,5458	0,5201	6,03	0,0849	120,79
trajectoryZed_90_30.txt	0,5821	0,5052	4,19	0,0608	146,41
trajectoryZed_90_45.txt	0,3443	0,3141	2,33	0,0755	115,48
trajectoryZed_90_60.txt	0,5121	0,3572	7,19	0,0942	94,03
trajectoryZed_90_75.txt	0,8553	0,5067	2,43	0,1113	104,21
trajectoryZed_90_90.txt	1,4028	0,9530	1,79	0,1261	120,01
trajectoryZed_105_0.txt	1,2650	0,9862	14,43	0,0704	160,13
trajectoryZed_105_15.txt	1,3378	0,9388	1,08	0,0926	170,34
trajectoryZed_105_30.txt	1,5730	1,1407	11,23	0,0983	171,62
trajectoryZed_105_45.txt	1,4402	1,1738	21,51	0,0793	201,28
trajectoryZed_105_60.txt	0,8806	0,6748	2,15	0,0537	174,98
trajectoryZed_105_75.txt	1,8926	1,4310	6,08	0,0982	150,60
trajectoryZed_105_90.txt	2,1462	1,4390	1,96	0,1029	312,62
trajectoryZed_120_0.txt	1,6907	1,3578	18,29	0,0454	279,14
trajectoryZed_120_15.txt	1,6035	1,2254	7,15	0,0632	286,14
trajectoryZed_120_30.txt	1,8564	1,4132	7,66	0,0370	346,54
trajectoryZed_120_45.txt	1,7088	1,2269	72,15	0,0132	326,23
trajectoryZed_120_60.txt	1,9241	1,5752	9,92	0,0522	172,51
trajectoryZed_120_75.txt	2,0263	1,6008	10,95	0,0504	317,85
trajectoryZed_120_90.txt	2,3006	1,2820	5,65	0,0867	344,26
trajectoryZed_135_0.txt	0,4955	0,2330	10,51	0,0593	346,65
trajectoryZed_135_15.txt	1,4872	-0,1901	46,73	0,0540	318,35
trajectoryZed_135_30.txt	2,0883	1,7535	2,87	0,0161	72,68
trajectoryZed_135_45.txt	1,9315	1,7894	33,43	0,0181	344,94
trajectoryZed_135_60.txt	1,9231	1,6706	10,28	0,0133	346,59
trajectoryZed_135_75.txt	1,4017	1,3577	18,92	0,0281	346,65
trajectoryZed_135_90.txt	0,8108	0,4894	8,42	0,0589	324,97