Satellite formation and instrument design for autonomous meteor detection

A thesis accepted by the
Faculty of Aerospace Engineering and Geodesy of the University of Stuttgart
in partial fulfilment of the requirements for the degree of
Doctor of Engineering Sciences (Dr.-Ing.)

Submitted by

Jona Petri

born in
Göttingen, Germany

Main referee:	Prof. Dr. Sabine Klinkner
Co referee:	Jérémie Vaubaillon
Oral exam date:	09.09.2022

Institute of Space Systems

University of Stuttgart

2022

Contents

Kurzfassung

Meteore sind Leuchterscheinungen, welche auftreten, wenn ein Meteoroid in die Erdatmosphäre eintritt. Meteoroide sind Fragmente von Kometen und Asteroiden, welche seit den Anfängen des Sonnensystems so gut wie unverändert sind. Daher können durch eine Beobachtung von Meteoren Rückschlüsse auf die Entwicklung des Sonnensystems gezogen werden. Außerdem kann durch Meteorbeobachtung der Meteoriden Fluss (Anzahl der Meteoriden pro Zeit- und Flächeneinheit) bestimmt werden. Dieser ist wichtig für die Planung und Durchführung von Aktivitäten im Weltraum.

Im Vergleich zu bodengestützten Beobachtungen hat eine satellitengestützte Beobachtung von Meteoren mehrere Vorteile: Zum einen ist die Abdeckung größer, d.h. es können mehr Meteore beobachtet werden. Weiterhin ist eine satellitengestützte Beobachtung unabhängig vom Wetter und es findet keine Dämpfung des vom Meteor erzeugten Lichts durch die Atmosphäre statt.

In dieser Arbeit wurden zwei Beispielmissionen verwendet, um die jeweilige Mission und ein Instrument für die visuelle Beobachtung von Meteoren zu konizipieren. Die erste Mission ist Stuttgart Operated University Research Cubesat for Evaluation and Education (SOURCE), ein *CubeSat*, der für Technologiedemonstrationen, Wiedereintrittsuntersuchungen und Meteorbeobachtungen eingesetzt wird. Im Rahmen der Mission soll das Instrument zur Meteorbeobachtung qualifiziert und getestet werden. Die zweite Mission heißt Formation for Analysis of Cosmic Particles (FACIS), eine Mission, die aus einer Formation von zwei Kleinsatelliten besteht. Das Missionsziel ist Meteorbeobachtung und Messung von Staubpartikeln. Da zwei Satelliten verwendet werden, soll diese Mission Meteore stereoskopisch beobachten, um die Trajektorie der Meteore zu bestimmen.

Die zwei wesentlichen Herausforderungen einer satellitengestützten Meteorbeobachtung wurden in dieser Arbeit behandelt. Die erste ist die Missionsanalyse und die Auslegung des Instruments. Dies beinhaltet eine Analyse des Einflusses der Satellitenbus Parameter (beispielsweise die Genauigkeit der Lagebestimmung) und der Formationsparameter (beispielsweise der Satellitenabstand) auf die wissenschaftlichen Ergebnisse einer (stereoskopischen) Meteorbeobachtungsmission. Weiterhin muss das Instrument zur Meteorbeobachtung abhängig von der wissenschaftlichen Zielsetzung der Mission ausgelegt werden. Daher wurden im Rahmen dieser Dissertation verschiedene Simulationen entwickelt und verwendet, um die wissenschaftlichen Ergebnisse in Abhängigkeit von den Satellitenbus- und Formationsparametern sowie den Instrumentenparametern zu bestimmen. Diese Simulationen wurden verwendet, um die Mission zu planen, einschließlich der Auslegung des Instruments und dem Aufstellen von Anforderungen an den Satellitenbus. Außerdem konnten die idealen Formationsparameter in Abhängigkeit von der wissenschaftlichen Zielsetzung und dem Instrumentendesign bestimmt werden.

Die zweite Herausforderung bei einer weltraumgestützten Meteorbeobachtungsmission ist die Begrenzung der Datenmenge, die ein Satellit zur Bodenstation senden kann. Dies erfordert die Verarbeitung der Bilddaten auf dem Satelliten, damit nur Bilder, die einen Meteor enthalten, zur Bodenstation gesendet werden. Dies geschieht durch einen Algorithmus, welcher alle aufgenommenen Bilder verarbeitet und Meteore detektiert. Bestehende Detektionsalgorithmen können nicht verwendet werden, da sich der Satellit während der Meteorbeobachtung bewegt und somit auch der Hintergrund in Bewegung ist. Daher wurde im Rahmen dieser Arbeit ein neuer Algorithmus zur Meteorerkennung entwickelt, basierend auf der Berechnung des optischen Flusses. Dieser Algorithmus wurde getestet, indem eine künstliche Meteorsimulation entwickelt und verwendet wird, um Testdaten zu erzeugen. Weiterhin wurde ein Prüfstand entwickelt und gebaut, um diese Daten anzuzeigen und

mit dem Instrument abzubilden. Eine Herausforderung für die Entwicklung des Algorithmus war die begrenzte Rechenleistung auf dem Satelliten. Daher wurde der entwickelte Algorithmus durch die Verlagerung von Funktionen auf spezielle Hardware des verwendeten Bordcomputers beschleunigt.

Letzendlich können das Beobachtungskonzept und die potentielle wissenschaftlichen Leistungsfähigkeit beider Missionen unter Berücksichtigung der Instrumentenparameter, der Parameter des Satellitenbusses und der Formation sowie der Performanz des Algorithmus bestimmt werden. Für beide Missionen konnte ein Konzept entwickelt werden, das eine signifikante Beobachtung von Meteoren ermöglicht.

Zusammenfassend zeigt diese Arbeit, dass eine weltraumgestützte Meteorbeobachtungsmission mit Hilfe von Kleinsatelliten möglich ist. Die wichtigsten Aspekte des Missionsdesigns wurden analysiert, um zwei verschiedene Missionen zu entwerfen. Die Hauptaspekte, die in dieser Arbeit behandelt wurden, sind das Instrumentendesign, die Anforderungen an den Satellitenbus und die Formation sowie die Algorithmenentwicklung. Alle drei Aspekte konnten erfolgreich abgehandelt werden, die Missionen wurden geplant und ein Prototyp des Instruments wurde entwickelt und qualifiziert. Darüber hinaus konnte das Instrument für die SOURCE Mission gebaut werden, einschließlich der Entwicklung des Detektionsalgorithmus.

Abstract

A meteoroid entering the Earth atmosphere causes a light phenomenon called meteor. Meteoroids origin from comets or asteroids, these small particles are mostly unchanged since the formation of the solar system. Therefore, observing meteors gives insight on how our solar system evolved and from what materials it consists. Additionally, meteor observations are used to improve meteoroid flux models, which are needed to safely plan space activities.

Meteor observations from space offer several advantages compared to ground-based meteor observations, such as greater coverage and an unobstructed view to the meteor as well as weather independence. Using two satellites for meteor observations allows to calculate the trajectory of a meteor and determine its parent body.

In this thesis two example satellites are used to develop and design a mission and an instrument for the visual observation of meteors. The first mission is called Stuttgart Operated University Research Cubesat for Evaluation and Education (SOURCE), a three unit *CubeSat* dedicated to technology demonstrations, demise investigation and meteor observation. Here, a visual monochromatic camera is used to observe meteors and qualify the instrument. The second mission is called Formation for Analysis of Cosmic Particles (FACIS), which consists of a formation of two identical small satellites dedicated to meteor observation and dust measurements. Since two satellites are used, meteors are observed stereoscopically to determine meteor trajectories.

Two main challenges of spaceborne meteor observation are addressed in this thesis: The first one is the design of the mission and the instrument. This includes analysing the influence of the satellite bus parameters (e. g. attitude knowledge accuracy) and formation parameters (e. g. satellite distance) on the scientific output of a

(stereoscopic) meteor observation mission. Furthermore, the instrument parameters must be analysed and optimized depending on the scientific objective of the mission. Thus, different simulations are developed to evaluate the scientific output depending on satellite bus and formation parameters as well as instrument parameters. These simulations were used to develop the mission, including the design of the instrument and deriving requirements for the satellite bus. Furthermore, the ideal formation parameters depending on the scientific objective and instrument design could be determined.

The second challenge of a space based meteor observation mission is the limited downlink capacity of a satellite. This requires onboard processing of the image data. An algorithm must be used, to identify images containing a meteor and downlink only these images. Existing detection algorithms can not be used, since the satellite moves during an observation and thus, the background is moving as well. Therefore, a new meteor detection algorithm called *Spaceborne MEteor Detection ALgorithm (SpaceMEDAL)* based on optical flow calculations is developed. This algorithm is tested by developing and using an artificial meteor simulation called *Artificial Meteorvideo Simulation Software (ArtMESS)* to generate test data and a test bed to display and image this data. A challenge for the algorithm is the limited processing power. Thus, the developed algorithm is accelerated by moving functions to dedicated hardware of the used on board computer.

Finally, the observation concept and scientific output of both missions can be determined, by taking into account the instrument parameters, satellite bus and formation parameters as well as the algorithm performance and operational constraints of the satellite. For both missions a concept resulting in a significant observation of meteors could be developed.

All in all, this thesis shows that a space based meteor observation mission using small satellites is possible. The main aspects of the mission design were analysed in order to design two different missions. The main aspects addressed in this thesis are the instrument

design, the satellite bus and formation requirements and the algorithm development. All three aspects could be successfully revised and the missions and the instrument could be developed. Furthermore, a demonstrator instrument for the SOURCE mission could be built, including the development of the detection algorithm.

Preface and Acknowledgments

When I started working on at the Institute of Space Systems in 2017, I did not know where this journey would lead me. To be honest, I still don't know now either. But I what I do and did know is that I am intrigued by sciences and space technology. Thus, I am very thankful to Prof. Dr. Sabine Klinkner for giving me the opportunity of working within the awesome small satellite department at the Institute of Space Systems and pursuing my PhD. Sabine Klinkner did not only provide a job, but also a great working environment. Furthermore, I am thankful for the valuable feedback she provided during my thesis.

I also want to thank Jérémie Vaubaillon for his advice and discussions about space based meteor observations and providing the SWARMS software which I could adapt and use in my thesis. Also, I want to thank Jérémie for co-supervising this thesis.

Generally the Institute of Space Systems is full of people I enjoyed working with. Especially I want to thank Stefan Löhle for establishing the idea of space based meteor observation at the Institute in the first place, his insights to meteor sciences and establishing contact to Jérémie. I also want to thank Thomas Albin for the useful discussions in the early phase of my PhD and providing the CILBO data base for usage in my thesis.

I want to thank the whole small satellite department, which is full of wonderful people and great colleagues, which made working at the Institute a great experience. I thank all current and former colleagues for their encouragement, useful discussions as well as fostering a great working environment. Especially I want to thank Michael Lengowski for making me feel welcome from the very first day, helping me navigate around all the small and big challenges and issues occurring when working at the Institute. I also want to

thank Michael for his help during several thermal vacuum tests and generally providing a lot of support and setting up a great working environment. Thanks also to Susann Pätschke for proof reading my thesis and our mutual motivation and discussions during our traditional coffee and tea break. Thanks to Steffen Gaißer I was able to understand and use the Flight software framework, which was very helpful during my thesis.

Furthermore, this thesis was supported by many students writing their thesis about specific parts of a space borne meteor observation. I want to thank Julia Zink, Jens Polzin and Niklas Geier for writing their thesis within the scope of my thesis. I also want to thank all Students working in the SOURCE team for their contribution to the satellite development. Especially I want to thank the Payload subsystem, for all their work. Namely I thank, in no specific order, Tristan Meyer, Philipp Hoffmann and Marcel Liegibel for supporting the camera tests, calibrating the testbed and developing the meteor simulation.

Finally, I want to thank my Family for providing support and encouragement during all my studies and especially during my PhD. Specifically I want to thank my mother Judith Petri for inspiring me from a young age for technology, maths and science and always supporting me. I want to thank my father Dr. Wolfgang Petri for his encouragement, for many philosophic discussions at the campfire and proof reading the thesis. I also want to thank my brother Micha Petri, for always looking out for me. There are many more reasons I can not properly include in this paragraph why I am thankful for having the three of you as my family.

Acronyms

ACS Attitude Control System

ADC Analog-to-digital Converter

AOCS Attitude and Orbit Control System

API Application Programming Interface

ArtMESS Artificial Meteorvideo Simulation Software

ASTOS Analysis, Simulation and Trajectory Optimization Software for Space Applications

CAD Computer-Aided Design

CAMS Cameras for Allsky Meteor Surveillance

CCD Charge-coupled Device

CDF Cumulative Distribution Function

CILBO Canary Island Long-Baseline Observatory

CME Coronal Mass Ejection

CMOS Complementary metal-oxide-semiconductor

COTS Commercial off-the-shelf

CPU Central Processing Unit

CVB Common Vision Blox

DTB Device Tree Blob

ECSS European Cooperation for Space Standardization

EMVA European Machine Vision Association

ESA European Space Agency

FACIS Formation for Analysis of Cosmic Particles

FLP Flying Laptop

FN False negative

FOV Field of View

FP False positive

FPGA Field Programmable Gate Array

fps frames per second

FRIPON Fireball Recovery and InterPlanetary Observation Network

FSFW Flight Software Framework

FWHM Full Width Half Maximum

GIMP GNU Image Manipulation Program

GNU GNU's Not Unix

GPS Global Positioning System

GSD Ground Sampling Distance

HDF Hardware Description File

HDMI High Definition Multimedia Interface

HSV Hue, Saturation, Value

IAG Institute of Aerodynamics and Gas Dynamics

IFOV Instantaneous Field of View

IMCCE Institut de Mécanique Céleste et de Calcul des Éphémérides

IP-Core Intellectual Property Core

IRS Institute of Space Systems

ISS International Space Station

JWST James Webb Space Telescope

KSat e.V. Small Satellite Student Society at the University of Stuttgart

LEO Low Earth Orbit

LTAN Local Time of Ascending Node

MeSHCam Meteor observation, Star and Horizon tracking Camera

MetDet algorithm Meteor Detection Algorithm

MetObs camera Meteor Observation Camera

MetObs controller Meteor Observation Controller

MetObs System Meteor Observation System

MOTS Meteor Orbit and Trajectory Software

MSX Midcourse Space Experiment

OBC On Board Computer

OpenCV Open Source Computer Vision Library

PCDU Power Control and Distribution Unit

PERC Planetary Exploration Research Center

PLOC Payload On-Board Computer

PPS Pulse per Second

PRIma PR Imager

PUS Packet Utilisation Standard

S-CUBE Shootingstar Sensing Satellite (S^3)

SDK Software Development Kit

SNPP Suomi National Polar-orbiting Partnership

SNR Signal to Noise Ratio

SoC System on a Chip

SOURCE Stuttgart Operated University Research Cubesat for Evaluation and Education

SpaceMEDAL Spaceborne MEteor Detection ALgorithm

SPOSH Smart Panoramic Optical Sensor Head

SSO Sun-synchronous Orbit

STD Standard Deviation

STL Standard Template Library

SWARMS Simulator for Wide Area Recording of Meteors from Space

SWARMSv2 SWARMS Version 2

SWIMSat Space Weather and Impact Monitor Satellite

tbd to be defined

TP True positive

TT&C Telemetry, Tracking, and Command

UHF Ultra High Frequency

UV ultraviolet

UVISI Ultraviolet and Visible Imagers and Spectrographic Imagers

VIIRS Visible Infrared Imaging Radiometer Suite

List of Symbols

D	Aperture diameter
$\mu_{e\_background\_px}$	Background noise
k	Bitdepth of the sensor
μ_I	Dark current in $e\text{-}/(px \cdot \mathrm{s})$
Δm_{dis}	Magnitude reduction due to distance to meteor
Δm_{move}	Magnitude reduction due to meteor movement
t_{dwell}	Dwell time
DR	Dynamic range of a camera
t_{exp}	Exposure time
FF	Pixel fill factor
$\Delta\lambda$	Filter bandwidth
f	Focal length
FR	Frame rate of a camera
μ_{e_full}	Full-well capacity of one pixel
$FWHM$	Full Width Half Maximum
E	Irradiance reaching the target
L	Radiance emitted from light source
c	Speed of light
L_λ	Spectral radiance emitted from light source

τ	Luminous efficiency, describes how much kinetic energy is transfered to light
I	Luminous Intensity of a meteor
m_{abs}	Absolute magnitude
m_{app}	Apparent magnitude
μ_y	Mean grey signal in *DN* (Digital Numbers)
$\mu_y - \mu_{y.dark}$	Mean photo-induced grey signal in *DN* (Digital Numbers)
M	Meteoroid mass
V_m	Meteor speed
μ_e	Sensor signal in *e-/px*
φ	Phase angle between two satellites
px	Pixel
A_{PX}	Area of a pixel
W_{PX}	Width of a pixel
h	Planck constant
η	Quantum efficiency
ξ	Radiant angle
R	Range to meteor
α	Tilt angle of a satellite, angle between nadir and optical axis
μ_{e_sat}	Saturation capacity of one pixel
μ_{p_sat}	Saturation irradiation of one pixel
μ_{p_min}	Absolute sensitivity threshold of one pixel
r	Sensor pixel resolution
SNR	Signal to noise ratio
K	Camera system gain in *DN*/*e-* (Digital Numbers/Electrons)
α	Camera tilt angle

T	Lens transmission
TR	Tumbling rate of a satellite
λ	Wavelength

List of Figures

List of Tables

As I stand out here in the wonders of the unknown at Hadley,
I sort of realize there's a fundamental truth to our nature:
Man must explore!

— David Scott, 1974

1. Introduction

The term meteor describes the lightning phenomenon occurring when a meteoroid enters the Earth atmosphere. Meteors have been observed for centuries, beginning with naked eye observations, later with binoculars, photo plates and today also with digital cameras. Meteoroids are small particles from comets or asteroids, which are mostly unchanged since the formation of the solar system. Therefore, observation of meteors gives insight on how our solar system evolved and from what materials it consists. Furthermore, determining the meteoroid flux by observing meteors is important to improve existing meteoroid flux models. Those models are crucial for planning safe space activities such as astronauts extra-vehicular activities or satellite missions. For example, recently the new James Webb Space Telescope (JWST) was hit by an unexpectedly large meteoroid (see [1]). While JWST still fulfils all requirements, this incident underlines the importance of understanding the meteoroid environment for space missions.

The technological advancement generally allows to built more capable instruments, which in turn gives more insight to the objects observed. This is also true for meteor observation: Digital cameras with low noise in combination with powerful processing algorithms, allow the continuous observation of meteors during all nights, detecting faint meteors and thus helping to characterize meteor phenomena.

In recent years, satellites became more affordable. The reasons for this is mainly the miniaturization of electronics and usage of Commercial off-the-shelf (COTS) components, e. g. in small satellites for the Low Earth Orbit (LEO). As a result, it is now technically and economically feasible to carry out a satellite mission with the objective of meteor observation.

1.1. Advantages of meteor observation from space using small satellites

While ground observation of meteors are common for quite some time, ranging from naked eye to video observations, a space borne instrument has several advantages and can aid the ground observations.

First, a space borne instrument can cover a much greater area than a ground based system, due to the higher distance from orbit. This allows for an observation of more meteors and thus enhancing the available data base and leading to more robust statistic results. Of course, this depends on the instrument parameters and its orientation. This was for example analysed by Bouquet et al. (see [2]).

Furthermore, a satellite based instrument is independent from the weather, while ground based systems can only observe during a clear night.

Also, the instruments observe the meteor from above the atmosphere, meaning the light produced is not absorbed by the atmosphere. This gives an unobstructed view to the meteor and gives the possibility to observe in different wavelength bands (e. g. in the ultraviolet (UV) band, see [3]). This allows the detection of Carbon in the UV spectrum, allowing identification of extra-terrestrial material.

Existing ground based instruments are mostly located in the northern hemisphere. As a result, some meteor showers are not well observed, due to the usually bad weather conditions when the shower peaks (e. g. Quadrantids in January, see [4, p.357]). Furthermore,

the radiant (apparent location from where the meteors of one shower seem to originate) is below the horizon during the best observation time (midnight). A spaceborne instrument is independent from the seasons and the location of the radiant.

Even more insights to meteors can be gained if two or more satellites are used to observe the same meteor. In this case, the trajectory of the meteor can be determined, which is used to determine the parent body of the meteoroid.

In conclusion, a spaceborne observation of meteors can aid the ground observations, enhance the statistical basis of meteor research and allow for new insights to meteor physics. Besides contributing to fundamental research about our solar system, the data can be used to improve meteor shower prediction models in order to protect assets in space or safely plan extra-vehicular activities.

1.2. Technical challenges of meteor observation from a satellite

As outlined in the previous section, a spaceborne meteor observation offers many advantage. However, some challenges are associated with instruments on satellites in general and with an meteor observation instrument specifically.

First, the satellite and the instrument are constantly moving around the Earth, which means attitude and position change constantly. This makes the evaluation of the data more complicated, since the position and attitude are important to determine the direction of the meteor and thus the assignment to a meteor shower. In case of a stereoscopic meteor observation, this data is also needed to determine the meteor trajectory. Finally, position and attitude data have a certain noise level, which decreases the accuracy of trajectory determination.

Second, due to the orbit altitude of the satellites, meteors appear fainter compared to ground based observations. The typically altitude in LEO is around 300 km to 800 km, meteors occur in an

altitude of about 100 km (see [4]). Due to the higher distance, the signal reaching the camera is weaker.

Third, onboard processing of the data is needed. While meteor showers can be predicted, meteors still occur randomly and it can not be predicted when exactly a meteor occurs. This means an instrument must constantly observe the Earth during eclipse. Daylight observations are possible, but meteors are usually too faint to be observed during the day with visual cameras.

Visual cameras are widely used for meteor observation, but they are producing a large quantity of data. Even with a low framerate and resolution about 4 GB can be produced in 30 min. The downlink capacity of a satellite is limited and is depending on the available resources (e. g. available power). Thus, the data must be processed onboard, in order to only downlink data containing a meteor observation.

Here, the fourth challenges becomes apparent: The volume and power available on a satellite is limited, therefore the processing power of the onboard computer is limited too. The algorithm must be able to process all data with the available resources. For comparison, the Central Processing Units (CPUs) used for payload processing are usually less or equally powerful as CPUs used in modern smartphones.

Finally, the satellite can only be controlled from the ground by sending commands. This must be taken into account when developing the instrument software (control software and algorithm). The software must be able to run autonomously and allow for remote adaption of parameters.

1.3. Example missions: SOURCE and FACIS

In this thesis, two missions dedicated to meteor observation are used as an example to design the instrument. The first satellite is the *Stuttgart Operated University Research Cubesat for Evaluation and Education (SOURCE)*, which is a three unit *CubeSat* developed at

the Institute of Space Systems (IRS) in cooperation with the Small Satellite Student Society at the University of Stuttgart (KSat e.V.). The scientific mission goals are atmospheric and re-entry science using different sensors as well as meteor observation using a visual camera.

The second mission is the *Formation for Analysis of Cosmic Particles (FACIS)*, a formation of two micro satellites (about 30 kg). Besides serving as a technology demonstration, the main scientific objectives are the stereoscopic observation of meteors as well as dust measurements using a dust sensor.

Both missions are presented in more detail in the following sections.

The SOURCE mission

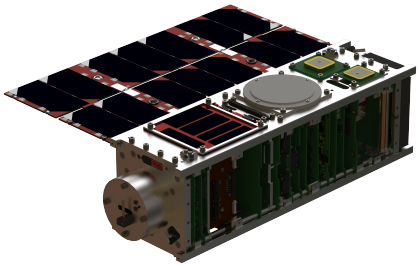


Figure 1.1.: CAD drawing of the SOURCE satellite with one solar panel already deployed. The second panel is not shown. (Credit: SOURCE Team)

The *SOURCE* satellite as a Computer-Aided Design (CAD) model is shown in Figure 1.1. The following information is taken from the documentation (see [5]) prepared by the *SOURCE* team of which the author is a member.

The *SOURCE* mission goals has three primary and one secondary objective. The primary objectives are:

1. Student education

2. Satellite platform development for future missions
3. Technology demonstration

The project gives students the opportunity to work on a satellite project and gain technical and procedural experience. Furthermore, a satellite platform for future missions should be developed. Finally, the satellite serves as a platform for technology demonstration. This includes e. g. smart heaters, new solar cells and a multifunctional sandwich structure which includes different sensors (gyroscope and accelerometer, radiation sensors).

Table 1.1.: SOURCE mission facts

Property	Value
Orbit	Sun-synchronous/ISS orbit
Orbit altitude	450 km to 500 km
Mass	$\sim$ 5 kg
Size	30 cm $\times$ 10 cm $\times$ 10 cm
Mission duration	$\sim$ 1 *year*
Attitude determination	Sun sensors, magnetometer, experimental Startracker
Attitude control	Magnetorquer
Pointing accuracy	5°
Position determination	Global Positioning System
Power generation	10 solar panels (327 $\times$ 80.25 $\times$ 1 mm) á 7 cells
Maximum power generation	30 W
Payload data rate per day	100 MB using S-band

The secondary objective is the demise investigation, which is divided into two parts:

1. Atmospheric and re-entry science

2. Meteor observation

A sensor suite is used for atmospheric and re-entry science. This sensor suite should verify the simulations software PICLas (developed by IRS and Institute of Aerodynamics and Gas Dynamics (IAG)). PICLas is a flexible particle-based plasma simulation suite for the numerical simulation and modelling of re-entry flow. The sensors suite includes two FIPEX sensors to measure atomic oxygen, pressure sensors as well as two kinds of heat flux sensors. During the re-entry of *SOURCE*, these sensors deliver data about the atmosphere, which is downlinked using the satellite network Iridium. Using this network is necessary, since the re-entry is short and most likely not over a ground station.

For the meteor observation, a visual monochromatic camera is used. Since *CubeSats* are generally limited in terms of available power and downlink capacity, the meteor observation is a demonstration for future missions. This allows to demonstrate the feasibility of space based meteor observations as well as testing the camera and on-board meteor detection algorithm. The import mission parameters are summarized in Table 1.1.

The FACIS mission

The *FACIS* mission is a joint mission planned by the IRS of the University of Stuttgart and the Technische Universität Berlin. While the satellite bus is based on the TUBiX20 platform developed by TU Berlin, the IRS provides the payload and the data downlink system. The mission consists of a formation of two identical satellites (30 kg each, see Figure 1.2) in Low Earth Orbit dedicated to the stereoscopic meteor observation and dust measurements. A miniature dust telescope is used to measure the dust around the satellite and delivers important data of the dust environment during meteor observation. For the meteor observation a visual camera system should be used. Since two satellites with overlapping instrument Field of Views (FOVs) are used, a stereoscopic meteor observation is possible. This allows to calculate the corresponding meteor trajectory and thus, to determine the parent body. In the following, the term

Table 1.2.: FACIS satellite facts

Property	Value
Orbit	Sun-synchronous/ISS orbit
Orbit altitude	300 km to 565 km
Mass	$\sim$ 30 kg
Size	36 cm $\times$ 36 cm $\times$ 34 cm
Mission duration	$\sim$ 2 *year*
Attitude determination	Sun sensors, magnetometer, startracker, gyroscopes
Attitude control	Magnetorquer, reaction wheels
Position determination	Global Positioning System
Pointing accuracy	**tbd**
Attitude determination accuracy	36″
Maximum power generation	90 W using Solar Panels
Payload data rate per day	1.6 GB using X-band

instrument describes the camera, detection algorithm as well as a computer to control the camera and store images.

In Table 1.2 the main mission data is summarized, the data is taken from the paper about Phase A (see [7]), which is based on a Master thesis (see [6]). Currently the project is in Phase A, the data should be seen as a baseline. In this thesis, further details regarding the satellite bus requirements as well as mission parameters are detailed. Furthermore, the idea of this mission serves as an example to develop requirements for future stereoscopic meteor observation missions using two satellites.

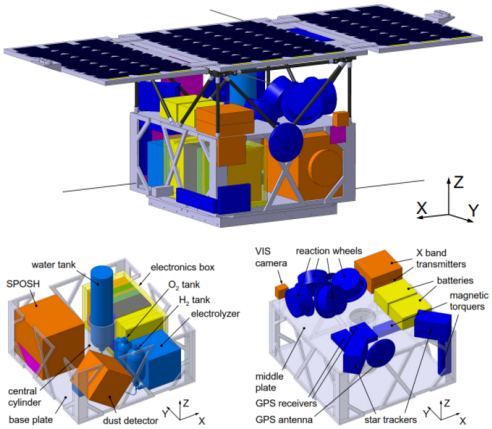


Figure 1.2.: Preliminary CAD drawing of the FACIS satellite (Credit: MA Fabian Hufgard [6])

1.4. Further planned meteor observation missions and instruments

Due to the advantages of space borne meteor observations described in the previous section, several missions and instruments have been proposed. Furthermore, some meteors could already be observed from space. In this section, a short overview of successful and proposed missions as well as instruments should be given as a context for the missions described in this thesis (see Section 1.3).

METEOR The project METEOR (see [8]) is a Japanese project from Planetary Exploration Research Center (PERC). In the course of the project, a high definition colour camera was developed and deployed onboard of the International Space Station (ISS). The camera has a wide FOV of 57.8° and is very sensitive due to the low f-number of 0.95. The camera observes meteors in the visual light

spectrum. Several meteors could be observed and videos of meteors have been published on Youtube [1].

Meteorix The Meteorix satellite is a three unit *CubeSat* currently developed by the Institut de Mécanique Céleste et de Calcul des Éphémérides (IMCCE), Sorbonne Université and other partners (see [9]). The satellite development is done by a student team and supervised by University researchers. The satellite is equipped with a camera developed by *3D plus*, dedicated to the observation of meteors in the visual light spectrum. Due to the limited data downlink capacity, it is planned to process all image onboard and downlink only images with meteors by using an meteor detection algorithm based on optical flow. The main goal of this mission is to observe sporadic and shower meteors over the course of one year to improve meteor models and get good statistical data about meteoroids and space debris entering the atmosphere.

SPOSH The Smart Panoramic Optical Sensor Head (SPOSH) camera is a satellite instrument designed for the observation of faint transient noctilucent phenomena (see [10]), which includes meteors. The camera sensors has a high sensitivity, which is needed for the observation of faint meteors. Furthermore, it has a large FOV of 120°. The camera is also equipped with a processing unit to detect the desired phenomena. Currently this algorithm is optimized for the detection of meteors at a frame rate of 3 *fps*. The instrument is designed for a use on a spacecraft, but is currently also used for ground based observations.

SWIMSat Space Weather and Impact Monitor Satellite (SWIMSat) is a six unit *CubeSat* (see [11]). Contrary to the previous missions, it should be stationed in a geostationary orbit and equipped with a propulsion system. The two proposed missions goals are the

[1] `https://www.youtube.com/channel/UC1dWPtTzsDqiLXsrJSocinA/videos`

observation of Coronal Mass Ejections (CMEs) and impacts of large meteoroids. Two instruments are used for this purpose: The Coronagraph for CME observation and a visual camera for meteor observations. Furthermore, it is proposed to develop a reduced version of this satellite (three units) without a propulsion system and the Coronagraph. Then the satellite is placed in a LEO to allow for the observation of smaller meteors. Since the satellite is half the size of the larger version, it is also considered to built two of them to allow for meteor trajectory determination (see [12]). Autonomous event detection is also planned for this mission.

As outlined above, several missions have been proposed, but currently no dedicated meteor observation satellite was launched. However, successful observations from the ISS could be made, showing the feasibility. Furthermore, the multiple proposals from different working groups show the general interest and need for a dedicated mission for spaceborne meteor observation.

1.5. Thesis objective and outline

The main goal of this thesis is the feasibility analysis of a spaceborne meteor observation mission, based on the results the design of the meteor observation instrument (camera as well as algorithm) and the derivation of requirements for the satellite bus and formation. As outlined above, a spaceborne meteor observation is proposed by several institutions, but so far no detailed analysis of the satellite and formation requirements and instrument development for a stereoscopic observation mission has been done. Therefore, this thesis aims to answer the question, if a stereoscopic meteor observation mission with small satellites is feasible and what scientific output can be expected. This also includes the development of the onboard meteor detection algorithm, which is crucial for all planned meteor observation missions.

Therefore, this thesis shall provide a major step towards the first dedicated stereoscopic meteor observation mission. In detail,

it should be analysed how the scientific output of the two example missions *FACIS* and *SOURCE* can be optimized, by defining satellite bus and formation parameters depending on the scientific objective and instrument design. Therefore, a comprehensive analysis of the mission is required, to identify design driving parameters of the satellite bus, instrument and formation on the scientific output. Finally, their effect on the scientific output can be quantified in order to define requirements necessary for a successful mission. Here, the challenge is to first identify the design driving parameters, which requires setting up simulations in order to also quantify their effect on the scientific output. This is complex, since a wide variety of parameters exists, which influence each other and can not be considered isolated. For example, some instrument parameters influence the ideal formation parameters, thus the formation parameters can not be analysed without taking into account the instrument.

Furthermore, the instrument should be developed, which includes deriving scientific and instrument requirements, camera and lens selection as well as qualification and testing. Additionally, the software required to control the camera and process the images should be developed, this includes the meteor detection algorithm. All in all, a working prototype of the complete instrument, including software should be developed in the scope of this thesis. Here, the main challenge is the development of the detection algorithm. A complete new concept is required, since ground based detection algorithms can not be applied due to the movement of the background. Additionally, the algorithm must cope with the limited resources onboard of the satellite while still allowing to process a sufficient amount of images.

1.5.1. Thesis structure

In order to consider all aspects outlined above, the thesis is divided into four main parts (see Figure 1.3):

The first part gives an introduction to meteor sciences and provides an overview of current ground based meteor observations and

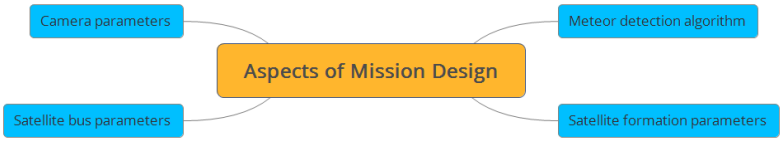


Figure 1.3.: Aspects of Mission Design

their methods. Also, an existing database containing the data of stereoscopically observed meteors is analysed. This analysis is used to define the scientific mission goals of *FACIS* and *SOURCE*. Furthermore, the mean meteor properties from this analysis are later used to design the instrument.

Second, parameters for the instrument, the satellite bus and formation are derived by using a set of simulations. The parameters are specified, in order to maximize the scientific output of the mission. Instrument parameters are for example the field of view of the lens. Satellite bus parameters include the attitude knowledge while the formation parameters take into account the orbit altitude and satellite distance.

Third, the meteor detection algorithm is described. As mentioned, the algorithm is a critical part of the instrument, since the downlink capacity of the satellites is limited. This chapter describes the working principle, implementation and test of the algorithm.

The last main part describes the development and final design of an instrument for meteor detection for the *SOURCE* and *FACIS* mission. This includes choosing a suitable camera and processing unit as well as testing and qualifying the chosen parts. The trade-off takes into account the conclusion of the requirements and parameters derivation as well as the algorithm development to optimize the scientific output, while taking into account the limitations of the according satellite bus. Finally, an estimation of the expected scientific output and performance is given.

1.5.2. Standing on the shoulder of giants

As outlined above, this works aims to analysis the effects of instrument, satellite bus and formation parameters in the scientific output in order to design two satellite missions. Furthermore, a meteor detection algorithm should be developed.

As usual in sciences, this work relies also on ideas and work done by others. In the according chapters detailed references are given. Here, only the most important references are given credit.

Since the scientific output is the main aspect when evaluating the effect of different parameters, a tool to calculated the scientific output is required. Therefore, the Simulator for Wide Area Recording of Meteors from Space (SWARMS) tool is adapted and used, which was kindly made available by the authors of the Python tool. This tool calculates the number of observable meteors for a given instrument and orbit parameters.

Furthermore, the scientific output is also evaluated in terms of trajectory accuracy. Here, the Meteor Orbit and Trajectory Software (MOTS) algorithm is used, which is implemented and adapted for a observation from satellites instead for ground based observations. The working principle of the tool was taken from the published paper.

Finally, the data base from Canary Island Long-Baseline Observatory (CILBO) was used in this thesis, which was made available by one of the team members. In this thesis, the data base is used to derive meteor properties, which are used in the SWARMS simulation.

2. Overview of meteor science and scientific mission goals

Meteors have been observed for a long time by different cultures, one of the earliest records available today origin from 687 BC during the Chou dynasty period (this as well as the following paragraphs are taken from [4]). A lot of myths are connected to meteors, from interpretation of a soul going to the hereinafter to a bad omen for upcoming times. For a long time, the nature of meteors was unknown and it was believed meteors are as sporadic as the weather and not predictable.

Observations of meteors from two different sites, conducted in Göttingen in 1798 (see [4, p. 39]), resulted in the discovery, that meteors occur high in the atmosphere and origin from outside the Earth. With more and more meteor observations and Newton's theory about gravity and the movements of objects in orbits, it became clear that meteor showers could be related to a parent comet. Comets are remains of the earliest building blocks of our solar system. By studying them, scientist are able to draw conclusions about the origin and formation of our solar system. Observing meteors, which are in fact formed by small parts of comets, can therefore help to understand the properties and formation of our solar system.

Observing meteors is not only possible on Earth, they occur on every planet with an atmosphere. Therefore, an observation of meteors on different planets (e.g. Mars) can help the scientist to not only study the meteor properties but investigate the planet's atmosphere as well. This will also help to identify other dense meteoroid streams [13]. All in all, an observation of meteors is an important part to investigate the formation and origin of our solar system.

Furthermore, detailed models of meteor and prediction of meteor showers are needed, to asses the danger for satellites. Our society depends more and more on technology, with satellites being an important part of the infrastructure. This includes for example, navigation systems (like Galileo) or communication systems. The observation of meteors is important to refine the meteor flux models and thus help to mitigate the risk of a satellite being damaged.

2.1. Meteor phenomena

In this section different terms in the context of meteors, as visualized in Figure 2.1, should be explained. Those terms are important to understand the simulations and calculations in later chapters.

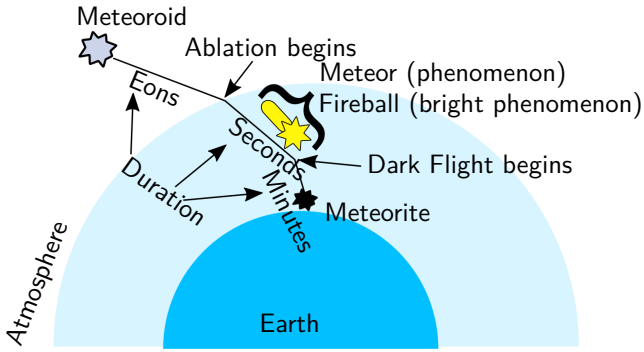


Figure 2.1.: Meteor Terminology. Illustration according to [14, Figure 2, p.330].

To begin with, the term *meteor* describes the visual phenomenon when particles enter the Earth atmosphere and melt due to their high velocity. These dust particles, before entering the atmosphere, are called *meteoroids*. Meteoroids vary in size and weight, from a few millimetres up to 100 m in size and from a few gram up to several tons. They originate from bodies like asteroids or comets (see [4]). Some are also debris from collision impacts.

If parts of a meteoroid survive the entry in Earth atmosphere, the particles reaching the ground are called *meteorites*. These remnants are collected and analysed, which give the chemical composition.

Important parameters describing the meteor phenomenon include the *mass*, the *size* and the *velocity* of the meteor. Most meteoroid have a mass between 1×10^{-3} g to 1×10^{-7} g before entering the atmosphere (see [15]), with larger particles resulting in brighter meteors. However, very bright meteors (called fireballs or bolides) can reach higher masses. Regarding the size of a typical meteoroid, the range is usually between 1×10^{-4} m to 1×10^{1} m (see [16]). The brightness depends also on the velocity (usually 15 km/s to 60 km/s, see [17]).

The *brightness or luminosity* of a meteor is another important property. Usually, it is assumed that the brightness is proportional to the kinetic energy. The following Equation 2.1, describing the relation between meteor magnitude, velocity, entry angle and mass, is taken from [4, p.46]:

$$\log_{10}(M) = 6.31 - 0.4 * m_{abs} - 3.92 * \log(V_\infty) - 0.41 * \log(\sin(\xi)) \quad \textbf{(2.1)}$$

This formula allows to calculate the meteoroid mass (M in g) from the brightness of the meteor (m_{abs} in the magnitude system), the entry velocity (V_∞ in km/s) and the entry angle (ξ). The *luminous efficiency* τ describes how much of the kinetic energy is transformed into light (usually about 0.1 % to 1 %, [4]).

The brightness of a meteor is given in magnitudes, which is widely used in astronomy to describe the brightness of objects (following section based on [18]). This scale was invented by the Greek astronomer Hipparchus, who used a scale of 1 to 6 to describe the brightness of stars visible to the naked eye. During the advancement of Astronomy, the scale has been adapted and different systems are nowadays used. It is important to notice, that this scale is reverse (lower numbers describe a higher brightness) and logarithmic. The

brightness of an object is described by comparing it with the brightness of a defined zero point of the scale as shown in Equation 2.2 (adapted [19, p. 25, Eq.2-1]).

$$m_1 - m_2 = 2.5 * \log_{10}\left(\frac{L_2}{L_1}\right) \quad \textbf{(2.2)}$$

A reference luminosity $L_1 = L_{reference}$ is defined to be magnitude m_1 zero, which results in $m_2 = -2.5 * \log_{10}(\frac{L_2}{L_{reference}})$. Due to historical reasons, the star Vega is used as a reference point. It has a magnitude of (close to) zero in most magnitude scales. Furthermore, the factor between a difference of 1 *magnitude* is $\sqrt[5]{100} \approx 2.512$.

Additionally, the brightness described by the scale has to be clearly defined in radiometric units. Since the brightness of an object varies depending on the wavelength, a photometric system is needed. A photometric system describes different filter bands, in order to classify objects by their brightness in different bands. The photometric system used in this thesis is the Johnson-Cousins UBVRI System (see [20]), which is widely used in astronomy. In this system the spectral irradiance for magnitude 0 is defined in different filter bands (U,B,V,R,I). The important filter bands for this thesis are the B,V and R bands, since those are in the visual spectrum detectable with a visual camera. The data of each filter is summarized in Table 2.1.

Table 2.1.: Johnson filter system (from [4])

Johnson Filter	Central wavelength (nm)	FWHM (nm)
B	444.3	83.1
V	548.3	82.7
R	685.5	174.2

In the context of the filter data, Full Width Half Maximum (FWHM) is the width of the transmission peak, at the half of the maximum

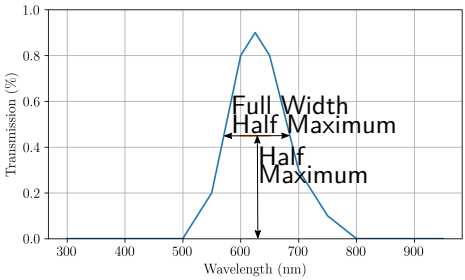


Figure 2.2.: Illustration of the FWHM for a filter

transmission in %. This is visualized in Figure 2.2. Furthermore, the central wavelength describes the wavelength at which the filter curve has the maximal transmission.

When giving the magnitude of a meteor or other object, it is always important to distinguish between the *apparent* and *absolute* magnitude. This is due to the fact, that the perceived brightness depends on the distance between observer and object. A higher distance results in a fainter perception of the object. Therefore, the apparent magnitude describes the brightness as seen from the observer, while the absolute magnitude gives the brightness of the object at a defined distance. Since most meteors occur at an altitude of 100 km above the Earth, the absolute magnitude for meteors describes the brightness of the meteor as seen from this distance. Another important magnitude is the *limiting magnitude*, which is used to describe the faintest stars or meteors which can be observed by an instrument. This quantity depends on the instrument parameters as well as on the observation conditions (e. g. cloud coverage, background noise and distance to the meteor).

This leads to an another important characteristic: The *distance* to the meteor. As mentioned, the light emitting process starts usually at around 100 km, depending on the speed and entry angle of the meteoroid. Usually an observer or instrument detects the meteor

at higher distances, since most meteors do not appear at the local Zenith. This is visualized in Figure 2.3.

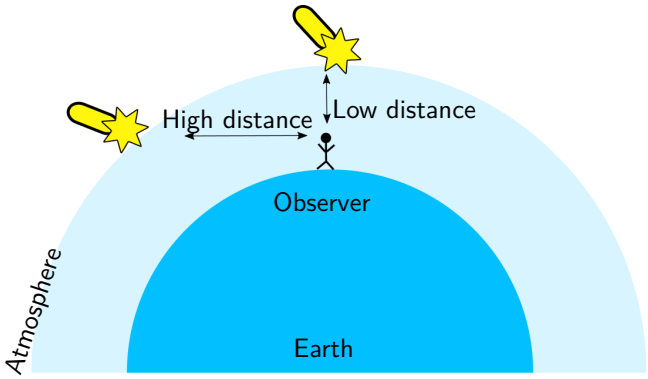


Figure 2.3.: Visualisation of meteor brightness distance dependency: The apparent brightness of the meteor at higher distance (left) is much lower than the brightness of the meteor appearing the local Zenith.

The brightness is also depending on the *entry angle*, which describes the angle between the meteor trajectory and the local horizon. A steeper angle results in brighter meteors.

Summing up, the *brightness* depends mostly on the velocity and the mass of the particle as well as the entry angle and the distance to the observer.

Finally, meteors can be divided into *shower* and *sporadic* meteors. Sporadic meteors occur all over the sky during night and day, moving in all directions. In contrast, shower meteors seem to appear from one direction in the sky, the so called *radiant*.

Shower meteors occur when the Earth intersects the orbit of dust particles, originating from a specific parent body (usually a comet, see [4] for more details). Thus, the meteors appear to originate from one direction in the sky. Usually the shower is named after the stellar constellation from which the meteors appear. For a ground based observation of a meteor shower, it is useful to observe close

to the direction of the radiant[1] to maximize the number of observed meteors.

While sporadic meteors occur from all directions, there are also directions in which more meteors can be observed. This is due to the Earth rotation and position of sources of disturbances (e. g. the moon). For example, more meteors can be observed in the Apex direction (direction in which the Earth moves along its orbit). This is due to the Earth movement (see [21]). Another source for sporadic meteors are the helion and anti-helion direction (see [22]), which are the directions towards and away from the sun.

2.2. Meteor observations methods and goals

Meteors are observed with different methods, each of them with the goal to study a specific property of a meteor or meteor shower. The instrument design depends on what aspect of meteors should be studied. Meteor observation methods are naked eye, photographic, radar observations and visual observations with digital cameras. Since digital cameras became affordable and more sensitive, they are widely used for automatic meteor observations. In the scope of this thesis, meteors should be observed with a visual camera, to aid observations made from the ground with visual video camera systems. Usually these camera systems consist of one or more cameras pointing to different parts of the sky. If the camera systems are distributed over a wider area and observe overlapping parts of the sky, it is possible to determine the trajectory of the meteor.

Examples are the networks called *Fireball Recovery and Inter-Planetary Observation Network (FRIPON)* (see [23] and [24]) and *Cameras for Allsky Meteor Surveillance (CAMS)* (see [25]) as well as the *Canary Island Long-Baseline Observatory (CILBO)* (see [26]). Depending on the scientific observation goal, the instrument design

[1] See `https://www.imo.net/observations/methods/visual-observation/major/`

varies. Three possible observation goals are explained in the following sections using the *FRIPON* and *CILBO* camera networks as an example.

2.2.1. Trajectory determination

If a meteor is simultaneously observed from two different points in space, the according trajectory can be calculated. The back propagation of the trajectory leads to an orbit of the meteor, which then allows to draw conclusions to the parent body of the meteor. *CILBO* and *FRIPON* are both used to calculate trajectories, but since the scientific goal differs, the instrument design differs as well.

In short, *FRIPON* is dedicated to observe bright meteors (fireballs), calculate the corresponding meteoroid orbit and recover the meteorite for laboratory analysis. This allows to combine information about the meteorite origin with its physical and chemical properties to study the formation and evolution of our solar system. To achieve this goal, multiple cameras are deployed, mainly in France and in other European countries, but also world wide. Since fireballs a rare events, a large area must be covered. This is achieved by using multiple cameras with a large FOV. Furthermore, for orbit determination, overlapping of the cameras FOVs are required.

The *CILBO* observatory is a ground based meteor observatory instrument stationed in Tenerife (see [27]). It is dedicated to the stereoscopic observation of meteors using intensified CCD cameras and focuses on the precise determination of trajectories for fainter but more frequent meteors. The two stations are located about 100 km apart, to allow for a trajectory determination. *CILBO* has a relativity small field of FOV, which increases the accuracy of the meteor position determination.

Both examples illustrate already, how the scientific goals influences the instrument design.

The accuracy of the trajectory determination depends on multiple factors, a bias analyses has been made for the *CILBO* system (see [28]). While the accuracy of a system is hard to simulate, some

factors have a great influence: A larger FOV results in a large FOV per pixel (Instantaneous Field of View (IFOV)), which reduces the accuracy of the meteor position determination. In contrast, a smaller FOV reduces the total amount of meteors observed. Also the speed measurement of the meteor depends on the frame rate, with a lower frame rate the speed determination becomes less accurate.

2.2.2. Flux measurement

The goal of flux measurements is the determination of the number of meteors hitting the Earth sporadically and during a meteor shower. This is useful to determine annual variation of meteor showers, drawing conclusion to the properties of the comet orbit and its dust trail. Furthermore, new meteor showers can be identified, by observing sporadic meteors. These kinds of measurements are made by ground based video systems, by deploying multiple cameras with a large FOV.

For flux measurements, a large FOV and high sensitivity of the camera are important design drivers to detect as much meteors as possible. This is the case for the *CAMS* system, which deploys several cameras in California. It is designed to confirm meteor showers, which requires sensitive cameras to observe faint meteors. Furthermore, a large FOV is needed, since unconfirmed showers usually have a low meteor flux. While *CILBO* and *FRIPON* can also determine the flux, they are not designed for this type of measurements as outlined above.

2.2.3. Spectrum measurement

When a meteoroid hits the Earth atmosphere, several mechanisms contribute to the emission of light which is observed. One of those mechanisms is the excitement of atoms from the meteoroid, e.g. Iron. Depending on the composition of the meteoroid, different atoms are excited and emit light of a specific wavelength. If those

emissions lines are measured, conclusions to the meteoroid composition can be drawn. This spectrum measurement can be made without additional measurements or aid the flux and trajectory measurement by providing additional data. For example, an additional *CILBO* camera is equipped with a grating to measure the spectrum of meteors.

2.3. Meteoroid mass models

Meteoroid mass models are used to describe the number of meteoroids of a specific mass. Since the brightness of the meteor depends on the kinetic energy and thus the mass of the meteoroid, those models can be used to estimate the number of meteors of a specific brightness. Together with the properties of a meteor observation instrument (coverage, sensitivity), these models are important to evaluate the scientific performance of the instrument.

The mass models are given as Cumulative Distribution Functions (CDFs), which describe the number of meteoroids above a certain mass *m* per unit area and unit time (meteoroid flux). The two models used in this thesis are the models from *Halliday* and *Grün* (see [29] and [30] respectively). Both models give the expected meteoroid flux, but have been developed using different methods and therefore differ quite widely.

The Halliday model was developed from data of the Canadian camera Network. 259 fireballs (very bright meteors) have been observed and their velocities, height, orbits, brightness and masses were calculated.

$$\log_{10}(N_{\text{Halliday}}) = \begin{cases} -0.48 \cdot \log_{10}(m) + 3.3 & \text{for } m < 2.4\,\text{kg} \\ -1.06 \cdot \log_{10}(m) + 5.26 & \text{for } m > 2.4\,\text{kg} \end{cases} \tag{2.3}$$

Depending on the meteoroid mass, two different functions are used to describe the CDF (see Equation 2.3). In this equation N_{Halliday} is given in $\textit{Events}/(\textit{year} \cdot \text{km}^2 \cdot 10^6)$, the meteoroid mass m is given in g.

The Grün model is derived from in-situ dust measurements as well as from studying lunar micro craters. This model describes the meteoroid flux depending on the distance from the Sun. At 1 *AU* the flux is described by Equation 2.4

$$N_{Gruen}(m) = (2.2 \cdot 10^3 \cdot m^{0.306} + 15)^{-4.38} \quad \textbf{(2.4)}$$

Here, the flux is given in $\mathit{Events}/(\mathrm{s} \cdot \mathrm{m}^2)$, m is given in g.

The meteoroid flux from both models is plotted in Figure 2.4 for initial masses 0.01 mg to 10 000 kg. As can be seen, the Grün model predicts an order of several magnitudes higher flux at low masses. For heavy, but rare meteoroids, the Halliday model gives a larger number of meteoroids, although the difference is smaller than for lightweight meteoroids. The Halliday model is biased for heavier meteoroids and not accurate for lightweight meteoroids, since the model is derived from bright and thus heavy fireballs.

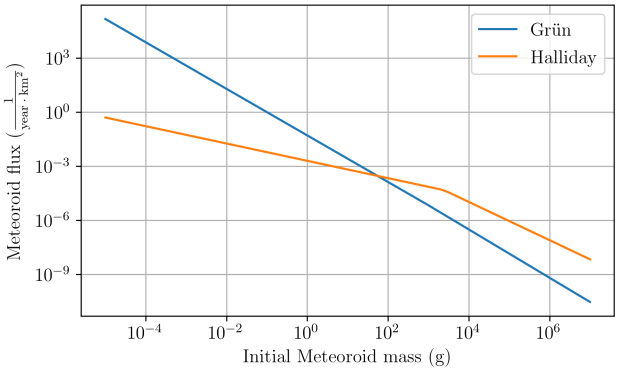


Figure 2.4.: Comparison of Meteoroid flux from Haliday and Grün model

While both models are derived from observations, they are still only modelling an average expected meteoroid flux. Furthermore, the meteoroid flux during a meteor shower can be significantly higher. Thus, any prediction made with those models must be used

with caution and only give an estimation of the expected meteoroid flux. The meteoroid flux calculated from observations made with the *CILBO* observatory fit the Grün model well (see [31] and [32]). After de-biasing the observed data, the slope of the meteoroid flux depending on the meteoroid mass fits the Grün model, although the measured flux is slightly higher. De-biasing the raw observational data is needed, because fast and large meteoroids result in bright meteors which are more likely to be detected.

All in all, the Grün model seems to be the better choice for predicting the number of meteors which could be observed with a spaceborne instrument:

First, the Halliday model is based on Fireball observations. The instrument designed in this thesis is a visual camera dedicated to the observation of sporadic and shower meteors and not only fireballs.

Furthermore, the Grün model fits observations made with the visual cameras from the *CILBO* observatory. Nevertheless, the Halliday model can still be used in parallel with the Grün model in order to give a range of the possible number of meteor observations.

2.4. Derivation of meteor properties from the CILBO database

Important meteor properties in the context of visual meteor observation are meteor velocity, brightness and direction of movement (includes re-entry angle). These properties are needed to determine whether a meteor can be observed with a given camera system. Furthermore, the distribution of these properties is needed to determine the number of meteors observable with a given camera system.

The data from the *CILBO* observatory has been analysed, in order to get realistic values for the distribution of meteor properties. Thanks to T. Albin, the database of cleaned meteor observations can be used to determine meteor properties and their distribution.

The database contains 12 045 meteors, simultaneously observed between January 2013 and August 2015 from both stations. The meteors are observed within at least four frames and the altitude each meteor is at least 80 km. More details about *CILBO* and the database can be found in [33] and [34] as well as the dissertation of T. Albin (see [35]).

2.4.1. Approach

The cleaned *CILBO* database contains the data of each meteor observed from both *CILBO* cameras (ICC7/cam1 and ICC9/cam2). It only contains valid meteors from which the trajectory could be calculated as well as different parameters. The ones used in this analysis are velocity, distance, apparent and absolute magnitude. Also, the angle between camera boresight respectively zenith and meteor radiant are analysed. Since each property is derived from several images and two cameras, four values actually exist in the database for each meteor property: The mean value, the standard deviation, the median value and the absolute deviation of the median value. For this analysis only the mean value of each meteor property is used.

The analysis is done by creating histograms for each property and calculating the mean value, standard deviation as well as minimum and maximum value for each distribution.

The data base is further filtered, to only include meteors with an velocity from 10 km/s to 100 km/s, an altitude below 130 km, an absolute magnitude > -3, a maximal distance to the meteor at maximum brightness below 200 km and an angular distance between radiant and zenith smaller than 90°. This is done to exclude extreme events such as fireballs from the analysis. Furthermore, excluding meteors far away and with a high angular distances increases the accuracy since such meteors are measured less accurate.

2.4.2. Results

In the following, the histograms showing the distribution of different meteor properties are analysed. The histograms are generated by sorting each meteor into a bin according to its value. The histograms are shown in the Appendix Section A.1. As mentioned, the results of this evaluation are used later in this thesis to analysis the observability of a typical meteor by a given instrument.

Angles Two different angles are measured by *CILBO*: The zenith angle describes the angle between the radiant and the zenith direction (see Figure 2.5). The boresight angle is the angle between the camera boresight and the radiant of the meteor. For further calculations the boresight angle is relevant, since this angle influences the perceived brightness of the meteor in the instrument. The distribution is Gaussian like and has a mean value of 62.55°.

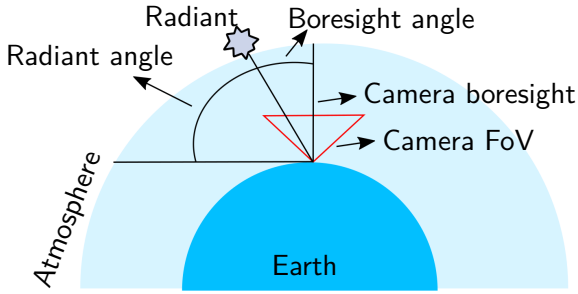


Figure 2.5.: Meteor angle terminology of boresight and zenith angle.

Velocity The distribution has two peaks at 29 km/s and at 58 km/s. The peaks are explained in [36] with the average velocity of two meteor showers. For one meteor shower the velocity distribution is Gaussian like, but not the distribution of sporadic meteors or multiple meteor showers. The velocity distribution of sporadic meteors is according to the ECSS standard logarithmic (see [37]). In case

the velocity is needed for further simulations, either the logarithmic distribution from the ECSS standard, the overall mean velocity of the *CILBO* data base or the mean velocity of one shower can be used. This depends on the purpose and objective of the simulation.

Distance Two different meteor distances are available in the *CILBO* data base. The first one is the mean distance to the meteor during the time the meteor was observed. This distribution is not Gaussian, but has a mean value of about 101 km as expected. The other distribution shows the distance to the meteors at their maximum magnitude. In this case, the mean value is shifted to higher values (119 km) than the expected 100 km, but the distance is Gaussian distributed. All in all, it is valid to assume a Gaussian distribution with a mean value of 101 km, even if the mean distance deviates from an ideal Gaussian distribution.

Table 2.2.: CILBO trajectory accuracy

	Min. trajectory error (m)	Max. trajectory error (m)	Mean trajectory error (m)	Median trajectory error (m)
All	0.08	1000.00	73.17	46.00
Filtered	11.90	260.00	66.20	56.91

Trajectory error The mean, the median, the minimal and the maximum trajectory errors of the *CILBO* data base were calculated as shown in Table 2.2. The trajectory error is defined as the difference between the trajectory and the meteor coordinates used to calculate the trajectory. A large difference means the trajectory could not be well fitted due to large errors in the meteor coordinate measurements. These values are needed as a comparison to evaluate the performance of a satellite based trajectory determination.

Two different approaches are used to calculate the values in Table 2.2. First, the values are simply determined by using all available measurements in the cleaned data base. However, this also includes measurements with a high error.

Thus, in a second approach, the data base is filtered and only meteor observations with a certain meteor velocity and measurement accuracy of the meteor position are taken into account. The meteors included have a velocity from 10 km/s to 45 km/s, a magnitude between -1 to 3 and a measure accuracy of meteor position between $0.01'$ to $0.06'$. As shown in Table 2.2, this only effects the minimal and maximal trajectory error. The mean and median trajectory are both in the same order of magnitude.

Meteor duration For later calculations, the average meteor duration is also needed and derived from *CILBO* data.

Therefore, the number of frames for each observation are taken from the data base and multiplied with the frame rate of the *CILBO* cameras (25 *fps*). The number of frames for a meteor can be determined for each camera by two methods:

First, the average number of frames can be used.

Second, since the distribution is not Gaussian like, the maximum of the histogram (see Figure 2.6) can be used. The results for both methods are shown in Table 2.3.

Table 2.3.: Meteor duration calculated from CILBO data. For each camera, the maximum of the histogram as well as the average number of frames are used to calculate the meteor duration.

	Histogram camera 1	Histogram camera 2	Average camera 1	Average camera 2
Number of frames	4.75	5.77	9.18	9.85
Meteor Duration (s)	0.19	0.23	0.37	0.39

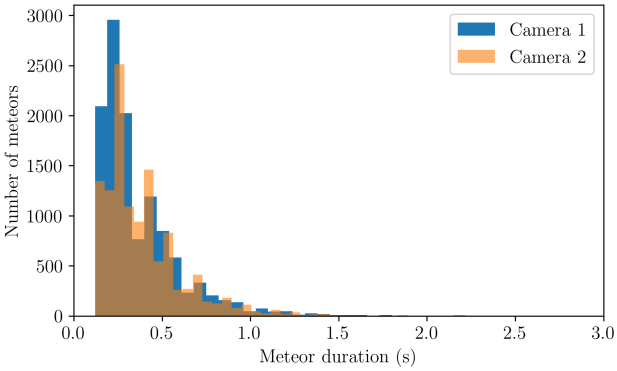


Figure 2.6.: Histogram of number of frames per meteor from CILBO data

Summary

In Table 2.4 mean, median as well as maximum and minimum values for each distribution analysed previously are given. Those values are later used for further simulations and calculations, e. g. to determine the limiting magnitude of a given camera system (see Section 3.3) or the expected number of observable meteors (see Section 3.4.1). Furthermore, the data can be used to evaluate the performance of a satellite based instrument, by calculating which percentage of meteors observed by *CILBO* could be observed with a space borne instrument (see Section 5.1.2). Therefore, the *CILBO* data base provides important data for the design of a meteor observation instrument on a satellite. However, even if the data base is already cleaned (removing unrealistic measurements and de-biasing), some values are not realistic. This concerns the maximum velocity of 90 km/s, since only meteors from outside the solar system exceed the velocity of 72 km/s. This is the sum of the escape velocity of the Sun and the orbital velocity of Earth. However, the data can still be used, since the extreme values are not used.

Table 2.4.: Mean meteor properties according to *CILBO* data base

Property	Mean	Median	Standard-deviation	Max	Min
App Mag CILBO	3.17	3.28	1.03	5.7	-1.58
Abs Mag	2.1	2.26	1.27	5.29	-2.79
Angle boresight (°)	62.55	62.01	22.26	123	3
Angle zenith (°)	43.55	43.84	17.62	1.63	88.94
Velocity (km/s)	40.3	37.6	16.7	90	10.2
Distance max mag (km)	120	119	11.0	167	90
Distance altitude (km)	101	100	8.5	130	80
Duration camera 1 (s)	0.37	0.28	0.25	4.32	0.12
Duration camera 2(s)	0.39	0.32	0.24	3.44	0.12

2.5. Scientific mission goals and constraints

For the two missions *FACIS* and *SOURCE* the scientific objectives are different.

To begin with, the scientific goal of the *FACIS* mission is the determination of meteor trajectories. Therefore, two satellites flying in formation with overlapping field of views are used. Similar to *CILBO*, the instrument should be able to observe faint meteors. All major meteor showers should be observed, in order to better characterize the showers and get a solid statistic on the meteor flux.

This includes confirming the parent body of a shower by trajectory determination as well as measuring the meteor flux during the shower in order to verify prediction models. Especially showers not well observable with existing ground based observatories should be characterized. Furthermore, sporadic meteors should be observed, in order to improve existing meteor mass models and identify respectively confirm new meteor showers. Please refer to Chapter 1 for more details on meteor science.

SOURCE is mainly a technological demonstration, to qualify the instrument for use in space and test the meteor detection algorithm with actual images from meteors as seen from the satellite. Nevertheless, *SOURCE* is also capable of carrying out scientific observations. The main scientific goal is to observe a selection of important meteor showers, as stated in the observation concept. Furthermore, as much sporadic meteors as possible should be observed, taking into account the limitations of the satellite bus.

For both missions, a camera operating in the visual spectral range is suitable to fulfil the scientific goal. Additionally, to demonstrate the use of COTS components and reduce costs, a industrial commercial camera shall be used.

2.6. Scientific instrument requirements

Deriving requirements for the instrument and evaluating the effects of camera parameters on the scientific performance is an essential part of this thesis. Nevertheless, some high level requirements can be directly derived from the nature of the planned objective. The following requirements represent high level requirements, which do not state numbers. They should be seen as preliminary thoughts on important parameters.

For both missions, *FACIS* and *SOURCE*, the observation of meteors during eclipse requires a camera capable of observing faint objects. This translates to a sensor with high quantum efficiency and

low noise. Furthermore, the aperture of the lens should be large, to capture as much light as possible.

Other requirements are more mission specific: The *FACIS* mission should determine the trajectory of meteors, which requires the observation of meteors from two satellites. This implies the number of meteors observed from two satellites must be maximized. Furthermore, the velocity of meteors must be determined in order to calculate the trajectory. Since the velocity is derived from the position of a meteor in different frames, the accuracy of velocity determination increases with the number of frames with a meteor. Thus, a high frame rate is needed. The performance of *FACIS* should be similar to *CILBO*: *FACIS* should observe at least 75 % of the meteors from the *CILBO* data base.This should assure, that enough meteors are observed to get a good statistical database, while at the same time taking into account the reduction of stereoscopic observable meteors from a satellite. Furthermore, the trajectory determination accuracy should be in the same order of magnitude. However, this might not be possible, since the trajectory is determined from moving satellites, which involves more measurements contributing to the error. Thus, the trajectory error should be in the order of about 450 m. It is estimated, that this is a realistic error to be achieved by a satellite system, while also being close to the performance of ground based systems.

The *SOURCE* mission focus on the determination of meteor flux. Therefore, a high number of observed meteors is required.

All these requirements will be refined in the following chapters, furthermore the requirements will be verified using different simulations.

3. Instrument, satellite bus and formation parameters

In the previous Chapter 2, the scientific mission goals of *FACIS* as well as *SOURCE* have been defined: The *FACIS* instruments should be able to determine a meteor trajectory of meteors and characterize meteor showers, while the *SOURCE* instrument is mainly a technology demonstration and a testbed for the detection algorithm. More detailed requirements were derived from the *CILBO* data base, e. g. the properties of a typical meteor which the instrument should be able to observe.

In this chapter, the influence of certain parameters (instrument, satellite bus and formation) on the scientific output of the mission is analysed. These parameters have a high influence on the scientific output, because they influence the number of observed meteors and the trajectory accuracy. After the analysis, the high level requirements are broken down to instrument, satellite bus and formation requirements.

The instrument requirements describe the parameters of the instrument, e. g. camera noise or lens field of view, which are needed to fulfil the scientific goals. A simulation is set up to analyse the effect of different instrument parameters on the limiting magnitude, taking into account the meteor properties analysed in Section 2.4. This simulation gives the limiting magnitude of a given instrument and is described in Section 3.3.

The formation parameters (Section 3.4) also influence the scientific output. The influences can be divided into the following two aspects: First, the number of observable meteors depends not only on the limiting magnitude of the instrument, but also on the satellite orbit (*SOURCE* and *FACIS*) and on the distance between

the satellites (only *FACIS*). The *SWARMS Version 2 (SWARMSv2)* simulation is used to evaluate different formation and orbit parameters on the number of observable meteors (Section 3.4.1). The simulation is based on *Simulator for Wide Area Recording of Meteors from Space (SWARMS)*, see [2]. Second, the accuracy of the trajectory is also influenced by formation parameters (e. g. satellite distance). The effect of those parameters is determined by setting up a simulation (Section 3.4.2) based on *Meteor Orbit and Trajectory Software (MOTS)*, see [38]).

Finally, the scientific output also depends on the achieved trajectory accuracy. The highest achievable accuracy is defined through the satellite bus parameters (e. g. attitude knowledge). Therefore, the trajectory simulation is also used to determine requirements for the satellite bus (Section 3.5). Furthermore, the *CILBO* data is used, to validate the satellite bus requirements (Section 3.5.1).

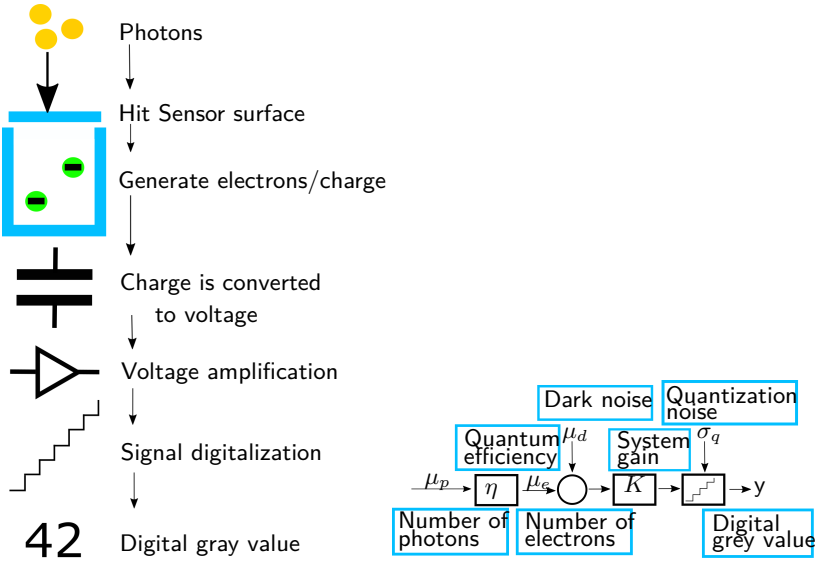
Finally, a camera parameter trade-off for *FACIS* and *SOURCE* is carried out based on the simulations and on the analysis (Section 3.6).

Before analysing instrument, satellite bus and formation parameters, a brief description of camera properties and working principles are given in the first Section 3.1.

3.1. Basic principles of cameras and optics

In order to understand the simulation of camera sensitivity, it is necessary to define and explain important terms used for cameras and in optics.

First, the camera model based on the *EMVA 1288* standard (see [39]) is explained. The *EMVA 1288* standard is an industrial standard defining how camera parameters should be measured based on the photon transfer method. It was developed by the European Machine Vision Association (EMVA). Some adaptions to the *EMVA 1288* formulas and notation have been made in this thesis. Furthermore, some parts of the standard are left out, which are not



(a) Physical camera model **(b)** Mathematical pixel model

Figure 3.1.: The *EMVA 1288* camera model, drawn according to [39, p. 5, Figure 1]

relevant in the context of this thesis. This allows for a consistent notations, however it does not change the model used in the standard. Second, important lens parameters are defined.

3.1.1. Camera model and parameters

The camera model used in the *EMVA 1288* standard is used in this thesis as a basis for the simulation of camera sensitivity. The following section is based on the standard specification document (see [39]) as well as on an article about camera noise from the company Thorlabs (see [40]).

In a camera sensor, photons are converted into an electrical signal. During the exposure time, *photons* hit the *pixel area* and generate

a number of *electrons* through the photoelectric effect. Those electrons represent a *charge* which is converted into a *voltage*. This voltage is *amplified* and *digitized* into a *value*. The value represents the *pixel value*. The more light hitting the sensor, the higher the pixel value. The process is visualized in Figure 3.1.

Using this physical model, some camera parameters can be introduced: First, the pixel area A_{px} is the area of one pixel. Due to electronic components on the pixels, not the whole area is sensitive to photons. The percentage of the pixel area actually converting photons is called the *fill factor*. Not all photons hitting the active pixel area generate a photon. The quotient between electrons generated μ_e and photons hitting the sensor μ_p is called the *quantum efficiency* η, see Equation 3.1.

$$\eta(\lambda) = \mu_e/\mu_p \tag{3.1}$$

As indicated in the formula, the quantum efficiency depends on the *wavelength* λ. In the following, a subscript *e* indicates a value regarding electrons, while *p* refers to photons.

The number of electrons generated μ_e represent the signal generated by a light source. This signal is important to later calculate the *signal to noise ratio*.

$$\mu_e = \frac{A_{PX} * E * t_{exp} * \eta * \lambda}{h * c} \tag{3.2}$$

Equation 3.2 shows the calculation of the signal, which depends on the quantum efficiency (η), the area of a pixel (A_{PX}), the exposure time (t_{exp}), the irradiance of the light source (E) and the wavelength of the signal (λ). Furthermore, h is the planck constant and c the speed of light.

This equation performs several steps at once: First, the irradiance given in W/m^2 is converted to power (Unit W) with the equation $P = E * A_{PX}$. Second, the resulting power can be converted into

a number of photons per second (N_p) by multiplying it with $\frac{\lambda}{h*c}$. This results in Equation 3.3.

$$N_p = \frac{P}{Energy_{Photon}} = \frac{P*\lambda}{h*c} = \frac{[\mathrm{W}]*[\mathrm{m}]}{[\mathrm{J}\cdot\mathrm{s}]*[\mathrm{m}/\sec]} = \frac{[\mathrm{J/s}]*[\mathrm{m}]}{[\mathrm{J}\cdot\mathrm{s}]*[\mathrm{m/s}]} = [1/\mathrm{s}] \tag{3.3}$$

Multiplying the number of photons with the quantum efficiency results in the number of electrons per second, which can be converted to a number of electrons by multiplying it with the exposure time. As a result, the number of electrons generated during exposure time per pixel is given, which represent the signal generated.

Noise sources

Unfortunately, different noise sources influence the signal. The first noise source is the *dark signal* (subscript *d*), which is the number of electrons present when no light is hitting the sensor (μ_d). The electrons μ_d are thermally induced electrons, the dark noise therefore depends on the temperature. Furthermore, it depends on the exposure time (t_{exp}). The calculation is shown in Equation 3.4.

$$\mu_d = \mu_{d.0} + \mu_I * t_{exp} \tag{3.4}$$

The dark signal depends linearly on the *dark current* μ_I, which in turn depends on the temperature. $\mu_{d.0}$ is the part of the dark signal generated at zero integration time, it can be seen as a *read out noise*.

To sum up, the *dark noise* σ_d is defined as:

$$\sigma_d^2 = \mu_{d.0}^2 + \mu_I * t_{exp} \tag{3.5}$$

The dark noise includes the dark signal as well as all noise sources from sensor read out and sensor electronics.

Due to the dark noise, the measured signal (the number of electrons) in the sensor is the sum of the noise μ_d and the signal μ_e. The *mean digital signal* μ_y depends on the sum of μ_d and μ_e. This

signal is generated by converting the electrons in a voltage, which is amplified and converted into a digital value (see right side of Figure 3.1). The relation between total number of electrons and the digital value is described by the *total system gain K*. The unit of *K* is *DN*/*e*-, where *DN* stands for digital number, used to describe the digital pixel value.

$$\mu_y = K * (\mu_e + \mu_d) = \mu_{y.dark} + K * \mu_e \tag{3.6}$$

The calculation of the digital signal is shown in Equation 3.6, $\mu_{y.dark}$ is the mean dark signal. The term $\mu_y - \mu_{y.dark}$ is also called the *mean photo-induced digital signal* and is calculated with $K * \mu_e$.

Combining Equation 3.2 and 3.6 results in Equation 3.7.

$$\mu_y = K*\mu_d + K*\eta*\mu_p = K*\mu_d + K*\eta*\frac{A_{PX} * E * t_{exp} * \lambda}{h * c} \tag{3.7}$$

Another noise source is the *shoot noise*, which describes the fluctuation of electrons due to the laws of quantum mechanics. The shoot noise σ_e depends on the number of electrons μ_e, more electrons result in a greater shoot noise.

$$\sigma_e^2 = \mu_e \tag{3.8}$$

The last noise source is the quantisation noise, which results from the quantisation of the signal into a digital value through an Analog-to-digital Converter (ADC). The quantisation noise is quite small (according to [39] the variance is about $\frac{1}{12}$ *DN*) and is therefore neglected in the following calculations.

Finally, all noise sources add up linear due to the linear camera model shown in Figure 3.1. This results in the following noise equation:

$$\sigma_y = \sqrt{\sigma_e^2 + \sigma_d^2} = \sqrt{\mu_e + \mu_{d.0}^2 + \mu_I * t_{exp}} \tag{3.9}$$

Again, $\mu_{d.0}$ is the read-out noise, $\mu_e = \eta * \mu_p$ the signal, μ_I the dark current and t_{exp} the integration time. In Equation 3.9 the square root of the sum of all *noise variances* σ^2 gives the total noise σ_y.

The word variance already indicates, that the noise is not constant. Due to quantum mechanics (shoot noise) and small changes in temperature, the noise is never constant, even at perfectly constant illumination. The total noise in Equation 3.9 is given in *e-/px*, but can also be calculated in *DN/px* by multiplying with the K.

Signal to noise ratio

The previous equations are used to calculate the Signal to Noise Ratio for a given camera.

$$SNR = \frac{\mu_e}{\sigma_y} = \frac{\eta * \frac{A_{PX} * E * t_{exp} * \lambda}{h * c}}{\sqrt{\mu_e + \mu_{d.0}^2 + \mu_I * t_{exp}}} \qquad \textbf{(3.10)}$$

As can be seen in Equation 3.10, the Signal to Noise Ratio (SNR) depends on several camera parameters (pixel size, quantum efficiency, dark current, read out noise) and settings (integration time), as well as on the light reaching the sensor (wavelength and irradiance). Contrary to the *EMVA 1288* standard, the SNR is calculated from signal and noise given in *electrons* instead of *DN*.

Additional noise sources can be taken into account as well, e. g. if another signal is present which is not of interest for the measurement (background noise). For a space based meteor observation, this could be light from cities. In this case, the background noise can be taken into account by adding the square of the noise to the other noise sources.

Sensor types

All equations described in this chapter are valid for digital cameras with a linear response, which means the digital signal depends linearly on the input signal. The equations are also independent from the sensor type. Two sensor types are used in current cameras, Charge-coupled Device (CCD) sensors as well as Complementary metal-oxide-semiconductor (CMOS) sensors. Both sensor types convert photons to electrons using the photoelectric effect (see [41]).

The difference between the sensor types is how the charge accumulated in each pixel is read out. In short, in a CMOS sensor each pixel has its own electronic and the charge is digitized in each pixel. Contrary, in a CCD sensor the charge is transported to the camera electronics outside the sensor and digitized there.

Another difference between both sensor types occurs when the *Binning* functionality is used. This function summarizes the value of pixels and thus increases the signal, reduces read out noise and decreases resolution. Depending on the type of sensor, this function has different effects (see [42] and [43]). In both cases, the signal is increased and the resolution is reduced. In the CCD, the read out noise is reduced, since the charge is only digitized once. For a CMOS, the signal increases with the same factor as in the CCD, but the read out noise is increased as well, because each pixel is read out independently.

3.1.2. Lens parameters

The previous section described the sensor part of a camera, but a useful image can only be produced if an optic is used. The field of optics in general and lenses in particular are a wide and complex field. This chapter therefore focuses only on the parameters of a lens relevant for the observation of meteors.

The most important parameter is the *focal length (f)* of the lens. This is the distance between the focal point of the camera (the sensor) and the centre of the lens. It indicates how large the observed area is, namely the *Field of View*. A smaller focal length results in a larger field of view and thus in a larger area observed. The field of view describes the opening angle of the lens. From a given focal length and physical dimensions of the sensor, the field of view can be calculated. Since sensors are usually not square, a horizontal and vertical field of view is calculated.

$$FOV = 2 * \arctan\left(\frac{l}{2 * f}\right) * \frac{180}{\pi} \qquad \textbf{(3.11)}$$

Equation 3.11 (see [44, p. 16]) gives the FOV in $\deg$ with l being the physical dimension of the sensor and f the focal length (both in m).

For point light sources, e. g. stars or meteors, the light collecting area is important. For a lens, the opening in the front, allowing light to enter is called the *aperture*. The relation between aperture and focal length is the so called *f-number (F#)*, which is defined as:

$$\mathsf{F\#} = \frac{f}{aperture} \tag{3.12}$$

Usually the f-number can be changed on the lens in a certain range, a lower f-number results in a larger aperture. A large aperture is desired for meteor observations, since the light collecting area increases. This area is calculated with:

$$Area = \pi * \left(\frac{aperture}{2}\right)^2 = \pi * \left(\frac{f}{2 * \mathsf{F\#}}\right)^2 \tag{3.13}$$

3.2. Simulation overview

After introducing important camera and lens parameters, the different simulations used are presented. An overview of the simulations and calculations is shown in Figure 3.2, these are used to derive instrument, satellite bus and formation parameters. Most tools are used for a single satellite meteor observation mission as well as a formation of two satellites. However, some calculations are only useful for a stereoscopic meteor observation mission.

In a first step, the *CILBO* data base is used to derive average meteor properties (see Section 2.4) as well as to analyse the influence of the attitude knowledge on the trajectory determination accuracy. The average meteor properties serve as an input for the *Sensitivity* simulation. This simulation is developed in the scope of this thesis and gives the limiting magnitude for a given set of instrument parameters. It is also used to determine the influence of different instrument and meteor parameters on the limiting magnitude in order to derive instrument requirements.

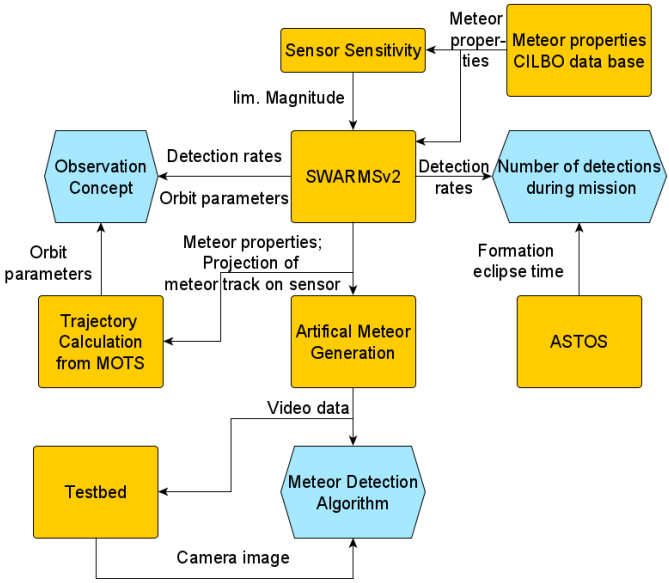


Figure 3.2.: Overview of simulations used for meteor observation mission design

The limiting magnitude is needed for the *SWARMSv2* simulation, in order to calculate the meteor detection rates depending on orbit and formation parameters. *SWARMSv2* is modified in this thesis based on *SWARMS* simulation (developed by J. Vaubaillon at al ([2])). Formation parameters include e. g. the satellite distance and the tilt angle. Their influence on the detection rates is evaluated, depending on different meteor mass models.

In combination with the orbit simulation tool Analysis, Simulation and Trajectory Optimization Software for Space Applications (ASTOS), the detection rates are used to calculate the number of meteor detections during the mission depending on different orbit parameters.

The detection rates are also used to derive an observation concept, which includes the satellite orientation and its altitude. This

concept takes into account the influence of those parameters on the detection rates as well as on the trajectory determination accuracy.

The effect of satellite bus and formation parameters on the trajectory determination accuracy is determined with a simulation based on the *MOTS* algorithm. The simulation developed in the scope of this thesis is used to analyse the ideal formation parameters for a low error on the trajectory determination as well as setting requirements for the satellite bus. As an input for this simulation, the properties of detected meteors as determined by the *SWARMSv2* simulation are used.

Those informations can also serve as an input for the generation of artificial meteor videos. The artificial meteor videos are necessary to test the meteor detection algorithm. The videos are either directly used as an input for the algorithm or displayed on the Testbed and imaged by the camera for an hardware in the loop test (see Chapter 4).

3.3. Simulation of camera sensitivity

In order the analyse the scientific output of a meteor observation mission, the sensitivity of the instrument must be determined. Thus, the *Sensitivity* simulation was developed. It is written in *Python*, with the objective of determining the limiting magnitude of a camera system with given parameters. It is used for two objectives:

First, the influence of each camera parameter on the limiting magnitude should be quantified. This is done to derive camera requirements for meteor observation.

Second, after selecting a camera for meteor observation, this simulation can be used to determine the limiting magnitude of this camera. The limiting magnitude is an important parameter for the scientific performance of a meteor observation instrument. As described in Chapter 2, the number of meteors grows exponentially with

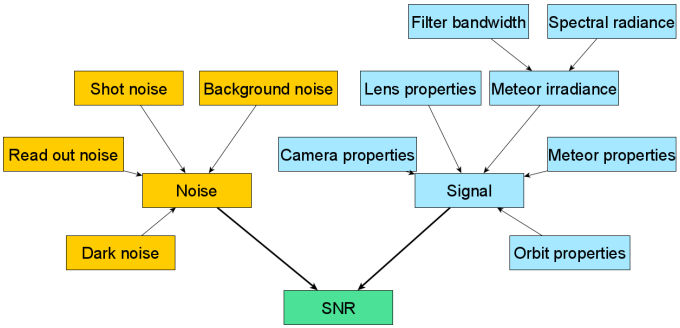


Figure 3.3.: Schematic of Sensitivity simulation program showing the SNR calculations. Furthermore, parameters taken into account for signal and noise calculations are shown.

decreasing mass and therefore brightness. A higher limiting magnitude enables the observation of fainter meteors and thus increasing the number of observations.

In this section, the working principle, the limitations and the verification of the simulation are presented. Furthermore, the significant camera parameters for meteor observation are given. Finally, the limiting magnitude of a camera is calculated, which serves as an example camera for further analysis.

3.3.1. Working principle

The *Sensitivity* simulation calculates the Signal to Noise Ratio (SNR) for a given camera system as well as meteor and orbit properties at different meteor magnitudes. Therefore, three steps are performed:

First, the radiance a meteor emits is derived from the magnitude. Here, different effects and meteor properties are taken into account, such as the distance to the meteor and the meteor speed.

Second, the noise is calculated. This includes the camera noise as described in Section 3.1.1 as well as the background noise from e. g. city lights.

Finally, the limiting magnitude can be calculated, which is defined as the minimal magnitude needed to reach a given SNR. The general principle is visualized in Figure 3.3, which shows the parameters needed for signal as well as noise calculations.

Signal calculations

In Section 3.1, equations to calculate the signal and noise of a camera were derived. Those are now used to calculate the signal to noise ratio for a camera system in orbit observing a point source. Equation 3.10 is the basis for this calculation, but additional factors have to be taken into account. First, the power (*radiant flux* Φ_s in W) or *Irradiance* (E in W/m^2) reaching the sensor from a target with given *radiance* (L_λ in W/(m^2 · µm · sr)) must be calculated. The subscript s indicates the value at the sensor. This is done using Equation 2.18 from [44, p. 33]:

$$E = \frac{\pi * T * \Delta\lambda * L_\lambda * \cos(\alpha)^4}{4 * f_\#^2} = \frac{\pi * T * \Delta\lambda * L_\lambda * FF * \cos(\alpha)^4 * D^2}{4 * f^2} \tag{3.14}$$

This equation takes into account the *transmittance of the lens* T, the aperture size (via the f-number) and the *angle* (α) between optical axis and the nadir direction (tilt angle). With these numbers, the target radiance L_λ is converted to the irradiance E reaching the sensor. Since the L_λ depends on the wavelength, the wavelength band $\Delta\lambda$ is needed to convert the radiance given in W/(m^2 · µm · sr) to W/(m^2 · sr). Furthermore, the fill factor FF is considered.

Now, Equation 3.14 (giving the Irradiance) can be combined with Equation 3.2 (giving the camera signal), resulting in Equation 3.15, which gives the resulting signal S_λ in *electrons/pixel*.

$$S_\lambda = \frac{K[-]*t_{exp}[\mathsf{s}]*\eta[\%]*\Delta\lambda[\mathsf{nm}]*A_{PX}[\mathsf{m}^2]*FF[\%]*L_\lambda[\mathsf{W}/(\mathsf{m}^2 \cdot \mathsf{sr} \cdot \mathsf{nm})]*\lambda[\mathsf{nm}]}{h[\mathsf{J} \cdot \mathsf{s}{=}\mathsf{W} \cdot \mathsf{s}^2]*c[\mathsf{m/s}]}$$
$$K = \frac{\pi*T*\cos(\alpha)^4}{4*\mathsf{F\#}^2} = \frac{\pi*T*\cos(\alpha)^4*D^2}{4*f^2} \qquad \textbf{(3.15)}$$

All values needed for the calculation are summarized in Table 3.1. Equation 3.15 is later used in the simulations, to calculate the signal (number of electrons in one pixel) generated from a meteor.

All parameters in this equation are natural constant or camera parameters, except the radiance L_λ. Unfortunately, the brightness of a meteor is usually given in magnitudes and not in radiometric units as e. g. the radiance. This means, the more challenging part of the signal calculation is the conversion of meteor magnitude to radiometric units. This is due to the fact, that different scales to measure the meteor brightness exists, additionally not a lot of data to convert the magnitude to radiometric units exists.

As described in Chapter 2, the Johnson magnitude scale defines the brightness of an object in different filter bands. Since the instrument should observe meteors in the visual spectrum, the filter bands *B (blue), V (visual)* and *R (red)* are used. For each filter band, the magnitude must be converted into a radiometric unit. Then, the radiance can be used in Equation 3.15, in order to calculate the number of electrons generated per pixel by the meteor in this filter band. The total number of electrons is calculated by adding the electrons generated in each filter band (see Equation 3.16).

$$S = \sum S_\lambda = \frac{Elek}{px}_{tot} = \sum_{x=,B,V,R} \frac{Elek}{px}_x \qquad \textbf{(3.16)}$$

In [4, p.592] the spectral radiance for a meteor of magnitude 0 is given for each filter band. The filters are used to measure only the radiance of a specific wavelength band. Ideally, the transmission of the filter would be 100 % inside the wavelength band and

Table 3.1.: Values needed for SNR calculation

Symbol	Unit	Description
t_{exp}	s	Exposure time
t_{dwell}	s	Dwell time of a meteor on one pixel
η	%	Quantum efficiency
A_{PX}	µm	Pixel size
r_X/r_Y	*px*	Sensor resolution X and Y direction
T	-	Transmission of lens
μ_I	*electrons*/(*pixel* · s)	Dark current
μ_e	*e-*/*px*	Sensor signal
E	W/m^2	Irradiance reaching the sensor
L_λ	W/(m^2 · µm · sr)	Radiance emitted from light source
λ	nm	Wavelength
$\Delta\lambda$	nm	Wavelength range
F#	-	f-number
f	m	Focal length
D	m	Aperture diameter
α	deg	Angle bet. camera optical axis and nadir
ξ	deg	Angle bet. meteor radiant and optical axis
V_m	km/s	Meteor speed
R	km	Range to meteor
FF	%	Pixel fill factor
c	m/s	Speed of light
h	J · s	Planck constant

Table 3.2.: Johnson filter system and Meteor (spectral) irradiance values (from [4, p.592])

Johnson Filter	Central wavelength (nm)	$FWHM$ (nm)	Spectral radiance $L_\lambda(\mathrm{W}/(\mathrm{m}^2 \cdot \mathrm{nm}))$	Radiance $L(\mathrm{W}/\mathrm{m}^2)$
B	444.3	83.1	6.40×10^{-11}	5.32×10^{-9}
V	548.3	82.7	3.67×10^{-11}	3.30×10^{-9}
R	685.5	174.2	1.92×10^{-11}	3.34×10^{-9}

0 % outside. Unfortunately, this is not the case. Therefore, two parameters are used to describe the filter transmission curve: First, the central wavelength is the wavelength at which the filter has the highest transmission. Furthermore, the filter bandwidth is defined as the *FWHM* in nm. This is the width of the wavelength band, at the half of the maximum filter transmission. The filter data as well as the spectral radiance taken from [4] is shown in Table 3.2. The radiance is calculated by multiplying the FWHM with the spectral radiance as shown in Equation 3.17.

$$L = L_\lambda * \Delta\lambda = L_\lambda * FWHM \tag{3.17}$$

This means, the filter FWHM is used as the filter band $\Delta\lambda$ in Equation 3.15.

The filter bands B,V and R are chosen, because the corresponding wavelengths of the filter lay all in a range were the camera is sensitive. The wavelengths range of all three filters are 403 nm to 947 nm, which can be calculated by adding, respectively subtracting, the FWHM of the filter. This represents the sensitivity range of a typical sensor which can be found in [45, p. 17]. The filter curves as well as the spectral response are shown in Figure 3.4 and 3.5.

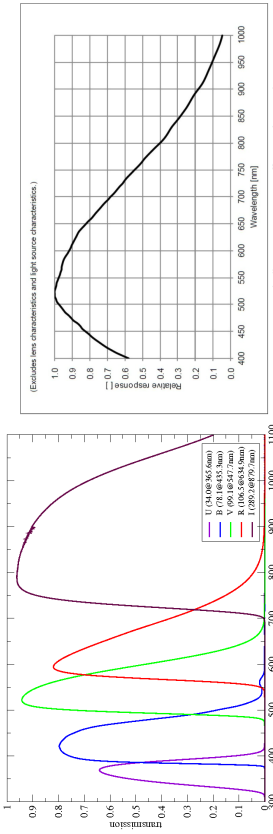


Figure 3.4.: Johnson filter band-width [46]

Figure 3.5.: Spectral response of *GenieNano M1920* camera [45]

All in all, it is now possible to calculate the number of electrons in a sensor generated by a meteor. This is done by converting the magnitude to radiance and using it in Equation 3.15. The result is the number of electrons in one pixel for the according filter band generated by a magnitude 0 meteor.

Before adding the electrons generated in each filter band, the meteor properties have to be taken into account: First, the spectral radiance values stated in Table 3.2 are valid for a meteor in a distance of 100 km. Consequently, the signal has to be adapted to the actual distance between camera and meteor, taking into account the orbit altitude. Second, the meteor moves during the observation and thus spreads the signal over several pixels. This means, the speed of the meteor has to be taken into account as well.

The *magnitude reduction due to distance* is calculated by using the inverse square law and applying it to Equation 2.2 (p. 18). The inverse square law states, that a point light source observed in twice the distance has only a quarter of the brightness. The brightness L is anti proportional to the square of the distance d (see Equation 3.18).

$$\frac{L_1}{L_2} = \frac{d_2^2}{d_1^2} \tag{3.18}$$

Applying this equation to Equation 2.2, gives the magnitude reduction Δm_{dis} for an meteor observed in the distance R instead of the lower reference distance of 100 km:

$$\Delta m_{dis} = \left\| 2.5 * \log_{10} \left(\frac{100\,\text{km}}{R} \right)^2 \right\| \tag{3.19}$$

The distance R in the simulation is the reference distance of 100 km minus the orbit altitude, which is the distance to the meteor from the satellite. This implies that the meteor is observed in the centre of the FOV and the camera is pointing towards nadir direction.

The *magnitude reduction due to meteor movement* results from the spreading of the photons emitted from the meteor over several pixels. This reduces the number of electrons generated per pixel

and thus the signal. Therefore, the magnitude reduction depends on the distance the meteor moves on the sensor. In [47, p. 69, Eq.3] a formula is presented to calculate the magnitude difference between stellar (no movement) and meteor magnitude Δm_{move} (see Equation 3.20).

$$\begin{aligned} \Delta m_{move} &= 2.5 * \log_{10}(d_{travel}) \\ &= 2.5 * \log_{10}\left(\frac{180 * r * V_m * t_{exp} * \sin(\xi)}{\pi * FOV * R * FWHM}\right) \end{aligned} \tag{3.20}$$

The following parameters are needed to calculate the magnitude reduction due to the meteor travel distance d_{travel} in the sensor: One factor contributing to d_{travel} is the *angle between the meteor radiant and the optical axis* (ξ). An angle of $0\,\deg$ means the meteor travels directly towards the sensor. Thus, the radiance is not spread over several pixels.

Next, the FOV of the lens and the field of view of one pixel (*IFOV*) are taken into account. The IFOV is the angular resolution of one pixel. A large angular resolution results in a longer time the meteor spends on one pixel (*dwell time* t_{dwell} is large) and thus a smaller travel distance on the sensor.

More factors influencing the travel distance are the *meteor speed* V_m, the *range to the meteor* R, both influencing the apparent angular velocity. A higher distance and a lower velocity result in a lower angular velocity and thus travel distance.

The number of pixels is also taken into account via the *resolution of the sensor* r. A large number of pixels, at the same sensor size, implies smaller pixels and thus a larger travel distance.

The *exposure time* t_{exp} directly influences the signal: A higher integration time results in a larger travel distance on the sensor. Although a longer integration time increases the amount of radiance hitting the sensor, this is only true as long as the integration time is lower than the dwell time of a meteor on one pixel. Due to the high speed of a meteor, the dwell time is usually an order of magnitude lower than the integration, which means increasing the integration

time is not increasing the signal. The dwell time can be calculated from the travel distance during the integration time.

The last parameter is the *FWHM*. This parameter describes how a point source (e. g. a star) is imaged on the sensor. Due to the physical limitations of the lens, a point source is spread over several pixels. The FWHM describes over how many pixel a point source spreads, by giving the width (in *pixel*) of the point source at half of the maximum measured intensity. Both, camera sensor and lens influence the value of the FWHM, which can be measured which can be measured using star images.

To summarize the signal calculations, Equation 3.15 is used with the values for radiance in Table 3.2 to calculate the signal for meteor of absolute magnitude 0. Since distance and meteor velocity reduce this signal, the absolute magnitude is reduced as shown in Equation 3.19 and 3.20. This means, with both equations the apparent magnitude of the meteor is calculated, taking into account meteor properties as well as camera and orbit parameters:

$$m_{app} = m_{abs} - \Delta m_{move} - \Delta m_{dis} \tag{3.21}$$

Finally, the signal should be calculated for different absolute meteor magnitudes. The adaption is done by applying the factor between to magnitudes of $\sqrt[5]{100} \approx 2.512$ to the signal. Since the signal is linear to the luminosity, in Equation 2.2 L can be substituted with the signal μ_e in *e-/px*.

$$\begin{aligned} m_1 - m_2 &= 2.5 * \log_{10}(\tfrac{\mu_{e2}}{\mu_{e1}}) \\ \Delta m &= 2.5 * \log_{10}(\tfrac{\mu_{e2}}{\mu_{e1}}) \\ \Delta m * 0.4 &= \log_{10}(\tfrac{\mu_{e2}}{\mu_{e1}}) \\ 10^{(\Delta m * 0.4)} &= \tfrac{\mu_{e2}}{\mu_{e1}} \\ \mu_{e1} &= \mu_{e2}/10^{(\Delta m * 0.4)} \end{aligned} \tag{3.22}$$

Equation 3.22 allows to calculate the signal μ_{e1} for a meteor of magnitude m_1 from the signal μ_{e2} generated from a meteor with magnitude $m_2 = 0$. In this case, Δm equals m_1.

Noise calculations

The noise sources can be divided into two types of noise sources, camera noise and background noise.

Camera noise, as described in Section 3.1.1, depends only on camera parameters and the signal generated by the meteor. Camera noise taken into account in the simulation includes dark noise and shoot noise. The readout noise is usually small and can be neglected, the same applies to the quantisation noise.

The noise calculations described in this section focuses on background noise, e. g. from city lights. Compared to the camera noise, the background noise is smaller than the shoot noise, but contributes significant to the noise level. However, the background noise is difficult to estimate: The source of background noise is light emitted from sources on the ground towards the camera. These are mainly artificial sources like city lights, which are the main contributor. Other sources include transient phenomenons like airplanes or lightning as well as reflections from the moonlight. Those transient sources must be taken into account when developing the meteor detection algorithm, but do not significantly contribute to the constant background noise. Therefore, only the background noise from artificial and quasi constant sources are taken into account.

The next problem in estimating the background noise is the wide range of background noise: Dense populated areas emit orders of magnitude more light than rural areas, over the ocean the background noise is neglectable. Inevitably, the background noise can only represent an average noise. Hence, the noise can be lower or higher, depending on the area the satellite is currently observing. Consequently, the limiting magnitude simulated is an estimation of the camera performance.

The background noise can be considered in two ways:

First, a city can be treated as a point source. In this case, the city is much brighter than the typical meteor, hence a meteor occurring directly over a city can not be observed. Thus, this is not taken into account in the simulation.

The second way of considering city lights is treating them as a diffuse light source: The city light is scattered in the atmosphere, increasing the sky brightness. This so called sky brightness or light pollution is also an problem for ground based astronomy. The approach chosen in the simulation is using an average value for the sky brightness and treating the city noise as a diffuse light source increasing the brightness of the background the camera is observing.

Hereafter, the term background noise refers to noise originating from artificial sources on ground, mostly city lights, increasing the sky brightness due to diffuse light scattering. The background noise is taken from [48, p. 3]. In this paper, the sky brightness is calculated from satellite data using models to calculate the light propagation in the atmosphere. The authors created a map showing the sky brightness around the world, with a colour coding indicating the magnitude of the brightness. For the simulation, the maximum value of $\mu_{p\_Sky} = 2.32 \times 10^9$ *photons*$/(\text{cm}^2 \cdot \text{s} \cdot \text{sr})$ in the astronomical *V band* is used. In order to calculate the background noise from the sky noise, the amount of photons reaching the sensor is calculated by taking into account the solid angle Ω and the lens aperture $A = \pi * (D/2)^2$ (see Equation 3.23). The solid angle is calculated from the FOV using $\Omega = 4 * \arcsin(\sin(FOV_H/2) * \sin(FOV_V/2))$. Furthermore, the quantum efficiency in the V band η_V and the fill factor FF is needed.

$$\mu_{e\_background\_Sensor} = \mu_{p\_Sky} * \left(\frac{1}{10^{-2}}\right)^2 * \Omega * A * \eta_V * FF \quad \textbf{(3.23)}$$

The noise generated per pixel is approximated by dividing the number of electrons generated on the sensor by the total number of sensor pixels (r_X and r_Y are the resolution in X and Y direction).

$$\mu_{e\_background\_px} = \mu_{p\_Senso}/(r_X * r_Y) \quad \textbf{(3.24)}$$

The background noise $\mu_{e\_background\_px}$ from Equation 3.24 is given in *e-*$/(px \cdot \text{s})$, thus it has to be multiplied with the exposure time t_{exp} to calculate the background noise in terms of *e-*$/px$.

To sum up, the largest noise source considered is the shoot noise, followed by the background noise and finally the dark current.

SNR and limiting magnitude calculations

After describing the calculation of signal and noise, the SNR can be calculated. To recapitulate, the signal is derived from the meteor radiance, taking into account camera and orbit parameters. First, the signal is the sum of all electrons generated by the radiance of a meteor with magnitude 0 in the three filterbands B,V and R (Equation 3.16). Next, this signal is adapted to different magnitudes using Equation 3.22. The signal itself is calculated by combining Equation 3.14 (giving the irradiance) with Equation 3.2 (giving the camera signal) resulting in Equation 3.15. The apparent magnitude is calculated by taking into account the reduction due to meteor velocity and orbit altitude (Equation 3.21). Combining all equations result in Equation 3.26 with Equation 3.28 giving the magnitude reduction.

$$SNR = \frac{Signal\ (S)}{Noise\ (N)} \tag{3.25}$$

$$\begin{aligned} S(m_{app}) &= \frac{Elek}{px}_{tot} \\ &\Big| \\ &= \left(\sum_{x=,B,V,R} \frac{E_x * A_{PX} * t_{exp} * \eta_x * \lambda_x}{h * c} \right) / 10^{(m_{app} * 0.4)} \\ &\Big| \\ &= \left(\sum_{x=,B,V,R} \frac{\frac{\pi * T * \cos(\alpha)^4 * D^2}{4 * f^2} * FWHM_x * FF * L_{\lambda_x} * A_{PX} * t_{dwell} * \eta_x * \lambda_x}{h * c} \right) / 10^{(m_{app}) * 0.4} \end{aligned} \tag{3.26, 3.27}$$

with

$$\begin{aligned} m_{app} &= m_{abs} - \Delta m_{move} - \Delta m_{dis} \\ &\Big| \\ &= m_{abs} - 2.5 * \log_{10} \left(\frac{180 * r * V_m * t_{exp} * \sin(\xi)}{\pi * FOV * R * FWHM} \right) - \left\| 2.5 * \log_{10} \left(\frac{100\,\mathrm{km}}{d} \right)^2 \right\| \end{aligned} \tag{3.28}$$

$$N = \sqrt{S(m_{app}) + \mu_{e\_background\_px} * t_{exp} + \mu_I * t_{exp}} \tag{3.29}$$

For the noise calculation, camera noise as well as background noise is taken into account (see Equation 3.29). All parameters are shown in Table 3.1. Furthermore, a more detailed derivation of the signal equation is shown in the Appendix (see Appendix Section A.2).

The simulation calculates the SNR for a given camera system depending on the meteor magnitude. The meteor magnitude is calculated for ground and space based systems as shown in Figure 3.6. Furthermore, the stellar magnitude is used as well. The determination of the limiting magnitude is done by calculating the intersection between the desired SNR value (in this case $SNR = 5$). The intersections gives the stellar limiting magnitude, the meteor limiting magnitude from ground and from orbit.

3.3.2. Simulation input parameters

Before giving details about results, the parameters for the simulation are given in Table 3.3. The stated parameters are needed to calculate the limiting magnitude m_{lim} of a given camera system. They include camera parameters and settings (e. g. quantum efficiency and exposure time) as well as meteor properties and orbit parameters.

The parameters can be adapted to simulate different camera systems or orbits. Furthermore, the meteor properties can be varied. This can be used to evaluate the effect of certain parameters on the limiting magnitude or to get the limiting magnitude for a specific mission and instrument.

As a starting point, the camera properties of the industrial camera for machine vision *GenieNano M1920* from the company *Teledyne DALSA* was used. The camera parameters stated in Table 3.3 are taken from the camera manual (see [45]), except the quantum efficiency, dark current and FWHM, which were measured in laboratory and real sky tests. The camera is simulated with the lens

Table 3.3.: Simulation parameters of the sensitivity simulation with default values from *GenieNano M1920* and *Schneider Kreuznach Cinegon 1.4/12*

Parameter	Default value	Unit
Camera parameters		
Exposure time t_{exp}	$\frac{1}{10}$	s
Quantum efficiency η	B: 73.9, V: 82.9, R: 67	$\%$
Pixel size A_{PX}	5.86	µm
Transmission of lens T	90	$\%$
Dark current μ_I	7	$electrons/(pixel \cdot \mathrm{s})$
Sensor res. r_X/r_Y	1936×1216	*px*
Tilt angle (camera optical axis/nadir) α	0	deg
Pixel fill factor FF	100	$\%$
Binning	2	-
FWHM	4	*px*
f-number F#	1.4	-
Focal length f	12.7	mm
Orbit parameters		
Altitude	565	km
Meteor parameters		
Speed V_m	40.3	km/s
Altitude	100.87	km
Radiant angle ξ	62.55	deg

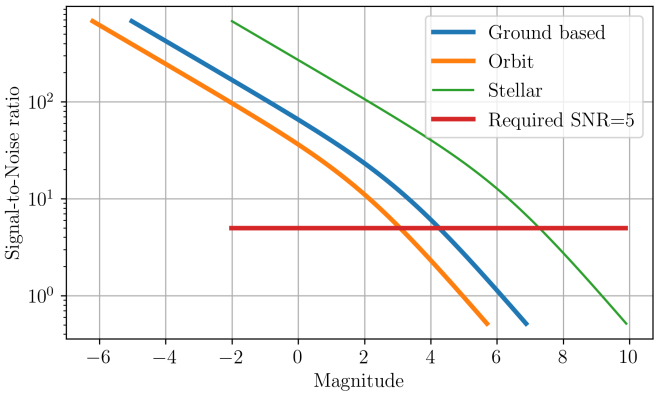


Figure 3.6.: Example SNR diagram from the *Sensitivity* simulation. The intersection between required SNR (red line) and calculated SNR indicates the stellar, the ground based and the orbital limiting magnitude.

Schneider Kreuznach Cinegon 1.4/12 (see [49]). The meteor properties are taken from the analysis of the *CILBO* data base (see Section 2.4).

Determination of Full Width Half Maximum In this context, the FWHM is defined analogue to the FWHM of the filter, but instead of filter transmission, the brightness of an object (in this case a star) is used: The FWHM is the width of the star, at the half of the maximum star brightness measured in *pixel*. This is visualized in Figure 3.7.

For the simulation, first an assumption of a FWHM of 4 *px* was used. Later, after procuring the camera and lens, the FWHM was measured, by taking images on the roof of the IRS at the University of Stuttgart. The Python package *photutils* is used to determine the FWHM. This package provides functions to evaluate and to process astronomical images, here the star finder function is used.

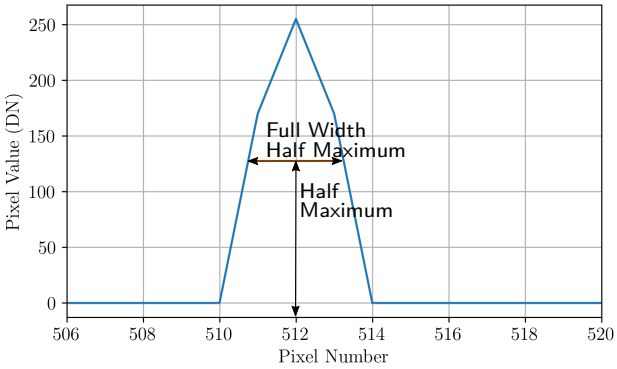


Figure 3.7.: Illustration of the FWHM using the values of a line of pixel through the center of a star on the sensor. The image was taken with the *GenieNano M1920* on the roof of the IRS with binning 2.

The star finder function takes several arguments, one is the minimal FWHM a star needs to have in order to be detected. As a result, the function returns the points in the image identified as a star with the given FWHM threshold. By using different FWHM thresholds, the FWHM can be derived. This is shown in Table 3.4, which gives the number of stars above the threshold.

Furthermore, the detected stars can be marked, as shown in Figure 3.8. A few stars have a FWHM larger than 4 *px*, most of the stars detected with a lower threshold are at the bottom of the image and in fact noise. Thus, a FWHM of 4 *px* is realistic and can be used in the simulation. The measurement is independent from the binning setting: Binning only increases the signal strength, but does not effect the FWHM.

This measurement was confirmed, by manually measuring the FWHM in the image using the image editing program GNU Image Manipulation Program (GIMP).

Table 3.4.: Number of stars with a minimal FWHM

FWHM Threshold (px)	Number of stars
2	374
3	381
4	113
5	29
6	26
7	12
8	4
9	1
10	0

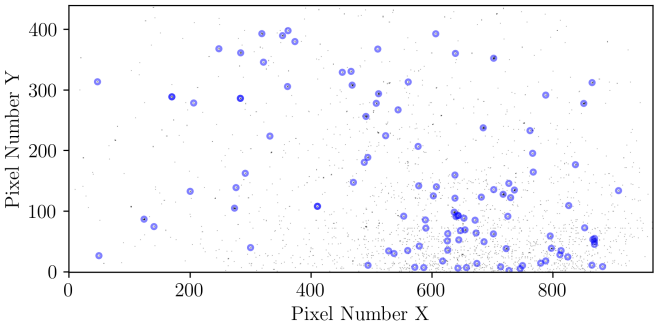


Figure 3.8.: Image of stars taken on the roof of IRS showing stars above a FWHM of 4 *px*. The image is inverted, blacker parts indicate higher brightness. Points detected as stars are marked with blue circles.

Determination of quantum efficiency and dark current First, the quantum efficiency and dark current of a different camera with the same sensor was used since this data is not provided by the manufacturer (see [50]). Later, both values were measured by performing an *EMVA 1288* test. The assumed values and the measured values for both parameters are in the same order of magnitude, with the measured value being better than the assumption.

Meteor properties As a starting point the average meteor properties, as determined by the *CILBO* observatory (see Section 2.4), are used. Later, these parameters are varied, to determine their influence on the limiting magnitude.

Orbit The orbit altitude for the *FACIS* and *SOURCE* mission are used, which is in the range of 300 km to 565 km.

Camera settings The camera settings include binning and exposure time. Binning reduces resolution, but increases the SNR (see Section 3.1.1). The exposure time effects the SNR, but also the *frame rate* (FR) of the camera. The frame rate (given in frames per second (fps)) can not be higher than the reciprocal of the exposure time, e. g. with an exposure time of $\frac{1}{5}$ s, the maximal frame rate is 5 *fps*. As mentioned in Section 2.6, the frame rate influences the scientific performance as well as the requirements for data processing. Therefore, the frame rate is set to 10 *fps* for *FACIS* and 6 *fps* for *SOURCE*.

3.3.3. Results

The simulation is set up with the camera *GenieNano M1920*, equipped with the *Schneider Kreuznach Cinegon 1.4/12* lens. These components were preliminary chosen and represent a possible option for the instrument. The camera has a high quantum efficiency as well as low noise and thus is suitable for the observation of faint objects.

The lens is part of a product line offering lenses with a wide range of focal length and respectively FOV. Furthermore, all lenses are ruggedized, focus as well as f-number can be set and fixed with a screw. Also, those lenses have been used on different satellites[1].

The parameters stated in Table 3.5 have been systematically changed from the values stated in Table 3.3. After each change, the simulation is run and the results are evaluated. This includes the limiting meteor magnitude, but also the observed area, the travel distance of the meteor on the sensor (on how many pixels is the meteor imaged?) and the dwell time (how much time does the meteor spend on one pixel?). The results are summarized in Table 3.5.

Field of View (FOV) A smaller FOV results in less observed area, which does not affect the limiting magnitude, but potentially the number of meteors observed. A larger area observed results usually in a higher number of observed meteors. This effect is further analysed in Section 3.4. Furthermore, the FOV is determined by the focal length, which, together with the f-number, influences the aperture area. A smaller FOV is the result of a larger focal length. According to Equation 3.12, the aperture is reduced in this case, given a constant f-number. A smaller aperture reduces the limiting meteor magnitude, since less light from the meteor hits the sensor. Contrary, the meteor travel distance is reduced, since a smaller area is imaged on the same number of pixels. This is increases the limiting magnitude, since the signal is spread on less pixel. Nevertheless, the limiting magnitude is reduced with a smaller FOV, because the smaller aperture area has a greater effect on the limiting magnitude.

Exposure time Increasing the exposure time does not increase the limiting magnitude. This is due to the fact that the meteor travels very fast. As long as the exposure time is not shorter than the dwell time of a meteor on one pixel, increasing the exposure

[1] see [51], further information received via personal Email correspondence with manufacturer

Table 3.5.: Instrument parameters and their effect on the performance. The change of the parameter is only given in the more favourable direction, although for some parameters this depends on the desired scientific objective.

Parameter	Change	Effect
FOV	Smaller	Smaller area covered (-), smaller travel distance (+), smaller aperture area (-)
Exposure Time	Larger	More dark noise (-), less image data (lower framerate) (+), worse velocity determination (-)
Pixel size	Larger	Longer dwell time (+), more sky noise (-), larger limiting magnitude (+)
Binning	Larger	Smaller image size (+), Longer dwell time (+), larger limiting magnitude (+), Less accurate determination of meteor position (-)
Orbit altitude	Higher	smaller travel distance (+), higher distance to meteor (-), smaller limiting magnitude (-), larger area covered (+)
Resolution	Smaller	Smaller travel distance (+)
Frame rate	Higher	Accurate meteor velocity determination (+), Higher amount of data (-)
Quantum efficiency	Higher	Increases signal strength (+)
f-number	Smaller	Larger aperture (+)

time only increases the dark noise. Decreasing the exposure time can therefore increase the limiting magnitude due the reduction of noise (dark noise and background noise).

A lower integration time also enables a higher frame rate, which benefits the later data evaluation: With a higher frame rate the speed of a meteor can be determined more accurately.

Pixel size and Resolution Special care has to be taken when dealing with the pixel size: If only the pixel size is increased without changing the resolution, the sensor sizes increases. This is due to the effect, that the sensor size is calculated from both values. A larger sensor has different effects. First, the lens could not fit to the sensor, vignetting could occur. Furthermore, due to the larger sensor more background noise is accumulated, due to the larger field of view of one pixel (IFOV). This effect can not be fully compensated by the smaller travel distance of the meteor over the sensor, which means the overall limiting magnitude is reduced. Also, with a different sensor size, the FOV changes. All in all, it is not suitable to only change the pixel size. However, a large pixel size is desirable. This can be seen if two sensors with the same physical dimensions, but different resolutions and pixel sizes are compared: A sensor with less resolution, but larger pixels has a greater limiting magnitude. This is basically the effect of turning binning on, but binning additionally decreases the read out noise (although this effect is smaller for CMOS sensors, see Section 3.1.1). In Table 3.5 it is assumed the pixel size is changed together with the resolution.

Binning Binning increases basically the pixel size. This results in a longer dwell time of the meteor on one pixel, reduced read out noise and therefore increases signal and limiting magnitude. Furthermore, the image size is reduced, resulting in less data to be stored, processed and downlinked. However, due to the increased pixel size,

the meteor position can not be determined as accurately. This influences the later trajectory determination in case of a stereoscopic meteor observation as well as the meteor velocity determination.

Orbit altitude Increasing the orbit height has two opposing effects on the limiting magnitude: On the one hand, the travel distance of one meteor on the sensor is reduced, which increases the limiting magnitude. This positive effect is compensated by the higher distance between meteors and the camera. The result is a reduced limiting magnitude. However, the area covered is increased, which increases the number of meteors observed.

Frame rate The maximum frame rate depends on the exposure time as explained in Section 3.3.2. The higher the frame rate, the more accurate the meteor velocity and in turn, the trajectory can be determined. Unfortunately, a higher frame rate also results in more data being generated.

Results The limiting magnitude for the *GenieNano M1920* equipped with *Schneider Kreuznach Cinegon 1.4/12* lens is 2.88. In Table 3.6 the effect on the limiting magnitude for different parameters is shown.

Effect of meteor properties The simulation can also be used to determine the effect of meteor properties on the limiting magnitude. Besides the meteor brightness, the meteor radiant and the velocity both influence how well the meteor can be observed. A higher meteor velocity results in a lower dwell time on one pixel and therefore reduces the limiting magnitude. A larger radiant means the meteor moves more horizontally over the sensor instead of moving towards the sensor. This also reduces the dwell time and hence the limiting magnitude.

Table 3.6.: Limiting magnitude for the *GenieNano M1920* equipped with *Schneider Kreuznach Cinegon 1.4/12* for different parameter settings. Deviations from settings stated in Table 3.3 are given. Meteor velocity and radiant are changed by adding the standard deviation of those values to the mean value as observed with the *CILBO* observatory (see Section 2.4).

Parameter Change	Limiting meteor magnitude	Magnitude change
None	2.88	0.00
Focal length 20 mm	3.37	+0.49
Exposure time $\frac{1}{6}$ s	2.65	−0.23
Binning/Pixel size 1	2.66	−0.22
Orbit 300 km	3.80	+0.92
Quantum efficiency $-10\,\%$	2.78	−0.10
Angle Radiant 84.81 deg	2.75	−0.13
Meteor velocity 57 km/s	2.50	−0.38
f-number 2.4	1.04	−1.84

Noise Beside evaluating different parameter options, the simulation can also be used to address the different noise sources. Looking at the different noise sources, the main contributor is the signal itself, the so called shot noise. The shot noise is about 20 times larger then the background noise. However, the background noise is still 180 times larger than the dark current. This means, the dark current is almost negligible.

Summary The most important camera parameters for meteor observation are high quantum efficiency, large aperture, low dark noise and short exposure time. The meteor properties have a significant influence on the limiting magnitude as well: A faster meteor results in a lower SNR. Furthermore, the geometry of the observation is important. A meteor moving towards the sensor is more likely to be observed than a meteor moving across the sensor. The orbit has a huge influence as well, a lower orbit results in a better signal. Nevertheless, the orbit also influences the number of meteors observed, which is analysed in Section 3.4. Finally, calculating the limiting magnitude and determining the requirements is a multidimensional problem. Several combinations of parameters can be suitable to fulfil the scientific requirements.

3.3.4. Verification

Before the simulation results can be used for further calculations, the simulation must be verified. The verification of the simulation shall ensure that the results are in a realistic order of magnitude.

Therefore, the stellar limiting magnitude was simulated and compared with measurements. Images of stars were taken inside the *Planetarium Stuttgart* and on the roof of IRS using the *GenieNano M1920*. Those images were used to determine the limiting magnitude.

For the the real sky and the planetarium tests, the camera *GenieNano M1920* is equipped with the lens *Kowa LM6JC* in contrast to the *Schneider Kreuznach Cinegon 1.4/12* to limit costs. Details

Table 3.7.: Limiting stellar magnitude from measurements and simulation

Source of lim. mag.	Settings	Lim. stellar magnitude
IRS roof	Binning 1, Exp 810 014 µs	5.26
Simulation	Binning 1, Exp 810 014 µs	6.70
IRS roof	Binning 2, Exp 250 002 µs	4.00
Simulation	Binning 2, Exp 250 002 µs	5.32
Planetarium	Binning 2, Exp 166 365 µs	6.54
Simulation	Binning 2, Exp 166 365 µs	5.09
Simulation	Binning 2, Exp 166 365 µs, no noise	6.94

about the measurements can be found in [52], here only the results are presented. The limiting stellar magnitude was determined by comparing the image to a map of stars and determining the magnitude of the faintest visible star. This method is widely used in astronomy, but has some limitations. The measurement depends on the weather and the light conditions. Furthermore, the availability and the selection of the stars influence the measured limiting magnitude.

As can be seen in Table 3.7, the simulation values match the measured values, although the simulation predicts higher magnitude values for the real sky tests. This is likely due to the fact that the simulation takes only sky noise in the V-Band into account. Furthermore, the light pollution was high during the image campaign on the roof. This resulted in a lower limiting magnitude than predicted. For the planetarium tests, the calculated magnitude is lower than the measured one. This can also be explained with the noise levels present. In the planetarium no light pollution is present, but the simulation takes it into account. Without the simulated noise, the values of measurement and simulation are in the same order of magnitude.

All in all, the simulated and the measured stellar limiting magnitudes are close, indicating the simulation is working as expected. The remaining difference can be explained with the uncertainties in the simulation as well as in the measurements.

3.3.5. Discussion of results

Despite verifying the simulation and therefore proving the results reliable, the *Sensitivity* simulation has some limitations, mostly due to necessary assumptions. Those shall be discussed in the following.

First, the conversion from magnitude to radiance is not precise. This is due to several reasons: The meteor irradiance is only taken into account in the B, V and R filter bands. This reduces the radiance actually reaching the sensor, since most camera sensors are also sensitive outside of the B filter band. Furthermore, using the FWHM of each filter to calculate the radiance means not taking into account all radiance. On the other hand, the complete filter bandwidth can not be used. This would result in an overestimation of the radiance, since the filters bands are overlapping. Also, it is assumed that the meteor radiance is constant over the filter band, but meteors do not have the same spectral radiance over the whole FWHM wavelength bandwidth. The accuracy of the magnitude to radiance conversion could be increased by using the actual filter transmission curve and integrating the spectral irradiance over the filter transmission. This would require the knowledge of the meteor irradiance for all wavelengths. Overall, the conversion of the apparent magnitude to the corresponding irradiance is an approximation.

Second, the background noise is calculated from the sky brightness in a city inside the V band. This represents a worst case when the satellite passes over a city. However, noise in other bands are not taken into account. Furthermore, it is assumed that the sky noise is the same noise reaching the satellite, but in reality the noise could differ. While this assumption does not concern the comparison of different cameras, it does limit the accuracy of the limiting magnitude for later calculations.

Third, the limiting magnitude is calculated for a meteor passing horizontally over the centre of the sensor. This limits the validity of the results to this meteor, since a meteor on the edge of the FOV suffers from two additional magnitude losses: First, a meteor on the edge of the FOV has a larger distance to the camera, thus the magnitude loss due to distance is increased. Second, the meteor does not appear on the optical axis, which results in vignetting (see [34]) and reduction of lens collecting area.

Fourth, the quantisation noise is not taken into account, which results in a slight underestimation of the noise. However, compared to the previous points the effect is negligible.

Finally, the simulation parameter must be chosen carefully. This regards the camera and the lens parameters on the one hand and the meteor property parameters on the other. If the camera and the lens parameters are not measured, the values stated by the manufacturer must be used. Sometimes not all necessary parameters are provided, so assumptions must be made. Especially the quantum efficiency is not always stated and must be estimated using similar sensors and the sensors responsivity curve. The FWHM must always be measured or estimated. Furthermore, the meteor parameters must be chosen carefully and always specified, when stating a limiting meteor magnitude. This is due to the fact that the magnitude strongly depends on the meteor speed and direction.

All in all, the simulation does have some limitations, but gives accurate results within the given limitations. The *Sensitivity* script can be used to compare the performance of different cameras and lens combinations, to determine important camera parameters and to estimate the limiting stellar and meteor magnitude for further calculations and simulations.

3.4. Formation Analysis: Optimization of scientific output

The limiting magnitude of a camera system, as calculated in the previous section, is an important parameter to maximize the scientific output of a meteor observation mission. However, for the *FACIS* mission the formation parameters also influence the scientific output. In this context, the influence of orbit parameters as well as the distance between the satellites on the scientific output are analysed. The scientific output is determined by two parameters: The number of observable meteors as well as the accuracy of the trajectory determination. In this section, the *SWARMSv2* simulation is used to analyse the effect of different orbit and formation parameters on the scientific output. Furthermore, the effect of these parameters on the accuracy of the trajectory is analysed by setting up a trajectory simulation based on the *MOTS* algorithm and analysing the data from *CILBO*.

3.4.1. Simulation of number of observable meteors

The *SWARMS* simulation (Simulator for Wide Area Recording of Meteors from Space) is a Python based tool to calculate the number of meteors observable with a satellite. The source code was provided by J. Vaubaillon, details about the simulation can be found in [2]. The analysis by Bouquet et al. was done for a single satellite mission to determine ideal camera and mission parameters. The main result for a single satellite mission is that a high orbit and a large FOV is desirable to increase the number of observed satellites. Furthermore, tilting the camera would increase the observed area and therefore the number of observed meteors.

In the original version, only one satellite is used. Therefore, the simulation was adapted for a stereoscopic meteor observation mission. Other adaptions concern the data evaluation and integration of the *Sensitivity* script which is described in the previous section. Those adaptions started in the scope of a bachelor thesis under the

supervision of the author (see [53]) and continued afterwards. The original simulation (by J. Vaubaillon et al.) is called *SWARMS*. The version used in this thesis (adapted among others by the author) is called *SWARMSv2* to distinguish between the original version and adapted version.

In this chapter, the working principle is presented before explaining the necessary adaptions and presenting the results of the simulations. Finally, the results are discussed.

Working principle

The goal of this simulation is to determine the number of meteors observable with a satellite based camera. Therefore, the user defines mission characteristics, such as the mission duration and the orbit altitude, as well as camera characteristics, such as the FOV, and meteor characteristics, such as the meteoroid masses. Depending on the input data, the program calculates the occurring number of meteors and those detectable by the specified camera. The following section is a short summary of the according paper (see [2]), more details can be found there.

The first step is the generation of meteoroids, which is done as follows: The number of meteoroids occurring in the camera FOV is determined using a settable meteoroid mass model (e. g. from Halliday or Grün). Those models give the number of sporadic meteoroids per time and area occurring in a settable mass range. Besides its mass, the following meteoroids properties are assigned to each meteoroid: Velocity, position respectively distance to the satellite, density, and ablation coefficient. The positions of meteoroids are randomly chosen inside the FOV, since meteoroids can occur anywhere. The velocities are randomly assigned, but the velocity distribution is Gaussian with a settable mean and standard deviation. This applies to all other properties (densities and entry angle β (slope between horizon and trajectory)) as well.

Since the physical meteoroid properties are randomly assigned, each simulation delivers unique results. Therefore, the program allows to conduct various simulations with the same input data and then uses the average value of detected events for the detection rates.

In the second step, it is determined whether a meteor, resulting from the given meteoroid properties, can be detected. Therefore, the kinetic and luminous energy of each meteoroid as well as duration are calculated using the velocity and the luminous efficiency. The luminous efficiency describes the percentage of kinetic energy converted to luminous energy. In order to determine the detectability of a meteor, the luminous energy released by the meteoroid is compared to the minimal energy detectable by the camera system. If the released luminous energy is larger than the minimal detectable energy, the meteor is considered to be detected. The minimal detectable energy is derived from the limiting magnitude of the camera as well as the exposure time and the minimal number of frames on which the meteor should occur. A larger exposure time and a higher number of frames means the energy of the meteoroid must be larger in order to be detected, since it must maintain the light emission over a longer time period.

After the detectability for all meteoroids is determined, the total number of meteors detected by the camera systems, as well as the meteor detection rate (given in *meteors*/h) can be calculated. This allows to estimate the number of observable meteors using different meteoroid mass models and to determine the influence of different camera (exposure time, field of view,...) and mission parameters (orbit altitude, minimal number of frames for detection) on the meteor detection rate.

Adaptions

The main adaption of the *SWARMS* simulation is the introduction of a second satellite. This is needed, since for a trajectory determination a meteor must be observed by both satellites. Thus, the

detection rates of meteors observed by two satellites is of interest. Furthermore, this detection rate also depends on the formation parameters and the orientation of the satellites. The adaption allows to determine the effect of satellite distance and satellite orientation on the detection rates for a stereoscopic meteor observation. The second satellite is in the same orbit as the first, the orientation (tilt angle) is settable by the user. A range of satellite distances can be given and the simulation is run for each distance.

For the determination of the meteor detection rate, the area observed by both satellites is used. The covered area depends on the orientation and the distance of the satellites:

The orientation of the satellites is described by the tilt angle, which is the angle between the optical axis and the nadir direction. A higher tilt angle results in a larger area covered by each satellite. The more the satellites are tilted towards each other, the further apart the satellites must be, in order to maximize the area covered by both satellites. Thus, the distance between the satellites (baseline) is calculated depending on the satellite tilt angle.

The baseline is calculated with trigonometric functions and the law of cosines (see Figure 3.9). Since the orbit is circular, the distance is represented with the phase angle φ. The ideal phase angle depends on the FOV, the satellite tilt angle α and the orbit height R_{orbit}.

The ideal baseline and according phase angle to maximize the area covered by both satellites can be derived from basic geometry. The calculations are shown in the appendix (see Equation A.8 to Equation A.17). The Equations are valid for tilt angles $\alpha \leq \alpha_{\mathsf{max}}$, where α_{max} is the maximum angle at which the camera does not look over the planet's horizon. It is defined as:

$$\alpha_{\mathsf{max}} = \arcsin(\frac{r_{\mathsf{planet}}}{r_{\mathsf{orbit}}} - \frac{\mathsf{FOV}}{2}) \qquad \textbf{(3.30)}$$

In Figure 3.10, the ideal satellite distance to maximize the overlapping FOV according to the presented calculations is shown. The

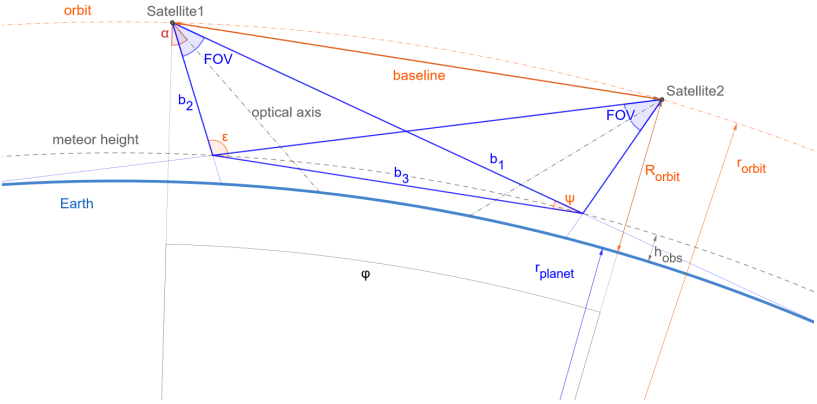


Figure 3.9.: Geometry for determination of optimal angle φ. Schematic adapted from [53].

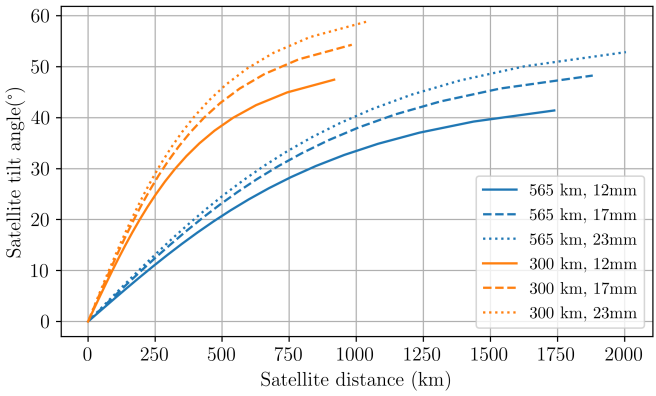


Figure 3.10.: Ideal satellite distance for maximum overlapping FOV for different orbits and focal length options

calculations are done for the minimal and maximal *FACIS* orbit altitude of 300 km and 565 km respectively. As can be seen, a higher tilt angle requires a higher satellite distance in order to maximize the area covered by both satellites. Furthermore, a higher orbit requires less distance for the same angle. However, a lower orbit allows higher tilt angles, before the FOV extends over the horizon. The above described adaptions were done in the scope of a bachelor thesis (see [53]).

Another important adaption is the implementation of the *Sensitivity* script (see Section 3.3), which gives the limiting magnitude of the instrument. This allows to quickly simulate different cameras and lens combinations. Furthermore, the limiting magnitude as well as the number of observable meteors depend on the exposure time, thus a change in exposure time is taken into account for both values. Finally, the *Sensitivity* script also gives the reference angular velocity, for which the limiting magnitude is valid. The difference between actual meteor and reference angular velocity is taken into account in the *SWARMS* simulation, to determine whether a meteor can be detected.

The third major adaption is the option to simulate an Attitude Control System (ACS) error. Since the satellites can not be perfectly orientated, the area covered by both satellites can change due this error and thus influence the meteor detection rate. Using this option, it is possible to simulate the effect of a range of attitude errors on a given satellite axis.

More adaptions include mainly the export of results and the generation of different plots for evaluation. All results, settings and data (e. g. the meteor properties) can be exported into an Excel file for later evaluation or use in other simulations.

Furthermore, it is possible to generate different plots, e. g. showing the satellite formation and covered area, as well as the projection of the meteor track on the sensor.

All important adaptions are summarized in the following list:

- Addition of a second satellite

- Calculation of detection rates for one satellite or formation of two satellites
- Calculation of optimal satellite distance to maximize the covered area
- Calculation of limiting magnitude using sensitivity script (see [54])
- Option to simulate Attitude Control System error
- Export data and plots of results and formation geometry
- Projection of meteor track on sensor
- Implementation of rectangular Field of View
- Addition of lateral meteor trajectory angle β
- Calculation of meteor magnitude
- Implementation of different meteor mass distribution models
- Option to change meteor speed
- Option to use European Cooperation for Space Standardization (ECSS) speed distribution (see [37])
- Implementation of more plots of satellite formation and results
- Option to simulate different parameters in one simulation run

Approach

The *SWARMSv2* simulation is used to evaluate the effect of different orbits, tilt angles and FOVs on meteor detection rates. These rates depend on various parameters (see Table 3.8). The parameters are not independent and influence each other. Therefore, it is difficult to evaluate the effect of a single parameter.

For the evaluation, the meteor detection rates are plotted against the satellite tilt angle. The tilt angle indicates the area covered by both satellites, a higher tilt angle results in a larger observation area (see previous section). Since the satellite tilt determines the satellite distance, this approach allows to evaluate the effect of different parameters depending on the satellite tilt without minding the satellites distance.

It is important to note that the detection rates in the following plots are only used to evaluate the relative effect of a parameter change on the detection rates. The meteor detection rates do not

represent actually expected rates, since some preliminary assumptions about the mission and camera properties have been made at the time the simulation was conducted. Updating the simulation is not required, this only the effect of parameter changes should be evaluated. The expected detection rates for *FACIS* and *SOURCE* are estimated in Section 5.1.2.

Different camera, orbit and meteor parameters must be set before a simulation can be started. The settings used for the following simulations are shown in Table 3.8. The meteor properties are derived from the *CILBO* data base (see Section 2.4) except the speed distribution, which is set up according to the ECSS standard. For each parameter on the top of the table, a simulation is conducted for all tilt angles. The parameters examined include orbit altitude, focal length, meteoroid mass model and mass range of meteoroids.

Table 3.8.: Simulation parameters for SWARMSv2 simulations

Parameter	Values
Parameter changed for each simulation	
Orbit altitude	High: 565 km, Low: 300 km
Focal length, F#	Wide FOV: 12.7 mm, 1.4, Narrow FOV: 35 mm, 1.9
Models	Grün, Halliday
Mass range Grün	Low: 0.01 g to 10 000 g High: 0.1 g to 10 000 g
Mass range Halliday	0.01 g to 30 000 g
Grid steps latitude/longitude	200/1000
Parameter constant for each simulation	
Tilt angle	0, 5, 10, 15, 20, 25, 30, 35, 40 and 42 deg
Camera	*GenieNano M1920*
Number of simulation runs	2
Obser. time Grün	40 h
Obser. time Halliday	674 h ($\times 2$ for low orbit)
Meteor angle γ mean	$-62.65\,\mathrm{deg}$
Meteor angle γ sigma	$22.48\,\mathrm{deg}$
Meteor angle β mean	$90\,\mathrm{deg}$
Meteor angle β sigma	$20\,\mathrm{deg}$
Speed distribution	From ECSS standard
Number of frames for detection	3
Exposure time	$\frac{1}{6}$ s

Results

After the simulation runs, the results are evaluated for each parameter by plotting the satellite tilt against the meteor detection rate of both satellites. Generally, a high detection rate is desirable, thus it is used to determine the ideal parameter.

Effect of orbit altitude and operation time In general, the greater the covered area, the higher the meteor detection rates. Thus, a higher orbit results in a greater coverage and detection rates as seen in Figure 3.11. The number of detected meteors fluctuates due

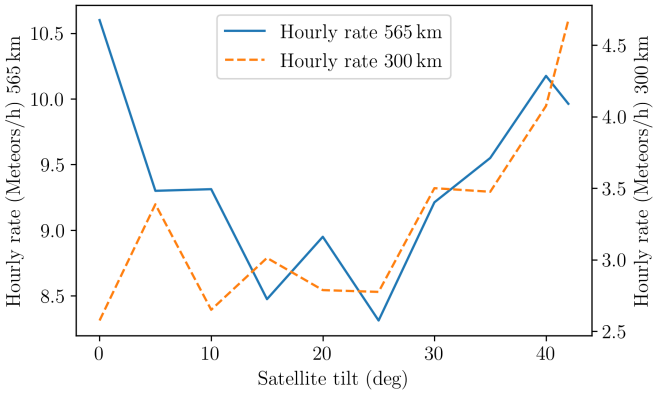


Figure 3.11.: Hourly rate for 565 km and 300 km orbits using the 12 mm lens and the Grün model with a minimal meteoroid mass of 0.1 g

to the random assignment of meteor properties.

While a higher orbit increases the coverage and thus the detection rate, the orbit altitude also influences the operation time. The operation time is the time, during which both satellites are in eclipse. A higher orbit decreases the eclipse time. Therefore, it was analysed whether the greater coverage or the longer operating time has a

larger impact on the number of detections. Furthermore, the satellite distance also influences the operation time and was taken into account as well. The higher the satellite distance, the shorter the time at which both satellites are in eclipse.

The analysis was done by determining the total operation time during the mission for each scenario with ASTOS and then calculating the number of detected events during the mission using the hourly detection rates from *SWARMSv2*.

As a scenario, the 565 km orbit and 300 km are simulated with different Local Time of Ascending Nodes (LTANs). Furthermore, different satellite distances and according tilt angles are simulated. As explained before, the tilt angle and distance are depending on each other. Therefore, suitable distances and tilt angles are simulated as outlined in Section 3.4.1. The parameters are summarized in Table 3.9.

Table 3.9.: ASTOS simulation settings, the camera tilt angle also defines the phase angle respectively satellite distance

Parameter	Value
Altitude	300 km and 565 km
Orbit type	SSO
LTAN	10, 11 and 12
Phase angle 565 km orbit	7.97°, 9.02°, 10.31°, 12.02° and 14.58°
Satellite distance 565 km orbit	964 km, 1091 km, 1247 km, 1452 km and 1760 km
Camera tilt angle 565 km	33.7°, 35.9°, 38.2°, 40.4° and 42.6°
Phase angle 300 km orbit	4.02°, 4.52° and 5.13°
Satellite distance 300 km orbit	468 km, 526 km and 597 km
Camera tilt angle 300 km	38.2°, 40.4° and 42.6°
FOV	48.14° × 31.34°

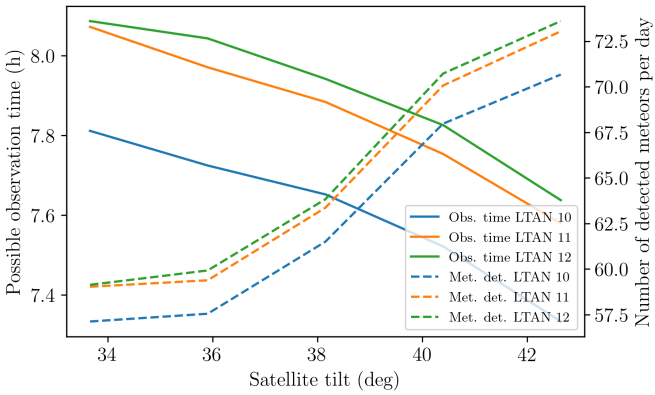


Figure 3.12.: Phase angle vs number of meteors and observation time for different LTANs (565 km orbit)

In order to evaluate all possible scenarios fast, a Python script was developed. In this script, different orbit parameters (altitude, LTAN, satellite distance, FOV) are set and used in an ASTOS scenario. After ASTOS finished simulating and determining whether the satellites are in eclipse, the results are exported and analysed in Python. The operation times per day for each scenario do not differ much for different LTANs ($\sim$ 9 h/d for the 300 km orbit and $\sim$ 7.8 h/d for the 565 km orbit). Details are shown in the Appendix (Table A.8 and A.9).

However, the operation time per day is only one aspect defining the scientific output. The other one is the detection rate, which is also influenced by the parameters tilt angle and altitude. Thus, the *SWARMSv2* simulation is used to determine the detection rates using the above mentioned scenarios and settings according to Table 3.8. The satellite distance and tilt angle are optimized to maximize the number of observed meteors while simultaneously allowing for an accurate trajectory determination (see Section 3.4.3). The resulting detection rates are used with the operation time to calculate the number of observable meteors per day during the 2 *year* mission.

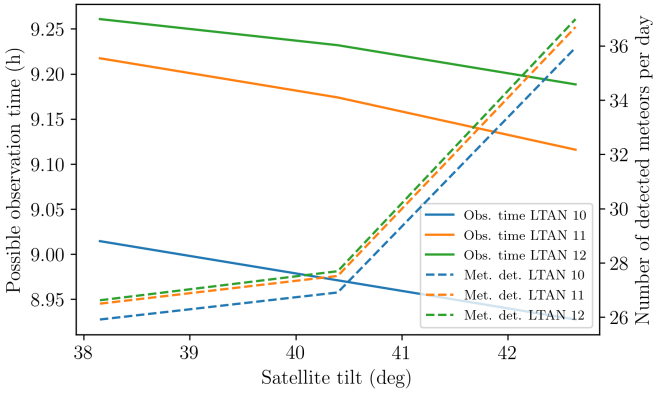


Figure 3.13.: Phase angle vs number of meteors and observation time for different LTANs (300 km orbit)

As can be seen in Figure 3.12, the LTAN has only a little effect on the observation times. A later LTAN results in slightly longer observation time per day, but has little influence on the overall detections. For all LTANs, a larger tilt and phase angle angle results in lower observation time per day, since with a greater distance, the time in which both satellites are in eclipse is reduced. However, since the detection rate increases due to a larger area covered, this effect is compensated and the number of meteors detected increases with the tilt angle.

The same applies when comparing the 300 km orbit (see Figure 3.13) with the 565 km orbit: At each tilt angle and LTAN, the observation time in the 300 km orbit is longer. However, the greater coverage of a higher orbit results in a larger meteor detection rate and compensates for the shorter operation time.

To sum up, a higher orbit is more suitable in terms of detected meteors during the mission, taking into account coverage and possible observation time. The LTAN has only a little effect and can be chosen depending on the available launch opportunity. If it is

possible to chose the LTAN, a later one should be chosen due to the longer observation time.

Effect of tilt angle The greater the tilt angle, the greater the covered area and thus the number of meteors inside the FOV. However, the area increases faster than the number of meteors detected. This is due to the greater distance to meteors at the edge of the FOV, which limits the detectability of low mass, and therefore faint meteors (see also next paragraph and Figure 3.16). As a result, the hourly rate decreases first with an increasing tilt angle. Furthermore,

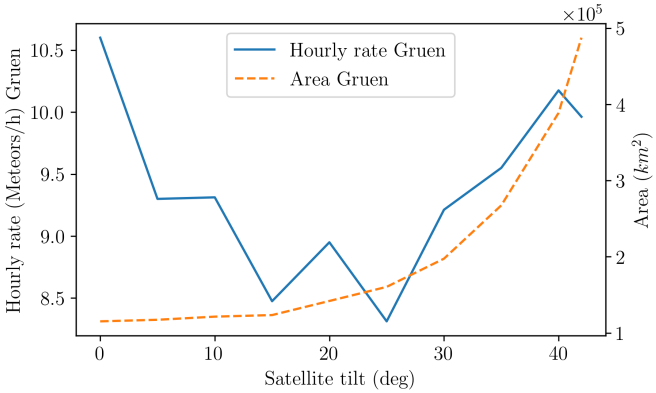


Figure 3.14.: Hourly rate and covered area for an 565 km orbit

Figure 3.14 shows that the increase in area is low at the beginning, only with a larger tilt angle the area increases significantly. This is the case for tilt angles greater than 20 deg. That explains why the hourly rate decreases first with tilt angle (due to greater distance) and then increases due to greater coverage. A useful tilt angle must therefore be greater than 20 deg. The same effect is observed in [2, Fig. 9]. The maximum usable angle is reached when the edge of the FOV hits the horizon.

Effect of meteor mass distribution and tilt angle In Figure 3.15 the hourly rates using the Halliday [29] and Grün [30] model are plotted against the tilt angle. Generally, the Halliday model generates

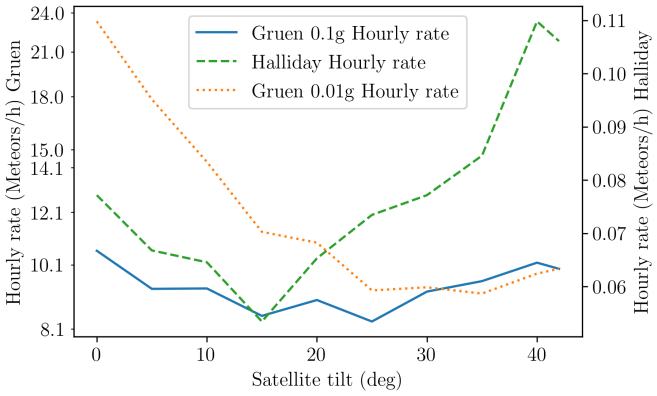


Figure 3.15.: Hourly rate for 565 km orbit and different mass distribution models and settings

less lightweight meteoroids and more heavier meteoroids. The Grün model however, generates more meteors in total, especially towards lower meteoroid masses. In the simulations, the lower masses are set to 0.1 g (Halliday and Grün) and 0.01g (Grün). The mass distribution models do not only have a huge effect on the absolute number of meteors detected, but also affect the relation between tilt angle and detection rate: The Halliday model and the Grün model ($m > 0.1\,\mathrm{g}$) show first a decrease and then an increase in the hourly rate, but those changes are small. In contrast, when using the Grün model ($m > 0.01\,\mathrm{g}$), the hourly rate decreases with the tilt angle, despite a greater coverage. Furthermore, a lower minimum mass results in a higher hourly rate, due to more fainter meteors being generated.

Those faint meteors, together with the geometry of the formation, lead to the decrease of the hourly rate with greater tilt angle. Before explaining this effect, the geometry of the formation needs to be

recapitulated: The tilt angle is set by the user. Depending on this angle, the distance of the two satellites is determined to maximize the covered area. The greater the tilt angle, the greater the distance between both satellites.

The covered surface area increases with tilt angle, therefore the number of meteors inside the FOV increases. But the area increases faster than the number of detected meteors, because faint (and therefore lightweight) meteors far away from the camera can not be detected. This is due to the reduction of apparent brightness from the greater distance to the meteor. If only heavier meteors are simulated (which is also the case when using the Halliday model), this effect is negligible. But for small meteor masses and greater tilt angle, this effect becomes significant. For a great tilt angle, the meteors close to the camera on the first satellite are far away from the camera on the second satellite and vice versa. Therefore, small meteors can only be detected by one camera (see Figure 3.16 for a visual representation). For a small tilt angle, the distance to the meteors are in the same order of magnitude for both satellites. As a result, the hourly rate decreases with increasing tilt angle, due to the greater distance to meteors for one satellite. This effect is stronger at higher orbits.

A higher tilt angle is beneficial for the observation of bright meteors, due to the increased area. For faint meteors a smaller tilt angle is more suitable. Furthermore, a larger tilt angle results in a higher average mass and brightness of the observed meteors, since faint and lightweight meteors are not detected. This effect is significantly seen for higher orbits, at lower orbits the effect is reduced due to the generally reduced distance to the meteors.

Effect of Field of View The effect of the size of the FOV is straight forward: A wider FOV covers a greater area and therefore increases the hourly rate. Beside this obvious effect, the FOV also influences the scientific output of the instrument:

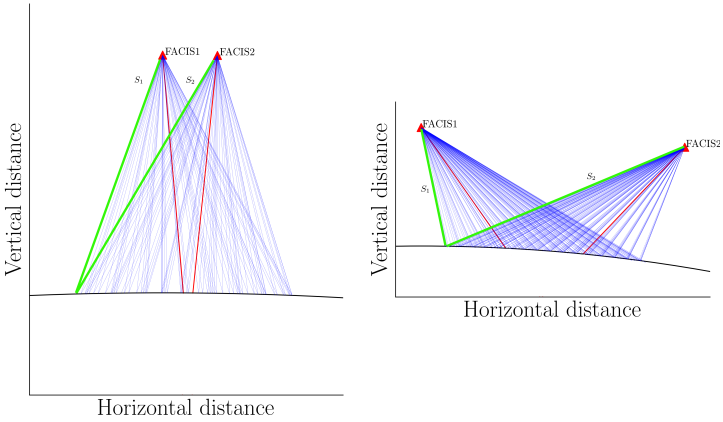


Figure 3.16.: Geometry for a tilt angle of 5 deg (left) and 30 deg (right). Left: The distance to meteors is in the same order of magnitude for both satellites, even for meteors at the edge of the FOV ($S_2 \approx S_1$). Right: The distance from a meteor at the edge of the FOV to satellite one is significantly shorter than to satellite two ($S_2 >> S_1$). Thus, faint meteors at the edge are only detected by one satellite.

A wide FOV should be chosen, if the meteor flux should be determined. In this case, a greater covered area is desirable, to observe as much meteors as possible. For the determination of the meteor trajectory, a narrower FOV results in a greater angular resolution and therefore preciser determination of meteor velocity and trajectory. The angular resolution describes the Field of View of one pixel, the so called Instantaneous Field of View. With a higher angular resolution (small IFOV), the meteor position between consecutive frames also increases. The meteor velocity is determined by the distance between two meteor positions. Since the distance in pixel is larger, the distance calculation is more accurate and thus also the velocity determination.

To sum up, the FOV must be chosen depending on the scientific objective. Furthermore, regarding the trajectory determination, a trade off between number of observable meteors and accurate trajectory determination must be done. Finally, the FOV influences the ideal distance between the satellites (see Figure 3.10), which also influences trajectory determination. Generally, a smaller FOV requires a larger tilt angle to achieve the same satellite distance. In other words: At the same tilt angle, the ideal satellite distance is smaller for a smaller FOV.

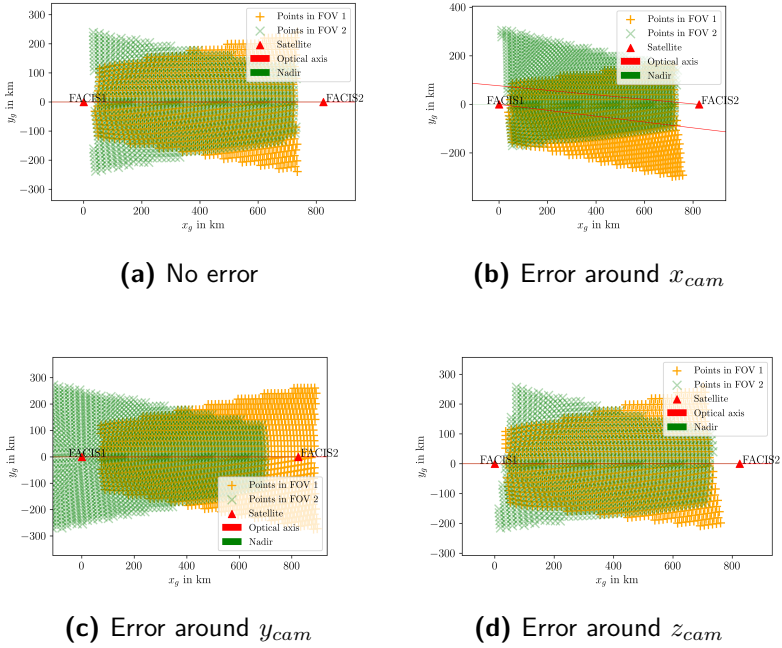
Effect of exposure time and frame rate In the *SWARMSv2* simulation, a meteor is detected if the energy (E) emitted by the meteor is high enough to maintain a luminous energy (I) for a user settable number of frames (see [2, section 2.3.4] and Equation 3.31). E_{min} is the minimal energy a meteor must emit in order to be detected.

$$E_{min} = I * t_{exp} * n_{Frames} \tag{3.31}$$

Since the number of frames for a useful detection is fixed ($n_{Frames} = 3$), a higher exposure time increases the minimal energy needed for detection. Therefore, a higher integration time results in lower detection rate, since meteors are not detected in the required number of frames. Furthermore, a higher exposure time reduces the limiting

magnitude of the camera system, since the dwell time of a meteor on one pixel is usually shorter than the exposure time. Thus, a longer exposure time does not increase the time a useful signal is measured. Finally, in the *SWARMSv2* simulation is is assumed that the frame rate is the inverse of the exposure time. Thus, a shorter exposure time results in a higher frame rate, which is useful for a more accurate trajectory determination.

Effect of attitude errors The *SWARMSv2* simulation allows to evaluate the effect of a settable ACS error. Therefore, one of the camera axis (x_{cam},y_{cam} and z_{cam}) is turned by the settable ACS error.



(a) No error

(b) Error around x_{cam}

(c) Error around y_{cam}

(d) Error around z_{cam}

Figure 3.18.: Top view of coverage (XY plane) for an Attitude Control System (ACS) error of 4°, tilt angle of 30° in an 565 km orbit

In the simulation, the ACS error around one axis at a time can be simulated. However, a range of ACS errors can be simulated in one simulation run. In order to simulate the worst case effect, the satellites are turned in opposite directions. Thus, the area observed by both satellites is reduced. This effect can be seen in Figure 3.18.

In this figure, the top view (looking down to Earth) of the observed area from both satellites is shown. Depending on the axis to which the error is applied, the observed area is differently effected. The largest effect has the same error on the x_{cam} axis, followed by y_{cam}. The lowest change in observed area has the error around the z_{cam} axis, since this is the optical axis.

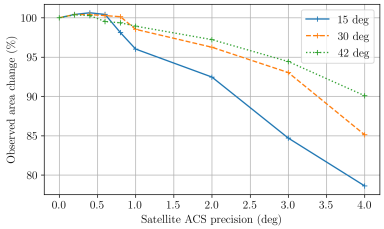
For a more detailed analysis, different tilt angles and ACS errors must be simulated.

In the following simulations, three tilt angles are used to simulate ACS errors between 0° to 4°. As shown later, a minimal tilt angle is needed to allow for an accurate trajectory estimation. Therefore, the tilt angle of 30° is chosen. In order to simulate the effect of a larger and a smaller tilt angle, the tilt angles 15° and 42° are simulated as well. The later one is the largest tilt angle possible, before the FOV extends over the horizon.

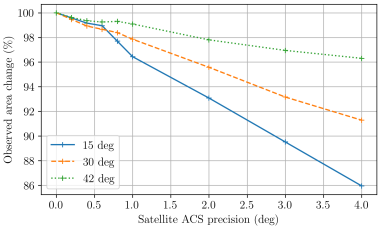
In Figure 3.19, the effect of the ACS error on the area observed from both satellites is shown for each axis. The area for an error of 0° is used as reference (set as 100 %). As can bee seen, small ACS errors have little to no effect on the observed area. Only for high errors the area is reduced significantly. Also, for higher tilt angles, the ACS error has a lower effect.

Furthermore, for small errors the observed area increases a little bit. This is likely to the inaccuracies of the method, which is used to determine the observed area.

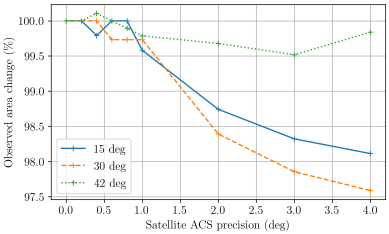
All in all, the effect of the ACS error on the observed area is negligible, because only large attitude errors reduce the area significantly. Even for an 1° attitude error, the area is reduced by less than 5 %. Small satellites usually reach higher attitude accuracies than 1°. For example the Flying Laptop (FLP), the first satellite of IRS,



(a) Percentage of observed area depending on Attitude Control System error around x_{cam} for different tilt angles



(b) Percentage of observed area depending on Attitude Control System error around y_{cam} for different tilt angles



(c) Percentage of observed area depending on Attitude Control System error around z_{cam} for different tilt angles

Figure 3.19.: Effect of 0° to 4° Attitude Control System (ACS) error on observed area for tilt angle of 15°, 30° and 42° in an 565 km orbit on all three axis.

University of Stuttgart (launched 2017) reaches pointing errors below 150″ (see [55]). Since for trajectory determination star trackers are needed anyway to determine the attitude accurately, it is valid to assume the satellite reaches an attitude accuracy better than 1°, because a high precision attitude sensor is usually combined with a high precision actuator. Furthermore, for trajectory determination a larger tilt angle is needed (see Section 3.4.3), thus reducing the effect of attitude errors on the observed area.

Finally, only for large area reductions ($> 10\,\%$) the meteor detection rate is significantly reduced, since the meteor detection rate correlates with the observed area. Refer to Figure A.5 for effect on detection rates and Figure 3.19 for the effect on the observed area.

Therefore, the ACS error in a range typical for small satellites has no significant effect on the meteor detection rate.

Summary

The *SWARMS* simulation has been developed and used to analyse a single satellite mission to observe meteors (see [2]), other single satellite missions have been analysed as well (see [10]). The main result from this analysis is that a greater coverage increases the resulting hourly rate.

However, a mission comprised of two satellites is more complex and differs from a single satellite mission. Therefore, the *SWARMSv2* simulation is developed and used to analyse the effect of camera and orbit parameters on the scientific output of the mission.

For a satellite formation a greater coverage is also desirable, since the coverage has a huge influence on the detection rates. A higher coverage results in higher detection rates, except if the higher coverage is achieved by tilting the camera. This is due to the effect, that tilting the satellites also means increasing the distance between the satellites to increase the overlapping area. The increased distance results in a large distance to the meteor for at least one satellite. As a consequence, the detection rate decreases, since a detection is in this context defined as a detection by both satellites. A

greater tilt angle increases coverage, which benefits measurements of bright meteors, but at the same time decreases the capability to observe faint meteors. Therefore, the tilt angle must be chosen with the scientific objective in mind. For trajectory determination, bright meteors result in a better signal, thus a larger tilt angle is suitable. In this case, the tilt angle should be at least 25°, otherwise the detection rate is at the lowest possible value (see Figure 3.14 based on data from Table A.2). If the scientific focus is also on meteor flux determination, a smaller tilt angle is preferable to observe as many and faint meteors as possible. The orbit altitude has also a significant influence on the coverage and detection rates. A high orbit should be chosen, which increases the total number of detected meteors during the mission.

The selection of the FOV size is a trade-off between accuracy of speed determination and coverage.

For the *FACIS* mission, the highest possible orbit (565 km) should be used to increase the coverage and therefore meteor detection rates. The tilt angle should be at least 25°, since some tilt angle is needed to increase the distance and allow for trajectory determination and a lower angle would reduce the meteor detection. For a lower tilt angle, the detection rate is reduced.

Discussion of results

The *SWARMSv2* simulation is successfully used to determine the effect of different camera and orbit parameters on the scientific output of the mission (detection rates and type of meteors observed). However, two points must be kept in mind when using the simulation and analysing the data: First, the simulation settings must be chosen carefully in order to obtain realistic results. Second, the context of the simulation and the interpretation of the data must be considered.

The settings include the simulation time (what time frame is simulated), the grid size as well as the meteor properties. The simulation time effects the type of meteors generated. If the time

is to low, heavy and therefore bright meteors are not generated, due to the fact that heavy meteors are rare. This is especially true for the Halliday model, since this model generates less meteors in general. Here, a large enough simulation time is crucial, to ensure the generated meteor masses and numbers are representative and large enough to see an effect of parameter change on the detection rates. However, the simulation time should not be too large, since this also increases the run time of the simulation. Simulating 40 h using the Grün model respectability 674 h for the Halliday model results in a high enough number of generated meteors to see effects of parameter changes while keeping the simulation time to around 30 min.

The grid size influences mainly the accuracy of the observed area calculation and should therefore not be to small. Furthermore, meteors appear in the simulation on grid intersections, thus the grid must be closed-meshed to ensure a representative meteor distribution in the FOV. On the other hand, the grid size should not be too large in order to limit the run time of the simulation.

The meteor properties influence whether a meteor can be detected or not and thus the meteor detection rate. Therefore, it is important to choose realistic values in order to evaluate the effect of orbit and camera parameters properly. Furthermore, the simulation is also used to determine which type of meteors can be detected (e. g. low or high speed), which is only possible with realistic input values. Finally, the chosen meteor mass distribution model has a huge influence on the effect of certain parameters. For example, the satellite tilt angle has different effects on the detection rate, depending on the chosen model and simulated meteoroid masses. Thus, it is important to use both models and keep the properties of each model in mind when evaluating the simulation results.

The second point, besides the simulation settings, which must be reminded is the context and significance of the results: The simulation is based on several assumptions (regarding meteor properties) and models (e. g. meteor mass model). Those assumptions are valid, since the meteor properties are taken from *CILBO* data. However,

the meteor properties are still biased, due to instrument biases. The meteor mass models are realistic, but do not take into account meteor showers and the resulting detection rates differ widely. Thus, the presented detection rates are realistic, but should not be taken as the expected detection rate in orbit. In the context of this section, the detection rates are only used to determine the effect of different camera and orbit parameters and are not suitable for the prediction of the scientific output. In order to predict the scientific output, an observation concept must be developed. This concept needs to take into account satellite bus constraints, meteor showers, consider both meteor mass models as well as the performance of the detection algorithm. Such a concept is developed in the scope of this thesis and presented in Section 5.1.2.

3.4.2. Simulation of meteor trajectory accuracy

In the previous section, the ideal distance between the two satellites to optimize the number of observed meteors was calculated (see Section 3.4.1). This is the first aspect of the scientific output for a stereoscopic meteor observation mission. However, the distance between the satellites also influences the trajectory determination. Therefore, in this section the distance between the satellites is optimized to minimize the trajectory error.

The basic principle of a stereoscopic meteor observation is to combine the two dimensional information about a meteor from different observation stations to calculate the three dimensional movement (trajectory) of a meteor. In the scope of the *FACIS* mission, two stations, e. g. satellites, are used. The distance between the stations is important. A low distance does not allow for accurate trajectory determination since the information from the different stations does not differ significantly. A large distance would reduce the number of meteors observed from both stations, since the meteors must be bright enough to result in a sufficient SNR for both stations. Therefore, the distance between the stations must be optimized in order

to maximize the accuracy of the trajectory. In the following, the term station refers to a meteor observing satellite.

Working principle

The simulation has three objectives: The first two are to determine the ideal satellite distance for a low trajectory error and to determine the expected trajectory error. The results and the approach are given in this section. Third, the simulation is used to evaluate the effect of satellite bus parameters (e. g. attitude knowledge) on the trajectory determination. The third aspect is covered in Section 3.5.2. Since the used simulation is the same, the approach to simulate the effect of satellite bus parameters is outlined is this section as well.

In order to fulfil these objectives, the simulation is set up by adapting a meteor trajectory algorithm for ground based to spaced based meteor observation. The Meteor Orbit and Trajectory Software (MOTS) algorithm (see [38]) is used to determine the effect of different parameters on the calculated trajectory. In this context, parameters are the satellite distance, orbit altitude as well as satellite bus parameters (attitude and position knowledge, clock accuracy).

This algorithm is successfully used for evaluating *CILBO* data and is well documented. After adapting and implementing the algorithm in a Python script, a meteor with settable properties (velocity, direction, position) is generated. Two satellites are positioned according to the user settings (e. g. orbit altitude). The needed parameters to calculate the meteor trajectory are the position of the two satellites as well as the meteor position during different time steps. Different error sources can now be applied to these values in order to simulate the effect of e. g. a satellite position knowledge error on the final trajectory. Before describing the simulation setup in more detail, the MOTS algorithm is briefly explained.

MOTS working principle and adaptions The MOTS algorithm is a software program developed by Koschny and Diaz del Rio, which

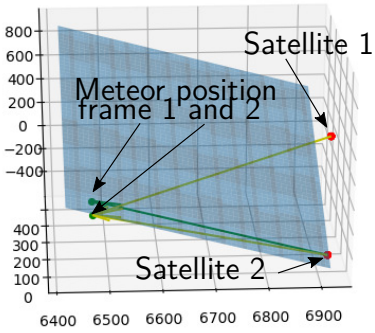


Figure 3.20.: Visualization of the MOTS algorithm. Shown in blue is the plane calculated with satellite 2 as part of the plane and the two vectors (green and yellow) from satellite 2 to two meteor positions. The intersection of this plane with the viewing direction to one meteor as seen from satellite 1 gives the three dimensional position.

calculates the trajectory of a meteor observed from two ground based stations.Only a short description on the working principle is given here, more details can be found in [38]. Generally, the algorithm uses the position of the two stations, their viewing directions and the (two dimensional) position of the meteor to calculate the three dimensional meteor position. See Figure 3.20 for visualization.

The algorithm works as follows: First, a plane is constructed from a point inside the plane and a normal vector of the plane. The point in this plane is the position of Station 2. The normal vector is calculated by calculating the cross product of two vectors. The two vectors are the viewing directions from Station 2 to the meteor in different frames. The meteor position, as seen from Station 2 is derived from the camera orientation. In order to improve the normal vector, the average cross product of all possible vector combinations is used and the average taken as the normal vector. This is only possible if the meteor is visible in more than two frames. The length

of the normal vector is not yet known, because an observation from one station is not sufficient to calculate a three dimensional position. The data of the first Station is required, to calculate the meteor position. This is done by calculating the intersection between the plane and the viewing vector to the meteor as seen from Station 1 (vector from Station 1 to the meteor). The intersection point is calculated for all frames in which the meteor is visible from Station 1. In doing so, the three dimensional meteor position is calculated for different points in time. The trajectory is calculated by fitting a line through the calculated meteor positions. The timestamp of each frame is essential, to calculate the velocity and the trajectory of the meteor.

This process is repeated with the roles of Station 1 and 2 reversed.

For this simulation the algorithm is adapted: Station 1 and 2 are replaced with Satellite 1 and 2. The satellite position is known via Global Positioning System (GPS), the meteor position, as seen from a satellite, can be calculated from the satellite position and attitude. The MOTS algorithm is basically used as described above. In contrast to ground based observations, the satellite moves during the observation. Therefore, the average satellite position is used to set up the plane. After all meteor positions as determined by each satellite is calculated, a line is fitted through all calculated meteor positions, giving the trajectory.

The trajectory error is calculated in the same way as in the *CILBO* project: The median distance of the calculated meteor position and the fitted line is used as the trajectory error. A shorter median distance means the line could be fitted better to the positions. The line and the trajectory error is calculated two times, the second time with the reversed roles of Satellite 1 and 2. The mean error of both trajectory errors is used for the evaluation and called simply trajectory error.

Trajectory simulation setup The simulation is implemented in Python and works as follows: First, the meteor position is calculated

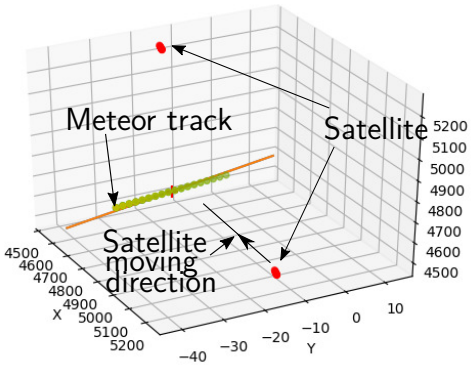


Figure 3.21.: Visualization of the satellite and meteor track

from given properties at different times during the event duration. The times for which the meteor positions are calculated is called the true time, which represents the reference time. The settable meteor properties include lateral and horizontal angle, speed, altitude, position and duration. A simple linear motion is assumed for the short time the meteor is visible. This simulated meteor position is hereinafter called the true meteor position, which is used as a reference.

The two satellites are positioned close to the meteor, with the same and settable distance from the meteor track. The satellite position is set in a way that the middle point of the meteor track is perpendicular to the satellite position (see Figure 3.21). The satellite orbit can be set as well.

From those settings, the satellite position is calculated for the same points in time as used for the calculation of the meteor position (the true time). Those positions are hereinafter referred to as the true satellite position.

This basic setup allows the calculation of the trajectory, the necessary data (satellite position, meteor position as seen from the satellite and time of observation) is given in the simulation. However, the calculated trajectory from stereoscopic meteor observations

is affected by different errors. In this simulation the effect of the satellite position and attitude knowledge accuracy as well as clock accuracy are evaluated. All three errors can be set independently in the simulation. Before calculating the meteor positions with the MOTS algorithm, the input data is altered by a settable error. The error is given as the minimal and maximal deviation (in percent) between the true value and the measured value by the satellite. Three errors can be set: satellite position, meteor position and clock error. The meteor position as seen from the satellite (including the error) is called the satellite meteor position. The satellite position as measured by the satellite (including the error) is called the satellite position. The time as measured by the satellite (including the error) is called the satellite time (see Table 3.10). To each true position and time value an individual error is applied. The error is settable in percent by the user as a high and low value. The high and low value are the limits in which a random number, representing the percentage error, is generated. The percentage error is now multiplied with the true position or time. This gives the random absolute error, which is added to the true position/time and finally gives the position/time as seen from the satellite.

Table 3.10.: Trajectory simulation parameter naming convention

Parameter	Parameter with error
True meteor position	Satellite meteor position
True satellite position	(measured) Satellite position
True time	Satellite time

As stated earlier, the effect of satellite bus parameters (position, attitude and time knowledge) on the trajectory should be evaluated. The bus parameter position knowledge is directly evaluated by setting the error on satellite position.

The second error, which should be investigated, is the satellite attitude knowledge accuracy. This value is derived by calculating

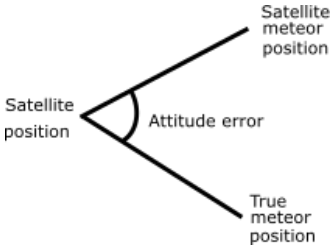


Figure 3.22.: Visualisation of the attitude error in the trajectory simulation. The attitude error is defined as the angle between true and satellite meteor position as seen from the satellite.

the angle between the true meteor position and the meteor position as seen from the satellite (see Figure 3.22). Therefore, the error on the satellite meteor position is treated as the satellite attitude knowledge accuracy. This is valid since in reality the meteor position is calculated from the satellite attitude.

The bus parameter time knowledge can be directly set as the clock error, similar to the satellite position error. However, the clock error does not influence the calculations directly. Instead this error influences both, satellite and meteor position and can be seen as an additional error to both positions:

When calculating the satellite position, orbital mechanic is used to calculate the position for different times. For the calculation the satellite time, which includes the error, is used and affects the accuracy of the satellite position. This simulates the clock error, which comes into effect when assigning a timestamp to a frame. Due to the clock error the timestamp is erroneous, but used to derive the satellite position during exposure. The satellite position is needed for trajectory determination, thus the clock error influences the trajectory determination via the satellite position.

The clock error also influences the satellite meteor position: In reality the satellite meteor position is derived from the meteor image and the satellite attitude. Thus, each image is assigned a time stamp

in order to derive the satellite attitude during exposure. The clock error results in wrong assigned attitude value and thus meteor position. In the simulation the difference between the true time and satellite time is used to calculate the moving distance of the meteor during this time. This distance can be negative or positive and is added to the satellite meteor position as an additional error.

Simulation parameters and evaluation method

As explained in the previous section, the satellite bus parameters are evaluated by setting the according errors. In order to evaluate the effect of formation parameters, orbit altitude and satellite distance can be set as well. While for the *FACIS* mission an orbit altitude between 300 km and 565 km is planned, the simulation includes also higher and lower orbits. For each orbit, different satellite distances are simulated. Each combination of orbit and distance is a separate simulation which is run 200 times to get good average values. This is necessary, because the errors are randomly assigned. The standard settings for the parameters are stated in Table 3.11.

The errors were chosen to represent realistic values: The satellite meteor position error results in an average angle between true meteor position and satellite meteor position of about $12''$. As explained, this angle represents the satellite attitude knowledge error. For a small satellite, this knowledge error is challenging, but possible. For comparison, the satellite FLP reaches a knowledge error between $10''$ to $30''$, depending on the pointing mode (see [56]). A more accurate pointing is feasible, but requires more accurate sensors and actuators as well as a thoroughly tested control algorithm. Another example of a small satellite with a low knowledge error is the TET-1 satellite, which reaches an attitude knowledge of $10''$ (see [57])).

The satellite position error results in an average position error of about 170 m, which is a very conservative value for GPS accuracy.

Table 3.11.: Trajectory simulation standard settings

Parameter	Setting
Meteor Altitude	100 km
Meteor speed	40 km/s
Meteor start angle	45°
Meteor lateral angle	90°
Meteor slope angle	0°
Time step	0.5 s
Meteor duration	2 s
Exposure time	0.167 s
Satellite distances	1°, 2°, 3°, 4°, 5°, 6°, 7°, 8°, 9° and 10°
Orbit	200 km, 300 km, 400 km, 500 km, 565 km, 600 km and 700 km
Simulation runs	200
Low error satellite position	8×10^{-6} %
High error satellite position	4×10^{-5} %
Low error meteor position	0.000 11 %
High error meteor position	0.000 13 %
Low error clock	0.01 %
High error clock	0.02 %

According to the GPS specifications[2], the error should be < 7.8 m. This only takes into account the space segment of the system, however a position error below 50 m is reasonable.

The last error to be set is the satellite time error. The satellite time is set to be accurate between 10 ms to 20 ms, which is challenging but feasible for a small satellite. In this context, the error relates to the difference between satellite time and actual time. In reality the actual time reference would be the GPS time. The time of the satellite can be synchronized by using the GPS Pulse per Second (PPS) signal. Using this method, the FLP is designed to reach an

[2] `https://www.gps.gov/systems/gps/performance/accuracy/`

error of $\sim$ 100 ms, which is indirectly confirmed due to the exact pointing during laser communication passes. An accuracy smaller than 100 ms should be feasible using the same technique[3].

The meteor and camera values are not changed, this includes meteor altitude and speed, duration of meteor event and exposure time. Some values (e.g. meteor angles) were changed only for dedicated simulations to evaluate their effect. Generally, the evaluation is done by plotting the trajectory error against the distance between the satellites for each orbit. The trajectory error is the mean distance between meteor positions as determined by the satellites and the trajectory line. The lower the error the better a line could be fitted through the calculated meteor positions (see Section 3.4.2). The simulation is also used to evaluate the effect of each error and their combination. This is done by setting two or one of the errors to zero.

Results and Summary

In Figure 3.23 the trajectory error is plotted against the distance between the satellites for different orbit altitudes. As can be seen, there is a minimal error depending on the satellite distance for each orbit. The higher the orbit, the higher the distance between the satellites must be in order to achieve a minimal trajectory error. Furthermore, the minimal trajectory error is in the same order of magnitude for each orbit (about 580 m, see Table 3.12 for the exact numerical values). It is important to notice, that the trajectory error is low for a certain range of satellite distances. Therefore, it is possible to vary the satellite distance and still achieve a low trajectory error. Furthermore, the lowest and highest possible satellite distance is calculated. This is done by fitting a function to the data shown in Figure 3.23 and calculating the satellite distances at which the trajectory errors differs by maximal $+5\,\%$ from the minimal trajectory error (see Figure A.18 for the fit function). This gives a range of

[3]Personal communication with Steffen Gaißer, expert in FLP software design and operations, IRS, University of Stuttgart on 2021-07-26.

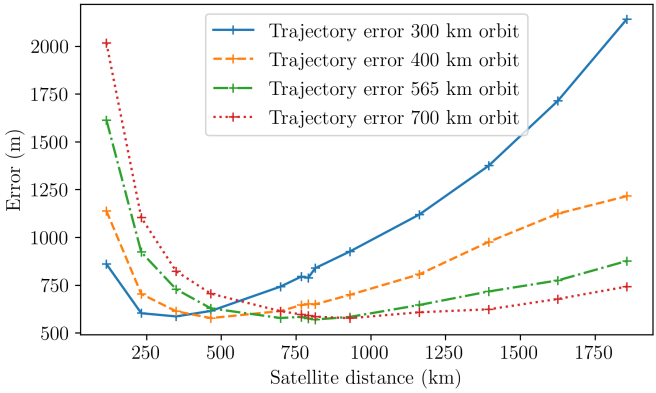


Figure 3.23.: Trajectory error for different orbits and satellite distances

satellite distances at which the trajectory error is in an acceptable range (see Table 3.12).

All in all, the trajectory simulation based on the *MOTS* algorithm gives the ideal distance between the satellites for a given orbit altitude. While doing so, the simulation takes into account satellite bus parameters (attitude and position knowledge error as well as clock error). The effect of different satellite bus parameters is evaluated in Section 3.5.2. The achievable minimal trajectory error is independent from the orbit altitude and amounts to about 580 m with the expected satellite bus parameters. Furthermore, the satellite distance can be set in a certain range, since the trajectory error is low for a certain range of distances.

Discussion of results

In the previous section, the simulation is used for two objectives: Determine the expected trajectory error and determine the ideal satellite distance for trajectory calculation. The simulation fulfils both objectives. However, it has to be kept in mind, that not all errors

Table 3.12.: Satellite distances for a low error in trajectory determination for different orbits. The minimal error satellite distance gives the distance needed to achieve the minimal trajectory error. At the lowest and highest satellite distance, the trajectory error deviates by $+5\,\%$ from the minimal trajectory error.

Orbit (km)	Minimal trajectory error (m)	Minimal error sat. distance (km)	Lowest sat. dist. (km)	Highest sat. dist. (km)
200	589	115	115	212
300	586	349	262	512
400	577	473	343	685
500	581	480	390	845
565	570	847	460	890
600	574	803	456	925
700	577	986	555	1505

sources are taken into account. This is especially true for the trajectory error, which is an estimate of the expected performance. In reality, more error sources influence the trajectory accuracy such as the determination of the meteor position on the image sensor. The meteor position on the image sensor is not accurate, since the meteor moves during the exposure. Hence, a long exposure time reduces the meteor position accuracy. Furthermore, the meteor velocity determination is important to calculate the trajectory and determine the initial orbit of the meteoroid before entering the atmosphere. The velocity determination accuracy is also not taken into account and depends as well on the exposure time. A longer exposure time results in a less accurate speed determination, since the meteor position can not be determined accurately.

Furthermore, the meteor properties are constant for the current simulation and the meteor always appears in the middle of both satellites. In reality the meteor position and properties vary, thus it has to be evaluated which effect different meteor properties have on the trajectory error. Also, the trajectory error should be estimated using a range of different meteor properties. This is done in Section 5.1.2.

For future work more recent algorithms (e. g. [58]) could be used, in order to take into account the movement of the satellites during observation, which results in more accurate determination of the trajectory error.

3.4.3. Ideal Formation Parameters

In order to determine ideal formation parameters, two aspects must be taken into account: First, the number and properties of observable meteors, which is analysed using the *SWARMSv2* simulation (see Section 3.4.1). Second, the accuracy of the trajectory, which is analysed using the trajectory simulation (see Section 3.4.2). Besides other parameters, both simulations vary the satellite distance and orbit altitude. In order to maximize the scientific output, both parameters and their influences on the two aspects (trajectory accuracy and number of meteors) must be taken into account. Furthermore, the tilt angle of the satellites depends on the satellite distance and must therefore also be considered.

Orbit altitude The orbit altitude has no influence on the minimal trajectory error achievable. Only the satellite distance needed to achieve this error changes with orbit altitude.

In order to increase the number of observable meteors, a higher orbit is desirable. A higher orbit increases the observed area and thus the number of observations.

This means, the orbit altitude should be as high as possible, since the trajectory error is not affected, but the number of observable meteors increases. Therefore, the *FACIS* mission should use the 565 km orbit.

Satellite distance and tilt angle For the trajectory error, a range of satellite distances exist, in which the trajectory error is within $+5\,\%$ of the minimal error. The satellite distance should be in the range of 460 km to 890 km for an 565 km orbit (see Table 3.12), while the ideal distance is 847 km. Since the orbit altitude decreases due to the residual atmosphere, the distance for ideal trajectory determination must be adapted as well (262 km to 512 km for an 300 km orbit).

The ideal satellite distance to increase the number of observed meteors depends on the tilt angle of the satellite. A higher tilt angle requires higher distances, in order to maximize the area observed by both satellites (see Section 3.4.1). If the tilt angle and distance between satellites is increased, the number of detected meteors decreases first, before increasing again. Thus, the tilt angle should be larger than 25° (see Figure 3.15) and the distance therefore at least 614 km (565 km orbit) respectively 246 km (300 km orbit). The tilt angle should always match the distances, otherwise the maximum possible area observed is not reached and thus the number of observed meteors is reduced.

Taking into account both distances, the minimal distance ideal for a low trajectory error is slightly larger than the ideal distance for increasing the number of observed meteors. Thus, the trajectory error determines the distance needed, since a lower distance would increase the error. This means, the tilt angle must be increased, in order to bring the ideal distance for number of meteor observation to the ideal distance for a low trajectory error. Using a tilt angle of 32°, the ideal distance for a large number of observed meteors is increased to 892 km. This is close to the ideal distance for

a low trajectory error. Furthermore, the number of observed meteors increases as well with a larger tilt angle, respectively, distance. Therefore, in the following the tilt angle needed is set to 32°. For a 300 km orbit the tilt angle of 38° increases the ideal distance to 467 km and is used for the following calculations.

Conclusion As outline above, both scientific objectives, a low trajectory error and a large number of observed meteors, can be achieved with the same formation parameters (see Table 3.13). The tilt angle should therefore be at least 32° (565 km orbit) respectively 38° (300 km orbit). A consequence of the higher tilt angle and increased distance for the meteor observation is the reduced number of observed faint meteors. Due to the higher distance between the two cameras, a faint meteor can only be detected by one camera. The distance to the other camera is too high and not enough light reaches the camera. On the other hand, the total number of observed meteors is increased due to the increased observed area. Depending on the scientific focus of the mission, a decision has to be made whether the trajectory determination or the observation of faint meteors is more important. However, the tilt angle and distances for ideal satellite distance from the *SWARMSv2* simulation are only valid for the 12 mm focal length lens, which was used for

Table 3.13.: Ideal formation parameters for low and high orbit valid for the *Schneider Kreuznach Cinegon 1.4/12* lens.

Distance range trajectory	Minimal tilt angle\distance from SWARMS	Ideal tilt angle	Ideal distance
	565 km orbit		
460 km to 890 km	25°\614 km	32°	892 km
	300 km orbit		
262 km to 512 km	25°\246 km	34°	389 km

the preliminary analysis of the mission. For a larger or smaller focal length, new simulations determining the ideal distance and tilt angle are necessary (see Figure 3.10). The ideal distance for the *FACIS* mission is analysed in Section 5.1.2.

3.5. Satellite bus requirements for trajectory determination

As mentioned in the Section 3.4.2, the trajectory of a meteor is calculated from the satellite position, the meteor position as seen from the satellites as well as the time of observation. All three parameters are directly measured or indirectly influenced by the satellite bus. Thus, in order to design a satellite bus suitable for a stereoscopic meteor observations, certain requirements regarding the satellite bus must be derived. The satellite bus parameters influencing the trajectory determination are the accuracy of the satellite position and attitude knowledge as well as clock accuracy. Therefore, calculations are needed, in order to estimate the effect of different knowledge accuracies on the meteor trajectory.

The knowledge of the satellite position and attitude is required to determine the meteor position, by relating pixels to coordinates. The position knowledge accuracy describes how accurate the satellite can determine its position. In a Low Earth Orbit a satellite positioning system (e. g. GPS) is often used. The attitude knowledge describes how well the satellite can determine its attitude, which is needed to determine the camera pointing direction. The satellite attitude can be determined with different sensors, e. g. sun sensors or star trackers.

The clock accuracy describes how accurate the time can be measured. An accurate time measurement is important to assign time stamps to images and to correlate telemetry data (e. g. satellite attitude) with the image in order to calculate the meteor position.

The effect of different satellite bus parameters on the trajectory accuracy is analysed with two approaches: First, the *CILBO* data

is used to estimate the effect of attitude knowledge accuracy. The achievable trajectory accuracy for a typical satellite attitude knowledge is calculated. Second, the previously introduced trajectory simulation based on *MOTS* (see Section 3.4.2) is used to derive requirements for satellite position and attitude knowledge accuracy.

3.5.1. Satellite bus requirements derivation from CILBO data

The *CILBO* observatory and data base was introduced in Section 2.4 and is used to derive meteor properties for further simulations. In this section, the data base is used to estimate the expected accuracy of the meteor trajectory determined by satellite measurements depending on the attitude knowledge accuracy (see also [59]). The attitude knowledge accuracy influences the accuracy of the meteor position determination. A higher accuracy of the meteor position results in a higher accuracy of the meteor trajectory.

This also applies to the *CILBO* trajectory determination. The accuracy of the meteor position should be noticeable in the trajectory accuracy. Thus, the following data base values are used for this analysis: The first value is the measure accuracy of the meteor position, which describes how well the meteor position could be derived from its position on the sensor of each camera. It is stated in arcminutes ($'$). Second, the velocity of the meteor is needed. Finally, the accuracy of the calculated trajectory is used. This value states how well the trajectory line is fitted to the measured meteor positions. The accuracy is the mean distance of those meteor positions to the trajectory line. Thus, the fitting error is treated as an error in the trajectory.

As mentioned before, the data base contains different statistical data for each entry, since most parameters are derived from several images. In this analysis the mean and median values are used.

Approach

The objective of this approach is to analyse the relation between accuracy of the meteor position measurement and the trajectory accuracy. Since the meteor position accuracy has a huge influence on the trajectory accuracy, it is assumed that there is a relation between the parameters. This relation should be used to derive requirements for the satellite attitude knowledge. Therefore, it is assumed that for the *CILBO* data the accuracy of the trajectory only depends on the accuracy of the meteor position determination. Thus, the relation is assumed to be $Trajectory\ accuracy\ (in\ \mathrm{m}) = Function(Meteor\ pos.\ acc.\ (in\ '))$.

Once this relation is derived from the *CILBO* data, another assumption regarding the satellite requirements is made. It is assumed that the meteor position error is only influenced by the attitude knowledge, other error sources are neglected. Thus, the meteor position error (given in $'$) is treated as the attitude knowledge error. This gives a relation between the trajectory accuracy and the satellite attitude knowledge error.

In order to have realistic assumptions, only meteors with the same properties (velocity and magnitude) should be used to derive the relation between trajectory accuracy and meteor position accuracy from the *CILBO* data. The only difference between the meteors used is the accuracy of the meteors position, the other parameters are in a defined range.

This is achieved by filtering the data base on putting each selected meteor in an according bin. Regarding the velocity, the meteors are separated into velocity ranges from 10 km/s to 45 km/s in 5 km/s steps.

The magnitude is between -1 and 3 for all meteors. A separation into magnitude bins would result in an insufficient amount of meteors in each bin.

The meteors *within each velocity range* are then categorized into a meteor position accuracy bin. The bins range from $0.01'$ to $0.06'$ while each bin is $0.002'$ wide. For each meteor position accuracy

bin the median and mean trajectory accuracy is calculated from all meteors inside this bin. This gives the relation between the meteor position and the trajectory accuracy. The resulting diagram for a one velocity range is shown in Figure 3.24.

As mentioned, the median and mean values for the meteor position is available in the *CILBO* data base. If the median value of the trajectory error is used for a bin, then the median value of the bin is calculated. In case the mean values are used, the mean value of a bin is calculated. The whole process is done for both cameras of *CILBO* separately.

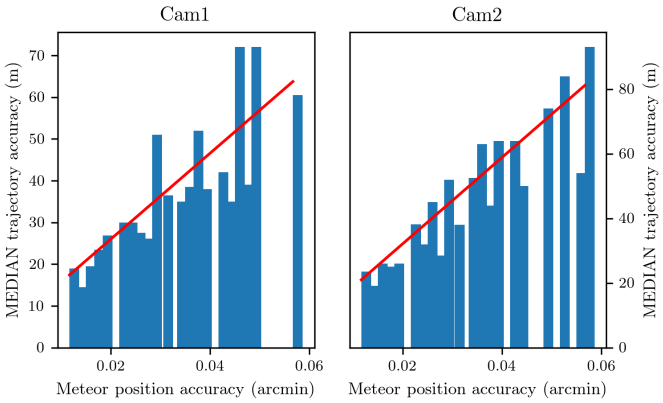


Figure 3.24.: The median meteor position accuracy for a velocity range of 15 km/s to 20 km/s and the median trajectory error. The fitted slope of the median trajectory is shown with a red line.

Results

As can be seen in Figure 3.24, a high error in the determination of the meteor position results in a high trajectory error. Furthermore, the relation between meteor position and trajectory accuracy seems

linear, therefore a line is fitted to the histogram data. The slope of the line is the function needed to derive satellite bus requirements.

Besides of the slope, the standard deviation for each line is calculated and given in absolute and relative values. The exact numerical values are shown in Table A.3 and A.4. These calculations are done for both cameras and each velocity range, all diagrams are shown in the appendix (see Appendix Section A.4). For some velocity ranges, an insufficient amount of meteors have been observed by *CILBO*. This results in bad fit of the line to the histogram.

Generally, the median values can be fitted better to a line. This can be seen when looking at the mean fitting error (standard deviation) in percent, which is calculated using all velocity ranges (see Table 3.14). Therefore, the median values are used for the following calculations.

Table 3.14.: Comparison of mean percentage fitting error for mean and median values

Data basis	Mean std error (%)
Median Cam1	16.0
Mean Cam1	20.4
Median Cam2	19.7
Mean Cam2	22.9

For the median values, the slope is between 1029 m/′ to 1972 m/′, both measured by camera 1. The error in estimating the slope is 16 % for the smaller slope and 12 % for the larger slope. Due to the higher fitting error, the data from camera 2 is not used.

The chosen slope values can be used to estimate the expected error due to the limited attitude knowledge accuracy of a satellite. To recapitulate, the two assumptions made are: The trajectory error is solely a result of the meteor position accuracy error. The meteor position error is solely a result of the attitude

knowledge error. Thus, the expected trajectory error is calculated from different attitude knowledge accuracies. A small satellite, which shall be used for the *FACIS* mission, can reach attitude knowledge accuracies of about 12″ (e. g. the FLP, see Section 3.4.2). In this case, a trajectory error due to attitude knowledge accuracy of 205 m to 394 m can be expected. This is calculated by $Trajectory\ error = slope * attitude\ knowledge\ error$, with the slope between 1029 m/′ to 1972 m/′. The trajectory error is also calculated for different satellite attitude knowledge errors, the results can be found in Table 3.15.

Table 3.15.: Trajectory errors calculated from slope camera 1 median values (1029 m/′ and 1971 m/′) for different satellite attitude knowledge errors

Attitude knowledge accuracy (′)	Trajectory error min (m)	Trajectory error max (m)
0.08	82	157
0.10	102	197
0.12	123	236
0.13	133	256
0.15	154	295
0.17	174	335
0.18	185	354
0.20	205	394
0.22	226	433
0.23	236	453
0.25	257	492
0.27	277	532
0.28	288	552
0.30	308	591
0.32	329	630

Discussion of results

In order to keep the trajectory error below 450 m (see Section 2.6), the satellite attitude knowledge should be better than $0.22' = 13.2''$.

However, this values must be treated with caution due to several reasons. First, the data base used for this analysis contains already processed data. This means, a lot of error sources have already been accounted for. On the one hand, this allows for a more precise error estimation, since the analysis is not influenced by other error sources. However, those compensated error sources in the *CILBO* data base, also occur on a space based instrument. These errors include mainly the non homogenous sensitivity of the camera over the FOV as well as the influence of the pointing direction on the SNR (see [33] and [34]).

Therefore, it has to be kept in mind, that other errors (e. g. velocity determination) also contribute to the trajectory error. Thus the total trajectory error will likely be larger than estimated with this method, but still in the same order of magnitude. Thus, it is desirable to increase the attitude knowledge as best as possible.

All in all, the analysis of the *CILBO* data base gives an estimate of the expected trajectory error due to attitude knowledge accuracy. Furthermore, with a state of the art attitude determination system the trajectory error is in a range, which allows for scientific valuable data acquisition.

3.5.2. Satellite bus requirements derivation from trajectory simulation

The trajectory simulation, as explained in Section 3.4.2, can also be used to evaluate the effect of satellite bus parameters on the trajectory error. The simulation is used to derive satellite bus requirements. Those requirements ensure the trajectory error is in the order of magnitude as defined by the scientific requirements.

Approach

The three errors applied in the simulation are the satellite position and attitude knowledge accuracy error as well as the clock accuracy error. All errors can be set and evaluated independently. For the evaluation, the trajectory is simulated for different satellite distances and the minimal trajectory error is calculated. As explained in Section 3.4.2, the minimal error achievable only depends on the satellite distance and the applied errors. Thus, the effect of different errors can be evaluated by calculating and comparing the minimal error for each simulation case.

The simulation cases are numbered from 0 to 7, with Case 0 being the Baseline case with all errors set to zero. Subsequently, the three different errors are applied in different combinations and the minimal trajectory error is calculated for each case. Each simulation case is simulated for a range of orbit altitudes. Ultimately, the effect of different error combinations is determined, by calculating the minimal achievable trajectory error.

The magnitude of the errors are not changed and are chosen to represent realistic satellite bus parameters. The simulation settings as shown in Table A.5 are the same as used for the previous simulation to derive ideal formation parameters.

Before evaluating the simulation results, it has to be verified that the relative errors given in percent actually convert into the desired absolute errors. The satellite position knowledge error should be around 170 m, the attitude knowledge error around $12''$ and the clock accuracy around 15 ms. All errors refer to the difference between the actual value and the value measured by the satellite bus. Furthermore, those values represent achievable values for a small satellite as outlined in Section 3.4.2.

The verification of the position error is done by calculating the difference between true satellite position and measured satellite position. This gives the satellite position knowledge error, which is shown in Figure 3.25. The position error increases with orbit altitude, since the relative error is applied by multiplying the true

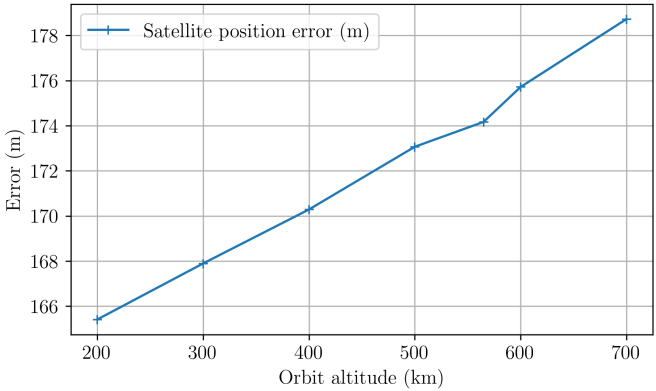


Figure 3.25.: The difference between true satellite position and measured satellite position for different orbits (satellite position knowledge error)

satellite position with the relative error. A higher orbit increases the values of the coordinates of the satellite position, because the centre of the coordinate system is the Earth centre. Applying the same relative error to increasing values results in a larger absolute error. However, the effect is small and negligible. Furthermore, the error is in the range of 165 m to 178 m, which is around the target value of 170 m.

In Figure 3.26, the angular distance between true and satellite meteor position is shown. The distance is about 12″. Since the angular distance is treated as the satellite attitude knowledge error as explained in Section 3.4.2, the desired attitude knowledge error of about 12″ is reached by the relative error.

Verifying the clock error is not necessary, since the time error is directly applied to the true time, giving the satellite time as measured by each satellite. However, the clock error does not directly influence the trajectory, since it is not directly used in the calculations of the trajectory. Instead, this error influences both, satellite and meteor position, and can be seen as an additional error to both

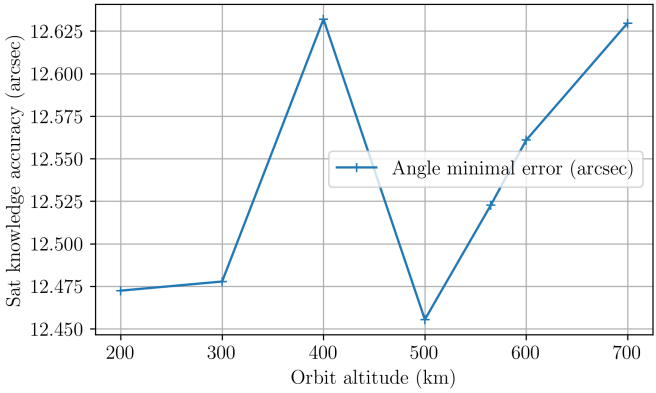


Figure 3.26.: The difference between true meteor position and satellite meteor position for different orbits (satellite attitude knowledge error)

positions. This is explained in Section 3.4.2. In both cases, the clock error simulates the error occurring when satellite bus telemetry data (position and attitude) is correlated with an image timestamp in order to derive the meteor position in the image. Since the bus data and the payload data are recorded by different systems (satellite bus and payload subsystem), the time reference can differ. This error is simulated with the clock error. For the simulation cases, one simulation is run in which the clock error only influences the satellite position and one simulation where the clock error influence additionally the meteor position and therefore the attitude knowledge. A table showing all simulation cases together with the calculated trajectory accuracy is given in the following results section.

Results

First, each of the three errors is evaluated individually before evaluating different combination of errors (see Table 3.16). As can be seen, the attitude error has the highest influence. The clock error

Table 3.16.: Influence of different error sources on trajectory error for realistic error magnitudes and a 565 km orbit

Case	Mean min. trajectory error (m)
Baseline	0.57
Clock error	49.22
Clock error complete	286.7
Position error	88.11
Attitude error	436.04
Attitude position error	444.2
Attitude position clock error	446.22
Attitude position clock error complete	583.24

has the second highest influence, if the error is also applied to the meteor position determination. The position error has the lowest influence. The errors do not add up linearly, which is due to the fact that the errors can cancel each other out due to the random nature of the error. This can be seen when looking at Case 3 to 5: The position error and the attitude error combined (Case 5) is not the sum of the position error (Case 3) and attitude error (Case 4). In the following, the results for each error are analysed in more detail.

Effect of satellite attitude knowledge In the simulation a realistic satellite attitude knowledge error of 12″ is used. It is analysed, which trajectory error could be achieved with this attitude knowledge error. As mentioned before, the satellite attitude knowledge is derived by calculating the angle between the true meteor position and the meteor position as seen from the satellite.

In Figure 3.27 the minimal achievable trajectory error for different orbit altitudes is shown. As analysed in Section 3.4.2, the trajectory error depends on the satellite distance, but the minimal error achievable is in the same order of magnitude for each altitude. As can be seen, the minimal trajectory error achievable is around 436 m, which

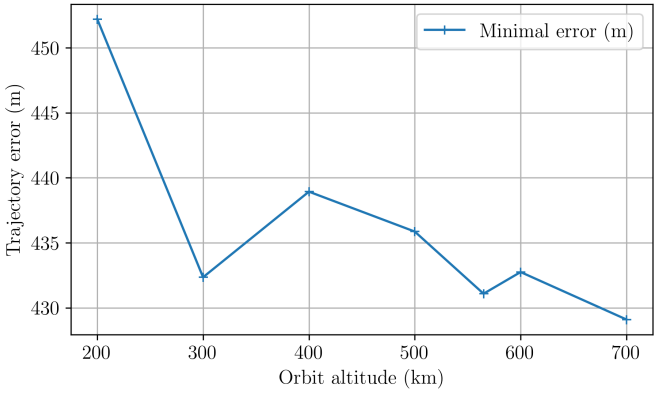


Figure 3.27.: Minimal achievable trajectory error for different orbits caused by attitude knowledge error

is only caused by the attitude knowledge error of 12″. This result is in agreement with the analysis of *CILBO* data, which estimated the trajectory error in the range of 205 m to 394 m (see Section 3.5.1).

Effect of satellite position knowledge The effect of the satellite position knowledge error is evaluated analogously to the attitude knowledge: Meteor position (attitude knowledge) and clock error were set to zero and the high/low error for satellite position were set to result in a conservative satellite position error of about 170 m. This satellite position error results in a trajectory error of about 88 m.

Effect of satellite clock error As mentioned before, the clock error influences the attitude knowledge, and therefore the meteor position determination, as well as the satellite position. Thus, two different scenarios were simulated: In the first scenario, the clock error only influences the satellite position determination (Case 1). In this scenario, the mean trajectory error is about 49 m. In the second

scenario, the clock error also influences the meteor position determination. This results in a total error of about 288 m. Therefore, the clock error has a higher effect on the trajectory error via the influence on meteor position determination than via the influence on satellite position determination.

Summary In Case 5 to Case 7, different combinations of realistic values for each error source (position and attitude knowledge as well as clock accuracy) are simulated and the effect compared to each other. The results are shown in Table 3.16.

When combining realistic values for all three errors (satellite position and attitude knowledge as well as the clock accuracy), the trajectory error is about 580 m (Case 7). This is an estimate of the trajectory error to be expected from the satellite bus parameters for a stereoscopic meteor observation. It is assumed that the satellites have the ideal distance to minimize the trajectory error and the *MOTS* algorithm is used for calculation.

Regarding the satellite bus parameters, it is obvious that the attitude determination error has the highest effect on the trajectory error. The used attitude knowledge of 12″ is realistic but still a challenge for a small satellite. The used position error is more conservative and a better performance can be expected. Even with a optimistic attitude determination and a conservative position determination error, the effect of the attitude determination error is larger. Thus, determining an accurate attitude is essential for a low trajectory error.

Furthermore, a precise time is also important, since the exact observation time has a huge influence on trajectory calculation. Therefore, during instrument development the time synchronisation via GPS has to be considered carefully.

Discussion of results

When considering about the expected trajectory error about 580 m, it is important to mention the limitation and context of the simulation. The calculated error only takes into account the accuracy of the satellite bus sensors for attitude, position and time.

However, it has to be noted that the trajectory error is higher in a real application due to additional errors not included in the simulation. For example, the determination of the photometric centre of a meteor on the image sensor further increases the trajectory error. The same applies to the error in velocity determination.

Furthermore, the error also depends on meteor properties, namely velocity, position, direction of travel and brightness. For this simulation run, average meteor properties derived from *CILBO* data are used. In order to estimate the trajectory error for a given satellite mission, different meteor properties need to be taken into account. This is done in Section 5.1.2, where the expected scientific performance of *FACIS* is estimated.

The expected performance of the satellite formation and the comparison with the performance of existing meteor observation systems is another important question when dealing with the trajectory error. Therefore, the trajectory errors of the *CILBO* data base (as shown in Table 2.2, Section 2.4) can be compared with the expected performance of the satellite formation.

As can be seen, the expected trajectory error of a satellite formation is between the minimal and maximal error of the *CILBO* system: However, the mean and median values of the *CILBO* system are significantly lower compared to the expected error for the satellite formation. Although, it has to be noted that the cleaned data base only contains meteors for which a good trajectory determination was possible. Furthermore, different error sources reducing the accuracy were already removed. However, since the trajectory determination from a satellite based observation systems requires more indirect measurements, it is only logical that the trajectory error is higher compared to ground based systems. For example, in

a ground based system the meteor position can be directly measured by comparing the meteor position with stars in the image. In a satellite instrument, this information must be derived from the measurement of satellite attitude. While the first method also involves errors, the second one involves an additional measurement and therefore a larger error.

3.5.3. Summary of satellite bus requirements

The satellite bus requirements are determined with two approaches:

First, the *CILBO* data was used to estimate the effect of the attitude knowledge error on the trajectory. This method estimated a trajectory error about 205 m to 394 m for an attitude knowledge of 12″.

Since various assumptions were made for this approach and other satellite bus parameters (position knowledge and clock accuracy) influence the trajectory as well, another simulation was set up. The second approach is based on the *MOTS* algorithm and evaluates the effect of three satellite bus parameters on the trajectory error. The three errors are errors on attitude and position knowledge as well as clock accuracy. First, this simulation confirms the evaluation of *CILBO* data regarding the effect of attitude knowledge error. Furthermore, the attitude knowledge error had the largest influence of the total trajectory error, with an error more than twice as large as position and clock error.

Thus, for the satellite bus the attitude determination knowledge is the most important parameter regarding trajectory determination. As defined in Section 2.6, the trajectory error should be in the order of 450 m. A trajectory error around 580 m is achievable with a satellite attitude knowledge error below 13.2″, a position knowledge error below 170 m and a clock precision below 15 ms. A small trade off between the parameters is possible, since different combinations of errors result in a trajectory error below 580 m. The clock precision refers to the difference between an accurate time measurement (e. g. GPS time) and the actual timestamp assigned

to an image. All requirements can be meet with a typical small satellite, the position knowledge using a Global Navigation Satellite System should even be better than 170 m. However, the currently used star trackers for *FACIS* achieve only an accuracy of about 36″. Therefore, the star trackers should be upgraded to allow for a better attitude determination. A lower trajectory error would require a very accurate attitude measurements, which is challenging for a small satellite. Thus, a slightly higher error than the defined 450 m must be accepted when assuming typical attitude measurements for small satellites.

However, the final trajectory accuracy depends on other factors as well. This includes the velocity and meteor position determination accuracy (also depending on camera properties) and the distance between satellites. Thus, the final trajectory accuracy will be larger than 580 m.

Another aspect of the satellite bus requirements is the pointing accuracy of the satellite. The attitude pointing error (difference between actual and command attitude), has no significant effect on the trajectory accuracy. However, a high precision attitude sensor (like star trackers) are usually combined with a high precision attitude actuators (reaction wheels). Any pointing accuracy achieved with reaction wheels and a small satellite are suitable for this mission.

3.6. Simulation summary

An overview of simulations and calculations used to design a meteor observation mission is shown again in Figure 3.28, a detailed description can be found in Section 3.2. Here, only the most important results of the previous sections should be summarized, in order to determine ideal parameters for the *SOURCE* and the *FACIS* mission. Furthermore, some aspects not covered in the previous sections are discussed.

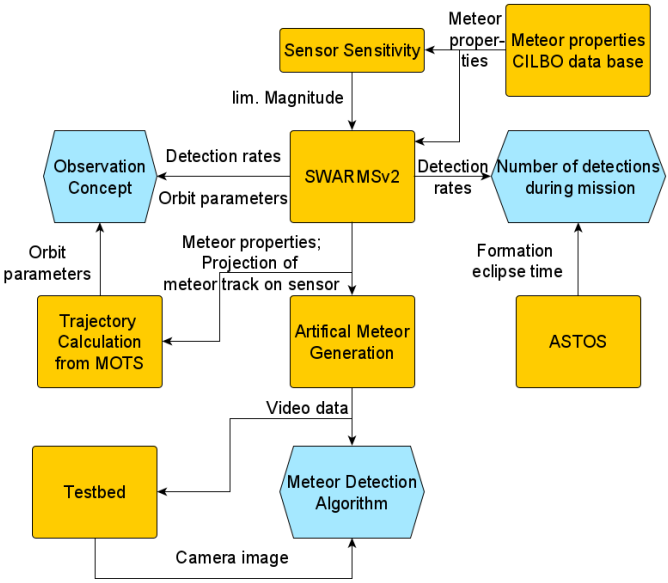


Figure 3.28.: Overview of simulations used for meteor observation mission design

In the following, the ideal formation and orbit parameters (see Section 3.4.3), the instrument parameters (see Section 3.3 and Section 3.4.1) as well as the satellite bus parameters (see Section 3.5) are given for each mission. Since each mission has a different scientific objective, each section is divided into Subsections. The *FACIS* parameters are given in the "Trajectory determination" Subsection while the *SOURCE* parameters are given in the "Flux determination" Subsection.

3.6.1. Formation and orbit parameters

The formation and orbit parameters include the orbit altitude as well as the satellite distance and orientation.

Flux determination In order to increase the number of observable meteors, the orbit should be as high as possible, as long as the FOV does not extend over the horizon. This increases the coverage and therefore the number of meteors occurring inside the FOV. As a drawback, faint meteors can not be observed as good as bright meteors. This bias towards brighter meteors must be taken into account during the evaluation of the data.

During observation the satellite should point nadir or slightly tilted. A tilt angle increases coverage, but results in an uneven sensitivity over the FOV. Faint meteors can only be observed when they occur close to the instrument, which is only possible for a small part of the FOV. This uneven sensitivity is another bias, which must be taken into account when evaluating the data.

Trajectory determination For a trajectory determination instrument, two factors have to be taken into account for the orbit and the formation parameters: The trajectory accuracy and the number of observable meteors (see Section 3.4.3). The orbit altitude only affects the number of observable meteors. As with the Flux determination instrument, the orbit should be as high as possible.

The satellite orientation and distance on the other hand influence both factors. Furthermore, distance and orientation depend on the altitude.

Regarding the number of observable meteors, a higher tilt angle results in a larger distance between the satellites to ensure maximal coverage. However, a tilt angle below $25\,\mathrm{deg}$ results in less observed meteors compared to no tilt angle. Therefore, the tilt angle should be larger than $25\,\mathrm{deg}$.

Regarding the trajectory error, the satellite distance should be high enough to ensure a low trajectory error.

Both aspects can be taken into account with a tilt angle of $32\,\mathrm{deg}$ and a distance of 892 km for a 565 km orbit.

A disadvantage of the necessary distance between the satellites is, that faint meteors are only observed with one satellite. Thus, trajectories can only be calculated for bright meteors.

3.6.2. Instrument parameters

The instrument parameters mainly influence the sensitivity of the instrument and the number of observable meteors. Regarding the trajectory determination, the instrument parameters also influence the trajectory error.

Flux determination For a high sensitivity, the most important parameters are a large aperture and pixel size, a high quantum efficiency and low sensor noise. Furthermore, the exposure time should not exceed the typical dwell time of a meteor on one pixel.

A longer dwell time of a meteor on one pixel increases the SNR, therefore a larger pixel size is desirable.

A high number of observable meteors can be achieved with a large FOV .

Trajectory determination The requirements for sensitivity and number of meteors are the same as for the flux determination.

However, the FOV and the exposure time also influence the trajectory accuracy. A smaller FOV results in a better angular resolution. This increases the accuracy of the speed determination, which in turn is needed for the trajectory determination. Therefore, a huge FOV is not desirable and a trade off between trajectory determination and number of observable meteors is needed.

The exposure time also influences the trajectory determination: With a lower exposure time, higher frame rates are feasible. This results in a better speed determination, since the meteor is imaged in more frames, resulting in more measurement points per meteor. Furthermore, a meteor must appear at least in 3 frames to determine its speed. A higher frame rate increases the number of meteors appearing in one frame.

However, a lower exposure time and thus higher frame rate impacts the design of the meteor detection algorithm and also increases the number of images which need to be processed and downlinked. This has to be taken into account for the instrument design as well.

3.6.3. Satellite bus parameters

The satellite bus parameters include the attitude knowledge, the attitude error, the satellite position knowledge as well as the clock accuracy. A general aspect is the pointing stability of the satellite. In all cases the drift in attitude should be limited to less than the Ground Sampling Distance (GSD) of the instrument to avoid smearing of the image.

Flux determination Generally, the flux determination has low requirements for the satellite bus. The attitude error has not much influence on the number of observable meteors, as long as the FOV does not extend over the horizon. A nadir pointing mode would simplify data evaluation, but is not mandatory. The ACS should be accurate enough, to avoid tilting the satellite too much, since the FOV should not extend over the horizon. Furthermore, a larger tilt angle would influence the scientific output, since faint meteors are less observed due to the higher distance.

The satellite attitude and position is needed if the measurements should be combined with ground observations. In this case, a detailed analysis of the satellite bus requirements is needed. This also applies to the clock accuracy. Information about the satellite attitude is also needed, in case the satellite is tilted, to take into account the sensitivity change over the FOV.

The attitude error (difference between commanded and actual position) has little to no effect on the number of observable meteors.

Trajectory determination The satellite bus parameters have a large influence on the accuracy of the trajectory. The largest influence on this error has the attitude knowledge error, therefore it is the most important parameter. It is followed by the clock error, which is required to get an accurate time stamp for all images. Since all three parameters influence the trajectory error, a trade off between those parameters is possible. For example, a worse attitude knowledge can be compensated with a better position knowledge. However, this is only possible to some degree, since the attitude knowledge has a larger influence on the trajectory error. A combination of an attitude knowledge of 13.2″, a typical GPS position determination accuracy and a clock accuracy of 15 ms is sufficient to achieve a trajectory error below 580 m.

The attitude pointing error (difference between commanded and actual attitude) has little to no effect on the trajectory accuracy and the number of observable meteors. Thus, the attitude pointing error achievable with typical small satellites using reaction wheels is suitable for this mission. However, the pointing should be stable enough, in order to avoid image blurring.

4. Meteor Detection Algorithm

The meteor detection algorithm is a crucial part of the missions. Without a reliable algorithm, a space based meteor observation mission is not feasible. Since the algorithm runs on board the satellite, it is regarded as a part of the instrument.

This chapter describes the meteor detection algorithm, focusing on the design and the performance. Before giving details about the algorithm, the challenge of the algorithm development and its significance for the mission is outlined. Furthermore, basic images processing techniques are explained.

4.1. Technical challenges and significance of onboard meteor detection

The need for an on board meteor detection algorithm arises from two aspects of a space based meteor observation mission: First, the downlink capacity for any satellite mission, and especially small satellites, is limited. Second, due to the nature of meteors, it is necessary to continuously observe the Earth and therefore generate image data. A huge amount of data is inevitably generated which can not all be downlinked. Thus, an onboard detection is needed. The algorithm is needed to process all images generated by the camera and to select images containing a meteor.

The algorithm performance is crucial for a space based meteor mission. If the detection algorithm does not perform as intended, two scenarios are possible. Either the algorithm does not detect all meteors and thus reduces the scientific output or, if to many phenomena are declared as meteors, to much data is generated. In reality no algorithm works perfectly, it is expected that some

meteors are not detected while simultaneously some false detections also occur. Thus, the algorithm must be tested and set up in a way, to balance these two aspects.

Besides this general challenge, the meteor detection algorithm faces several intrinsic technical challenges: Meteors are usually fast (several km/s) and can be faint, thus detecting them is challenging due to the short and low signal. The fact that the number of meteors increases exponentially with decreasing brightness, highlights the importance to detected faint meteors (see Section 2.3). Furthermore, the satellite is moving as well, thus simple detection algorithms based on movement detection as used on ground based system do not work. Also, the background noise from clouds, spikes, airplanes and cities must be taken into account by the detection algorithm.

Those challenges are intrinsic to space based meteor observation, more aspects arise from the satellite bus: The satellite resources are usually limited, which must be taken into account when developing the algorithm. This includes the limited processing power as well as the thermal budget. However, processing data on board a satellite and transmitting only useful data is needed for other missions with limited downlink capacity (see [60]). This includes for example deep space missions. Thus, developing technology for on board processing is beneficial for other missions as well and not limited to meteor observation missions.

4.2. Existing meteor detection algorithms and approach for space based detection algorithm

For ground based meteor observation systems, several meteor algorithms have been developed (see [61]). Different approaches are used to detect a meteor in an image or consecutive frames, depending on the used instrument. Factors influencing the approach are the instrument frame rate, the integration time as well as the FOV.

However, most algorithms try to detect a moving cluster of bright pixels or a line in an image formed by a meteor (see [62])). In most cases, the images are processed before, usually this includes operations to remove the background. This is usually done by applying a "threshold operation to obtain pixel exceedances that rise above the background due to the passage of a meteor" [see 62, Section 2]. Thus, the algorithm assumes a static or only a slowly changing background in order to remove stationary objects like stars (see [63]).

Furthermore, in some algorithms, consecutive frames are summarized in one image to allow for line detection (see [64]). In order to be able to summarize different frames, the background must be static. The same approach is used for the detection of space debris from ground based systems (see [65]), here also consecutive frames are summarized, in order to detect regions not belonging to the static background.

For a satellite based observation, a new approach is needed, since the background changes fast and can not be removed with the methods currently used. Furthermore, city lights would also be recognized as a moving cluster of bright pixels or be detected as a line segment in consecutive frames. Finally, it is not possible to summarize frames into one image for line detection, due to the moving background.

The central challenge for a satellite based meteor detection algorithm is to distinguish between satellite and meteor movement. Thus, the movement of the satellite needs to be considered, before meteors are detected.

A feasible approach is to use optical flow algorithms, which is also established by other teams (see [66] and [67]). Optical flow calculations give the movement of every pixel or a region of pixels between consecutive frames. It is usually used for object tracking and movement estimation. The calculation of optical flow works independently from the movement of the camera, thus it is applicable in the case of a moving background. Therefore, the algorithm developed in the scope of this thesis is based on optical flow calculations.

Details about the algorithm as well as optical flow calculations are given in the following sections.

4.3. Basics of image processing

The algorithm needs to process images in order to detect meteors. Thus, a short introduction to the basics of image processing is given here.

Image processing is usually computational intensive, therefore an efficient implementation of all functions processing image data is needed. Therefore, the algorithm relies on functions provided by the Open Source Computer Vision Library (OpenCV). The *OpenCV* library provides functions for machine learning and computer vision and implements different computer vision and image processing functions (see [68]).

The most important functions used for the algorithm are optical flow and a simple blob detector. The basic principles and use cases for each function are explained in the following sections based on [69].

4.3.1. Basic functions and concepts

Here, the basic functions used in the algorithm are shortly explained, which do not require further elaboration.

Threshold The threshold function is applied to an image matrix, in order to filter pixel values. For example, all pixel values above a certain threshold can be set to zero or to the threshold value. This can be used for noise reduction. In the context of meteor detection, this function can be used to remove all pixels below a certain brightness. Furthermore, this function is used to create binary images.

Binary images These images contain only two pixel values, black (0) and white (maximum value, 255 in case of 8 bit images). They

are generated by using the threshold function and setting all values above the threshold to the maximum value and all values below to zero.

Image Pyramids As described in [69, p.302], image pyramids are used in a wide variety of image processing applications. Here, they are used internally by the optical flow functions. Therefore, it is important to understand what image pyramids are. In an image pyramid, a single image is downsampled (meaning reduced in resolution) multiple times. Important parameters are the number of images in the pyramid (called levels) as well as the scale between two pyramid levels. A level of the pyramid is created by first applying a Gaussian filter to smooth the image, and then removing the according rows and columns as defined by the scale. Usually a scale of 0.5 is used, meaning every second row and column is removed, thus the next layer is a quarter of the previous layer (see Figure 4.1). In the context of optical flow, different image scales (=layers of the pyramid) are useful to calculate the movement on different scales. In an image from the top layer of the pyramid (low resolution), the overall motion in the image is calculated. In a lower layer the detailed motion of specific regions can be determined.

4.3.2. Optical flow

Optical flow calculations are the core component of the algorithm. The main idea of optical flow is to determine the movement of specific or all points from one image to another (see [69, p. 498ff]). The result of an optical flow calculation is the displacement of all or a specific subsets of pixels between two images. The displacement is the moving distance of a pixel or a region between two images. It is given as an x and y component, so the direction of motion as well as the magnitude can be calculated. With a known frame rate, the magnitude can be converted into a velocity. Depending on the scenery, available resources and goal of the image processing, different types of optical flow can be calculated. One assumption of

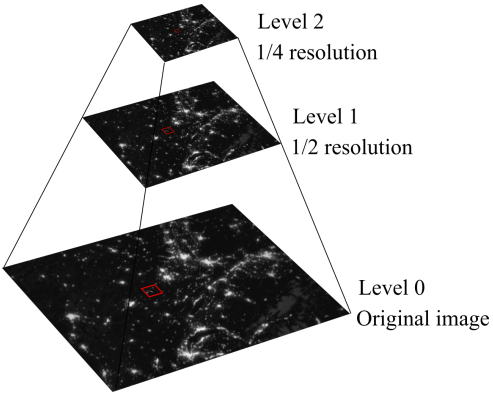


Figure 4.1.: Visualization of image pyramids. The original image (level 0) is downscaled by a factor of 0.5 between each level.

all optical flow methods is, that the brightness of an object moving between two frames is (almost) constant.

Sparse optical flow For the sparse optical flow, the movement of specific regions in the image is calculated. The idea is to select unique parts of an image (so called *keypoints*) and track them from one image to another (see [69, p.493ff and 511]). An advantage of the sparse optical flow is the reduced processing power needed, since only the movement of the selected keypoints is calculated. It is crucial that the selected keypoints are as unique as possible, so that they can be recognized in the consecutive image. However, if the scenery is dark, like the Earth at night, no usable keypoints can be detected. Thus, this type of optical flow can not be used.

Dense optical flow The dense optical flow calculates the movement of every pixel from one image to another (see [69, p.588]). The result of a dense optical flow calculation is a two channel matrix with the same resolution as the original image. Each channel

gives the motion of the according pixel in x, respectively, y direction. This image is also called the motion image. As with the sparse optical flow, the dense optical flow also relies on tracking points from one image to another. However, some regions of an images are not well recognisable from one image to another, this mostly concerns uniform regions. For those regions, the dense optical flow calculations rely on the interpolation of the movement from well trackable regions. This interpolation requires processing time, which makes the dense optical flow orders of magnitude slower than the sparse optical flow.

OpenCV provides two functions to calculate the optical flow, the one used here is called `calc-Optical-Flow-Farneback`. This function calculates the optical flow using the *Farnebäck Polynomial Expansion algorithm*, which works as follows (according to [69, p.588]): It is assumed, that the image can be described as a continuous surface. The image is described by locally fitting a polynomial to every point. Therefore, every pixel value in one part of the image is described by a quadratic polynomial.

Since the image is assumed to be a continuous surface, a small change (displacement) between two images results in a change of the polynomial exponents. From the change of the exponents it is possible to determine the magnitude of the displacement.

Since this works only with small displacements, image pyramids are used: First, the low resolution pyramids are used for the displacement calculations. The displacement is small, because the resolution is small as well. Then, this displacement is used as an estimate for the higher resolution pyramid image. Now, the polynomial exponents are not compared at the same image location, but in the location as estimated by the displacement of the lower resolution pyramids. The number of pyramid levels can be set. Furthermore, at each level the calculations can be iterated to improve the displacement estimation. This can also be set by the user.

Other parameters used are shown in Table 4.1. Before the optical flow calculations, the image is smoothed by a Gaussian filter or

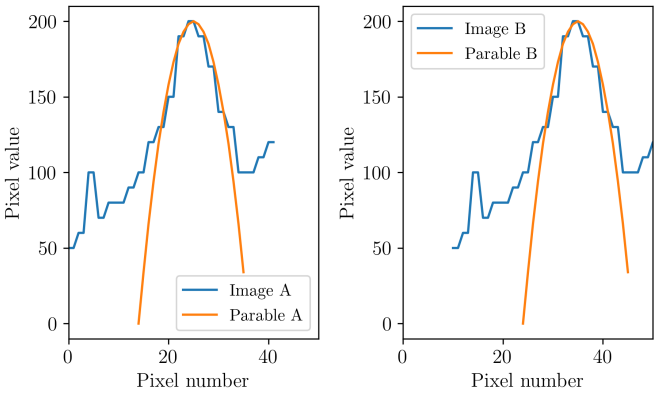


Figure 4.2.: Dense optical flow calculation according to *Farnebäck Polynomial Expansion algorithm*, image drawn according to [69, p.590, Figure 17-1]. A polynomial is fitted to the image data, in this case a parable with the function $f = a_{A/B} * x^2 + b_{A/B} * x + c_{A/B}$, the subscript indicates the according image. Since it is assumed that the parable shape does not change and the displacement d between the parable maxima is small, the distance can be calculated with $d = -\frac{1}{2} * a_A^- 1 * (b_B - b_A)$

Table 4.1.: Parameters for the dense optical flow calculations

Parameter	Description
Image A	First image
Image B	Second image
Result	Result image giving movement in x and y direction
levels	Number of image pyramid levels
pyr_scale	Scale between images of pyramid
winsize	Size of the window for image smoothing before calculations
iterations	Number of iterations of optical flow calculation per image pyramid
poly_n	Size of the area to which the polynomial is fitted
poly_sigma	Standard deviation used for smoothing derivatives during polynomial fitting
flags	Type of filter for image smoothing, use *Result* images as input for inital estimation of flow

sliding average filter. The type is controlled by the *flags* parameter, the window size used for smoothing can be set with *winsize*. The window size influences which types of features are taken into account. A large window is more robust to noise, decreases the calculation time and is able to detect fast motions. However, the resulting motion image is blurred, since the movement of several pixels are summarized in the window and described by one polynomial. A movement from different objects in different direction can not be determined accurately in case of larger windows.

poly_n and *poly_sigma* control how the polynomial is fitted to the image. According to the *OpenCV* documentation for *poly_n* =5, *poly_sigma* should be 1.1, for *poly_n* =7 *poly_sigma* =1.5 should be chosen. These represent values which have been found to work good. According to [69, p.592] *poly_sigma* should be a bit more than 20 % of *poly_n*. *poly_n* sets the area used for fitting a polynomial around one point, similar to the *winsize* argument used for smoothing. During fitting of the polynomial, derivatives of the image are calculated. *poly_sigma* sets the standard deviation, used to smooth the derivatives.

The *flags* parameter allows to set the type of filter used for image smoothing before the calculations. Usually, a box filter is used, however, it is possible to use a Gaussian filter, which gives better optical flow results, but increases the computation time. Furthermore, the *flags* parameters allow the algorithm to use the *Results* images as an input. In this case, the *Results* image contains the optical flow from a previous calculations. This flow is used as an estimation of the optical flow. The option is used in case consecutive images are analysed, in which it is assumed the current motion is similar to the one in the previous images.

4.3.3. HSV images

Hue, Saturation, Value (HSV) is a model to represent colours. In this model, the colour is represented by a colour value (hue), the saturation of the colour and the brightness (value). The colour (hue)

is encoded as an angle between 0° to 360° on a colour wheel. 0° is red, 120° is green and 240° is blue. The brightness is encoded as a percentage, where 1 is bright and 0 is dark. The saturation is encoded the same way, with 1 representing the pure colour.

This representation is useful to evaluate the optical flow. In this case, the result of the optical flow is encoded in a HSV image. The colour is representing the direction of motion since it is defined as an angle. The brightness represents the magnitude of the motion, saturation is not used. Therefore, magnitude of motion (brightness) and direction of motion (colour) can be seen as polar coordinates.

In order to get an HSV image, the result of the optical flow calculations needs to be converted. The optical flow calculation result is a two channel image, each channel gives the motion of the according pixel in *x*, respectively, *y* direction. Those values are interpreted as Cartesian coordinates, which are transformed to polar coordinates using the *cv::cartToPolar* function (see Figure 4.3). The result is two new images, one giving the direction of motion (angle), the other of the magnitude of motion. The angle can be directly used to construct the HSV image, the magnitude is scaled to values between 0 to 1.

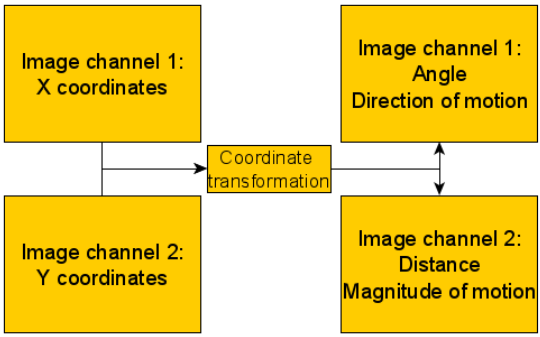


Figure 4.3.: Visualization of Cartesian to polar coordinate transformation

The constructed HSV images is, one the one hand, used for visualising the optical flow during the algorithm development. On the other hand, the detection algorithm evaluates the optical flow images based on the HSV representation.

4.3.4. Simple BLOB detector

A *blob* ([69, p.535]) in an image describes a region with a similar structure, e. g. a similar brightness or colour. A *blob* detector is used to detect such regions. Contrary to keypoints (see Section 4.3.2), *blobs* are less localised and cover a larger area. *Blobs* are used to determine regions, which could be of interested, e. g. for object recognition and tracking.

In the context of meteor detection, a *blob* detector can be used to determine regions in the optical flow images, which exceed a certain threshold. For example, in a matrix containing the magnitude of motion of the image scene, regions which move with a similar magnitude can be identified.

Different methods for *blob* detection exist, here the *cv::SimpleBlobDetector* ([69, p.534]) is used. The working principle is as follows: First, the input image is converted to a grey scale image. Then, a set of binary images is created by applying different thresholds to the image. In each binary image, connected components are detected using the *cv::findContours* function (see [69, p.407] for more details on contours). The components are called *blob* candidates. *Blob* candidates near each other in the same binary image and the adjacent binary images are grouped together. Each group of *blob* candidates are assigned a radius and center coordinates, which forms the final *blob*.

Different parameters can be applied to influence the *blob* detection. The construction of the binary images is influenced by setting the minimal and maximum threshold as well as the step size between thresholds. For an 8 bit image, the minimal threshold starts around 50 to 64, the maximum is set to around 220 to 235 and the

threshold steps are around 10 (taken from [69, p.536]). The parameter *minRepeatability* determines in how many consecutive binary images overlapping *blob* candidates must be found in order to be combined into a *blob*. In this context, *minDistBetweenBlobs* is used to define overlapping. The centre of two *blob* candidates must be below this distance to be considered belonging to the same *blob*.

After the *blob* detection is finished, the list of *blobs* can be filtered by different parameters (see Table 4.2). If the properties regarding area, inertia, circularity etc. of the *blobs* of interests are known, these filters can be applied (see [70]). If *filterByArea* is set, only *blobs* with an area equal or greater than *minArea* but smaller than *maxArea* are returned.

filterByCircularity filters by circularity, meaning how close the shape of the detected *blob* is to an circle. The circularity of a *blob* is defined as $\frac{4*\pi*blob\ area}{blob\ perimeter^2}$.

The inertia of a *blob* describes how elongated the *blob* shape is. A visualisation of circularity, convexity and inertia is shown in Figure 4.4.

Convexity is defined as $blob\ area/area\ blob\ convex\ hull$. A convex hull of a shape is the tightest convex shape that encloses the *blob*. A convex shape is a polygon in which all points of the polygon can be connected to each other without intersecting the polygon.

4.4. SpaceMEDAL: Meteor detection algorithm design

The challenges of a meteor detection from a satellite is to detect a faint and fast moving object from a moving platform. Thus, optical flow calculations are needed, in order to distinguish between the motion caused by satellite movements and meteors (see Section 4.2). Optical flow calculations are the core component of the algorithm called *Spaceborne MEteor Detection ALgorithm (SpaceMEDAL)* developed in the scope of this thesis.

Table 4.2.: Parameters for the Simple *blob* detector. For each filter, the minimal and maximum value of the according parameter can be set (e. g. minimal and maximal area). One exception is the colour filter, here, only the minimal value can be set.

Parameter	Description
minThreshold	Minimal Threshold for creating binary images
maxThreshold	Maximum Threshold for creating binary images
thresholdStep	Threshold step size between binary images
minRepeatability	Number of binary images in which a *blob* candidate must appear
minDistBetweenBlobs	Distance between *blob* candidates, candidates below this distances are merged
filterByArea	Filter *blob* by area
filterByCircularity	Filter *blob* by circularity
filterByInertia	Filter *blob* by inertia
filterByConvexity	Filter *blob* by convexity
filterByColor	Filter *blob* by colour
blobColor	Minimal value of *blob* brightness required

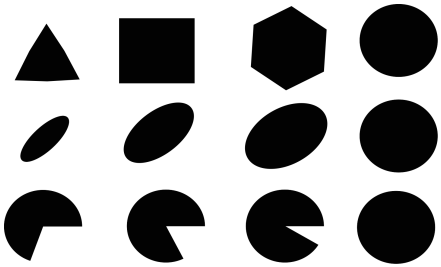


Figure 4.4.: Visualisation of Blob detection filter options[a]. The described parameter increases from left to right.

[a]Image drawn according to `https://learnopencv.com/blob-detection-using-opencv-python-c/`

In a bachelors thesis, under the guidance of the author, different optical flow algorithms were tested, in order to become familiar with the *OpenCV* framework and evaluate the suitability for meteor detection (see [71]). A result from this thesis is that algorithms based on sparse optical are not suitable, since meteors are not always detected as a keypoint and thus have no effect on the optical flow calculations. The algorithm must therefore be based on dense optical flow.

However, in this work no working algorithm was developed, it only gave an idea on how a meteor detection algorithm could work.

Thus, a software Testbed was developed, in order to develop the detection algorithm. Here, different ideas for algorithms and settings can be implemented in one software project. The parts needed for all algorithms, like reading images or remove noise, are only implemented once. This allows for fast testing of different ideas and approaches. Furthermore, the time needed to perform each part of any algorithms is measured to evaluate the algorithm speed. Also, the results of different processing steps can be visualized, e. g. the results of optical flow calculations or the original images with the detected meteors marked. This allows to debug the algorithm. Additionally, the data about detected meteors (position, frame number)

is exported. This is crucial to evaluate the algorithm in terms of the detection performance as well as the speed performance. Finally, all important parameters are set using a configuration files, this is useful to test different parameters of optical flow or blob detection without needing to recompile the program. All in all, the software Testbed is a crucial tool to develop the algorithm, by allowing to test algorithm prototypes and evaluating their performance.

After trying different approaches and thorough testing, two working concepts of the algorithm could be developed by the author, both based on dense optical flow calculations. The first concept evaluates the movement of each pixel individually and determines pixels whose movement magnitude and brightness is above a certain threshold. Those pixels are counted as meteors if pixels with a similar brightness and moving direction are detected in two other frames. The second concept applies a *blob* detector to the magnitude of motion part of the HSV image derived from the optical flow image. Potential meteor candidates are found by filtering the detected *blobs* and comparing their moving direction to the background movement direction. These candidates are confirmed by evaluating their brightness and the standard deviation of their moving direction. These working concepts of the algorithm were finalized in the scope of a master thesis, supervised by the author (see [72] and [73]), by setting up an automatic test environment. Thus, the prototypes and ideas previously developed in the software Testbed could be refined, debugged and a final algorithm could be chosen.

The final algorithm *SpaceMEDAL* combines the *blob* detection algorithm with some parts of the first algorithm (confirming a meteor in different frames). The HSV image is constructed from the dense optical flow calculations. It is used, since it is assumed that meteors usually have a different travel direction and a larger magnitude than the movement of the background caused by satellite motion. *This is the central assumption in the algorithm and the basis of meteor detection.* Thus, a meteor is detected by filtering the optical flow results to detect movements typical for a meteor. This is done by determining and removing the background motion

caused by satellite movement and then applying a *blob* detection to the remaining image parts in order to detect moving image regions typically caused by a meteor. Therefore, the simple *blob* detector is applied to the magnitude of motion part of the HSV images with the background motion removed.

Figure 4.5.: Flow chart of meteor detection algorithm *SpaceME-DAL*

In more detail, *SpaceMEDAL* (see Figure 4.5) works as follows: First, all variables and the export file for the results are initialized and the configuration file is read. The configuration file contains the parameters of the algorithm.

Second, the first image is read and some image processing is done, to reduce the noise. This takes place in the image class. Processing includes a median filter, applying the opening operator and applying a threshold. All values below the threshold are set to 0, this helps to remove faint parts of the background which in turn improves the optical flow calculations and results in a more precise background motion determination.

Next, the dense optical flow is calculated as described in Section 4.3.2. All parameters, like pyramid layers and window size can be changed via the configuration file.

Before constructing the HSV image, all displacements smaller than a settable threshold are set to 0. This is necessary, since a lot of pixels have a small displacement due to the uniform dark background. The small displacement causes an incorrect angle calculation. For the HSV image, the moving direction and magnitude of motion is derived from the displacement in x and y direction (see Section 4.3.3). Next, the mean and standard deviation of the moving direction and magnitude of motion is calculated. Regarding the magnitude of motion, only values above a threshold are taken into account to reduce noise. This also applies to the moving direction. Here, only pixels above a certain moving magnitude are considered to calculate the mean and standard deviation. However, the moving direction is given as an angle and thus circular statistic must be used to calculate the correct statistical values. In short, all angle values are converted to unit vectors and the average unit vector is calculated. Transforming the unit vector back to an angle gives the mean angle, the standard deviation is derived from the length of the average unit vector. Details can be found in [72, Section 6.4.3] and [74]. The results are the final HSV image, containing magnitude and

direction of motion, the mean magnitude $\mu_{magnitude}$ and its standard deviation $\sigma_{magnitude}$ as well as the mean direction $\mu_{direction}$ and its standard deviation $\sigma_{direction}$.

These results are processed in the detector class. As mentioned, the algorithm detects parts of the image not moving in the main direction and exceeding the background motion. Therefore, parts of the image belonging to the background are removed. The background is defined by pixels moving in the main direction with the main magnitude of movement, both are derived from the mean and standard deviation as outlined above. The main direction and magnitude are defined as:

$$main\ direction = [\mu_{direction} - \frac{3}{2} * \sigma_{direction}, \mu_{direction} - \frac{3}{2} * \sigma_{direction}] \quad \textbf{(4.1)}$$

$$main\ magnitude = [\mu_{magnitude} - \frac{3}{2} * \sigma_{magnitude}, \mu_{direction} - \frac{3}{2} * \sigma_{magnitude}] \quad \textbf{(4.2)}$$

Those values were experimentally found, excluding a large range of magnitude values increases the robustness of the algorithm. Again, mean and standard values are determined from all pixel values with a magnitude of motion greater than the set threshold.

All pixels moving with the background are set to 0 in the magnitude part of the HSV image, all values above are set to 255. Thus, a binary image is created before applying the simple *blob* detector to this part of the HSV image. The parameters described in Table 4.2 are used to filter the detected *blobs*. Due to different trials, the typical area, the convexity and the circularity of a *blob* in the magnitude images caused by a meteor are known. Thus, the returned *blobs* are already filtered by setting up the according parameters of the *blob* detector. This is used to remove *blobs* not representing a meteor.

The detected *blobs* are called keypoints and are further filtered, in order to determine meteor candidates. In order to be considered

as a meteor candidate, a *blob* must fulfil two conditions regarding its brightness. Thus, the mean and standard variation of the *blob* brightness in the original image taken by the camera (not the optical flow images) are calculated. The brightness must exceed a threshold, since it is assumed meteors are usually brighter than the background. This is done to avoid dark parts of an image being detected as a meteor. Furthermore, the standard deviation must exceed a threshold as well, since for a meteor the brightness variation inside the *blob* is usually high. This is due to the fact that the *blob* area is larger than the meteor and the background is usually dark. All meteors fulfilling the thresholds are stored as meteor candidates in a vector.

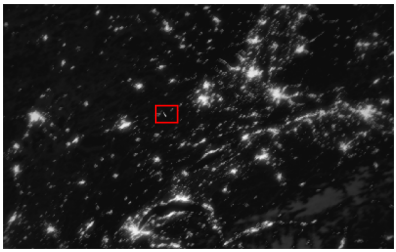
This vector of meteor candidates contains all candidates of the last 18 frames. It is used to confirm meteor candidates as meteors. Since it is assumed that the camera of *SOURCE* is running with 6 *fps* and a meteor last no longer than 3 s, the vector is able to contain all candidates belonging to one meteor. Since brighter meteors last longer, this must be adapted if longer meteors should be successfully detected.

The meteor confirmation takes place in the meteor class. Here, all candidates detected in the current frame are compared to the candidates of the previous frames. A meteor candidate is confirmed if it moves in the same direction as the candidate from another frame and the distance between both is below a threshold. In this case, both candidates are considered to belong to the same meteor. The distance threshold is necessary to avoid relating two candidates which move in the same direction, but are too far away from each other to be part of the same meteor. This distance threshold takes into account the number of frames between the candidates. The more frames between the candidates, the more time has passed and the higher the threshold. All candidates which are confirmed are stored in the confirmed meteor vector. This vector also contains the information, which candidate is related to which other candidate.

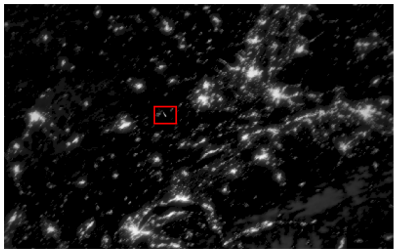
In a final step, the meteor candidates are verified. A meteor is considered to be detected if related candidates are found in at least

three different frames. In this case, the meteor is verified and stored in a vector for later export. Finally, all verified meteors are exported in a file, which contains the size, the brightness, the movement direction, the coordinates and the frame number of the meteor. This file is used for the evaluation of the detection algorithm. Later on the satellite, this file is used to determine which image (part) should be downlinked.

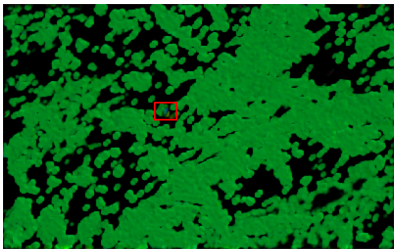
This completes the meteor detection algorithm for one frame. A visualisation of the processing steps is shown in Figure 4.6. Now, temporary variables are reset and the process is repeated with the next image.



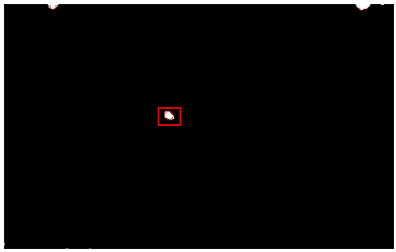
(a) Original frame



(b) Preprocessed frame



(c) Optical flow calculation results visualisation. Colour indicates moving direction, brightness magnitude of motion. In this case, green indicates the background motion direction.



(d) Detected *blobs* in magnitude of motion image with background motion compensated. Due to inaccuracies in the optical flow determination, some *blobs* are detected in the top of the image. Those are filtered out, since they are not consistent between several images.

Figure 4.6.: Visualisation of processing images with the detection algorithm. The meteor is marked with a red square in each image.

4.5. Methods for performance testing of SpaceMEDAL

During development it is necessary to test the algorithm, in order to evaluate the performance and to debug the software. The performance of the algorithm has multiple dimensions: The detection performance (e. g. number of false positive detections) as well as the speed performance of the algorithm. Furthermore, the complete detection chain (camera, algorithm and controller) needs to be tested and validated, once the final algorithm is developed. Depending on the development stage, different testing strategies with different purposes are deployed.

First, the algorithm is tested on the development machine, using existing data of meteors observed from space. In this step, different algorithms can be tested using the software Testbed (as described above). This gives a general idea of the challenges and which algorithms are worth to further investigate.

In a next step, data generated using the hardware Testbed (see Section 4.5.2) can be used. Here, the meteor data is imaged by the camera used for meteor observation. Thus, the algorithm uses realistic data, since camera properties and settings (like dark noise and frame rate) influence how the meteor is imaged. This allows to further improve the algorithms.

However, in order to evaluate the algorithms detection performance and test systematically, more data is required. Therefore, the artificial meteor simulation (see Section 4.5.1) is used, to generated a systematic test set of videos. In this simulation a wide range of videos showing meteors as seen from a satellite with varying backgrounds, velocities and brightness as well as other parameters are generated. Thus, first performance tests can be done, e. g. evaluating the detection rates for different types of meteors and backgrounds. Furthermore, the algorithm can be improved if certain types of meteors are not well detected. Also, the effect of

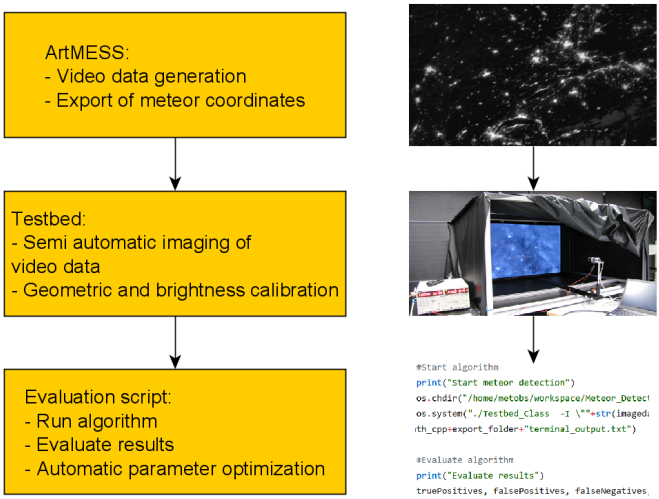


Figure 4.7.: Flow chart algorithm test concept: The videos are generated using *Artificial Meteorvideo Simulation Software (ArtMESS)*, imaged in the Testbed and finally evaluated.

different parameters on the algorithm performance can be evaluated. It is crucial to understand how different parameters influence the algorithm. Then, the algorithm can be fine tuned in orbit after the first images have been downlinked or the algorithm can be tuned to a specific type of meteors which should be detected. This concept is shown in Figure 4.7.

The final test (results are given in Section 4.7) is done with the optimized algorithm. It includes two steps: First, the detection performance using a test set generated to include all expected meteor types and backgrounds. Again, this test set is imaged by the camera in the Testbed. This allows to evaluate the detection rate in terms of false positive and negative as well as correct detections. Second, the algorithm is ported to the Payload On-Board Computer (PLOC), to measure how many images can be processed in a given

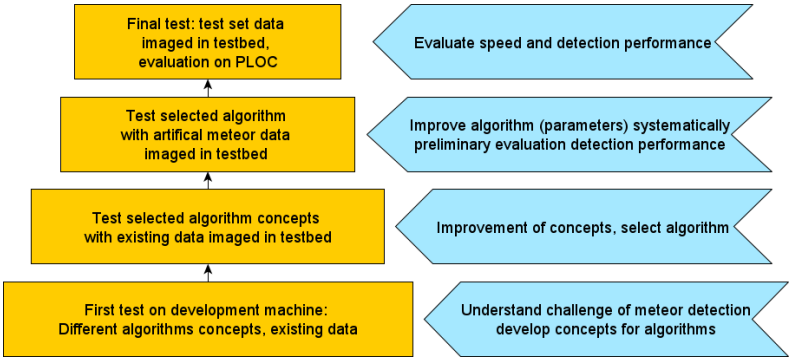


Figure 4.8.: Visualisation of the algorithm test concept

time frame. This is important for mission planning to estimate how much processing time is needed for a given observation time. All steps of the algorithm test concept are shown in Figure 4.8.

4.5.1. Meteor video generation for testing

Testing the algorithm requires data, in this case videos showing a meteor from the perspective of a satellite. However, no sufficient amount of data exists to develop such an algorithm. The only data available is taken from the ISS by the METEOR project from the PERC (see [75]). With this data it is only feasible to develop a prototype, since camera properties and settings (especially the frame rate) influence the representation of a meteor in the image and therefore the design of the detection algorithm. Furthermore, the videos available have a low resolution. In order to develop and test the algorithm it is necessary to generate realistic image data.

Therefore, in the scope of the *SOURCE* project a program to generate artificial meteor observation data was developed[1]. The program is called *ArtMESS* (see [76]). It is a Python based script, which

[1] Main developers who supported the author: Niklas Geier, Marcel Liegibel and Philipp Hoffmann

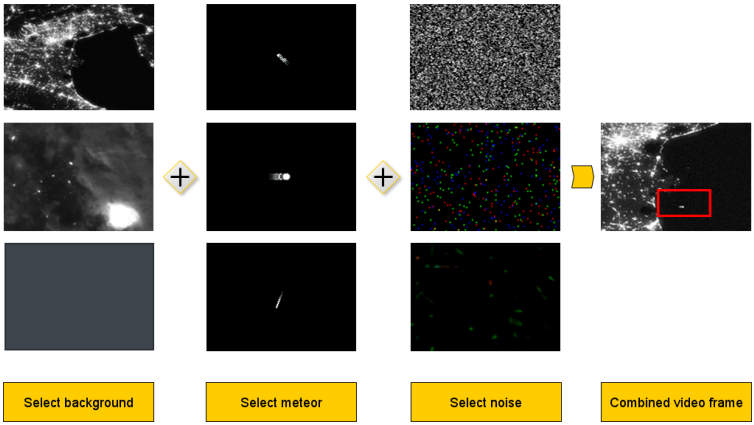


Figure 4.9.: Illustration of *ArtMESS* working principle: The selected background is combined with the selected meteor and noise into the final frame.

generates videos of a meteor as seen from a space based camera. Therefore, images containing a meteor are generated depending on various settings, which are combined with different available backgrounds (see Figure 4.9). This is done by overlaying the background image with the meteor image.

Meteor image generation The first step is to generate a set of consecutive images containing a meteor moving from a defined starting point over the set of images. For the image generation, different parameters of the meteor are considered. The most important ones for the detection algorithm are the velocity, the brightness and the duration of the meteor. Other settable meteor parameters include the meteor start position in the frame, the moving direction as well as the starting frame in the final video. The first step for each image is to generate the meteor mask image, which is later overlaid with the background image. The meteor mask image has the same size

as the background video and is an 8 bit gray scale image. All pixel values are initialized as black (pixel value 0).

The representation of the meteor in one image is generated by setting selected pixels to a specific brightness. The pixels are selected according to different shape masks (see Figure A.35). The pixel brightness is defined by the luminosity function, which gives the current brightness l_{curr} of the meteor depending on the duration of the meteor (z_{total}) and its maximum brightness (l_{max}). Currently, the light curve implementation is an inverted parable, which is orientated on actual meteor light curves measured (see [77]). It is implemented as shown in Equation 4.3, where z is the current time step.

$$l_{curr} = l_{max} - \frac{4}{z_{total}^2 * l_{max}} * \left(\frac{z - z_{total}}{2}\right)^2 \qquad \textbf{(4.3)}$$

Finally, the brightness of the meteor core, the bright area around the core and its tail is varied randomly for a more realistic representation.

The position of the meteor in the mask images depends on the start position, the moving direction and the velocity of the meteor. All steps described above are repeated until all meteor images are generated. The number of images is defined by the duration of the meteor.

Background image generation The background of the meteor video is crucial, since it must represent a realistic view of the Earth at night. In this program, the background is a video file from which the frames are extracted.

The first option is to use no background. In this case, only the meteor mask images are combined into a video. This video shows only black parts, except the meteor and its tail. Thus, these videos can be used to test algorithm concepts and to determine the effect of algorithm parameters on the detection performance. It can be evaluated, which types of meteors can be detected.

The second option is to use an existing video taken from the ISS by the METEOR project from PERC (see [75]) which already contains a meteor. In this case, an additional meteor is implemented.

This option offers a realistic view of the Earth including different lightning conditions, clouds and thunderstorms. A wide range of different background is crucial to test the detection algorithm thoroughly. However, the video quality (resolution and noise) is limited. Furthermore, no rotation of the video occurs, since the ISS is able to hold its attitude accurately. This may not be the case for small *CubeSats* with limited ACS systems.

Therefore, the third option is to generate the background video as well. An additional Python script allows to generate videos from the NASA black marble[2] images. Those images are taken from a satellite showing the Earth at night and are publicly available[3]. The images used are taken from the Suomi National Polar-orbiting Partnership (SNPP) satellite using the Visible Infrared Imaging Radiometer Suite (VIIRS) instrument. More details about this instrument can be found in [78]. Since the images are available in high resolution, it is possible to extract a region of the images and use this as a background. By moving a selection window over the image, a video can be generated. This process is called sliding window. *ArtMESS* allows to define the image to be used, the moving direction of the window as well as the window size. Furthermore, the amount of pixels the window is moved between each frames can be set. This defines how fast the background moves.

Additionally, this methods allows to take into account the rotation of the satellite, which is done by rotating the sliding window, after the meteor was implemented. This is necessary, since the original moving direction of the meteor relative to the background should be kept. This process is also explained in more detail in Appendix Section A.13.1.

Combination of meteor and background image In the final step, the meteor and the background images are combined. In principle, this is done by adding the meteor mask image to the selected

[2] `https://blackmarble.gsfc.nasa.gov/`
[3] `https://earthobservatory.nasa.gov/features/NightLights`

background frame. However, in order to get a realistic movement between meteor and background, the optical flow of the background must be calculated first. The resulting movement is used to correct the meteor movement, by adapting the meteor coordinates. This is needed, since the meteor movement must fit to the background movement. Otherwise, the meteor would seem to move sideways, instead of straight.

After correcting the meteor movement, different noise types can optionally be added to the background. This includes Gaussian noise, speckle noise, salt and pepper noise, poisson noise as well as hotpixels. For this purpose, the function *random_noise* from the *scikit* library[4] is used.

Finally, the meteor mask image is simply added to the background image. Since all parts in the meteor mask image except the meteor are dark (pixel value is zero), only the meteor becomes visible in the background images. All pixel values of the background and the meteor image are simply added, which is possible, since both have the same dimension.

The frame in which the meteor starts to appear can be set via a parameter. Furthermore, the apparent velocity of the meteor can be controlled. This is done by adding the meteor mask only every X number of background frames, where X depends on the required velocity. The larger X, the higher is the apparent velocity of the meteor, since its position changes more compared to the background.

Finally, all images generated are combined into a video file and the coordinates and the corresponding frame number of each meteor are exported into a file.

Test set generation The simulation *ArtMESS* also includes an option to generate multiple videos at once with different backgrounds and meteors. The generated test set of videos is used to evaluate the algorithm systematically, by setting different meteor

[4] see `https://scikit-image.org/docs/dev/api/skimage.util.html#skimage.util.random_noise`

Table 4.3.: Parameters for the test set generation

Parameter	Description
Meteor speed	Number of background frames between two meteor mask images
Meteor brightness	Maximum pixel value of meteor head, given in percent of 255
Meteor duration	Number of meteor mask frames
Meteor angle	Moving direction of meteor in degree

properties like brightness, velocity, duration and moving direction (see Table 4.3). This is needed to avoid optimizing the algorithm for one specific meteor property. Also, this approach helps to understand which types of meteors can be detected with the current detection algorithm. Furthermore, different parameters of the algorithm can be varied in order to evaluate how these parameters affect the detection of different meteor types.

Each test set can be generated with different backgrounds. Available options include any of the videos taken by the METEOR project from the ISS, any video generated by using the blackmarble images or no background.

Video upscaling As outlined in the description of the Testbed (see Section 4.5.2), the main component of the Testbed is a high resolution screen. The videos must be displayed in full screen to make sure the complete sensor of the camera is imaging the video. However, the generated videos have a lower resolution and would be scaled by the screen and the video player software if displayed in full screen. Therefore, the videos should have the same resolution as the screen.

This is achieved by upscaling the videos, after the generation is finished. The open source tool *Video2x*[5], delivers realistic results.

[5] `https://github.com/k4yt3x/video2x`

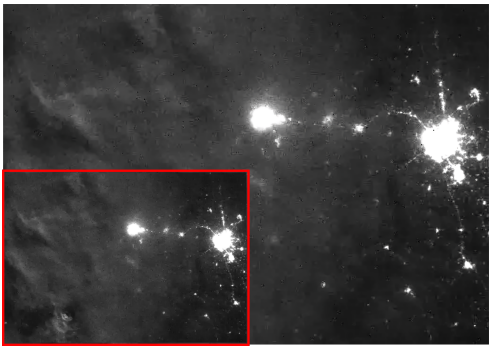


Figure 4.10.: Comparison of original and upscaled images. The original image is shown with a red border[a].

[a] Frames extracted from `https://www.youtube.com/watch?v=j5gdLovzp0Q`

For this tool a Python wrapper was developed, which automatically upscales all videos from one generated test set to the screen resolution of the Testbed. An example of an upscaled video from the METEOR project is shown in Figure 4.10.

4.5.2. Testbed development and calibration

The Testbed is used for hardware in the loop tests of the meteor detection chain, which consists of the camera, the detection algorithm and the processing unit on which the algorithm runs (usually called the PLOC). The idea is to display the generated videos on a screen, which is imaged by the camera. The images taken by the camera are either processed with the meteor detection algorithm in real time or stored for later processing. This allows to test the camera and optimize the algorithm by using realistic data, since the images are acquired with the same camera and settings as planned for the orbital observations.

Pre testing with a standard computer screen showed that those screens are not usable for this applications. Due to the backlight used in the screen, dark parts of the video still produce some light

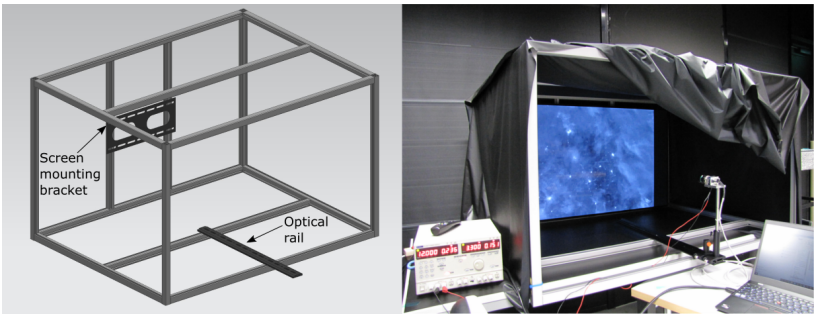


Figure 4.11.: Mechanical design of the Testbed: CAD drawing (left) and final setup with Blackout Fabric (right).

(high black level). Therefore, an OLED display is needed due to the superior black level. Those displays work without a backlight, instead LEDs for each pixel emit the light.

In the following the design and the calibration (brightness and geometric) of the Testbed is given.

Testbed design

The Testbed (see Figure 4.11) consists of the OLED TV screen *LG OLED 55B8LLA*[6] with a diagonal of 55 *inch*. The frame consists of aluminium profiles, which allow a flexible and low cost construction. On the profiles the screen mount is attached. The optical rail is mounted on the profiles as well, the position of the rail can be adapted back and forth as well as left and right. The complete Testbed is covered with Nylon Blackout Fabric to avoid stray light entering the camera. At the front, the fabric can be moved to allow access to the camera and screen.

The Testbed is designed to fulfil two requirements:

First, the size of the screen must be large enough, that the complete FOV of the camera images the screen at a useful distance.

[6] see `https://gscs-b2c.lge.com/downloadFile?fileId=bncFyVuYmqmuY7bo3kBQ`

A useful distance is defined as large enough, that the camera still can be focused (minimally 614 mm for the *Schneider Kreuznach Cinegon 1.4/12*), but small enough to keep the setup at reasonable dimensions. With the given screen size and the FOV of the *Schneider Kreuznach Cinegon 1.4/12* lens, the maximum useable distance is about 1.2 m. In the actual setup a lower distance of 1 m is used to keep the setup smaller.

Second, at the chosen camera distance, the screen resolution should be large enough. This means, one pixel of the camera image should image more than one screen pixel. Assuming a *4K* resolution of 3840 *px* $\times$ 2160 *px* for the screen and that the complete screen is imaged by the camera and, the number of screen pixels per camera pixels is given by $Screen_{res,x}/Sensor_{res,x}$. This results in 1.98 screen pixels per camera pixel. Since a smaller distance is used, the resolution is reduced to 1.46 screen pixels per camera pixel.

Testbed calibration

The Testbed calibration includes two important steps: First, the brightness of the screen must be calibrated in order to image the videos properly. This calibration includes measuring the absolute brightness and measuring the brightness received by the camera.

Second, is the geometric calibration. Here, the pixels of the screen are mapped to the pixels of the camera. This allows to determine if a meteor was detected correctly, since the position of the meteor in the video as well as in the camera image is known. Therefore, an automated test evaluation is possible.

Brightness calibration In the scope of the *SOURCE* project, the Testbed brightness calibration was conducted by the *SOURCE* Payload team[7], see [79].

Two main aspects should be achieved with the calibration: First, the absolute brightness of the screen should be measured. Second,

[7] Besides the author mainly from Philipp Hoffmann

a relation between the screen pixel value and the camera pixel value should be determined. This aspect is important for the video generation, since it allows to control the camera pixel value by setting the right pixel value on the screen.

The calibration was conducted by displaying different grey scale images with increasing pixel values on the screen. The resulting brightness was measured using an optometer or the camera imaging the screen. The image brightness and the measured brightness are plotted against the screen pixel value. For the calibration, different screen and aperture settings were tested, in order to find ideal settings.

First, the irradiance at the camera position was measured with the optometer. The conversion between screen pixel value and irradiance at the camera position is shown in Figure 4.12. The conversion of the measured data and displayed pixel value can be described with a 3th degree polynomial:

$$Irradiance(Screen\ pix.\ value\ x) = 6.76 \times 10^{-9}x^3 + 1.02 \times 10^{-5}x^2 + 6.10 \times 10^{-5}x + 0.013 \quad \textbf{(4.4)}$$

This equation is used to converse between screen pixel value and irradiance at the camera position.

However, two issues arise when analysing the measured optometer data: First, the brightness of the screen is orders of magnitude higher than the brightness of a meteor. A meteor of magnitude 0 has a brightness of about $1196 \times 10^{-9}\,\mu W/cm^2$ (see Table 3.2 on page 50), while the screen has a brightness in the order of $1\,\mu W/cm^2$. Therefore, the aperture of the lens must be further closed than in orbit to avoid saturation of the image. Thus, all other settings of the camera (mainly binning, exposure time and frame rate) can be kept the same. It is important to keep those camera parameters the same as later in orbit for realistic algorithm testing.

Second, the brightness does not increase linear with the pixel value. As a consequence, the camera signal is also not linear. This is

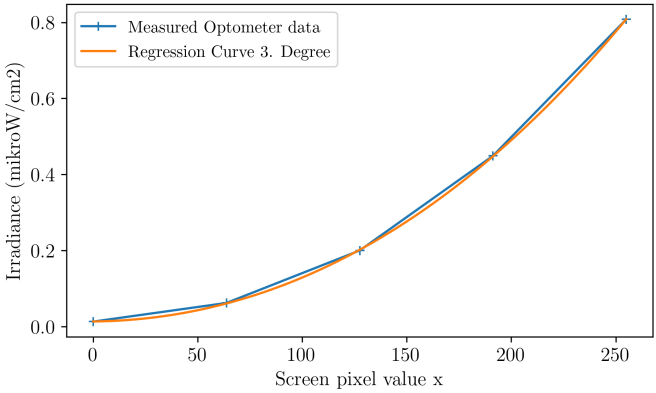


Figure 4.12.: Absolute brightness calibration Testbed, conversion of screen pixel values to irradiance output at camera position. Measured with screen brightness 70 and contrast 55.

a problem when imaging meteor videos, since the values in the video are imaged differently with the camera, which effects the algorithm. Furthermore, with the existing screen settings and available camera apertures, it is not possible to image the whole range of screen pixel values. Therefore, a relative brightness calibration is needed.

The idea of the relative brightness calibration is as follows: First, an aperture setting should be found, which allows to image the widest range of input brightness without over- or underexposing the image. As mentioned, it is not possible to image the complete input range with one aperture setting, however the best fitting setting should be determined. In a next step, the relation between screen and camera pixel value should be determined, in order to control the pixel value of the camera pixel.

The ideal aperture setting is determined by displaying and imaging different grey scale image on the screen using different aperture settings. The measured pixel values by the camera can be plotted against the input irradiance of the screen, changed by displaying

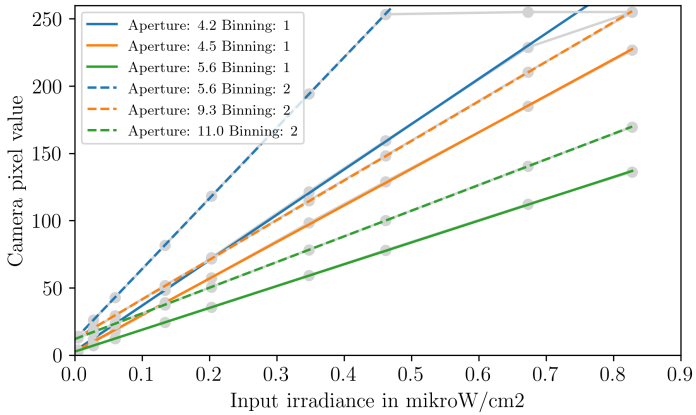


Figure 4.13.: Irradiance calibration Testbed. Here the measured pixel values of the camera image (determined in the centre of the image) are plotted against the screen irradiance values as determined with the absolute brightness calibration. The blue lines indicate over exposure, green lines under exposure while the orange line represent the ideal aperture setting.

the grey scale images (see Figure 4.13). The aperture setting with the widest range of pixel values without under- or overexposure is chosen. For binning 1 this is 4.5, for binning 2 aperture 9.3 is used.

The conversion between irradiance and camera pixel value is described by a simple linear equation. Depending on the binning setting and aperture, the equations are as follows:

$$Camera\ Pixel\ Value(Binning\ 1, Apertue\ 4.5) = 271.65 * Irradiance + 2.5 \quad \textbf{(4.5)}$$

$$Camera\ Pixel\ Value(Binning\ 2, Apertue\ 9.3) = 294.32 * Irradiance + 11.91 \quad \textbf{(4.6)}$$

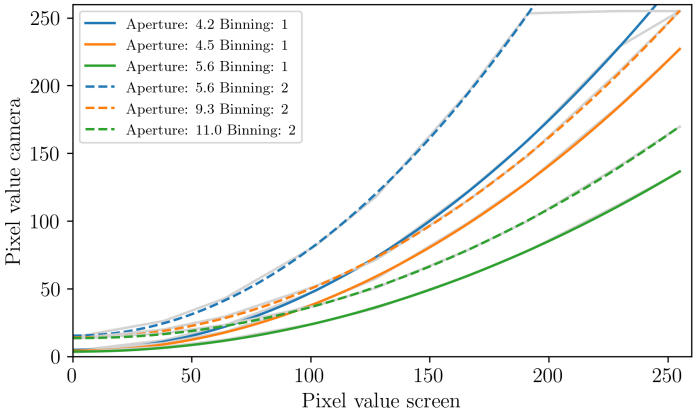


Figure 4.14.: Brightness calibration Testbed. Binning 1 is shown in solid lines, binning 2 in dashed lines. Blue lines indicate the aperture is too open, green lies indicate too closed. The orange line indicates the chosen aperture setting, for which almost the whole range of input values can be mapped to the complete range of camera output values.

In a final step, the relation between the camera and screen pixel value can be determined, by combining the optometer measurements (see Figure 4.12) with the irradiance calibration (see Figure 4.13). The result is shown in Figure 4.14.

As can be seen, a curve can be fitted to the result plot. This gives a function, which describes the relation between screen pixel value and resulting camera image value. The function is a second degree polynomial in the form of $A * X^2 + B * X + C = Y$, with X being the screen pixel value and Y being the target camera pixel value.

The function is fundamental for the relative brightness calibration. It is used to control the camera pixel value, by setting the screen pixel value. The pixel values of the video are adapted to ensure that the pixel value in the camera image is exactly the pixel value in

the original video. In the original video, the meteor brightness is set to a pixel value as predicted with the *Sensitivity* simulation (see Section 3.3) and thus related to a meteor magnitude. Using this equation ensures that the meteor brightness in the camera image is as predicted with the simulation. This function is implemented in the artificial meteor simulation, which adapts the resulting video pixel values. The tables shown in Appendix Section A.8 can be used to convert the apparent magnitude into the expected camera pixel value. Equation 4.7 shows the converted equation, which is used to convert the video pixels values. In this case, the desired target pixel value in the camera is the original pixel value of the video.

$$PixelValueVideo_{new} = \frac{-B + \sqrt{B^2 - 4 * A * (C - TargetPixelValueCamera)}}{2 * A} \quad \textbf{(4.7)}$$

The used coefficients differ, depending on the aperture and binning setting. The coefficients are given in Table A.13.

The conversion works, as can be seen in Figure 4.15, which shows a comparison between the original video frame, the adapted video frame and the image taken by the camera in the Testbed. The image taken by the camera is close to the original video frame, which is the goal of the adaption. Furthermore, the conversion was tested by comparing the expected camera pixel with the measured camera pixel value. The mean difference is 7.8 pixel values for binning 2 and 6 for binning 1 (see Figure 4.16). Since this error is low, the conversion is working and can be used for further tests.

Geometric The geometric calibration is used to relate the camera image pixels to the screen pixels. This is achieved by showing a test chart on the screen. The test chart is a chess pattern, in which each tile is numbered (see Figure A.36). The coordinates of each tile in both images can be related, thus, if the tile in which the meteor appears in the video is known, the according tile in the meteor image

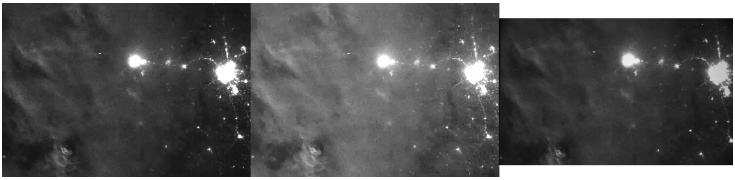


Figure 4.15.: Comparison of original[a] (left), converted (middle) and imaged video frame (right) for binning 2 using aperture 9.3

[a]Extracted from `https://www.youtube.com/watch?v=j5gdLovzp0Q`

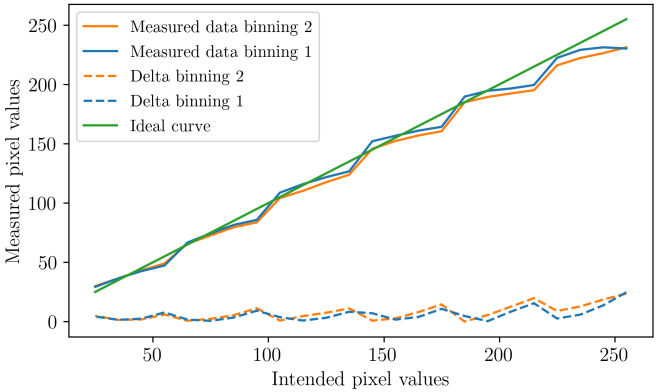


Figure 4.16.: Evaluation of brightness conversion for two binning settings

can be determined. Therefore, it can be checked if the meteor was correctly detected. Furthermore, the chess board pattern is used to properly align the camera before each test.

4.5.3. Algorithm test campaign

After setting up and calibrating the Testbed as well as setting up the meteor simulation, the algorithm test campaign can be started. The test campaign is done with a promising algorithm candidate, after testing different approaches using existing data (see Figure 4.8). The campaign took place in the scope of a master thesis (see [72]). Two goals should be achieved with the test campaign: First, the algorithm should be improved. This includes evaluating the effect of different parameters, selecting proper parameters as well as improving the algorithm by identifying deficiencies. Second, the detection and the speed performance should be evaluated.

The detection performance is expressed with three metrics (see [72]): Precision, recall and the F-measure value. These values are calculated with the True positive (TP), False positive (FP) and False negative (FN) results of the test. TP are the number of correct detections of an existing meteor, FP are the number of erroneously detected meteor and FN are the number of not detected existing meteors. The precision value P is calculated with:

$$P = \frac{TP}{TP + FP} = \frac{TP}{all\ detections} \tag{4.8}$$

Thus, the precision value states how many of the total detections are actually meteors and describes the algorithm's ability to detect only relevant objects. The recall value R is the fraction of meteors detected from all actual meteors. Therefore, the recall value states how well the algorithm is able to detected meteors. It is calculated with:

$$R = \frac{TP}{TP + FN} = \frac{TP}{all\ actual\ meteors} \tag{4.9}$$

The ideal algorithm as a high precision value to ensure only meteors are detected (reduce number of false positives) and a high recall

value to ensure all meteors are detected (reduce number of false negatives). The $F-measure$ is the harmonic mean of both values to evaluate the algorithm detection performance in a single value.

$$F-measure=\frac{2*P*R}{P+R} \tag{4.10}$$

In the scope of the meteor detection algorithm testing, the detection performance is evaluated based on the results of at least one test set. Thus, a TP is defined as a meteor correctly detected in a video of a test set in at least one frame. Consequently, a FN is a meteor not detected in a video and a FP is any wrong detection of a meteor. For a test set consisting of 10 videos, the maximum number of TP is 10, since each video contains exactly one meteor.

The speed performance is first measured on the development machine and later on the PLOC used for the *SOURCE* mission using Field Programmable Gate Array (FPGA) acceleration.

In this section, the test concept as well as the used test sets are elaborated. The results of the test are given in the following section.

Test set generation

In order to achieve the campaign goals, different test sets are generated with the meteor simulation *ArtMESS* as shown in Table 4.4. The main parameters influencing the detectability of a meteor in a test set are the background type, the meteor angle, the brightness and the velocity. Each test set serves a different purpose, thus different types of meteors and backgrounds are implemented. Four categories of meteor types can be implemented in a test set, which are listed in the following:

1. Slow and bright
2. Slow and dark
3. Fast and bright
4. Fast and dark

It is assumed that the difficulty to detect each meteor increases with the category. Depending on the purpose of the test set, one or

more categories are chosen for a test set. For each test set the same background is used and a meteor property is varied. Thus, only one parameter is of the test set is varied and its effect can be evaluated.

The purpose of each test set is as follows: The black background videos (test set V1) are used to determine which meteors could be detected in general. Since no noise or distortions from the background are present, the algorithm can be tested and issues identified. Issues could be problems with the detection of faint meteors or meteors moving in a specific direction. Thus, first the moving direction is varied using easily detectable bright and slow meteors. Next, the brightness is varied, to determine to which brightness a meteor still can be detected. The meteors are still slow in order to only evaluate the effect of brightness. The next varied parameter is velocity, used to adapt the optical flow calculations. Finally, both parameters are varied, to ensure all combinations are tested. This test set is also used to evaluated the effect of different algorithm parameters. Parameters include settings of the optical flow (see Table 4.1), the blob algorithm (see Table 4.2) as well as different threshold values. A complete list of available parameters can be found in Table A.25.

The test sets with backgrounds are used to determine how well the algorithm performs with noise and distortions present. The black marble backgrounds provide only city lights and are more static. They are used first, in order to evaluate the effect of a moving background on the detection rate. Since this test set is the first one with a background, easily detectable meteors are implemented. Due to the background, some parameters are adapted to reduce the number of false detection.

The ISS videos are less static, they also contain lightning and clouds and therefore represent the biggest challenge for the algorithm. Here, the final detection performance is evaluated using two videos taken by the METEOR project from the ISS: The Italy background video shows city lights and some noise, the West India background is the most challenging one providing a view of clouds and

Table 4.4.: Description of test sets used for algorithm test campaign

Test set name	Meteor types	Background	Videos	Varied parameter
Set V1	Slow and bright	All black	17	Angle
	Slow and dark		9	Brightness
	Fast and bright		13	Velocity
	Fast and dark		14	Brightness and velocity
Set V2	Slow and bright	Black marble V1	16	Angle
		Black marble V2	16	
		Black marble V3	16	
		Black marble V4	16	
Set V3	Slow and bright	ISS Italy	16	Angle
	Slow and dark		9	Brightness
	Fast and bright		13	Velocity
	Fast and dark		13	Brightness and velocity
	Slow and bright	ISS West India	14	Angle
Set V4	Slow and bright	Black Marble	8	Rotation rate

lightning[8]. For this test, the same meteors as used for test set V1 are implemented to ensure a wide variety of meteor properties.

The last test set V4 is used to evaluate the effect of a satellite rotation in case the attitude control system does not work accurately. Thus, a rotation rate between 0.2 °/s to 1 °/s is implemented in the videos.

For each test set the brightness conversion for binning 2 is applied and the videos are imaged in the Testbed with binning 2, aperture 9.3, exposure time 166 365 µs and frame rate 6. These are the settings intended to be used on the *SOURCE* mission (see Section 5.2). Currently, the algorithm is developed for this mission, since it is the first to fly. However, the test environment can be easily adapted for different camera settings. The only setting different from the flight setting is the aperture, which must be closed due to the high screen brightness. This does not effect the algorithm testing.

Evaluation method for test set data

Finally, each test set is imaged in the Testbed and processed with the algorithm. For the evaluation a Python script is used, which was developed in the scope of a master thesis (see [72]). This script allows to automatically set the parameters for the algorithm, to run the algorithm on the selected data and to evaluate the results by calculating the recall and precision values. In order to do so, the script first reads the algorithm parameters defined by the user and writes them into the configuration file for the algorithm. After the algorithm is called the results are evaluated, by determining which meteors were correctly detected.

Therefore, the script needs to know in which camera frames and at which position a meteor should be detected. The meteor positions in the original video displayed on the Testbed are given by the output file from the meteor generation. This file contains the information about frame number and position of the implemented meteor. Since

[8] West India video: `https://www.youtube.com/watch?v=j5gdLovzp0Q`, Italy: `https://www.youtube.com/watch?v=2DVsYWhvf0U`

image acquisition and video are always started at the same time when imaging the videos in the Testbed, a relation between video frame number and camera image frame number can be derived. Therefore, the frame rate of the video and the camera are required (see Equation 4.11).

$$MeteorFrame_{camera} = ceil\left(\frac{MeteorFrame_{video}}{FrameRate_{video}} * FrameRate_{camera}\right) \quad \textbf{(4.11)}$$

A frame margin of 4 is used to account for inaccuracy in starting the video and imaging. Thus, a meteor counts as correctly detected if the meteor is detected in the expected camera frame or in the 4 frames before or after the expected frame.

Furthermore, the script needs the coordinates of the meteors in the camera frame. These are derived from the coordinates of the meteor in the original video, which are converted to coordinates on the Testbed screen. This is necessary, since the resolution of the screen and video are not exactly the same due to different aspect ratios. Finally, the geometric calibration gives the relation between screen and camera image pixel location. For each meteor a bounding box is defined from the upper left and lower right meteor coordinate. This accelerates the later evaluation, since it only has to be checked whether the detected meteor is inside this box. The bounding box is also necessary since the camera frame rate is lower than the video frame rate and thus the meteor appears in the camera image on more pixels compared to the original video. A small pixel margin of 30 *px* is applied to each side of the meteor bounding box to account for inaccuracies in the geometric calibration. Thus, for each camera frame a meteor bounding box is calculated which defines the area in which the meteor should be detected.

Additionally, the script also needs the detected meteors from the algorithm, which is provided by the output file of the algorithm. This file contains informations about the detected meteor (frame and position).

Using this method, the script can automatically determine which meteors are detected and how many false positive detections occurred. This allows to test different videos and algorithm parameters fast, since manual evaluations are time consuming. The script is used first to evaluate the effect of different parameters on the detection performance and to identify the parameters that have the largest effect. Furthermore, suitable values for some parameters are identified, this includes for example settings of the optical flow algorithm. Those settings are kept constant after identifying suitable values. Other parameters, like thresholds for blob size and moving direction, effect the detection performance and depend on meteor properties.

The results of the test campaign are given in the following sections.

Summary of testing approach

The testing approach consist of the following steps:

1. Test set generation using artificial meteor simulation *Art-MESS*
2. Imaging of test set in Testbed
3. Automatic evaluation of algorithm performance

In the first step, different test sets are generated, with different backgrounds and meteor properties depending on the test objective. Since the generation requires only some input values from the user (meteor properties, background), the test set generation is fast and low effort is needed to generate the data. Furthermore, the upscaling is automated by using a Python script.

The second step, imaging the test set data, is supported by a Python script as well. This script automatically generates a folder for each test set video and controls the camera according to settings specified by the user. In doing so, a test set is imaged fast in the Testbed.

The third step, the evaluation, is automated as well, with the Python script described above. Therefore, the data can be easily evaluated and different algorithm parameters can be tested systematically.

All in all, the algorithm testing approach is automated as much as possible. Thus, the pipeline allows to:

1. Generate multiple test sets with varying meteor parameters and backgrounds
2. Image the test data with low effort
3. Evaluate the data with different algorithm parameters

This allows for fast testing and development of the algorithm.

4.6. Algorithm parameters and their effects

The algorithm, as described in Section 4.4, has three main parts: The optical flow calculation, the meteor candidate detection based on a *blob* detector and the verification of the meteor. Each part can be influenced with different parameters, settable via the configuration file. Resulting from the above described testing campaign (see also [72]) the parameters are set.

For some parameters only an ideal setting has to be found. This ideal setting works for all cases and thus changing it only worsens the detection performance. An example would be the parameters for the optical flow calculation. Other parameters, like the threshold for the meteor brightness to detect a meteor candidate, influence which types of meteors can be detected. Those parameters must be optimized to increase the overall detection performance of the algorithm.

The parameters for each main class (Image, Detection and Meteor class) are explained in the following. The parameters are divided into six different categories, depending on the algorithm step to which they are applied (see Table 4.5).

Table 4.5.: Parameters for meteor detection algorithm

Category	Parameter
Preprocessing	`thresNoise`, `ResizeFactor`
Optical flow	`lvl`, `scl`, `win_size`, `iterations`, `poly_n`, `poly_sigma`
Background motion	`magThresholdAngleMatMean`, `minDisplacementHSV`
Blob detection	`filterByArea`, `minArea`, `maxArea`, `filterByCircularity`, `minCircularity`, `maxCircularity`, `filterByInertia`, `maxInertiaRatio`, `minInertiaRatio`, `minThresholdBLOB`, `maxThresholdBLOB`, `thresholdStep`
Meteor candidate confirmation	`minStdBrightnessThresh`, `brightnessThresh`
Meteor confirmation and verification	`angleDev`, `maxBlobDistance`, `framesToConfirm`, `storageTime`, `thres_number_keypoints`

Preprocessing

Preprocessing takes place in the image class. Here, noise is removed in order to prepare the image for the following optical flow.

`thresNoise` First, a threshold is applied to the image, all pixel values below `thresNoise` are set to 0. This removes noise and improves the detection of *blobs*. During testing a value 20 allowed to detect also faint meteors.

`ResizeFactor` The image can be resized in order to reduce processing power required. This option is not used yet.

Optical flow

Choosing the right parameters for the optical flow is crucial, since these calculations are the core component of the algorithm. The effect of the parameters are already described in Section 4.3.2. The parameters must be chosen in a way, that the small movements (meteor does not cover many pixels) caused by meteors are detected. Optical flow calculations are also part of the image class.

`scl` and `win_size` The ability to detect small objects moving fast is mainly influenced by the scale of the image (`scl`) as well as the windows size (`win_size`). A small `scl` is required, since the meteor displacement is usually large due to the fast movement. Since the meteor covers usually only a few pixels, the windows size must be small. Values of $\texttt{win\_size} = 7$ and $\texttt{scl} = 0.25$ resulted in accurate optical flow calculations. These values are kept constant, but could be adapted in orbit.

`iterations` and `lvl` The number of images pyramids (`lvl`) and iterations of the calculation at each level (`iterations`) influences the accuracy of the optical flow calculations as well as the processing time. Since only the meteor movement must be detected, accurate

velocity determination is not required. Furthermore, the processing power is limited. Thus, both values are set to 1.

`poly_n` and `poly_sigma` Those values are set according to the documentation of the optical flow function to `poly_n` $= 7$ and `poly_sigma` $= 1.5$.

Background motion detection

Before detecting *blobs* in the magnitude of motion image, this image is processed: All parts of the image considered to be background should not be taken into account when detecting *blobs*. This is achieved by calculating the main direction and magnitude of motion, which depends on the mean and standard deviation of the direction and magnitude images (see Section 4.4). All pixels belonging to the background are set to 0, all others to 255, which creates a binary image. The background determination is part of the image class.

`minDisplacementHSV` The parameter is directly applied to the results of the optical flow calculations (displacement in x and y direction), before constructing the HSV image. All pixels which have a displacement below this threshold, will not be taken into account. Due to the dark background, a lot of pixels have no or very little displacement. Furthermore, these small displacements are not accurate since no keypoints can be determined for dark uniform parts of the image and the displacement is calculated by interpolation. If an angle is calculated from those values, the variation is very high and does not represent the actual movement of the pixels. The value should be at least 0.0001. Increasing this value, reduces the variation of $\sigma_{direction}$ and $\sigma_{magnitude}$. Thus, this parameter can also be used to ignore small movements, which may be caused by clouds and lightning. In this case a value larger than 0.8 is required.

`magThresholdAngleMatMean` This parameter is used to define which pixel values are considered in calculating the mean and standard deviation values from the HSV image. All pixels below this magnitude of motion are not considered for calculating the mean and standard deviation values for the direction and magnitude of motion. Masking these pixels out has two effects: First, the direction of motion can not be accurately detected without this threshold, since the small magnitude of motion values are not precise. This is due to the fact, that no keypoints can be determined for dark uniform parts of the image and the displacement is calculated by interpolation. Second, a larger value for `magThresholdAngleMatMean` reduces the variation in the calculation of the mean values, thus the standard deviation is reduced. The main direction of motion can thus be determined more accurately. The smaller the parameter value, the smaller the main direction of motion range. Therefore, less parts of the images are masked out, leading to more false positive detections. On the other hand, a wide range in the direction of motion excludes all meteors in this direction, which can not be detected. In contrast to `minDisplacementHSV`, this value only effects the calculation of the statistical values, while `minDisplacementHSV` determines which pixel values should be omitted for all following calculations.

Blob detection parameters

Once the HSV image has been constructed from the optical flow calculations and the statistical values for background motion have been calculated, the *blob* detection can be run. The *blob* detector is applied to the magnitude of motion image with the background masked out. These parameters are directly passed to the *OpenCV* function for the simple blob detector (see Section 4.3.4). They influence which types of *blobs* are returned by the function, by applying different filters. Selectable filters include area, circularity, convexity, inertia and intensity. Since the intensity is already used beforehand, by applying `magThresholdAngleMatMean`, this filter is not used.

Furthermore, testing showed that filtering by convexity is not useful. Each filter can be turned on and off by setting the `filterBy` option, furthermore the minimal and maximal thresholds can be set. The *blob* detection is part of the detection class.

`filterByArea` This parameter is used to return only *blobs* with an area between `minArea` and `maxArea`. Since the typical size of a meteor is known, this option reduces the number of false positive detections. Thus, the filter should be used with `minArea` $= 60$ and `maxArea` $= 1000$.

`filterByCircularity` Since meteors are usually imaged as elongated, but round shapes, this filter is used to reduce false positive detections. The faster the meteor and the longer its tail, the more elongated is the detected *blob*. However, mostly the meteor *blob* does appear more circular than *blobs* caused by the background. Thus, `minCircularity` $= 0.7$ reduces most false positive detections. Since meteors can also appear as circular *blobs*, `maxCircularity` should be set to 1.

`filterByInertia` This filter is not used, since meteor *blob* usually appear with a high inertia (very elongated). Thus, setting this options would require a very low `minInertiaRatio`, which does not reduce false positive detections. A higher value would discard meteors.

`minThresholdBLOB, maxThresholdBLOB, thresholdStep` The *blob* detection is applied to binary images, as outlined in Section 4.3.4. The creation of the binary image is influenced by this parameters. Since a binary image is already passed to the function, all parameters are not used. Again, the binary image is created by setting all pixels belonging to the background to 0 and all others to 255.

Meteor candidate confirmation

The list of *blobs* detected in one frame must be processed, in order to detected meteor candidates. This is part of the detection class and takes place after applying the *blob* detector.

`minStdBrightnessThresh` Since meteors are usually bright compared to the background and the area of the detected *blob* is larger than the meteor itself, the *blob* brightness has a high variation. In order to be detected as a meteor candidate, the *blob* brightness variation must be above this threshold. However, faint meteors result in a lower brightness variation, thus this value should not be set too high.

`brightnessThresh` Meteors are usually bright, thus this threshold filters out all *blobs* below this threshold. Currently this is not used, since black areas are already filtered by `minStdBrightnessThresh` and during preprocessing a threshold is applied (`thresNoise`) to the original image to reduce noise. Thus, the parameter is currently not used (set to 0), but could be useful for in orbit adaption in case `thresNoise` must be reduced.

Meteor confirmation and verification

All meteor candidates must be confirmed, by evaluating if the same candidate can be detected in another frame. Next, a meteor is verified, if it is detected in a settable number of frames (`framesToConfirm`).

`angleDev`, `maxBlobDistance` In order for a meteor candidate to be confirmed, it must be detected in another frame. Two candidates are considered related, if they move in a similar direction. Their difference in motion direction must be below `angleDev`. A value between 10 to 40 is working for most meteors. The second condition two candidates must fulfil, is the distance between them.

The distance should be below `maxBlobDistance`, in order to reject meteor candidates which moved an unrealistic distance between frames. Thus, the distance can be derived from the maximum expected moving distance between frames, which depends on the velocity of the meteor (V_m), the camera frame rate (FR), pixel width (W_{PX}), binning, focal length (f) and orbit altitude (see Equation 4.12).

$$\texttt{maxBlobDistance} = \frac{\frac{f}{FR * orbit\ \ altitude} * V_m}{W_{PX} * binning} \qquad \textbf{(4.12)}$$

Using this equation with the values from the *SOURCE* mission, 500 km orbit, $V_m = 72\,\text{km/s}$, binning 2, $FR = 6$ *fps*, $f = 12\,\text{mm}$ and $W_{PX} = 5.56\,\mu\text{m}$, results in a distance of about 24 *px*. However, the actual *blob* distance can be larger, since the meteor has a tail and the *blob* coordinates can differ from the meteor coordinates. Thus, `maxBlobDistance` is set to 33 after several tests.

`framesToConfirm` This parameters defines in how many frames a confirmed meteor must be detected, in order to be verified as a meteor. The value depends mainly on the frame rate, currently a value of 3 is used.

`storageTime` This parameter defines for how many frames all meteor candidates are stored. Since a meteor candidate is compared to all candidates of previous frames, the minimal storage time must be in the order of the expected meteor duration. With a frame rate of 6 *fps* and an expected meteor duration of 3 s, the candidates of 18 frames should be stored. This value influences the detection performance and can be adapted, it should be increased, if precision is high and recall is low.

`thres_number_keypoints` This value is intended to reduce the number of false positive detections. In case more *blobs* than indicated by `thres_number_keypoints` are detected in one frame, all *blobs* from this frame are rejected. Since all previous measures to

reduce the number of *blobs* not being a meteor are sufficient (like removing the background), this parameter is currently not used. However, this could change in orbit and a value of 20 *blobs* maximal per frame is suitable.

4.7. Algorithm parameter optimization and performance evaluation

The final algorithm performance is evaluated in terms of speed and detection performance. The detection performance is evaluated using the previously mentioned test set V1-V4 and an optimized set of algorithm parameters. The speed performance is evaluated on a desktop machine as well as on the PLOC intended for the *SOURCE* missions. The approach to determine both aspects of the algorithm performance is given in the following.

4.7.1. Detection performance and optimization of algorithm parameters

In the previous sections, the effect of different parameters on the detection performance for individual videos are analysed. Hence, the algorithm parameters were changed and optimized manually for each video. This approach was useful to determine a range of suitable parameters and analyse the effect of parameters. Furthermore, parameter values which can be kept constant independently from the type of background and meteor were identified. However, in order to detect as much meteors as possible with one parameter set, the parameters not kept constant must be optimized. Thus, the goal is to identify the parameter set which works best for all selected test set videos. For the optimization 113 videos from the test sets V2 and V3 (black marble V2-V3 and all ISS background videos) are selected, because they provide the most realistic representation.

In Table A.25 all parameters for the algorithm with their identified ideal values are listed. The parameters chosen for optimization with

Table 4.6.: Detection algorithm parameter selected for optimization with a suitable value range.

Parameter	Value
`magThresholdAngleMatMean`	0 to 1
`minDisplacementHSV`	0.0001 to 1
`minArea`	60 to 100
`minCircularity`	0.3 to 0.7
`minStdBrightnessThresh`	5 to 18

a suitable range is given additionally in Table 4.6. Only parameters actually used are given, some features are not used (e. g. erosion and dilation), therefore the parameters for these features are left out. The parameters selected for optimization were identified to have a large impact on the algorithm performance, because they depend on the background and meteor properties, thus they should be optimized.

In order to optimize these parameters, a second script is developed (see [72]). The purpose of this script is to optimize the algorithm parameters, in order to detected as many meteors from all test set videos with a lower number of false detections. Therefore, the algorithm is treated as a function with 6 parameters which returns the $F - measure$. As mentioned before the $F - measure$ is the harmonic mean value of precision P (describing the algorithms ability to detected only meteors) and recall R (describing the ability to detected all meteors). Thus, a high $F - measure$ indicates the ability of the algorithm to detect all meteors with a low number of false detections.

The optimization works as follows, see Figure 4.17 for a visualization: An initial set of algorithm parameters in a suitable range is selected. The algorithm is applied with those parameters to all suitable test set videos (Black marble V2-V4, all ISS Italy and West India videos, totalling to 113 videos). Afterwards, the performance

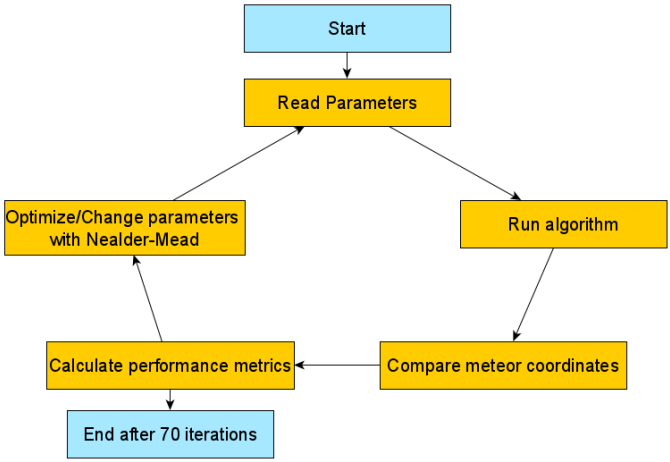


Figure 4.17.: Algorithm optimization loop: The algorithm is for each step applied to all videos.

metrics (P, R and $F-measure$) are calculated. The algorithm parameters are successively changed, to increase the $F-measure$.

This process of optimizing the output of a function with different parameters applied to a set of data is performed using the *Nelder-Mead method*. The method is part of the Python library *scipy*. All steps are iterated, the optimization is stopped after 70 steps.

The values of the selected parameters after optimization are found in Table A.26. Some values slightly exceed the range defined as useful from previous tests. This shows the need to optimize the parameters for all videos, instead of trying to optimize parameters manually. With the optimized values, 89 of the 113 meteors could be detected ($TP = 89$), while only declaring 5 frames wrongly as containing a meteor ($FP = 5$). This results in a P of 0.9468, meaning the 94.7 % of all detections were actually meteors (correct detections). Furthermore, 24 meteors could not be detected (FN). This gives a R value of 0.7876, which means 78.8 % of all meteors could be detected. The results are given in Table 4.7. The performance is

Table 4.7.: Final algorithm performance metrics

Parameter	Value
P	0.9468
R	0.7876
$F - measure$	0.8599

slightly reduced compared to the results given in the master thesis (see [72]), which is due to the fact the more and more diverse videos are used for the optimization.

The performance values indicate a very robust and reliable meteor detection algorithm. Since the selected videos contain meteors with varying properties and different background, the algorithm is expected to work in a wide range of scenarios. However, it has to be kept in mind that these values are determined using a limited set of videos. Thus, further tests with even more different backgrounds and meteor types are required. Additionally, the videos used for testing are rather short, thus the frequency of meteors is higher than in orbit. It is likely, that the number of false positive detections increases with longer observation time and a lower frequency of meteors. Nevertheless, the algorithm concept could be proven and can be used on a satellite based meteor observation mission.

Effects of background and meteor properties on algorithm performance

As mentioned, the detectability of a meteor not only depends on the algorithm parameters, but also on the chosen background and meteor properties of the implemented meteor. Here, the influence of those properties is given along with possible parameters changes to adapt the algorithm to a specific meteor or background property.

Meteor brightness Faint meteors are more frequent, but harder to detect. In order to detect faint meteors, the parameter `thresNoise` has to be lowered. However, this also increases false positive detections, since less noise is reduced, which could be detected as a meteor. Furthermore, the `minArea` could be increased as well, since faint meteors illuminate less pixels and result in smaller *blobs*.

Meteor velocity and tail length Fast meteors spread their signal on more pixels in a frame and thus appear elongated. This also applies to meteors with a long tail. Thus, the *blob* filter has to be adapted by lowering `minCircularity`. Furthermore, the displacement calculation of the optical flow is less accurate, since those meteors have a high variation in brightness along their illuminated tail. As a result, the direction is less accurate, resulting in lower probability of detection. Finally, the distance between *blobs* could increase, requiring a higher value of `maxBlobDistance` in order to confirm meteor candidates.

Meteor duration Short duration meteors are less likely to be detected, since they may not be detected as *blobs* for the required number of frames. Thus, `framesToConfirm` must be adjusted depending on the expected meteor duration. This could lead to higher false positive detections.

Clouds and lightning The last videos of test set V3 are using the West India background video. This video contains lightning as well as clouds. The large clouds introduce homogenous bright regions into the image. The optical flow calculation is not precise for those types of regions, thus the calculated displacements are smaller than the expected background motion. As a result, these regions influence the background motion determination, resulting in a lower estimation of the background motion. Consequently, other regions belonging to the background have a higher magnitude of motion than the calculated background motion and thus are not

masked out. By increasing `minDisplacementHSV`, small displacements from the optical flow calculations are omitted and replaced by 0. Therefore, small displacements do not influence the background motion calculation. Thus, the background motion is better determined and the detection improved. A second option is to increase `magThresholdAngleMatMean`, which excludes small magnitudes of motion before calculating the statistical values mean and standard deviation. Since these values are used to estimate the background motion, this also avoids using the displacements caused by clouds in the background motion.

Lightning events are short, thus they do not results in *blobs* which can be tracked over different frames. Thus, lightning does not result in false positive detections.

Rotation Due to the rotation of the satellite, a second direction of motion is introduced to the image sequence. Therefore, the background motion can not be accurately determined, which results in more false positive detections. These detections mostly occur on the edges of the image, since here the rotation results in a larger magnitude of motion of the pixels (rotation is applied around the centre of the image). Thus, these regions are not inside the determined ranges for main direction and magnitude of motions. Consequently, these regions are not masked out and lead to false detection of *blobs*. Since the *blobs* have a uniform movement and appear in multiple frames, they are confirmed as meteors. The higher the rotation rate, the more false positive detections occur.

A mitigation would be to not take into account the edge of the image and reject all *blobs* detected within a settable distance from the edge. Another approach would be to adapt the determination of the background motion. This could be done by increasing the range of main direction and magnitude. However, in this case meteors moving close to the main direction of motion are not detected.

4.7.2. Speed performance

The speed performance describes how many frames the algorithm processes per second. This is needed for the observation concept (see next chapter), in order to determine the time needed to process the images and in turn for how long meteors can be observed. The speed measurements were also done during development, in order to avoid developing inefficient algorithms and improve the speed of the algorithm.

However, the crucial performance is the speed on the used PLOC. As mentioned earlier, the algorithm is currently developed for the *SOURCE* mission. Thus, the performance is measured on the *Trenz TE0720-03-1CFA*.

Since this board has an FPGA, it is possible to increase the processing speed, by outsourcing parts of the algorithm into the FPGA. This means, some functions are implemented directly in the hardware of the FPGA and thus are not executed by the CPU. A function implemented in hardware works usually orders of magnitude faster.

The manufacturer of the FPGA offers the possibility to compile a program with some functions of the program implemented in hardware. Several steps and multiple software packages are required before the hardware acceleration is working, here only the results are given. The actual hardware acceleration is implemented in a software called *SDSoC*. This software is needed to compile the algorithm with selected function exported to hardware.

The existing algorithm source code (written in *C++*) is imported into SDSoC. Since the image processing functions (optical flow and blob detection) from *OpenCV* are the most time consuming, those are candidates for hardware acceleration. In a first step, the optical flow calculations are exported into hardware. The function call `calc-Optical-Flow-Farneback` is replaced by an identical function from *xfOpenCV*[9]. The *xfOpenCV* library provides kernels containing *OpenCV* functions optimized for the used FPGA. After

[9] See `https://github.com/Xilinx/xfopencv`

Table 4.8.: Final algorithm speed performance. The processing time for one frame is shown. The optical flow step includes the optical flow calculations as well as background subtraction. The *blob* detection steps include all steps needed for meteor candidate detection and confirmation.

Processing step	Without acceleration (ms)	With acceleration (ms)
Optical flow	2000	380
Blob detection	450	430
Complete	2450	810

replacing and selecting the according function for hardware acceleration, the algorithm is compiled and the binary file can be executed on the PLOC.

For comparison, the same algorithm without hardware acceleration is also run on the PLOC. As an example, one of the test set videos is processed with both versions and the time needed for different processing steps is measured. Here, the working algorithm as described above (see Section 4.4) is used as implemented in [72] with minor adaptions to include time measurement and hardware acceleration. The results are shown in Table 4.8.

As can be seen, the hardware acceleration of the optical flow calculations improves the overall runtime by a factor of 3, the optical flow calculations alone are 5.2 times faster. This is a significant acceleration and allows the algorithm to process images in a reasonable time. The current processing rate of $\sim 0.8\,\mathrm{s}$ per image can be further improved: Currently, no other speed optimization is done, for example reducing the number of copy operations of image data. Furthermore, more functions could be accelerated by hardware, e. g. the *blob* detection or the images processing (threshold as well as erosion and dilation). Thus, a processing rate of $\sim$

0.5 s per images is realistic. Higher rates can be achieved with more powerful PLOC hardware. However, currently no real time processing of the data is possible, thus the images must all stored in the mass memory of the PLOC.

4.8. Summary and outlook of algorithm development and testing

As outlined above, a working detection algorithm was developed, which is based on optical flow calculations and detection of regions not moving with the background. This algorithm was tested with a variety of different videos, showing meteors as seen from a moving satellite with different backgrounds and meteor properties. It was shown, that the meteor detection algorithm is working due to the high recall and precision values. Thus, a critical part of any satellite based meteor observation mission was developed.

However, more development effort is necessary. This includes further testing with even more backgrounds and meteor types. Additionally, the concept of running the detection algorithm on board the satellite must be developed. This could include multiple runs of the algorithm with different parameters to ensure all meteor types are detected. Mostly parameters concerning the *blob* detection need to be adapted. For the most resource intensive part, the optical flow calculations, no parameter change is necessary. Thus, algorithm could run after the optical flow implementation with different parameters without significantly influencing the required time to process the data. Furthermore, the algorithm needs to be adapted in order to work with a rotation of the satellite. This is especially crucial for the *SOURCE* mission, since a rotation is to be expected due to the lack of reaction wheels. This could be achieved by adapting the filter thresholds for the direction of movement of the detected *blobs* as well as the calculation of the average movement direction of the scene.

An option to further increase the autonomy of the meteor observation and increase the precision and recall, would be the automated adaption of parameters. Depending on the image properties, e. g. background noise and general brightness of the image, some parameters could be adapted. This would include e. g. `thresNoise`, which is used to threshold the image before optical flow calculations. The parameter value could depend on the average image brightness and mask out a settable percentage of the faintest pixels.

Additionally, a concept to adapt the algorithm once the satellite is in orbit needs to be developed. This could include downlinking additional telemetry data, like the meta data of all detected *blobs* (size, coordinates, brightness, frame number), even if they did not get verified as a meteor. In combination with additional data of the images, like some histogram data and the calculated mean and standard deviation of the magnitude and direction of motion, meteors could be identified and the parameters adapted. Furthermore, a series of images could be downlinked and used in the artificial meteor simulation, to adapt the algorithm on the ground.

Also, in a first step meteor observation should take place when the satellite is over an area with low light pollution (e. g. oceans or uninhabited areas). Here, the algorithm can be tested more easily since less noise (e. g. city lights) is present. Observations over areas with low light pollution are rather the rule than the exception, because most of the Earth is covered with water or uninhabited. This can be seen, when looking at a map showing the artificial sky brightness (see [80]) Furthermore, different sets of parameters should be used, depending on the expected background conditions.

Finally, setting the parameter for in orbit operation is a trade off, depending on the available resources: The parameters could be adapted to detect almost all meteors, which would also increase the number of false detections and thus the data which need to be downlinked. In this case, the R value would be increased, while P decreases. On the other hand, the parameters could be set to reduce false detections, which would also reduce the number of correct detections. In this case, the P value would be increased, while R

decreases. Currently, the optimization scripts uses the $F-measure$ to optimize the parameters. This means R and P are weighted equally. A new measure could be introduced, which weights one of the parameters more than the other. Thus, the script can be used to optimize the parameters for either a better R or P value.

5. Instrument design and scientific performance

In this chapter, the results of the two previous chapters are used to design a suitable instrument for each of missions *SOURCE* and *FACIS*. This includes selecting components (e. g. camera) and choosing parameters (e. g. framerate).

Furthermore, an observation concept, taking into account the scientific objectives as well as satellite bus and instrument parameters, is developed. The observation concept defines feasible operation times of the instrument, taking into account constraints from the satellite bus, the mission design, and the instrument. Finally, the expected scientific performance regarding trajectory accuracy (*FACIS* only) and number of observable meteors is analysed.

In the following different terms are used to describe the instrument and its parts. Generally, the designed instrument is called *Meteor Observation System (MetObs System)* and consists of two parts: The *Meteor Observation Camera (MetObs camera)*, which consists of a camera and lens. The second part is the *Meteor Observation Controller (MetObs controller)*, responsible for receiving commands from ground, controlling the camera and processing images. The general approach of selecting the parts is given first, before describing instrument details, parameters and observation concept for each mission.

Meteor Observation Camera The *MetObs camera* consists of the camera itself and a suitable lens. Both components are analysed with the *Sensitivity* script in order to determine important parameters for a meteor observation camera (see Section 3.3). According to the simulation, the camera should have a high quantum

efficiency, large aperture, low dark noise and a short exposure time. Furthermore, large pixels are desirable. These requirements are valid for *SOURCE* as well as *FACIS*. Additionally to these scientific requirements, the camera should also be low cost and demonstrate the feasibility of using COTS parts for a satellite mission. Since *SOURCE* and *FACIS* are both missions in a Low Earth Orbit, the radiation is low and non radiation hard hardware can be used. Therefore, different cameras for machine vision were analysed.

Machine vision cameras are usually used to control industrial processes, e. g. for quality control in the electronic industry[1]. These cameras are suitable due to their robustness and wide range of camera options and lens available. Furthermore, many cameras use the same interface for controlling the camera, making software development more independent from the actual hardware used. Also, a lot of manufacturers offer a Software Development Kit (SDK), including examples and support, which also reduces the effort needed for software development.

Finally, to reduce development effort and cost and since *SOURCE* should serve as a demonstration mission for *FACIS*, the same camera should be used on both missions.

Meteor Observation Controller The second part of the instrument is the *MetObs controller*, which is a critical part of the instrument and the detection chain. The controller is responsible for controlling the camera, storing images and forwarding them to the On Board Computer (OBC) as well as receiving commands from the OBC. The most important task is running the *Meteor Detection Algorithm (MetDet algorithm)*.

Thus, the basic requirements for the *MetObs controller* are: The controller has to offer interfaces to the *MetObs camera* and be able to run the software needed to control the instrument. Furthermore, interfaces to the OBC must be provided, this also includes software

[1] `https://www.stemmer-imaging.com/de-de/maerkte/`

to receive commands. Finally, sufficient processing power and memory (working memory as well as storage) to run the algorithm and store images must be provided.

This thesis focuses on the development of the *MetObs camera* as well as *MetDet algorithm* and deriving requirements for the satellite mission. Nevertheless, development effort is dedicated to the development of the *MetObs controller*, especially the control software and the efficient implementation of the *MetDet algorithm* (see previous chapter).

5.1. FACIS

FACIS, as introduced in Section 1.3, is a satellite mission consisting of two satellites in a formation. The most important parameters for the following sections are again mentioned in Table 5.1

Table 5.1.: FACIS satellite bus and mission parameters

Property	Value
Orbit	Sun-synchronous/ISS orbit
Orbit altitude	300 km to 565 km
Mission duration	$\sim 2\,year$
Pointing accuracy	**tbd**
Attitude determination accuracy	currently $36''$, required $< 13.2''$[a]
GPS position accuracy	required < 170 m
Time accuracy	required < 15 ms
Maximum power generation	90 W
Payload data rate per day	1.6 GB[b]

[a] The current design of the *FACIS* satellite has a lower accuracy (see [6]), however according to the analysis in this thesis a more accurate attitude determination is required. The following analysis uses the required value.

[b] This is the current assumed data rate. The X-band downlink system, planned for this mission and under development at the IRS, is most likely to downlink more data. Currently about 45 GB/d are feasible for a 600 km orbit (Information received via personal correspondence with Susann Pätschke on 28.07.2021 and [81]).

5.1.1. Instrument components

Meteor Observation Camera As mentioned, machine camera vision cameras are affordable and robust cameras, which should be used as a meteor observation camera. Several cameras were analysed using the *Sensitivity* script, one suitable candidate is *GenieNano M1920* due to the following reasons: The camera is equipped with the *Sony IMX249* sensor, which has a low dark noise, a high quantum efficiency and large pixels. Furthermore, the camera is using the machine vision standard *GigE Vision*, which allows to use different SDKs to developed the software. Since the camera is distributed by *Stemmer Imaging*, their SDK Common Vision Blox (CVB) can be used. This SDK offers a good documentation and examples in *C++*, which is needed to develop the control software using the IRS Flight Software Framework (FSFW). Finally, this camera has a *C mount* to attach the lens, for which a wide range of different lenses are available.

A similar sensor is the *Sony IMX174*, which is used in the *JAI GO-2400M-PGE* (see [82]). This sensor has the same resolution as the *Sony IMX249*, however the quantum efficiency is lower (74 % @550 nm compared to 82 % @525 nm for the *Sony IMX249*, see [83]). Furthermore, this camera has a lower operational temperature range from $-5\,°C$ to $45\,°C$. However, the power consumption is slightly lower ($\approx$ 2.64 W @12 V) and the dynamic range is larger. This camera could be an alternative, if for some reason in the further development the *GenieNano M1920* can not be used.

Regarding the lens, a large aperture (low f-number) is an important parameter. Furthermore, the FOV, depending on the focal length, is a trade off: A lower focal length increases the observed area, but reduces the trajectory determination accuracy.

Lenses options are available from a product line by the manufacturer *Schneider Kreuznach*, offering lenses with a wide range of focal length and, respectively, FOV. Most lenses are offered as ruggedized versions, which increases the robustness of the lens and reduces the risk of damages during launch. Furthermore, the focus as well as

f-number can be set and fixed with a screw, to avoid a change by vibrations during the launch. Also, those lenses have been used on different satellites[2].

Possible lens options for the *FACIS* missions are the *Schneider Kreuznach Cinegon 1.4/12* as well as *Schneider Kreuznach Xenoplan 1.4/17* and *Schneider Kreuznach Xenoplan 1.4/23*. Those lenses have a suitable FOV resulting from their focal length of 12 mm, 17 mm and 23 mm. Choosing the focal length is a trade off, depending on the scientific objective. Camera settings and processing capabilities of the *MetObs controller* must be considered as well. This is done in the following section.

[2] see [51], further information and confirmation received via personal Email correspondence with manufacturer

Table 5.2.: Parameters for the *GenieNano M1920* and *Schneider Kreuznach* lenses

Parameter	Value
Exposure time[a] t_{exp}	34.23 µs to 16 s
Frame rate[a] FR	1 *fps* to 38 *fps*
Quantum efficiency[e] η	73.9 % @444.3 nm, 82.9 % @548.3 nm, 67 % @658.5 nm
Pixel size[b] A_{PX}	5.86 µm
Dark current[b] μ_I	7 $electrons/(pixel \cdot s)$
Native sensor resolution[a] r_X/r_Y	1936 *px* × 1216 *px*
Binning[a]	1 to 2, vertical and horizontal
Temperature range (operational)[a]	−20 °C to 60 °C
Temperature range (non-oper.)[a]	−40 °C to 80 °C
f-number[d] F#	1.4 to 11
Focal length[d] f	12.7 mm, 17 mm and 23 mm
Temperature range (non-oper.)[d]	−25 °C to 70 °C

[a] From the *GenieNano M1920* data sheet (see [45]).
[b] Measured by an *EMVA 1288* test at the Institute of Space Systems.
[c] Measured during ground testing of the camera.
[d] From the *Schneider Kreuznach* data sheet (see [49], [84],[85])).
[e] Derived from responsivity curve (see [45]) and the quantum efficiency measured during *EMVA 1288* testing.

Chosen parameters After selecting the camera and lens product line, the final lens must be chosen and suitable camera parameters settings for meteor observation must be defined. Mainly, the frame rate and exposure time, as well as focal length and aperture setting must be chosen.

Increasing the frame rate allows for a better meteor velocity and position determination, both are needed for a precise trajectory determination. Furthermore, the meteor detection rate is increased, since more meteors are imaged in the minimal number of frames necessary for a useful observation. As a drawback, the amount of data to be downlinked and processed increases. Taking into account the processing capabilities of the *MetObs controller* (see following paragraph), the frame rate should not exceed 10 *fps*. However, it would be possible to increase the frame rate to 15 *fps* if longer times for processing those images are acceptable. This reduces the number of time slots dedicated to meteor observation (see Section 5.1.2).

The required processing power is also influenced by the binning setting. If binning is not used, images are taken at a higher resolution. A higher resolution requires more processing power and reduces the sensitivity, but allows for a more precise meteor velocity and position determination. For the camera settings, binning 2 is selected for two reasons: First, the camera is more sensitive to faint meteors, since 4 pixel values are summarized. Second, the resolution is reduced, reducing the amount of data to be downlinked and processed. However, the binning setting can be changed in orbit, allowing to trade longer processing times for a higher trajectory accuracy.

Regarding the focal length, it would be possible to use lenses with a focal length of 12 mm, 17 mm and 23 mm. Increasing the focal length reduces the area observed. On the contrary, the limiting magnitude is increased due to the increased aperture. In the end, the number of observable meteors is nonetheless reduced, since the observed area has a larger effect. The advantage of a larger focal length is an increased angular resolution and thus a more accurate meteor velocity and position determination. Hence, it is a decision about the scientific output: The 12 mm focal length results in more

meteor trajectories with less accuracies, while the 17 mm or 23 mm focal length results in less, but more accurate trajectory measurements.

Table 5.3.: Effect of diff. lens and frame rate options for *FACIS*

Parameter	12 mm[a]	17 mm[a]	23 mm[a]
FOV horizontal/vertical	48.1° × 31.3°	36.9° × 23.7°	27.7° × 17.6°
IFOV horizontal/vertical	0.0497° × 0.0515°	0.0381° × 0.0389°	0.0286° × 0.0290°
Framerate	10 and 15 *fps*	10 and 15 *fps*	10 and 15 *fps*
Meteor limiting magnitude [b]	2.88 and 3.02	3.17 and 3.34	3.50 and 3.69
Aperture diameter	9.07 mm	12.14 mm	16.43 mm
Meteor detection rate[c]	11.64 and 16.39 *met*/h	7.58 and 10 *met*/h	4.35 and 5.42 *met*/h
Minimal tilt angle 565 km[d]	18°	22°	27°
Useful tilt angle 565 km[e]	31°	36°	38°
Minimal tilt angle 300 km[d]	20°	25°	29°
Useful tilt angle 300 km[e]	34°	36°	38°

[a] Frame rates 15 *fps* and 10 *fps* are used to determine limiting magnitude and meteor detection rate.
[b] Calculated for a typical meteor (see Section 3.3.2) as seen from 565 km orbit for 10 and 15 *fps*.
[c] Using *SWARMSv2* settings from Section 3.4.1 with adapted focal length/framerate, tilt angle of 33.66°, 565 km orbit.
[d] Minimal tilt angle required to get overlapping FOV at minimal satellite distance for low trajectory error (349 km @ 300 km orbit respectively 847 km @ 565 km orbit). If the tilt angle is lower at this distance, there is no overlapping area.
[e] Minimal angle at which the ideal satellite distance for a max. overlapping FOV is larger or equal to the min. satellite distance required for a low trajectory error. Below this tilt angle, the area covered by both instruments is reduced.

In Table 5.3 the three lens options and two frame rate options are shown with their effect on the meteor detection rate. As can bee seen, a smaller FOV reduces the number of observable meteors. However, this can be partly compensated by increasing the frame rate.

Another aspect of choosing the focal length is the tilt angle and the distance of the satellites. As outlined in Chapter 3, two factors influence the distance: The trajectory error and the tilt angle. The minimal distance required is defined by the trajectory error, below this distance the error in trajectory calculation increases. The tilt angle influences the required distance, in order to maximize the area covered by both instruments. The larger the tilt angle, the larger the distance between the satellites. Since the minimal distance is defined by the trajectory error, the tilt angle must be large enough, in order for the ideal distance between the satellites to reach this distance defined by the trajectory error. Thus, the useful tilt angle is defined as the tilt angle at which the ideal distance for a maximum overlapping FOV is larger or equal to the minimal distance required for trajectory determination.

For a smaller FOV, the tilt angle must be larger at the same distance, in order to get the FOVs to overlap. This means, for a smaller FOV and larger focal length, the tilt angles increase and therefore only brighter meteors can be observed. All in all, a smaller FOV increases accuracy of the trajectory determination, but reduces the number of observable meteors and increases the required tilt angle and thus allows only the observation of brighter meteors.

If the processing unit is powerful enough to process the larger amount of data and it is feasible to downlink the increased number of images, using the 17 mm lens with 15 *fps* would be the ideal solution. This allows for a precise trajectory determination while also allowing to observe a significant number of meteors. Otherwise the 12 mm lens with 15 *fps* should be used to ensure the observation of a significant number of meteors. For both lenses the frame rate could be set, depending on the processing time and power available.

Table 5.4.: Chosen parameter values of the *GenieNano M1920* and *Schneider Kreuznach Xenoplan 1.4/17* for the *FACIS* mission

Parameter	Value
f-number F#	1.4
Exposure time t_{exp}	$\frac{1}{15}$ s to $\frac{1}{10}$ s
Frame rate	10 *fps* to 15 *fps*
Binning	2, horizontal and vertical

The remaining camera and lens parameters are set as follows: The lens f-number should be set as low as possible (1.4) to increase the amount of light reaching the sensor. The exposure time is simply the inverse of the frame rate, thus $\frac{1}{15}$ s to $\frac{1}{10}$ s. All settings are summarized in Table 5.4.

Meteor Observation Controller The *MetObs controller* has to fulfil two basic tasks: Control the camera and process the images in order to detect meteors. This implies requirements such as:

- providing interface for camera connection
- being able to run camera control software
- being able to process images in a reasonable time

The chosen camera requires a Gigabit Ethernet interface for camera control and a Linux operating system to run the camera control software using the SDK from the manufacturer. Furthermore, the meteor detection algorithm is based on optical flow calculations using the *OpenCV* library. Thus, the *MetObs controller* must be able to perform those operations reasonably fast. A faster processing increases the scientific output, at least 2 images should be processed per second.

As for the *MetObs camera*, it is possible to use COTS parts as part of a technology demonstration. Two candidates would be the *Trenz TE0720-03-1CFA* (see [86]) and the *Aitech A177* (see [87]).

The first one is a System on a Chip (SoC) based on the *Xilinx Zynq XC7Z020*, which is also produced with an extended temperature range for use in the automotive sector. Furthermore, the *Xilinx Zynq XC7Z020* chip is used for space applications[3]. The FPGA allows to perform functions directly in hardware and thus accelerating the calculations.

The *Aitech A177* is a ruggedized processing unit equipped with an *NVIDIA Jetson TX2 System on Module*. The processing unit is designed for harsh environments, but not space graded. However, the Nvidia chip provides high processing capabilities for optical flow calculations.

For the current instrument the *Trenz TE0720-03-1CFA* board is used, due to the flight heritage of the *Xilinx Zynq XC7Z020* and the suitability for the *SOURCE* mission due to the small form factor. However, for future mission a different processing unit could be chosen as the *MetObs controller*.

5.1.2. Observation concept and scientific performance

In the next two sections, first the observation concept is derived, which determines how much time can be dedicated to meteor observation during a 2 *year* mission. This information is then used to estimate the scientific performance of this mission in terms of number and types of meteors as well as trajectory accuracy.

From herein after, the *GenieNano M1920* equipped with *Schneider Kreuznach Xenoplan 1.4/17* is used as *MetObs camera* and the *Trenz TE0720-03-1CFA* as the *MetObs controller*. Furthermore, the analysis is done for a 565 km orbit with an LTAN of 11 and a tilt angle of 35.9° which results in a phase angle of 7.53° and a satellite distance of 910 km. This distance is chosen, because it is just over the minimal distance required for a minimal trajectory error. The tilt angle is calculated in order to maximize the area covered by both instruments. The data is summarized in Table 5.5. The analysis is

[3] see https://www.spacemicro.com/products/digital-systems/CSPCUBESATSPACEPROCESSOR.pdf

Table 5.5.: Chosen parameter values for camera, orbit and formation

Parameter	Value
Focal length	17 mm
f-number F#	1.4
Exposure time t_{exp}	$\frac{1}{15}$ s to $\frac{1}{10}$ s
Frame rate	10 *fps* to 15 *fps*
Binning	2, horizontal and vertical
Orbit altitude	565 km
LTAN	11
Tilt angle	35.9°
Satellite distance	910 km

also done for the lower orbit of 300 km (LTAN of 11, tilt angle of 42°), but since in those altitude the quality and quantity of the measurements are not satisfactory, the calculations are only done to cover all orbits initially selected for the *FACIS* mission. This data can be found in the Appendix (see Appendix Section A.14.1).

Observation concept

The observation concept defines how the meteor observation is done, e. g. how much time during the 2 *year* mission can actually be dedicated to meteor observation. This concept takes into account satellite bus constraints (e. g. data and power budget), additionally to the constraints by the instrument parameters (e. g. processing time needed) and the formation parameters (e. g. eclipse time). All aspects must be taken into account, in order to develop a suitable operational concept and later estimate the potential scientific performance.

The observation concept is derived by developing and using a Python based script which calculates the time needed to perform each step of the meteor observation. The steps include:

- Orientation of the satellite to nadir pointing
- Take images containing potential meteors
- Process the images by running the meteor detection algorithm
- Transfer images with a meteor to the OBC
- Downlink the images

For the observation concept it is also necessary to take into account the budgets available for data downlink and power. Thus, the script calculates how much data needs to be downlinked and how much power is needed for the observation. For the *FACIS* mission the constraints and settings are defined in Table 5.6. Some assumptions need to be explained, before results of the analysis are given:

First, the camera properties are either set (frame rate), measured (image size) or taken from the data sheet (power).

Second, the speed parameters are taken from measurements. The process speed is the duration needed to process one image with the algorithm using hardware acceleration. It is assumed, that the final speed optimized algorithm is able to reach this processing rate (see Section 4.5). The transfer speed is the time needed to transfer one image from the OBC to the *MetObs controller* and is measured using a serial terminal simulating the OBC and the *MetObs controller* prototype.

Third, the meteor parameters are taken from the *CILBO* data evaluation (event duration, see Section 2.4) or from the *SWARMSv2* simulation (meteor detection rate). The meteor detection rate is calculated for the ideal parameters as discussed earlier using the Grün model. Since using the Grün model results in high detection rates, the calculated amount of data is likely overestimated. The false event rate is an early estimation based on the performance of the algorithm and can be adapted once more accurate data is available.

Fourth, the satellite parameters are estimated for a hypothetical but realistic satellite bus. This concerns the daily data budget, downlink rate as well as time needed to orientate the satellite towards nadir.

Finally, the orbit duration and orbits per day are calculated from an ASTOS scenario, using the given satellite tilt angle and distance.

Table 5.6.: Settings and constraints for the observation concept

Setting	Value
Frame rate	10.00
Image size (MB)	0.40
Power PLOC (W)	3.50
Power Camera (W)	4.00
Process speed (s/Image)	0.50
Transfer speed (s/image)	1.00
Event duration (s)	1.00
Meteor detection rate (Met/h)	8.86
False event duration (s)	1.00
False event rate (Events/h)	2.00
Data budget (MB/day)	1600.00
Data downlink rate (Mbyte/s)	1.30
Satellite orientation time (min)	5.00
Orbit duration (min)	95.78
Orbits per day (-)	15.04
Limit observation time per orbit (min)	30.00
Number of observations per day (-)	10.00
Duration one observation (min)	10, 15, 20, 25 and 30

The last three settings in Table 5.6 can be set in order to evaluate different scenarios. The observation time limit can be set in order to take into account thermal constraints, in case the camera can only be operated for a given time before overheating. For the *FACIS* mission this is not taken into account and set for the typical Eclipse duration of ~ 30 min. The number of observations per day, together with the duration of one observation, gives the total observation time per day.

Further assumptions made for the calculations are as follows: Since the processing can take place during the sun phase, it is assumed that the batteries are charged during the processing and the power supply system is designed to supply enough power to continuously operate the *MetObs controller*. At a later stage of the mission planning, this has to be verified. Furthermore, 10 % of each day (144 min) are dedicated to other satellite tasks, such as ground station passes. Again, this has to be analysed in more detail in later phases. The concept presented in the following is thus an ideal scenario regarding the available time. However, the concept can be easily adapted to new constraints using the developed Python script.

Results As can be seen in Table 5.7, making 10 observations per day, each lasting 20 min, is the maximum possible scenario. In this case, 13.1 orbits of the available 15.04 orbits per day are used for tasks related to meteor observation. A longer observation time or more observation time would result in a backlog of images, since not all images can be processed and downlinked between observations.

Table 5.7.: Time budget for different observation times at 10 *fps* in a 565 km orbit and 10 observations per day. The selected option is marked in grey.

Observation time (min)	Observation time total (min)	Process time acc. (min)	Downlink time (min)	Image transfer time (min)	Orientation time (min)	Total time (min)	Total orbits
10	100	500	3	5	0.9	654	7.30
15	150	750	5	5	1.4	956	10.6
20	200	1000	6	5	1.9	1258	14.0
25	250	1250	8	5	2.3	1560	17.3
30	300	1500	9	5	2.8	1862	20.7

Next, the data budget must be analysed, as shown in Table 5.8. The required data budget is calculated by multiplying the number of expected meteors during the total observation time with the event duration and frame rate. Together with the image size, this gives the amount of data which needs to be downlinked. The same process is applied to determine the amount of data in the downlink budget due to false positive detections. This is also included in the data budget. For the same scenario, 200 min observation time per day, the amount of data to be downlinked is lower than the available data budget (128 MB used from 1600 MB).

Table 5.8.: Data budget for different observation times at 10 *fps* in a 565 km orbit. The selected option is marked in grey.

Observation time total (min)	Data generated (MB)	Images generated	Images with meteor	Data to downlink (MB)	Detected meteors
100	24000	60000	181	72	14.77
150	36000	90000	272	109	22.16
200	48000	120000	362	145	29.54
250	60000	150000	453	181	36.93
300	72000	180000	543	217	44.31

Regarding the power budget (given in Table 5.9), 84 Wh/*day* are needed to operate *MetObs System* for 200 min/*day*. The power demand is calculated by multiplying the power consumption of each device with its operation time. During processing only the *MetObs controller* is operated, the *MetObs camera* is turned off. The calculated power demand only takes into account the *MetObs System*, not the power needed to operate the satellite bus (e. g. the downlink system). A detailed analysis taking into account the complete satellite bus is necessary in a later phase. For comparison: The FLP

has an average bus power usage of 41 W[4] on a typical day. This means per day about 990 Wh are used. A typical day, from which the power usage was calculated, includes several passes with downlink, one uplink pass and nominal payload operation during the day. Even a slightly smaller satellite with less available power should be able to provide enough power for the *MetObs System*.

Table 5.9.: Power budget for different observation times at 10 *fps* in a 565 km orbit. The selected option is marked in grey.

Observation time total (min)	Energy consumption PLOC (Wh)	Energy consumption camera (Wh)	Energy consumption total (Wh)
100	50	7	57
150	75	10	85
200	100	13	113
250	125	17	141
300	150	20	170

For the chosen scenario (200 min observation time) and a frame rate of 10 *fps*, it is expected to observe ~ 25 *meteors*/*day*. However, for a higher trajectory accuracy it is desirable to increase the frame rate to 15 *fps*. In this case, more images need to be processed and downlinked, since the data generated increases with the frame rate. Hence, only 15 observations each lasting 10 min are possible during which ~ 27 *meteors*/*day* are detected. The detailed result table as well as the results for an 300 km are given in the appendix (see Appendix Section A.14.1). All in all, the observation time is limited by the time needed to process the images. Thus, it would be beneficial to increase the algorithm speed in order to increase the scientific output.

[4] Average value calculated from telemetry data on 23.07.2021. Telemetry value is PPTBUP00 PCDU Bus Power.

Table 5.10.: Chosen scenarios for the observation concept

Orbit	Frame rate	Total observation time per day	Observable meteors per day	Data budget
565 km	10 *fps*	200 min	25 *meteors*	128 MB
565 km	15 *fps*	150 min	27 *meteors*	196 MB
300 km	10 *fps*	200 min	13 *meteors*	77 MB
300 km	15 *fps*	150 min	11 *meteors*	96 MB

Potential scientific performance

The previous section dealt with the possible observation times for the given instrument and satellite bus constraints. In this section, the potential scientific performance of the mission regarding number of observable meteors and trajectory accuracy is given for the selected operational scenario.

ASTOS: Observation time The theoretical observation time per day is given by the time both satellites are in Eclipse (see Section 3.4.1). Since the actual observation time is limited by the processing time and not the theoretical observation time, these times are given in den Appendix (see Appendix Section A.10 and Figure A.19). Depending on the LTAN, the theoretical observation time for the *FACIS* mission ranges from 7.84 h/d to 8.12 h/d in a 565 km orbit (35° tilt angle) and 9.01 h/d to 9.26 h/d in a 300 km orbit (40° tilt angle).

SWARMS: Observable meteors numbers and types The simulation *SWARMS* is run with the lens and camera selected above as well as with the chosen orbit and tilt angle. Furthermore, the exposure time and frame rate has been adapted and only the Grün model is used. The Grün model is used because it shows good agreement

with the data measured by the *CILBO* observatory (see Section 2.3). The other settings are the same as mentioned in Table 3.8. The resulting meteor detection rates as well as observable meteors per day are shown in Table 5.11. For the scientific performance, it is

Table 5.11.: Meteor detection rates for chosen orbit (565 km) and tilt angle (35°). Detection rates calculated using the Grün model.

Framerate (*fps*)	Detection rate 565 km orbit (*meteors*/h)	Observation time per day (min)	Observable meteors per day
10	8.86	200	25
15	11.06	150	27

important to analyse the types of meteors which can be detected. Therefore, the mass, the magnitude, the velocity as well as the distance to the satellites distributions of the meteors detected by both satellites are analysed. As shown in Table 5.12, a higher frame rate allows to detect slower meteors as well as fainter and more lightweight meteors. All other properties are comparable. As explained in Section 3.4.1, the high tilt angle results in a larger mass and brightness of the detected meteors. The higher tilt angle and distance between the satellites require a larger brightness for a meteor, in order to be detected by both satellites. This bias towards brighter meteors needs to be taken into account when evaluating the data and calculating the meteor flux from the data provided by *FACIS*.

The histograms showing the distributions as well as the data for the 300 km orbit can be found in the Appendix (see Appendix Section A.11).

CILBO: Number of meteors observable In order to evaluate the expected scientific performance, the data base from *CILBO* is

Table 5.12.: Properties of detected meteors for chosen orbit (565 km) and tilt angle (35°)

	Min. values 10fps	Max. values 10fps	Mean values 10fps	Min. values 15fps	Max. values 15fps	Mean values 15fps
Mass (g)	0.10	4.91	0.42	0.10	4.56	0.40
Velocity (km/s)	11.50	71.50	24.15	11.50	71.50	22.70
Distance Sat 1 (km)	491.06	826.87	651.37	491.06	826.87	661.54
Distance Sat 2 (km)	504.57	864.23	655.47	504.57	864.23	645.93
Magnitude (absolut)	-11.15	2.09	-4.72	-10.64	2.42	-4.35

used. It is calculated how many of the meteors in the data base could be observed with *MetObs camera*. Therefore, the properties of each meteor are taken from the data base and the *Sensitivity* script is used to calculate the limiting magnitude of the *MetObs camera* for the given meteor properties. If the limiting magnitude is greater than the apparent magnitude of the meteor, the meteor can be observed. The apparent magnitude of the meteor is calculated from the absolute magnitude as determined by *CILBO*. Then, the magnitude reduction due to distance and angular velocity is subtracted. The distance is calculated from the satellite orbit and the meteor altitude measured by *CILBO*. Accordingly, the reduction due to angular velocity is also calculated from the satellite orbit, the camera parameters and the meteor velocity as measured by *CILBO*.

For the 565 km orbit, most meteors observed by *CILBO* can be observed from orbit as well (78 % using 15 *fps* and 77 % using 10 *fps*). The analysis shows that the larger distance of the satellites towards the meteors does not significantly influence the capability to observe the same meteors as observed from ground based systems. This is important, if the space based measurements should be combined with ground based observations. However, in this evaluation it is only calculated if one satellite can observe the meteor. The observability of a meteor from both satellites depends on the satellite distance as well as the brightness and position of the meteor. Since the distance between the satellites is different from the distance between the *CILBO* stations, the observability with both satellites can not be calculated without making assumptions about the position of the meteor.

Trajectory accuracy In order to evaluate the trajectory accuracy, the trajectory simulation based on *MOTS* is used. But instead of using constant meteor properties, the properties of the detected meteors from the *SWARMSv2* simulation, which in turn is based on average meteor properties of the *CILBO* data base, is used. This gives a more realistic estimation of the expected trajectory error.

Thus, a *SWARMSv2* simulation is run with the selected instrument parameters (frame rate 10 *fps* and 15 *fps*, focal length 17 mm, binning 2) and the high and low *FACIS* orbit. Based on the mean and standard deviation of the meteor properties from the *CILBO* data base the meteors for the *SWARMSv2* simulation are generated. The number of generated meteors is calculated according to the Grün model. Satellite distance and tilt angle are chosen as defined in Section 5.1.2. Within the simulation, all properties of the detected meteors (distance, velocity, position) as well as satellite properties (position, distance) are exported into a file.

This file is imported into the the trajectory simulation. The meteor and satellite data is used to evaluate the trajectory error for each meteor from the *SWARMSv2* simulation using the geometry

(satellite and meteor position) from *SWARMSv2* as well. All other simulation parameters, especially the satellite bus parameters (attitude and position knowledge as well as clock error), are unchanged (see Table A.5). For this analysis an attitude knowledge accuracy of sightly larger than 12″ but below 13.2″ is used according to the requirement stated in Table 5.1. Accordingly, a GPS accuracy of 170 m and a time accuracy between 10 ms to 20 ms is assumed.

The results are shown in Table 5.13. As can be seen, the mean

Table 5.13.: Expected trajectory error from trajectory simulation based on *MOTS* and meteor properties from *SWARMSv2* simulation

Tilt angle	Median trajectory error (m)	Mean trajectory error (m)	Max trajectory error (m)	Min trajectory error (m)	Number of meteors from SWARMS
10 *fps* 565 km	404	540	9915	285	366
15 *fps* 565 km	403	562	8237	280	445
10 *fps* 300 km	424	599	8006	301	150
15 *fps* 300 km	422	662	6793	305	175

error for both orbits is between 540 m and 660 m, the higher orbit has a slightly lower error. These values are in agreement with previous trajectory simulations as shown in Section 3.5.2. The mean error is increased, since some combinations of meteor properties result in high trajectory errors up to 10 km. In Figure 5.1, the expected trajectory error for each meteor observed is shown. Usually, the trajectory error is as expected, but some meteors cause an expectationally high trajectory error (up to several km). Therefore, for most observed meteors the trajectory error should be below the mean value. This is also the case for the 300 km orbit and the 15 *fps* frame rate (see Appendix Section A.12 for the according plots).

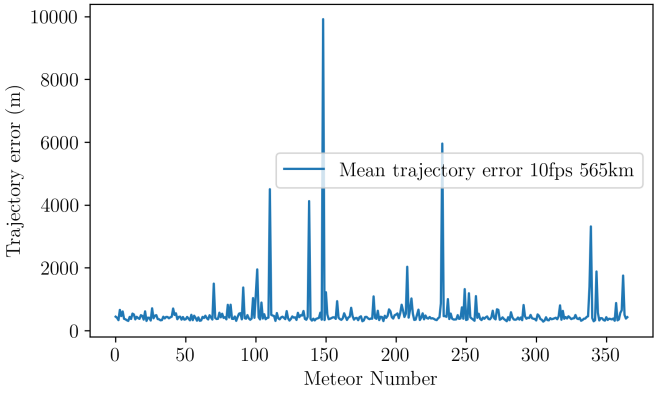


Figure 5.1.: Expected trajectory error for each meteor observed according to trajectory simulation based on *SWARMSv2* meteor properties for the 565 km orbit and 10 *fps*

5.2. SOURCE: Instrument design and development

The *SOURCE* mission serves as a technology demonstration for *FACIS* to show the feasibility of space based meteor observation and to improve the *MetDet algorithm*. If possible, the same camera should be used for both missions, to reduce development effort. Secondary, the mission should also generate useful scientific data by observing as much meteors as possible. However, since *SOURCE* is a *CubeSat*, the satellite bus has much more constraints in terms of available power budget, allowed heat dissipation and data downlink budget. On top of that, the meteor observation camera on *SOURCE*, called Meteor observation, Star and Horizon tracking Camera (MeSHCam), is not the only payload on the satellite and must share the limited resources of the satellite bus.

Furthermore, Meteor observation, Star and Horizon tracking Camera (MeSHCam) is also used as a star tracker, to demonstrate

the possibility of a low cost star tracker based on COTS parts. This aspect must be considered in the software design as well as in the component selection.

5.2.1. Instrument components

As mentioned above, the instrument should use the same components as the *MetObs System* used for *FACIS*. However, some adjustments are necessary, to adapt the instrument to the satellite bus.

Meteor Observation Camera For the *SOURCE* mission the *GenieNano M1920* can be used as well as the *MetObs camera*. In the following, the *MetObs camera* for *SOURCE* is called MeSHCam. As mentioned, the sensor used in this camera has a high quantum efficiency, low noise and large pixel required for meteor observations (see Table 5.2).

However, a different lens must be used, since MeSHCam is also used as an experimental star tracker. For a star tracker, the FOV should be small enough, to reduce the probability of blinding by bright objects (Sun, Earth, Moon) and reduce distortions, which occur at smaller focal length. On the other side, the FOV should not be too small, in order to image enough stars.

Regarding meteor observations, a large FOV is desirable, to increase the number of observed meteors.

Thus, the ruggedized *Schneider Kreuznach Cinegon 1.4/12* lens was chosen with a focal length of 12 mm. The FOV and focal length result in a large area covered to increase the number of observable meteors and stars. On the other hand, the FOV is still small enough to avoid pointing the FOV over the horizon due to the limited ACS precision of the *SOURCE* satellite (see Table 5.14). The small f-number allows for a wide aperture and thus the collection of more photons. Furthermore, the space in a *CubeSat* is limited, so a shorter lens (lower focal length) is desirable.

Table 5.14.: SOURCE satellite bus and mission parameters

Property	Value
Orbit	Sun-synchronous/ISS orbit
Orbit altitude	300 km to 500 km
Mission duration	$\sim$ 1 *year*
Pointing accuracy	5°
Maximum power generation	30 W
Payload data rate per day	100 MB

Chosen parameters The *Sensitivity* simulations with the *GenieNano M1920* and *Schneider Kreuznach Cinegon 1.4/12* were done with the settings mentioned in Table 5.15. The rationale behind those choices are as follows: The f-number is set as low as possible, to open the aperture as wide as possible and thus collect as much light as possible.

Table 5.15.: Chosen parameter values of the *GenieNano M1920* and *Schneider Kreuznach Cinegon 1.4/12* for the *SOURCE* mission

Parameter	Value
f-number F#	1.4
Exposure time t_{exp}	$\frac{1}{6}$ s
Frame rate	6 *fps*
Binning	2, horizontal and vertical

The frame rate is a trade off. Generally, a high frame rate is desirable to analyse the light emission of a meteor during re-entry. However, a high frame rate generates more data to be downlinked and the meteor detection algorithm needs to process more data. Therefore, the frame rate is selected as high as possible while still allowing for reasonable amount of image data and process time.

The SPOSH camera uses a similar frame rate (3 *fps* (see [10]), while some other observation systems use higher frame rates.

Binning 2 is selected for two reasons: First, the camera is more sensitive to faint meteors, since 4 pixel values are summarized. Second, the resolution is reduced, reducing the amount of data to be downlinked and processed. As a drawback, the reduced resolution also reduces the accuracy of meteor position determination. Since *SOURCE* should only measure the meteor flux and demonstrate the algorithm, this is not critical.

With those settings, the camera has a stellar limiting magnitude of 7.39, the limiting magnitude for a typical meteor and a 500 km orbit is 2.82. It has to be mentioned, that those parameters (with the exception of the f-number) can be changed during the mission.

Meteor Observation Controller The *MetObs controller* of the *SOURCE* mission is called PLOC (see Table 5.16) and is based on the same *Trenz TE0720-03-1CFA* as the *FACIS MetObs controller*. The board is mounted on the so called port expander, a circuit board designed by the *SOURCE* team.

Table 5.16.: Data sheet *MetObs controller* for *SOURCE*

Parameter	Value
Mass memory (SD Card)	32 GB
Working memory	1 GB
CPU	ARM dual-core Cortex-A9 MPCore
Interfaces	Gigabit Ethernet, USB 2.0, I2C, SPI
Power consumption	$\sim$ 3.5 W

5.2.2. Observation concept and and scientific performance

The observation concept and estimation of the scientific performance for the *SOURCE* mission is developed using the same approach as described Section 5.1.2. The observation concept is done using the *GenieNano M1920* equipped with *Schneider Kreuznach Cinegon 1.4/12* as *MetObs camera* and the *Trenz TE0720-03-1CFA* as *MetObs controller*. The analysis is done for a 500 km orbit, which is the highest possible orbit for *SOURCE*. It is assumed that the orbit has a duration of 90 min with 30 min eclipse time. Since the *SOURCE* satellite does not provide a proper nadir pointing mode, a tilt angle of 20° is assumed. Furthermore, due to limitations in the heat dissipation, the camera can only be operated for a maximum of 30 min per day. This number can change in the further development phases, therefore different maximum operation times are considered.

Observation concept The same Python script as used and explained in Section 5.1.2 is used here. The adapted settings are shown in Table A.23, see Section 5.1.2 for more details about the parameters.

As shown in Table 5.17, more than 3 observations each lasting 30 min are feasible while still being able to process the images during one day (using 5.1 of the avialable 16 orbits per day).

Table 5.17.: Time budget for different observation times at 6 *fps* in a 500 km orbit. The selected option is marked in grey.

Observation time (min)	Number of observations	Observation time total (min)	Downlink time (min)	Process time acc. (min)	Image transfer time (min)	Orientation time (min)	Total time (min)	Total orbits
10	3	30	90	16	15	1.8	183	2
15	3	45	135	24	15	2.6	252	2.8
20	3	60	180	32	15	3.5	321	3.6
25	3	75	225	40	15	4.4	389	4.3
30	3	90	270	48	15	5.3	458	5.1

However, the longest observation time is currently 10 min per observation due to the limited thermal budget. Therefore, the first scenario (3 observations, each 10 min per day) is analysed.

Next, the data budget must be analysed, as shown in Table 5.18. For the same scenario, 30 min observation time per day, the amount of data to be downlinked is lower than the available data budget (13 MB used from 100 MB). This leaves enough data budget for downlinking images to test and improving the algorithm.

Table 5.18.: Data budget for different observation times at 6 *fps* in a 500 km orbit. The selected option is marked in grey.

Observation time total (min)	Data generated (MB)	Images generated	Images with meteor	Data to downlink (MB)	Detected meteors
30	4320	10800	33	13	4.5
45	6480	16200	49.5	20	6.75
60	8640	21600	66	26	9
75	10800	27000	82.5	33	11.25
90	12960	32400	99	40	13.5

Regarding the power budget (given in Table 5.19), 10 Wh/*day* are needed to operate *MetObs System* for 30 min/*day*. This power only takes into account MeSHCam and *MetObs controller*, not the power needed to operate the satellite bus (e. g. the downlink system). A detailed analysis taking into account the complete satellite bus is necessary in a later phase. Furthermore, it needs to be analysed, if the time between the observations is sufficient to recharge the batteries.

Scientific performance Using the Grün model and the *SWARMSv2* simulation, the number of observed meteors is 4.5 *meteors*/d. However, since the model may not be accurate and the *MetDet algorithm*

Table 5.19.: Power budget for different observation times at 6 *fps* in a 500 km orbit. The selected option is marked in grey.

Observation time total (min)	Energy consumption PLOC (Wh)	Energy consumption camera (Wh)	Energy consumption total (Wh)
30	8	2	10
45	12	3	15
60	16	4	20
75	20	5	25
90	24	6	30

may not detect all meteors, the actual rate will be lower. Therefore, the scientific output is reduced, which is mainly due to the limitations of the satellite bus (thermal budget, power budget and downlink capacity).

The scientific performance can be increased by observing during major meteor showers. During a meteor shower the observation time could be increased, by observing e. g. 5 d around the shower peak and delaying the processing and downlink of the images after the shower. This reduces the thermal load on the satellite and the power needed on one day. Possible scenarios include a 3 d observation with 10 observations per day as well as 5 days of observation with 10 or 15 observations per day. The resulting energy consumption and required downlink capacity are shown in Table A.24.

Finally, *SOURCE* is an important demonstrator mission, to show the feasibility of space based meteor observation as well as to qualify the *MetObs System* and the *MetDet algorithm*. Thus, the scientific output is not prioritized in this mission.

5.2.3. Instrument development

The focus of this thesis is the mission analysis and instrument design for a space borne meteor observation mission. However, the instrument development is the next step towards realising the mission. Thus, here the current development status of the *SOURCE* instrument should be given. The development is divided into instrument qualification and testing and software design.

Software design

The general software outline is shown in Figure 5.2. The operating system used is Linux distribution Ubuntu 18.04 LTS, which allows to make use of the standard software repository to install necessary packages (e. g. required libraries needed for the image processing and camera control). Since the operating system is running on an embedded board, it is necessary to build a custom Linux kernel as well as the boot loader responsible for starting the system. This is done by configuring the hardware and using the hardware description to compile a custom Linux kernel with the required functions. The operating system runs the Autostart software as a system service, needed to start the PLOC software. The PLOC software is the term of the main software used for camera control and image processing.

As a basis for the PLOC software the Flight Software Framework (FSFW) developed at the IRS is used. More details about the FSFW can be found in [88]. The usage of FSFW allows for a faster software development, since the framework provides different functions and interfaces required for a satellite software. Furthermore, the CVB framework is used for camera control and *OpenCV* is used for image processing.

Currently, the main parts of the software are implemented and working, this includes basics like the start of the PLOC software using the Autostart software, controlling the file system as well as controlling the camera with the MeSHCam handler using commands. Furthermore, the software runs on the development board

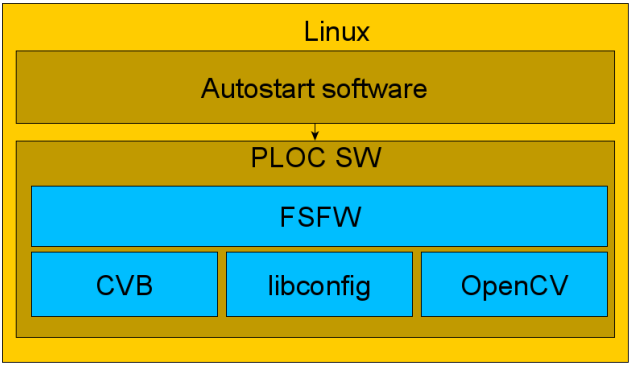


Figure 5.2.: Overview of SOURCE instrument software design

and can be remotely controlled and debugged. Regarding the *MetDet algorithm*, the handler is defined, but most functions still need to be implemented.

Instrument qualification and testing

The instrument qualification includes performing thermal vacuum tests of the cameras as well as of the PLOC development board. These tests proved that the instrument is working in vacuum and at all expected temperatures during the *SOURCE* mission. Furthermore, images were taken in vacuum in order to evaluate the effect of vacuum on the focus. No effect of the vacuum conditions on the focus could be determined.

Additionally, several camera characterisation tests were performed. This includes *EMVA 1288* tests to determine the linearity of the sensor, its quantum efficiency as well as the dark current and the temperature dependence of the dark current. Also, a radiometric calibration was performed to measure the relation between irradiance and measured pixel value. Finally, the geometric calibration is performed to evaluate the lens distortion.

All in all, the camera could be successfully qualified. Pending tests are vibration and shock tests, which are scheduled within the *SOURCE* project.

6. Summary and Outlook

The objective of this thesis was evaluate the possibility to design a (stereoscopic) space based meteor observation mission. Therefore, requirements for the instrument, the satellite bus and the satellite formation were derived. Different simulations were developed to derive requirements and to answer the question, which scientific performance of a space based stereoscopic meteor observation using small satellites is expected. It could be shown, that such a mission based on small satellites is feasible and able to generate a significant scientific output.

Two missions, were designed and used as an example: The first small satellite mission is called *FACIS*, which consists of two satellites for stereoscopic meteor observation. The second mission *SOURCE*, consisting of one *CubeSat*, is a demonstrator mission to test the instrument and the algorithm.

Furthermore, in the scope of the *SOURCE* mission a demonstrator instrument could be designed, built and tested. This includes the detection meteor algorithm as an integral part of the instrument design, since not all images can be downlinked and thus onboard detection of meteors is needed.

This thesis is a major step towards a spaceborne meteor observation mission, because for the first time a detailed analysis of satellite bus and formation parameters was performed. Also for the first time, a working prototype of a meteor detection algorithm could be developed, which is crucial for any space based meteor observation mission.

6.1. Requirements for space based meteor observations

In order to maximize the scientific output of a space based meteor observation mission, requirements must be defined. The requirements can be divided into orbit and formation, satellite bus as well as instrument requirements. However, most aspects are interdependent and can not be considered individually. All aspects must be traded off for a working mission design.

Orbit and formation Regarding a single satellite dedicated to meteor flux measurements, a higher orbit is desirable to increase the number of observations.

For a stereoscopic meteor observation, the orbit altitude also influences the ideal distance between the satellites and their tilt angle, to ensure an overlapping FOV of both instruments. Although a higher orbit increases the number of detected meteors, it also influences the types of meteors detected. A higher orbit requires a larger tilt angle and thus distance between the satellites. This results in less faint meteors being detected. For stereoscopic meteor observation this is acceptable, since more bright meteors can be observed. Thus, a high orbit with large tilt angle should be used, to observe as much meteors as possible. In case of an 565 km orbit and a lens with a focal length of 17 mm the tilt angle should be 36°.

Another aspect of the satellite distance is the trajectory determination accuracy. A minimal distance is needed, to ensure trajectory determination is possible. However, according to the simulation, the error on trajectory determination is low for a certain range of distances. Therefore, it is possible to chose a distance at which the observed area is maximized and at the same time the error on the trajectory determination is low.

Satellite bus The satellite bus has a huge influence on the scientific output. Mainly for *CubeSats*, like *SOURCE*, this is due to the

limited available power and thermal constraints. Both limit the time the instrument can be operated. But even with a limited time, e. g. 30 min per day, meteors can be observed during the 1 *year SOURCE* mission. For larger satellites, like *FACIS*, the instrument can be operated in all eclipse phases and thus generate more data. Here, the scientific output is limited by the ability of the algorithm to process the images fast enough.

The algorithm is needed since the downlink capacity is limited. The amount of data depends on the observation time, a minimal amount of 20 MB/d is needed for the *SOURCE* mission. However, with longer observation times, more data is generated. Furthermore, a larger data budget also allows to improve the algorithm during in orbit operation by downlinking all images generated. An amount of 300 MB/d allows to downlink all generated data for the *FACIS* mission, while still having some budget available to downlink images not containing a meteor. However, the data budget necessary depends of course on the used camera and algorithm performance.

To sum up, the power, thermal and downlink budget have a huge influence on the scientific performance. The larger all budgets, the larger the amount of useful data. However, even with a limited satellite bus a useful scientific mission is possible.

Another bus parameter is the accuracy of the ACS. Here, the requirements depend on the scientific objective: For a mission measuring the meteor flux, the attitude control is less important. It just has to be ensured, the satellite points the instrument towards the Earth and limits its rotation to avoid image blurring.

For a stereoscopic meteor observation, the attitude and position knowledge as well as clock accuracy are important. Especially the attitude knowledge has a huge influence on the accuracy of the meteor trajectory determination. This is the biggest challenge for a small satellite, since high precision attitude sensors must be available and properly integrated into the system. Regarding the clock accuracy, some development effort is necessary to provide the time

from the GPS receiver to the PLOC. The satellite position knowledge error should be below 170 m, the attitude knowledge error below 12″ and the clock accuracy below 15 ms.

To sum up to satellite bus requirements, all can be fulfilled by small satellites. However, for a *CubeSat* the challenge is to provide enough power and a good thermal design, to allow for long observation times. For a stereoscopic observation mission, the challenge for the satellite bus is to provide an accurate attitude measurement, to reduce the trajectory error below 450 m. Furthermore, development effort is needed to provide an accurate timestamp for the images taken.

Instrument The instrument consists of a camera, a processing unit and a detection algorithm. The algorithm is the critical part of the instrument. It has a huge influence on the scientific output, given by the detection performance and the speed performance. The algorithm needs to detect as much meteors as possible without a significant amount of false positives. Furthermore, the detection algorithm needs to run fast, in order to allow for longer observation time, since all images must be processed before the next observations starts. All these requirements must be fulfilled with the limited resources of a satellite. The requirement for the algorithm is to processes one image in 0.5 s.

Regarding the camera, a high quantum efficiency, low dark noise and large pixel size are needed to increase the Signal to Noise Ratio. The lens should have a low f-number to increase the area collecting light. The Field of View depends on the scientific objective: For meteor flux determination a large FOV is needed, to observe as many meteors as possible. For stereoscopic observations, a smaller FOV is more suitable to increase the accuracy of meteor velocity determination.

Depending on the available satellite resources, two instrument per satellite could be used for a stereoscopic meteor observation mission. One instrument can be dedicated to meteor flux measurements using

a wide FOV and one is dedicated for stereoscopic observations using a smaller FOV. This would allow to combine both measurements and increase the scientific output of the mission.

6.2. Scientific performance

As mentioned before, the satellite bus, the instrument and the algorithm performance influence the scientific output. All three aspects must be taken into account to develop an observation concept and to estimate the scientific performance.

For the *SOURCE* mission, the resources of the satellite (power, thermal and downlink budget) are limited due to the small form factor. Therefore, these resources limit the scientific output. Due to thermal constraints and available power, the instrument can only be operated for 30 min/d. With the designed instrument it is possible to observe up to 4.5 *meteors*/d. However, this number is only an estimate and not all meteors may be detected by the algorithm.

For the *FACIS* mission, the algorithm speed performance is critical. In order to observe meteors during all available eclipse times, the algorithm needs to process all images during the sun phase. In this case, the algorithm speed needs to be improved. However, even with the current status of the algorithm using the PLOC from *SOURCE* it is possible to observe a significant amount of meteors (up to 27 *meteors*/d).

The trajectory accuracy depends mainly on the knowledge of satellite attitude and an accurate time stamp for each image. For a small satellite it is possible, but challenging, to achieve trajectory errors below 500 m.

The accuracy from a space based measurement is reduced compared to ground based systems. This is due to the fact that additional measurements are necessary to determine the trajectory (e. g. the attitude and position of the satellite). The additional measurements and the movement of the satellite during observation reduce the

accuracy. However, the advantages prevail and a space based meteor observation mission would be a valuable complement to ground based observations.

6.3. Summary of instrument development

In the scope of the *SOURCE* mission an instrument was developed, including component selection, environmental testing and component qualification, software and algorithm development. The instrument (*MetObs System*) consists of a *MetObs camera* (*GenieNano M1920* and *Schneider Kreuznach Cinegon 1.4/12*), as well as algorithm (*MetDet algorithm*) and processing unit (*MetObs controller*). The components were selected based on the previous described simulations as well as satellite requirements. However, the instrument itself (camera and lens) is less complex and existing components can be used.

The instrument was build, tested and is being integrated in the satellite laboratory model (flat sat). Multiple tests were conducted to qualify and characterize the instrument. These tests include geometric tests to determine lens parameters, *EMVA 1288* tests to determine the camera properties (e. g. quantum efficiency), thermal vacuum testing for space qualifications, focus tests in vacuum as well as real sky tests and tests in the planetarium to observe meteors and stars.

In the scope of the instrument development, the software for the *MetObs controller* was developed, including setting up the operating system and configuring the FPGA. Also, the instrument software was developed and tested, which is responsible for processing commands send from the ground segment, controlling the camera and managing image processing and storing. The software is based on the FSFW and could be tested on the engineering model of the *MetObs controller*.

Furthermore, the algorithm, a critical part of the software, was developed and tested in a Testbed. The algorithm influences the

scientific output of the mission via the detection performance and the speed performance. Without the algorithm, the mission would not be possible due to limitations in the downlink capacity. The developed algorithm *SpaceMEDAL* is able to detect most meteors with a low number of false positives ($F - measure$ is 0.885). Furthermore, the algorithm runs fast enough (0.8 s per image) to allow for a significant amount of observation time. This is possible by making use of outsourcing resource intensive functions of the *OpenCV* framework into the FPGA.

The algorithm is based on optical flow calculations using the *OpenCV* framework. The main moving direction caused by satellite movement is identified, all parts moving in this directions are removed. A *blob* detection is applied to the remaining part, to identify meteor candidates. The candidates are filtered, e. g. by brightness, area and movement magnitude. Further filters are applied to confirm the meteor in multiple frames, e. g. the moving distances of *blobs* between consecutive frames.

The algorithm was tested in multiple stages, a test environment was set up for this purpose. The two main components of this test environment are the artificial meteor simulation and the Testbed. The artificial meteor simulation *ArtMESS* is used to generate image test data with varying meteor properties and backgrounds. This test data is either processed directly or displayed on the Testbed and imaged by the instrument.

The development of the algorithm running on limited resources is a crucial part for any space based meteor observation mission. The development also enables progress in the area of complex onboard processing of data with limited resources. This is beneficial for future deep space missions, which also rely on onboard processing and selecting data to be transmitted.

6.4. Conclusion and Outlook

In this thesis it could be shown, that a space based meteor observation mission using small satellites is possible. The work done in this thesis is a major step towards spaceborne meteor observation mission, because for the first time a detailed analysis of satellite bus and formation parameters was done and a working prototype of a spaceborne meteor detection algorithm could be developed.

The thesis used two example missions to address three main aspects of the mission: First *SOURCE*, a small *CubeSat* dedicated to demonstrate space based meteor observation, test the instrument and measure the meteor flux. Second *FACIS*, a stereoscopic meteor observation mission using a formation of two satellites. The main aspects addressed in this thesis are the instrument design, the satellite bus and formation requirements and the algorithm development. All three aspects could be successfully revised and the mission and the instrument could be developed. Furthermore, a demonstrator instrument for the *SOURCE* mission could be built, including the development of the detection algorithm.

This thesis is a significant step towards the first dedicated spaceborne meteor observation missions. Especially the progress in the development of the detection algorithm is crucial for all planned missions. However, some tasks still need to be done:

First, the algorithm *SpaceMEDAL* can be improved, by conducting more tests and thus increasing the detection rate. Currently, *SpaceMEDAL* processes one image in 0.8 s, but 0.5 s are realistic with further improvements. Additionally, the meteor simulation should be improved and made even more realistic. Especially the radiometry of the meteor re-entry should be investigated and a physical model of meteor re-entry should be implemented.

Furthermore, for the *SOURCE* mission, the PLOC software needs to be further developed and tested. This includes tests together with the OBC as well as complete hardware in the loop tests inside the Testbed. While this primarily concerns *SOURCE*, the developed software can be used for *FACIS* as well with minor adaptions.

For the *FACIS* mission, the trajectory error should be analysed in more detail. Especially the trajectory back propagation to an orbit of the meteoroid and the effect satellite bus parameters on the determined orbit should be analysed.

Finally, if the principle of space based meteor observation using small satellites and an onboard detection algorithm could be proven, successor missions could be possible. A scientifically interesting mission would be to observe the spectrum of a meteor, to analyse the composition of meteors. From a satellite, the spectrum could be observed in the UV spectrum, which is blocked by the atmosphere and thus not observable from ground based systems. This would give a detailed view of the composition of meteors, e. g. the detection of Carbon in the UV spectrum, allowing identification of extra-terrestrial material.

Bibliography

[1] "Thaddeus Cesari". *NASA JWST Blog*. https://blogs.nasa.gov/webb/2022/06/08/webb-engineered-to-endure-micrometeoroid-impacts/. Accessed: 10.06.2022. 2022 (cit. on p. 1).

[2] Alexis Bouquet et al. "Simulation of the capabilities of an orbiter for monitoring the entry of interplanetary matter into the terrestrial atmosphere". In: *Planetary and Space Science* 103 (2014), pp. 238–249. URL: https://doi.org/10.1016/j.pss.2014.09.001 (cit. on pp. 2, 36, 44, 74, 75, 87, 91, 95).

[3] Nicolas Rambaux et al. "Detection of spectral UV from meteors by a nanosatellite". In: *Proceedings of the International Meteor Conference, Giron, France*. 2014, pp. 182–184 (cit. on p. 2).

[4] Peter Jenniskens and Petrus Matheus Marie Jenniskens. *Meteor showers and their parent comets*. Cambridge University Press, 2006 (cit. on pp. 2, 4, 15–18, 20, 48, 50).

[5] The SOURCE Team. *Payload documentation SRC-DOC-PL-150-003*. Report. KSat e.V., 2021 (cit. on p. 5).

[6] Fabian Hufgard. "Phase A study of the academic satellite formation „Formation for Analysis of Cosmic partIcleS" (FACIS) (IRS-17-S-014)". German. Master Thesis. Stuttgart, 2017 (cit. on pp. 8, 9, 204).

[7] Fabian Hufgard et al. "Preliminary study of an academic micro-satellite formation mission for meteoroid trajectories

determination and dust mass flow measurement". In: *presented at the Small Satellites Systems and Services Symposium*. 2018 (cit. on p. 8).

[8] T Arai et al. "International Space Station-Based Meteor Observation Project: Initial Results". In: *Lunar and Planetary Science Conference*. 1964. 2017, p. 3034 (cit. on p. 9).

[9] Nicolas Rambaux et al. "Meteorix: a cubesat mission dedicated to the detection of meteors and space debris". In: *1st ESA NEO and Debris Detection Conference*. ESA Space Safety Programme Office. 2019, 9–p (cit. on p. 10).

[10] J Oberst et al. "The Smart Panoramic Optical Sensor Head (SPOSH)—a camera for observations of transient luminous events on planetary night sides". In: *Planetary and Space Science* 59.1 (2011), pp. 1–9 (cit. on pp. 10, 95, 229).

[11] Victor Hernandez et al. "SWIMSat: Space Weather and Meteor Impact Monitoring using a Low-Cost 6U CubeSat". In: *Collection of Technical Papers - AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference* (2016) (cit. on p. 10).

[12] Ravi teja Nallapu, Himangshu Kalita, and Jekanthan Thangavelautham. "On-Orbit Meteor Impact Monitoring Using CubeSat Swarms". In: *The Advanced Maui Optical and Space Surveillance Technologies Conference* (2018) (cit. on p. 11).

[13] Apostolos Christou et al. "Orbital observations of meteors in the Martian atmosphere using the SPOSH camera". In: *Planetary and Space Science* 60.1 (2012), pp. 229–235. URL: https://elib.dlr.de/73081/ (cit. on p. 15).

[14] Zdeněk Ceplecha et al. "Meteor phenomena and bodies". In: *Space Science Reviews* 84.3 (1998), pp. 327–471. DOI: 10.1023/A:1005069928850 (cit. on p. 16).

[15] Ulf von Zahn. "The total mass flux of meteoroids into the Earth's upper atmosphere". In: *17th ESA Symposium on European Rocket and Balloon Programmes and Related Research*. Ed. by Barbara Warmbein. Vol. 590. ESA Special Publication. Aug. 2005, pp. 33–39 (cit. on p. 17).

[16] Martin Beech and Duncan Steel. "On the definition of the term meteoroid". In: *Quarterly Journal of the Royal Astronomical Society* 36 (1995), p. 281 (cit. on p. 17).

[17] Althea V. Moorhead et al. "Fully correcting the meteor speed distribution for radar observing biases". In: *Planetary and Space Science* 143 (2017). SI:Meteoroids 2016, pp. 209–217. ISSN: 0032-0633. DOI: https://doi.org/10.1016/j.pss.2017.02.002. URL: https://www.sciencedirect.com/science/article/pii/S0032063316302689 (cit. on p. 17).

[18] Rothstein, Dave. *What is apparent magnitude?* https://web.archive.org/web/20150117205438/http://curious.astro.cornell.edu/question.php?number=569. Accessed: 13.04.2021 (cit. on p. 17).

[19] D. W. R. [Verfasser/in] McKinley, ed. *Meteor science and engineering*. Englisch. McGraw-Hill series in engineering sciences. Bibliogr. S. 284 - 296. New York [u.a.]: McGraw-Hill, 1961, IX, 309 Seiten (cit. on p. 18).

[20] Michael S Bessell. "Standard photometric systems". In: *Annu. Rev. Astron. Astrophys.* 43 (2005), pp. 293–336 (cit. on p. 18).

[21] Juergen Rendtel. "Visual Sporadic Meteor Rates". In: *WGN, Journal of the International Meteor Organization* (June 2006) (cit. on p. 21).

[22] Juergen Rendtel. "Sporadic meteors". In: *Proceedings of the International Meteor Conference, 25th IMC, Roden, Netherlands, 2006*. Ed. by Felix Bettonvil and Javor Kac. Dec. 2007, pp. 99–103 (cit. on p. 21).

[23] Colas, F. et al. "FRIPON: a worldwide network to track incoming meteoroids". In: *A&A* 644 (2020), A53. DOI: 10.1051/0004-6361/202038649. URL: https://doi.org/10.1051/0004-6361/202038649 (cit. on p. 21).

[24] François Colas et al. "French fireball network FRIPON". In: *Proceedings of the International Meteor Conference, Mistelbach, Austria*. 2015, pp. 27–30 (cit. on p. 21).

[25] P Jenniskens et al. "CAMS: Cameras for Allsky Meteor Surveillance to establish minor meteor showers". In: *Icarus* 216 (Nov. 2011), pp. 40–61. DOI: 10.1016/j.icarus.2011.08.012 (cit. on p. 21).

[26] Detlef Koschny et al. "A double-station meteor camera set-up in the Canary Islands – CILBO". In: *Geoscientific Instrumentation, Methods and Data Systems* 2 (Dec. 2013). DOI: 10.5194/gi-2-339-2013 (cit. on p. 21).

[27] D Koschny et al. "A double-station meteor camera set-up in the Canary Islands–CILBO". In: *Geoscientific Instrumentation, Methods and Data Systems* 2.2 (2013), pp. 339–348 (cit. on p. 22).

[28] Esther Drolshagen et al. "Meteor velocity distribution from CILBO double station video camera data". In: *Proceedings of the International Meteor Conference, edited by: Rault, J.-L. and Roggemans, P., IMO*. Vol. 1622. 2014, p. 1821 (cit. on p. 22).

[29] Ian Halliday, Arthur A Griffin, and Alan T Blackwell. "Detailed data for 259 fireballs from the Canadian camera network and inferences concerning the influx of large meteoroids". In: *Meteoritics & Planetary Science* 31.2 (1996), pp. 185–217 (cit. on pp. 24, 88).

[30] E Grün et al. "Collisional balance of the meteoritic complex". In: *Icarus* 62.2 (1985), pp. 244–272 (cit. on pp. 24, 88).

[31] Jana Kretschmer et al. "De-biasing CILBO meteor observational data to mass fluxes". In: *Proceedings of the International Meteor Conference, edited by: Rault, J.-L. and Roggemans, P., IMO*. 2015, pp. 209–213 (cit. on p. 26).

[32] Theresa Ott et al. "Meteoroid flux determination using image intensified video camera data from the CILBO double station". In: *Proceedings of the International Meteor Conference, edited by: Rault, J.-L. and Roggemans, P., IMO*. Vol. 2329. 2014, p. 1821 (cit. on p. 26).

[33] Thomas Albin et al. "Influence of the pointing direction and detector sensitivity variations on the detection rate of a double station meteor camera". In: *International Meteor Conference Mistelbach, Austria*. 2015, pp. 226–232 (cit. on pp. 27, 119).

[34] Thomas Albin et al. "Analysis of the technical biases of meteor video cameras used in the CILBO system". In: *Geoscientific Instrumentation, Methods and Data Systems* 6.1 (2017), pp. 125–140 (cit. on pp. 27, 73, 119).

[35] Thomas Albin. "Machine learning and Monte Carlo based data analysis methods in cosmic dust research". Englisch. Hochschulschrift. Stuttgart, 2019, xx, 246 Seiten (cit. on p. 27).

[36] Esther Drolshagen et al. "Meteor velocity distribution from CILBO double station video camera dat". In: *Proceedings of the International Meteor Conference, edited by: Rault, J.-L. and Roggemans, P., IMO*. Vol. 1622. 2014, p. 1821 (cit. on p. 28).

[37] European Cooperation for Space Standardization Space Engineering. *Space Envrionment ECSS-E-ST-10-04C*. ECSS Standard. 2008 (cit. on pp. 28, 80).

[38] D. Koschny and J. Diaz del Rio. "Meteor Orbit and Trajectory Software (MOTS) - Determining the Position of a Meteor with Respect to the Earth Using Data Collected with the Software MetRec". In: *WGN, Journal of the International Meteor*

Organization 30.4 (Sept. 2002), pp. 87–101 (cit. on pp. 36, 99, 100).

[39] European Machine Vision Association EMVA. *EMVA1288*. https://www.emva.org/wp-content/uploads/EMVA1288-3.1a.pdf. Accessed: 26.09.2018 (cit. on pp. 36, 37, 40).

[40] Thorlabs, Inc. *Camera Noise and Temperature*. https://www.thorlabs.com/newgrouppage9_pf.cfm?guide=10&category_id=&objectgroup_id=10773. Accessed: 09.12.2020 (cit. on p. 37).

[41] Stemmer Imaging. *CCD oder CMOS: Können CMOS-Sensoren in allen Fällen CCDs ersetzen?* https://www.stemmer-imaging.com/media/uploads/cameras/avt/de/de-Allied-Vision-WhitePaper-CCD-vs-CMOS-0416-KAVTO115-201604.pdf. Accessed: 09.12.2020 (cit. on p. 41).

[42] Jacob, Atik Cameras. *Binnning: The differences between CMOS and CCD*. https://www.atik-cameras.com/news/binnning-the-differences-between-cmos-and-ccd/. Accessed: 10.12.2020 (cit. on p. 42).

[43] TELEDYNE PHOTOMETRICS. *Imaging topics: Binning*. https://www.photometrics.com/learn/imaging-topics/binning-2. Accessed: 10.12.2020 (cit. on p. 42).

[44] George Joseph. *Building earth observation cameras*. CRC press, 2015 (cit. on pp. 43, 47).

[45] Teledynedalsa. *GenieNano User Manual*. http://info.teledynedalsa.com/acton/ppform/14932/0017/f-054e. Accessed: 26.09.2018 (cit. on pp. 50, 51, 59, 207).

[46] Leibinz-Institut für Astropyhsik Potsdam. *Johnson Filter Data*. http://www.aip.de/en/research/facilities/stella/instruments/data/johnson-ubvri-filter-curves. Accessed: 26.09.2018 (cit. on p. 51).

[47] A Kingery and RC Blaauw. "Determination of the meteor limiting magnitude". In: *Planetary and Space Science* 143 (2017), pp. 67–70 (cit. on p. 53).

[48] Pierantonio Cinzano, Fabio Falchi, and Christopher D Elvidge. "The first world atlas of the artificial night sky brightness". In: *Monthly Notices of the Royal Astronomical Society* 328.3 (2001), pp. 689–707 (cit. on p. 56).

[49] Schneider Kreuznach. *Cinegon 12 lens data sheet*. `https://schneiderkreuznach.com/application/files/1615/4115/0478/1001951_Cinegon_1-4_12.pdf`. Accessed: 08.01.2021 (cit. on pp. 61, 207).

[50] FLIR. *MONOCHROM-KAMERA SENSOR-EVALUIERUNG ver 2018 5*. May 2018 (cit. on p. 64).

[51] "Cubesat Shop managed by ISIS - Innovative Solutions In Space B.V." *SCS Gecko imager*. `https://www.cubesatshop.com/product/scs-gecko-imager/`. Accessed: 07.05.2021 (cit. on pp. 65, 206).

[52] Jona Petri. *Determination of limiting magnitude of GenieNano M1920*. Test report. IRS, 2018 (cit. on p. 71).

[53] Julia Zink. "Analysis of payload requirements for a small satellite formation to determine meteor trajectories and measure cosmic dust (IRS-18-S-129)". Englisch. Bachelor Thesis. Stuttgart, 2018 (cit. on pp. 75, 78, 79).

[54] Jona Petri et al. "Design and test of a COTS based imaging system for stereoscopic meteor observations". In: *Proceedings of IAA Smallsat conference*. 2019 (cit. on p. 80).

[55] Jonas Keim et al. "Commissioning of the Optical Communication Downlink System OSIRISv1 on the University Small Satellite "Flying Laptop"". In: *International Astronautical Congress (IAC), IAC-19,B4,3,2,x54132*. Oct. 2019 (cit. on p. 95).

[56] Steffen Gaisser et al. "Improvements in attitude determination and control of the small satellite Flying Laptop". In: *33rd Annual AIAA/USUConference on Small Satellites, SSC19-III-06* (2019) (cit. on p. 105).

[57] S Eckert et al. "The TET-1 Satellite Bus–A High Reliability Bus for Earth Observation, Scientific and Technology Verification Missions in LEO". In: *Pestana Conference Centre - Funchal, Madeira-Portugal* (2010) (cit. on p. 105).

[58] Denis Vida et al. "Estimating trajectories of meteors: an observational Monte Carlo approach – I. Theory". In: *Monthly Notices of the Royal Astronomical Society* 491.2 (Nov. 2019), pp. 2688–2705. DOI: 10.1093/mnras/stz3160. URL: https://doi.org/10.1093%2Fmnras%2Fstz3160 (cit. on p. 110).

[59] Jona Petri, Julia Zink, and Sabine Klinkner. "Optimizing the scientific output of satellite formation for a stereoscopic meteor observation". In: *International Meteor Conference Bollmannsruh, Germany*. 2019 (cit. on p. 114).

[60] Shan Mignot. "Towards a demonstrator for autonomous object detection on board Gaia". In: *Space Telescopes and Instrumentation 2010: Optical, Infrared, and Millimeter Wave*. Vol. 7731. International Society for Optics and Photonics. 2010, p. 773132 (cit. on p. 136).

[61] Sirko Molau and PS Gural. "A review of video meteor detection and analysis software". In: *WGN, Journal of the International Meteor Organization* 33 (2005), pp. 15–20 (cit. on p. 136).

[62] Peter S Gural. "Fast thresholding options for video meteor imagery to obtain pixel exceedances". In: *WGN, Journal of the International Meteor Organization* 47.5 (2019), pp. 151–155 (cit. on p. 137).

[63] Peter S Gural. "Algorithms and software for meteor detection". In: *Advances in Meteoroid and Meteor Science*. Springer, 2007, pp. 269–275 (cit. on p. 137).

[64] Peter Gural and Damir Segon. “A new meteor detection processing approach for observations collected by the Croatian Meteor Network (CMN)”. In: *WGN, Journal of the International Meteor Organization* 37 (2009), pp. 28–32 (cit. on p. 137).

[65] Huan N Do et al. “Robust foreground segmentation and image registration for optical detection of GEO objects”. In: *Advances in Space Research* 64.3 (2019), pp. 733–746 (cit. on p. 137).

[66] Maxime Millet et al. “Meteorix-A new processing chain for real-time detection and tracking of meteors from space”. In: *WGN, Journal of the International Meteor Organization* 49.6 (2022) (cit. on p. 137).

[67] Maxime Millet et al. “Meteorix: High performance computer vision application for Meteor detection from a CubeSat”. In: *44th Scientific Assembly of the Committee on Space Research (COSPAR)*. 2022 (cit. on p. 137).

[68] "OpenCV team". *OpenCV About*. `https://opencv.org/about/`. Accessed: 25.08.2021. 2021 (cit. on p. 138).

[69] Adrian Kaehler and Gary R. Bradski. *Learning OpenCV 3: computer vision in C++ with the OpenCV library*. Englisch. First edition. Beijing ; Boston ; Farnham ; Sebastopol ; Tokyo: O’Reilly, 2016. ISBN: 9781491937990 (cit. on pp. 138–142, 144, 146, 147).

[70] "Satya Mallick". *Blob Detection Using OpenCV*. `https://learnopencv.com/blob-detection-using-opencv-python-c/`. Accessed: 16.09.2021 (cit. on p. 147).

[71] Jens Plozin. “Entwicklung und Verfizierung von Algorithmen zur Echtzeitdetektion von Meteoren auf einem Satelliten (IRS-19-S-001)”. German. Bachelor Thesis. Stuttgart, 2019 (cit. on p. 149).

[72] Julia Zink. "Test and optimization of an on-board meteor detection algorithm for the CubeSat SOURCE (IRS-21-S-066)". Englisch. Master Thesis. Stuttgart, 2021 (cit. on pp. 150, 153, 175, 179, 182, 191, 193, 197).

[73] Jona Petri, Julia Zink, and Sabine Klinkner. "Hardware accelerated onboard image processing for space-based meteor observation - Concept and Implementation of SpaceMEDAL". In: Vilamoura, Portugal, 2022 (cit. on p. 150).

[74] Philipp Berens. "CircStat: A MATLAB Toolbox for Circular Statistics". In: *Journal of Statistical Software* 31.10 (2009), pp. 1–21. DOI: 10.18637/jss.v031.i10. URL: https://www.jstatsoft.org/index.php/jss/article/view/v031i10 (cit. on p. 153).

[75] T Arai et al. "Meteor observation HDTV camera onboard the international space station". In: *Lunar and Planetary Science Conference*. 1777. 2014, p. 1610 (cit. on pp. 160, 162).

[76] Marcel Liegibel et al. "Meteor observation with the SOURCE CubeSat – Developing a simulation to test on-board meteor detection algorithms". In: Barcelona, Spain, 2022 (cit. on p. 160).

[77] J Borovička, P Spurny, and P Koten. "Atmospheric deceleration and light curves of Draconid meteors and implications for the structure of cometary dust". In: *Astronomy & Astrophysics* 473.2 (2007), pp. 661–672 (cit. on p. 162).

[78] Steven D. Miller et al. "Suomi satellite brings to light a unique frontier of nighttime environmental sensing capabilities". In: *Proceedings of the National Academy of Sciences* 109.39 (2012), pp. 15706–15711. ISSN: 0027-8424. DOI: 10.1073/pnas.1207034109 (cit. on p. 163).

[79] Hoffmann, Philipp and Petri, Jona. *Testbed Calibration for MeSHCam SRC-CAL-310-001-1.0*. Report. KSat e.V., 2021 (cit. on p. 168).

[80] Fabio Falchi et al. "The new world atlas of artificial night sky brightness". In: *Science advances* 2.6 (2016), e1600377 (cit. on p. 199).

[81] Susann Pätschke and Sabine Klinkner. "Towards a highly adaptive software-defined radio transmitter for small satellite platforms". In: *Acta Astronautica* 186 (2021), pp. 50–59 (cit. on p. 204).

[82] JAI. *JAI GO-2400M-PGE User Manual*. https://www.jai.com/downloads/datasheet-go-2400m-pge. Accessed: 02.07.2021 (cit. on p. 205).

[83] FRAMOS GmbH. *SONY CMOS IMX174 and its little brother IMX249*. https://www.framos.com/en/news/sony-cmos-imx174-and-its-little-brother-imx249. Accessed: 02.07.2021 (cit. on p. 205).

[84] Schneider Kreuznach. *Xenoplan 17 lens data sheet*. https://schneiderkreuznach.com/application/files/6415/4115/1580/1001957_Xenoplan_1-4_17.pdf. Accessed: 02.07.2021 (cit. on p. 207).

[85] Schneider Kreuznach. *Xenoplan 23 lens data sheet*. https://schneiderkreuznach.com/application/files/1315/4115/0849/1001917_Xenoplan_1-4_23.pdf. Accessed: 02.07.2021 (cit. on p. 207).

[86] "Trenz Electronic". *TE0720 TRM Revision: v.85*. https://shop.trenz-electronic.de/trenzdownloads/Trenz_Electronic/Modules_and_Module_Carriers/4x5/TE0720/REV03/Documents/TRM-TE0720-03.pdf. Accessed: 14.05.2019. Nov. 2017 (cit. on p. 212).

[87] "Aitech". *A177 – Twister GPGPU Mini Supercomputer*. Rev 2.2. Jan. 2019 (cit. on p. 212).

[88] Bastian Bätz. “Design and implementation of a framework for spacecraft flight software”. Englisch. 1 Online-Ressource (xvi, 235 Seiten). Hochschulschrift. Stuttgart, 2020. URL: http://nbn-resolving.de/urn:nbn:de:bsz:93-opus-ds-112229 (cit. on p. 234).

A. Appendix

A.1. Meteor properties from the CILBO database

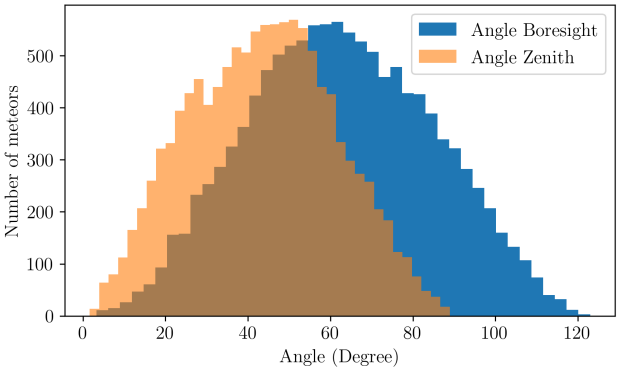


Figure A.1.: Angles to meteor (boresight and zenith) distribution from CILBO data

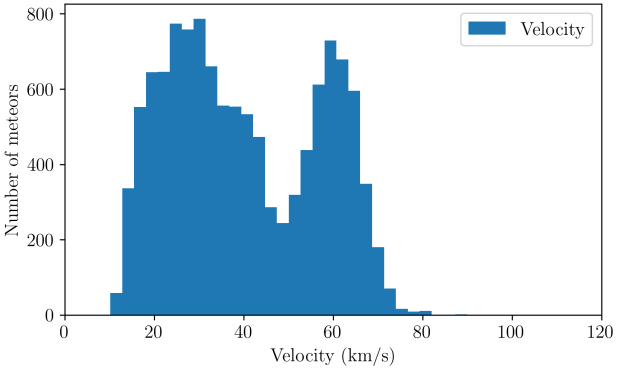


Figure A.2.: Velocity of meteor distribution from CILBO data

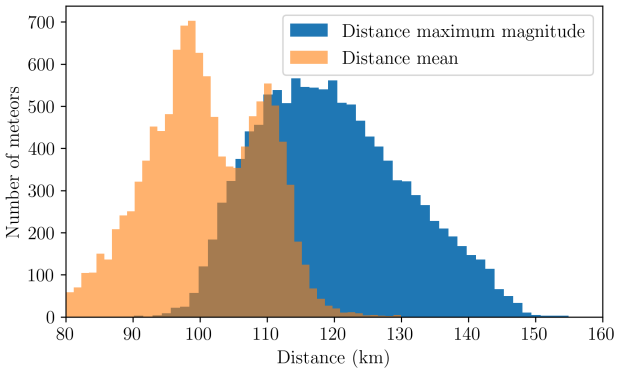


Figure A.3.: Distance to meteor distribution from CILBO data

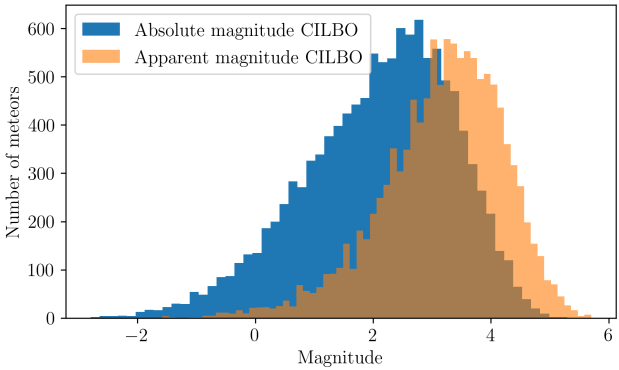


Figure A.4.: Meteor magnitude distribution from CILBO data

A.2. Sensitivity simulation

$$SNR = \frac{Signal}{Noise} \tag{A.1}$$

$$\begin{aligned} S(m_{app}) &= \frac{Elek}{px}_{tot} \\ &= \left(\sum_{x=,B,V,R} \frac{Elek}{px}_{x} \right) / 10^{(m_{app}*0.4)} \\ &= \left(\sum_{x=,B,V,R} \frac{E_x * A_{PX} * t_{exp} * \eta_x * \lambda_x}{h * c} \right) / 10^{(m_{app}*0.4)} \\ &= \left(\sum_{x=,B,V,R} \frac{K * \Delta\lambda_x * FF * L_{\lambda_x} * A_{PX} * t_{exp} * \eta_x * \lambda_x}{h * c} \right) / 10^{(m_{app}*0.4)} \\ &= \left(\sum_{x=,B,V,R} \frac{\frac{\pi * T * \cos(\alpha)^4 * D^2}{4 * f^2} * FWHM_x * FF * L_{\lambda_x} * A_{PX} * t_{dwell} * \eta_x * \lambda_x}{h * c} \right) / 10^{(m_{app})*0.4} \end{aligned} \tag{A.2, A.3}$$

with

$$\begin{aligned} m_{app} &= m_{abs} - \Delta m_{move} - \Delta m_{dis} \\ &= m_{abs} - 2.5 * \log_{10} \left(\frac{180 * r * V_m * t_{exp} * \sin(\xi)}{\pi * FOV * R * FWHM} \right) - \left\| 2.5 * \log_{10} \left(\frac{100\,\mathrm{km}}{d} \right)^2 \right\| \end{aligned} \tag{A.4}$$

and

$$L = \Delta\lambda * L_\lambda = FWHM * L_\lambda \tag{A.5}$$

$$N = \sqrt{S(m_{app}) + \mu_{e\_background\_px} * t_{exp} + \mu_I * t_{exp}} \tag{A.6}$$

A.3. Formation Analysis: Optimization of scientific output

$$\cos\alpha = \frac{2r_{\mathsf{orbit}}^2 - \mathsf{baseline}^2}{2r_{\mathsf{orbit}}^2} \tag{A.7}$$

$$r_{\mathsf{orbit}} = r_{\mathsf{planet}} + R_{\mathsf{orbit}} \tag{A.8}$$

$$\mathsf{baseline} = \sqrt{b_2^2 + b_1^2 - 2b_1 b_2 \cos(\epsilon - \psi)} \tag{A.9}$$

$$\varphi = \arccos\left(\frac{2 * r_{\mathsf{orbit}}^2 - \mathsf{baseline}^2}{2 * r_{\mathsf{orbit}}^2}\right) \tag{A.10}$$

$$b_1 = -\sqrt{a^2 + r_{\mathsf{orbit}}^2 \cdot \cos^2(\frac{\mathsf{FOV}}{2} + \alpha) - r_{\mathsf{orbit}}^2} + r_{\mathsf{orbit}} \cdot \cos(\frac{\mathsf{FOV}}{2} + \alpha) \tag{A.11}$$

$$b_2 = -\sqrt{a^2 + r_{\mathsf{orbit}}^2 \cdot \cos^2(\frac{\mathsf{FOV}}{2} - \alpha) - r_{\mathsf{orbit}}^2} + r_{\mathsf{orbit}} \cdot \cos(\frac{\mathsf{FOV}}{2} - \alpha) \tag{A.12}$$

$$b_3 = \sqrt{b_1^2 + b_2^2 - 2b_1 b_2 \cos(\mathsf{FOV})} \tag{A.13}$$

$$a = r_{\mathsf{planet}} + h_{\mathsf{obs}} \tag{A.14}$$

$$\epsilon = \arccos(\frac{b_2^2 + b_3^2 - b_1^2}{2b_2 b_3}) \tag{A.15}$$

$$\psi = 180\,\mathrm{deg} - \mathsf{FOV} - \epsilon \tag{A.16}$$

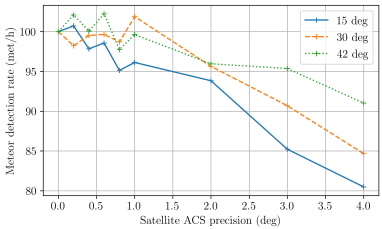
$$\tag{A.17}$$

Table A.1.: Settings for *SWARMSv2* simulation to determine effect of ACS error

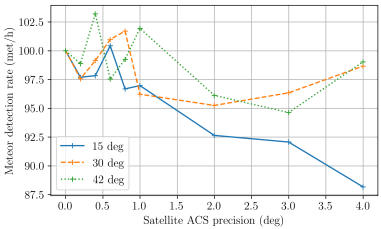
Setting	Value
Orbit	565 km
FOV	48°x31°
Tilt angles	15°, 30° and 42°
ACS error	0°, 0.2°, 0.4°, 0.6°, 0.8°, 1°, 2°, 3° and 4°
Axis	x_{cam},y_{cam}, z_{cam}
Grid steps latitude/longitude	800/2000

Table A.2.: Summary SWARMS simulation for different orbits. All simulations using the Grün model (minimal mass 0.1 g), *GenieNano M1920* with *Schneider Kreuznach Cinegon 1.4/12*, Binning 2, 6 *fps*.

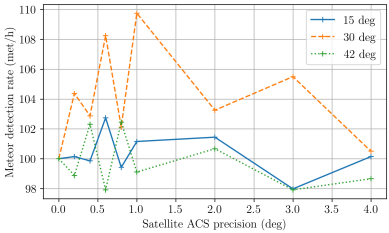
Tilt angle (deg)	Satellite distance (km)	Area observed (km^2)	Hourly rate (*meteors*/h)
	565 km orbit		
0	0	115175	6.61
5	107	117232	6.31
10	218	121345	6.09
15	335	123402	5.70
20	465	141912	5.99
25	614	160422	6.09
30	796	197443	6.46
35	1036	267371	7.48
40	1411	388716	7.94
42.64	1760	549139	9.53
	300 km orbit		
0	0	18510	1.90
5	44	24680	2.34
10	89	22624	2.13
15	136	24680	2.25
20	188	24680	1.95
25	246	28794	2.19
30	315	37021	2.70
35	400	43191	2.98
40	515	55531	3.08
42.64	597	76098	3.64



(a) Percentage of meteor detection rate depending on Attitude Control System error around x_{cam} for different tilt angles



(b) Percentage of meteor detection rate depending on Attitude Control System error around y_{cam} for different tilt angles



(c) Percentage of meteor detection rate depending on Attitude Control System error around z_{cam} for different tilt angles

Figure A.5.: Effect of 0° to 4° Attitude Control System (ACS) error on meteor detection rate for tilt angle of 15°, 30° and 42° in an 565 km orbit on all three axis.

A.4. Results trajectory error estimations using CILBO data

A.4.1. Results using median values

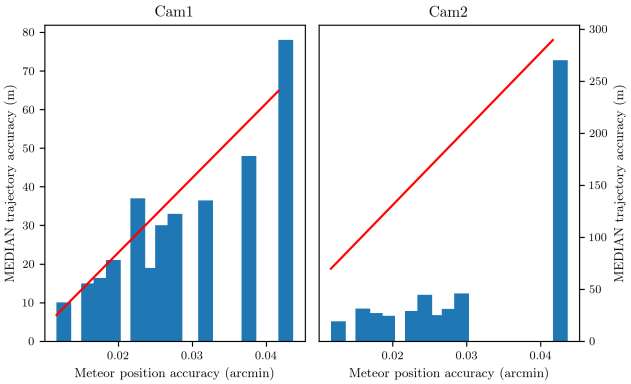


Figure A.6.: The median meteor position accuracy for a velocity range of 10 km/s to 15 km/s and the median trajectory error. The fitted slope of the median trajectory is shown with a red line.

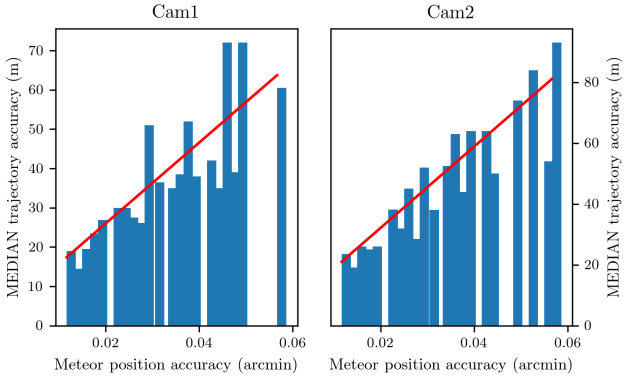


Figure A.7.: The median meteor position accuracy for a velocity range of 15 km/s to 20 km/s and the median trajectory error. The fitted slope of the median trajectory is shown with a red line.

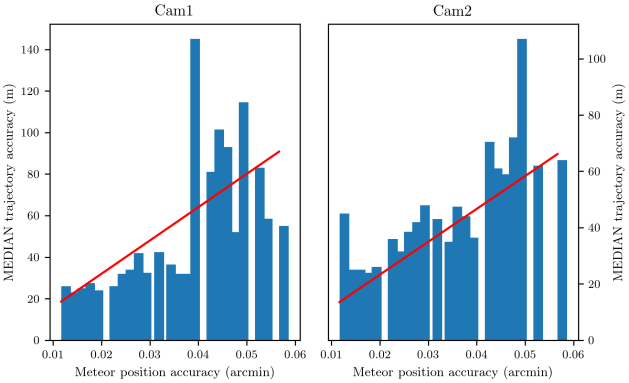


Figure A.8.: The median meteor position accuracy for a velocity range of 20 km/s to 25 km/s and the median trajectory error. The fitted slope of the median trajectory is shown with a red line.

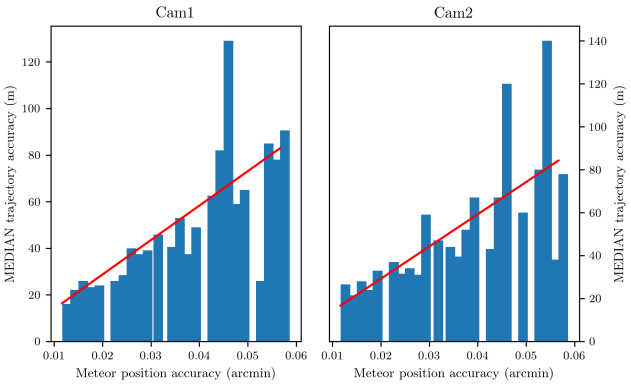


Figure A.9.: The median meteor position accuracy for a velocity range of 25 km/s to 30 km/s and the median trajectory error. The fitted slope of the median trajectory is shown with a red line.

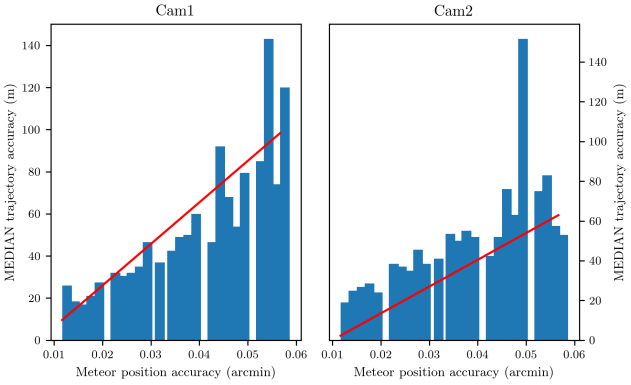


Figure A.10.: The median meteor position accuracy for a velocity range of 30 km/s to 35 km/s and the median trajectory error. The fitted slope of the median trajectory is shown with a red line.

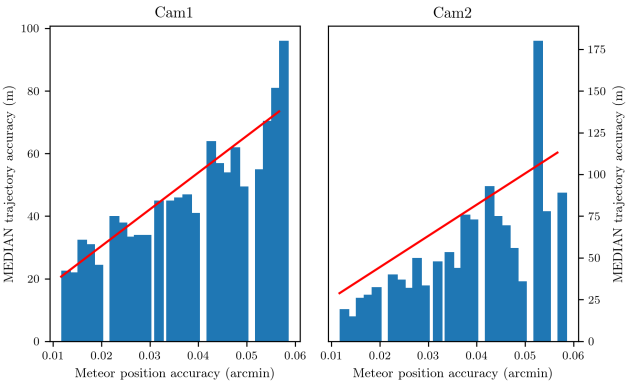


Figure A.11.: The median meteor position accuracy for a velocity range of 35 km/s to 40 km/s and the median trajectory error. The fitted slope of the median trajectory is shown with a red line.

A.4.2. Results using mean values

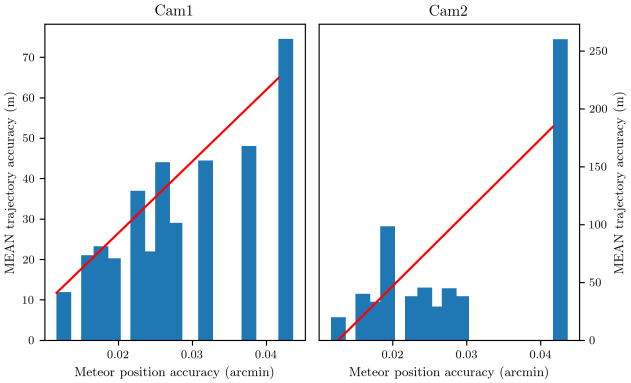


Figure A.12.: The mean meteor position accuracy for a velocity range of 10 km/s to 15 km/s and the mean trajectory error. The fitted slope of the mean trajectory is shown with a red line.

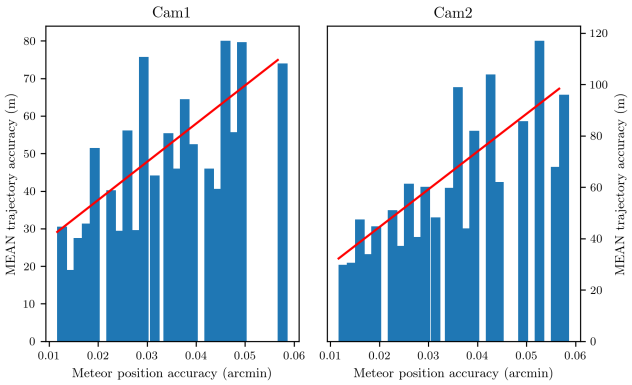


Figure A.13.: The mean meteor position accuracy for a velocity range of 15 km/s to 20 km/s and the mean trajectory error. The fitted slope of the mean trajectory is shown with a red line.

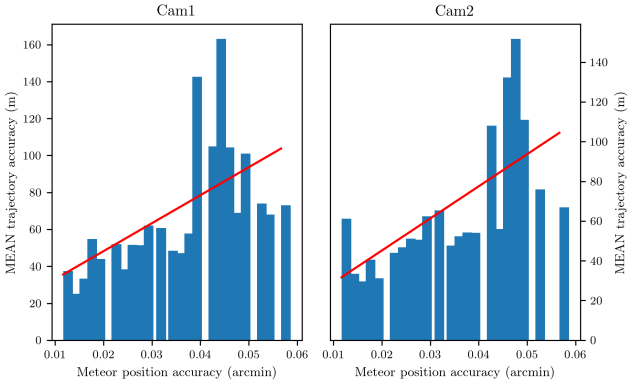


Figure A.14.: The mean meteor position accuracy for a velocity range of 20 km/s to 25 km/s and the mean trajectory error. The fitted slope of the mean trajectory is shown with a red line.

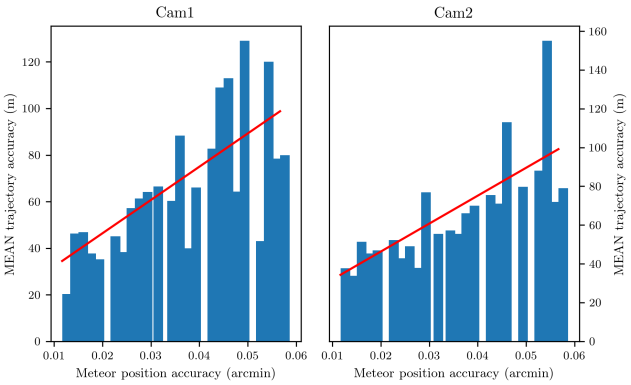


Figure A.15.: The mean meteor position accuracy for a velocity range of 25 km/s to 30 km/s and the mean trajectory error. The fitted slope of the mean trajectory is shown with a red line.

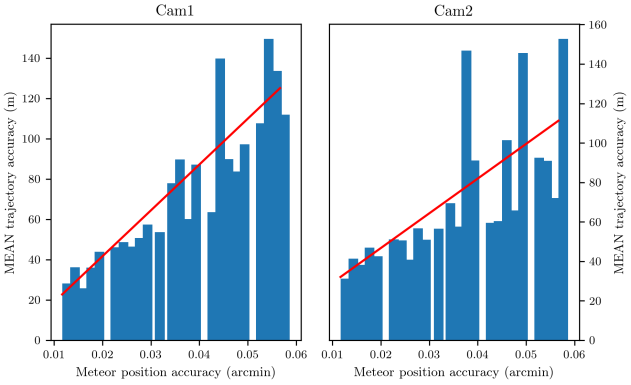


Figure A.16.: The mean meteor position accuracy for a velocity range of 30 km/s to 35 km/s and the mean trajectory error. The fitted slope of the mean trajectory is shown with a red line.

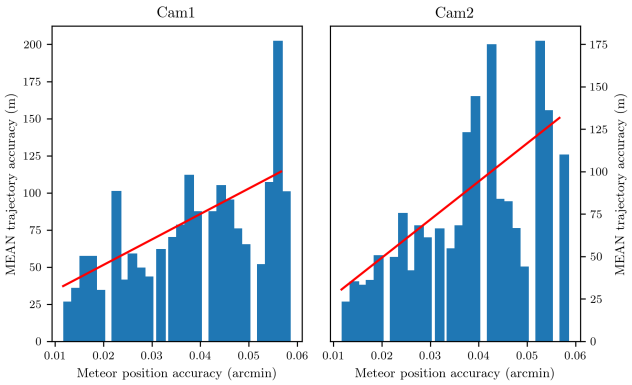


Figure A.17.: The mean meteor position accuracy for a velocity range of 35 km/s to 40 km/s and the mean trajectory error. The fitted slope of the mean trajectory is shown with a red line.

A.5. Formation Analysis: Trajectory error CILBO data

Table A.3.: Slope and standard deviation (stderr) for fitting the meteor position and trajectory error using *mean* values. The standard deviation describes the how well the line could be fitted to the data.

Velocity range (km/s)	Slope Cam1 (m/′)	Stderr slope cam1 (m/′)	Percentage stderr cam1 (%)	Slope Cam2 m/′	Stderr slope cam2 (m/′)	Percentage stderr cam2 (%)
10–15	1772	259	14.62	6357	1963	30.88
15–20	1018	226	22.19	1469	264	17.96
20–25	1515	435	28.74	1617	411	25.43
25–30	1429	313	21.91	1441	276	19.15
30–35	2276	239	10.48	1760	372	21.13
35–40	1714	424	24.72	2247	517	23.04

Table A.4.: Slope and standard deviation (stderr) for fitting the meteor position and trajectory error using *median* values. The standard deviation describes the how well the line could be fitted to the data.

Velocity range (km/s)	Slope Cam1 (m/′)	Stderr slope cam1 (m/′)	Percentage stderr cam1 (%)	Slope Cam2 m/′	Stderr slope cam2 (m/′)	Percentage stderr cam2 (%)
10–15	1934	264	13.66	7326	1820	24.84
15–20	1029	165	16.04	1336	155	11.60
20–25	1603	411	25.62	1168	211	18.05
25–30	1478	273	18.49	1502	323	21.54
30–35	1972	241	12.20	1347	294	21.86
35–40	1173	119	10.18	1872	384	20.52

A.6. Formation Analysis: Trajectory error simulation

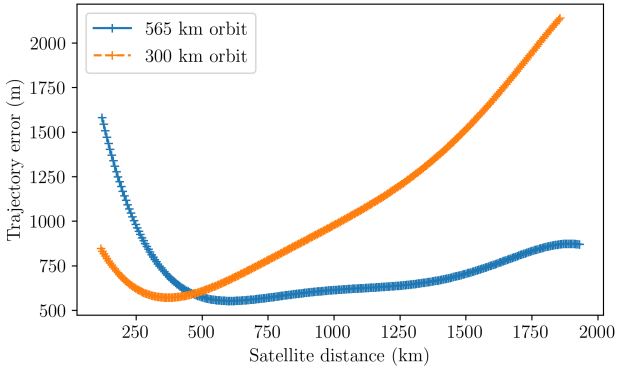


Figure A.18.: Trajectory error for different satellite distances calculated by fitting a function to the calculated simulation results for a 300 km and 565 km orbit.

A.7. Formation Analysis: Satellite bus requirements

Table A.5.: Trajectory simulation settings for satellite bus requirements

Parameter	Setting
Meteor Altitude	100 km
Meteor speed	40 km/s
Meteor start angle	45°
Meteor lateral angle	90°
Meteor slope angle	0°
Time step	0.5 s
Meteor duration	2 s
Exposure time	0.167 s
Satellite distances	1°, 2°, 3°, 4°, 5°, 6°, 7°, 8°, 9° and 10°
Orbit	200 km, 300 km, 400 km, 500 km, 565 km, 600 km and 700 km
Simulation runs	200
Low error satellite position	8×10^{-6} %
High error satellite position	4×10^{-5} %
Low error meteor position	0.000 11 %
High error meteor position	0.000 13 %
Low error clock	0.01 %
High error clock	0.02 %

A.8. Conversion of apparent magnitude to grey pixel value

Table A.6.: Conversion between magnitude, signal and image grey value for the *SOURCE* mission in a 400 km orbit using *GenieNano M1920* equipped with *Schneider Kreuznach Cinegon 1.4/12*, frame rate 6 *fps* and binning 2.

Magnitude Orbit	Signal Orbit (DN)	Signal Orbit (µW/cm^2)
-6.20	36303.76	7.55e-06
-6.10	33109.42	6.89e-06
-6.00	30196.15	6.28e-06
-5.90	27539.22	5.73e-06
-5.80	25116.06	5.22e-06
-5.70	22906.12	4.76e-06
-5.60	20890.63	4.34e-06
-5.50	19052.48	3.96e-06
-5.40	17376.07	3.61e-06
-5.30	15847.17	3.30e-06
-5.20	14452.79	3.01e-06
-5.10	13181.10	2.74e-06
-5.00	12021.30	2.50e-06
-4.90	10963.56	2.28e-06
-4.80	9998.89	2.08e-06
-4.70	9119.09	1.90e-06
-4.60	8316.71	1.73e-06
-4.50	7584.93	1.58e-06
-4.40	6917.54	1.44e-06
-4.30	6308.87	1.31e-06
-4.20	5753.76	1.20e-06
-4.10	5247.49	1.09e-06
-4.00	4785.77	9.95e-07
-3.90	4364.67	9.08e-07
-3.80	3980.63	8.28e-07
-3.70	3630.38	7.55e-07
-3.60	3310.94	6.89e-07
-3.50	3019.62	6.28e-07

Table A.6.: Conversion between magnitude, signal and image grey value for the *SOURCE* mission in a 400 km orbit using *GenieNano M1920* equipped with *Schneider Kreuznach Cinegon 1.4/12*, frame rate 6 *fps* and binning 2.

Magnitude Orbit	Signal Orbit (DN)	Signal Orbit (μW/cm^2)
-3.40	2753.92	5.73e-07
-3.30	2511.61	5.22e-07
-3.20	2290.61	4.76e-07
-3.10	2089.06	4.34e-07
-3.00	1905.25	3.96e-07
-2.90	1737.61	3.61e-07
-2.80	1584.72	3.30e-07
-2.70	1445.28	3.01e-07
-2.60	1318.11	2.74e-07
-2.50	1202.13	2.50e-07
-2.40	1096.36	2.28e-07
-2.30	999.89	2.08e-07
-2.20	911.91	1.90e-07
-2.10	831.67	1.73e-07
-2.00	758.49	1.58e-07
-1.90	691.75	1.44e-07
-1.80	630.89	1.31e-07
-1.70	575.38	1.20e-07
-1.60	524.75	1.09e-07
-1.50	478.58	9.95e-08
-1.40	436.47	9.08e-08
-1.30	398.06	8.28e-08
-1.20	363.04	7.55e-08
-1.10	331.09	6.89e-08
-1.00	301.96	6.28e-08
-0.90	275.39	5.73e-08
-0.80	251.16	5.22e-08
-0.70	229.06	4.76e-08

Table A.6.: Conversion between magnitude, signal and image grey value for the *SOURCE* mission in a 400 km orbit using *GenieNano M1920* equipped with *Schneider Kreuznach Cinegon 1.4/12*, frame rate 6 *fps* and binning 2.

Magnitude Orbit	Signal Orbit (DN)	Signal Orbit (µW/cm^2)
-0.60	208.91	4.34e-08
-0.50	190.52	3.96e-08
-0.40	173.76	3.61e-08
-0.30	158.47	3.30e-08
-0.20	144.53	3.01e-08
-0.10	131.81	2.74e-08
-0.00	120.21	2.50e-08
0.10	109.64	2.28e-08
0.20	99.99	2.08e-08
0.30	91.19	1.90e-08
0.40	83.17	1.73e-08
0.50	75.85	1.58e-08
0.60	69.18	1.44e-08
0.70	63.09	1.31e-08
0.80	57.54	1.20e-08
0.90	52.47	1.09e-08
1.00	47.86	9.95e-09
1.10	43.65	9.08e-09
1.20	39.81	8.28e-09
1.30	36.30	7.55e-09
1.40	33.11	6.89e-09
1.50	30.20	6.28e-09
1.60	27.54	5.73e-09
1.70	25.12	5.22e-09
1.80	22.91	4.76e-09
1.90	20.89	4.34e-09
2.00	19.05	3.96e-09
2.10	17.38	3.61e-09

Table A.6.: Conversion between magnitude, signal and image grey value for the *SOURCE* mission in a 400 km orbit using *GenieNano M1920* equipped with *Schneider Kreuznach Cinegon 1.4/12*, frame rate 6 *fps* and binning 2.

Magnitude Orbit	Signal Orbit (DN)	Signal Orbit (µW/cm^2)
2.20	15.85	3.30e-09
2.30	14.45	3.01e-09
2.40	13.18	2.74e-09
2.50	12.02	2.50e-09
2.60	10.96	2.28e-09
2.70	10.00	2.08e-09
2.80	9.12	1.90e-09
2.90	8.32	1.73e-09
3.00	7.58	1.58e-09
3.10	6.92	1.44e-09
3.20	6.31	1.31e-09
3.30	5.75	1.20e-09
3.40	5.25	1.09e-09
3.50	4.79	9.95e-10
3.60	4.36	9.08e-10
3.70	3.98	8.28e-10
3.80	3.63	7.55e-10
3.90	3.31	6.89e-10
4.00	3.02	6.28e-10
4.10	2.75	5.73e-10
4.20	2.51	5.22e-10
4.30	2.29	4.76e-10
4.40	2.09	4.34e-10
4.50	1.91	3.96e-10
4.60	1.74	3.61e-10
4.70	1.58	3.30e-10
4.80	1.45	3.01e-10
4.90	1.32	2.74e-10

Table A.6.: Conversion between magnitude, signal and image grey value for the *SOURCE* mission in a 400 km orbit using *GenieNano M1920* equipped with *Schneider Kreuznach Cinegon 1.4/12*, frame rate 6 *fps* and binning 2.

Magnitude Orbit	Signal Orbit (DN)	Signal Orbit ($\mu W/cm^2$)
5.00	1.20	2.50e-10
5.10	1.10	2.28e-10
5.20	1.00	2.08e-10
5.30	0.91	1.90e-10
5.40	0.83	1.73e-10
5.50	0.76	1.58e-10
5.60	0.69	1.44e-10
5.70	0.63	1.31e-10

Table A.7.: Conversion between magnitude, signal and image grey value for the *FACIS* mission in a 565 km orbit using *GenieNano M1920* equipped with *Schneider Kreuznach Xenoplan 1.4/17*, frame rate 15 *fps* and binning 2.

Magnitude Orbit	Signal Orbit (DN)	Signal Orbit ($\mu W/cm^2$)
-6.03	29143.85	7.55e-06
-5.93	26579.51	6.89e-06
-5.83	24240.80	6.28e-06
-5.73	22107.87	5.73e-06
-5.63	20162.62	5.22e-06
-5.53	18388.53	4.76e-06
-5.43	16770.54	4.34e-06
-5.33	15294.91	3.96e-06
-5.23	13949.12	3.61e-06
-5.13	12721.75	3.30e-06
-5.03	11602.38	3.01e-06
-4.93	10581.49	2.74e-06

Table A.7.: Conversion between magnitude, signal and image grey value for the *FACIS* mission in a 565 km orbit using *GenieNano M1920* equipped with *Schneider Kreuznach Xenoplan 1.4/17*, frame rate 15 *fps* and binning 2.

Magnitude Orbit	Signal Orbit (DN)	Signal Orbit ($\mu W/cm^2$)
-4.83	9650.44	2.50e-06
-4.73	8801.30	2.28e-06
-4.63	8026.88	2.08e-06
-4.53	7320.60	1.90e-06
-4.43	6676.47	1.73e-06
-4.33	6089.01	1.58e-06
-4.23	5553.25	1.44e-06
-4.13	5064.62	1.31e-06
-4.03	4618.99	1.20e-06
-3.93	4212.57	1.09e-06
-3.83	3841.91	9.95e-07
-3.73	3503.86	9.08e-07
-3.63	3195.56	8.28e-07
-3.53	2914.39	7.55e-07
-3.43	2657.95	6.89e-07
-3.33	2424.08	6.28e-07
-3.23	2210.79	5.73e-07
-3.13	2016.26	5.22e-07
-3.03	1838.85	4.76e-07
-2.93	1677.05	4.34e-07
-2.83	1529.49	3.96e-07
-2.73	1394.91	3.61e-07
-2.63	1272.18	3.30e-07
-2.53	1160.24	3.01e-07
-2.43	1058.15	2.74e-07
-2.33	965.04	2.50e-07
-2.23	880.13	2.28e-07
-2.13	802.69	2.08e-07

Table A.7.: Conversion between magnitude, signal and image grey value for the *FACIS* mission in a 565 km orbit using *GenieNano M1920* equipped with *Schneider Kreuznach Xenoplan 1.4/17*, frame rate 15 *fps* and binning 2.

Magnitude Orbit	Signal Orbit (DN)	Signal Orbit (µW/cm^2)
-2.03	732.06	1.90e-07
-1.93	667.65	1.73e-07
-1.83	608.90	1.58e-07
-1.73	555.32	1.44e-07
-1.63	506.46	1.31e-07
-1.53	461.90	1.20e-07
-1.43	421.26	1.09e-07
-1.33	384.19	9.95e-08
-1.23	350.39	9.08e-08
-1.13	319.56	8.28e-08
-1.03	291.44	7.55e-08
-0.93	265.80	6.89e-08
-0.83	242.41	6.28e-08
-0.73	221.08	5.73e-08
-0.63	201.63	5.22e-08
-0.53	183.89	4.76e-08
-0.43	167.71	4.34e-08
-0.33	152.95	3.96e-08
-0.23	139.49	3.61e-08
-0.13	127.22	3.30e-08
-0.03	116.02	3.01e-08
0.07	105.81	2.74e-08
0.17	96.50	2.50e-08
0.27	88.01	2.28e-08
0.37	80.27	2.08e-08
0.47	73.21	1.90e-08
0.57	66.76	1.73e-08
0.67	60.89	1.58e-08

Table A.7.: Conversion between magnitude, signal and image grey value for the *FACIS* mission in a 565 km orbit using *GenieNano M1920* equipped with *Schneider Kreuznach Xenoplan 1.4/17*, frame rate 15 *fps* and binning 2.

Magnitude Orbit	Signal Orbit (DN)	Signal Orbit ($\mu W/cm^2$)
0.77	55.53	1.44e-08
0.87	50.65	1.31e-08
0.97	46.19	1.20e-08
1.07	42.13	1.09e-08
1.17	38.42	9.95e-09
1.27	35.04	9.08e-09
1.37	31.96	8.28e-09
1.47	29.14	7.55e-09
1.57	26.58	6.89e-09
1.67	24.24	6.28e-09
1.77	22.11	5.73e-09
1.87	20.16	5.22e-09
1.97	18.39	4.76e-09
2.07	16.77	4.34e-09
2.17	15.29	3.96e-09
2.27	13.95	3.61e-09
2.37	12.72	3.30e-09
2.47	11.60	3.01e-09
2.57	10.58	2.74e-09
2.67	9.65	2.50e-09
2.77	8.80	2.28e-09
2.87	8.03	2.08e-09
2.97	7.32	1.90e-09
3.07	6.68	1.73e-09
3.17	6.09	1.58e-09
3.27	5.55	1.44e-09
3.37	5.06	1.31e-09
3.47	4.62	1.20e-09

Table A.7.: Conversion between magnitude, signal and image grey value for the *FACIS* mission in a 565 km orbit using *GenieNano M1920* equipped with *Schneider Kreuznach Xenoplan 1.4/17*, frame rate 15 *fps* and binning 2.

Magnitude Orbit	Signal Orbit (DN)	Signal Orbit ($\mu W/cm^2$)
3.57	4.21	1.09e-09
3.67	3.84	9.95e-10
3.77	3.50	9.08e-10
3.87	3.20	8.28e-10
3.97	2.91	7.55e-10
4.07	2.66	6.89e-10
4.17	2.42	6.28e-10
4.27	2.21	5.73e-10
4.37	2.02	5.22e-10
4.47	1.84	4.76e-10
4.57	1.68	4.34e-10
4.67	1.53	3.96e-10
4.77	1.39	3.61e-10
4.87	1.27	3.30e-10
4.97	1.16	3.01e-10
5.07	1.06	2.74e-10
5.17	0.97	2.50e-10
5.27	0.88	2.28e-10
5.37	0.80	2.08e-10
5.47	0.73	1.90e-10
5.57	0.67	1.73e-10
5.67	0.61	1.58e-10
5.77	0.56	1.44e-10
5.87	0.51	1.31e-10

A.9. Scientific output of FACIS mission Operational concept

Table A.8.: Number of observable meteors for different orbit parameters using 12 mm lens in 300 km orbit

LTAN	Phase angle (deg)	Tilt angle (deg)	Detection rate (met/h)	Observation time (h per day)	Number of meteors per day
10	4.02	38.15	2.88	9.01	25.92
10	4.52	40.40	3.00	8.97	26.91
10	5.13	42.64	4.03	8.93	35.93
11	4.02	38.15	2.88	9.22	26.50
11	4.52	40.40	3.00	9.17	27.52
11	5.13	42.64	4.03	9.12	36.69
12	4.02	38.15	2.88	9.26	26.63
12	4.52	40.40	3.00	9.23	27.70
12	5.13	42.64	4.03	9.19	36.98

Table A.9.: Number of observable meteors for different orbit parameter using 12 mm lens in 565 km orbit

LTAN	Phase angle (deg)	Tilt angle (deg)	Detection rate (met/h)	Observation time (h per day)	Number of meteors per day
10	7.97	33.66	7.31	7.81	57.12
10	9.02	35.91	7.45	7.72	57.55
10	10.31	38.15	8.04	7.65	61.50
10	12.02	40.40	9.04	7.52	67.98
10	14.58	42.64	9.64	7.33	70.68
11	7.97	33.66	7.31	8.07	59.03
11	9.02	35.91	7.45	7.97	59.38
11	10.31	38.15	8.04	7.88	63.37
11	12.02	40.40	9.04	7.75	70.07
11	14.58	42.64	9.64	7.58	73.05
12	7.97	33.66	7.31	8.09	59.14
12	9.02	35.91	7.45	8.04	59.92
12	10.31	38.15	8.04	7.94	63.83
12	12.02	40.40	9.04	7.83	70.73
12	14.58	42.64	9.64	7.64	73.61

A.10. Scientific output of FACIS mission ASTOS results

Table A.10.: Number of observable meteors for different orbit parameters using 17 mm lens in 565 km orbit

LTAN	Phase angle (deg)	Tilt angle (deg)	Detection rate (met/h)	Observation time (h per day)	Number of meteors per day
10	7.23	33.66	7.58	7.87	59.61
10	7.53	35.91	8.86	7.84	69.49
10	8.37	38.15	9.27	7.78	72.18
10	9.35	40.40	10.91	7.71	84.14
10	10.54	42.64	11.04	7.64	84.30
11	7.23	33.66	7.58	8.10	59.61
11	7.53	35.91	8.86	8.09	69.49
11	8.37	38.15	9.27	8.03	72.18
11	9.35	40.40	10.91	7.96	84.14
11	10.54	42.64	11.04	7.87	84.30
12	7.23	33.66	7.58	8.14	61.70
12	7.53	35.91	8.86	8.12	71.93
12	8.37	38.15	9.27	8.07	74.87
12	9.35	40.40	10.91	8.01	87.46
12	10.54	42.64	11.04	7.91	87.34

Table A.11.: Number of observable meteors for different orbit parameters 17 mm lens in 300 km orbit

LTAN	Phase angle (deg)	Tilt angle (deg)	Detection rate (met/h)	Observation time (h per day)	Number of meteors per day
10	3.83	40.40	3.77	9.01	34.03
10	4.19	42.64	3.99	8.99	35.83
11	3.83	40.40	3.77	9.22	34.03
11	4.19	42.64	3.99	9.19	35.83
12	3.83	40.40	3.77	9.26	34.96
12	4.19	42.64	3.99	9.25	36.87

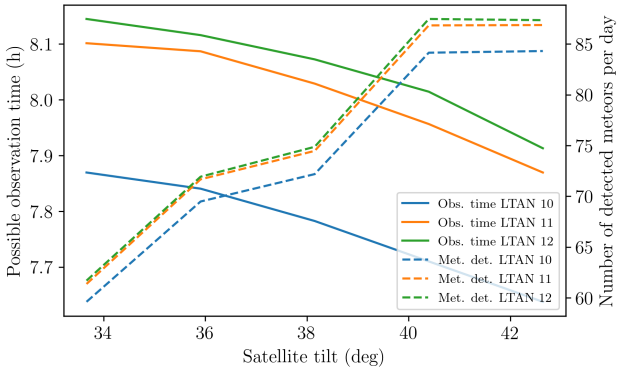


Figure A.19.: Phase angle vs number of meteors and observation time for different LTANs (565 km orbit) 17 mm lens

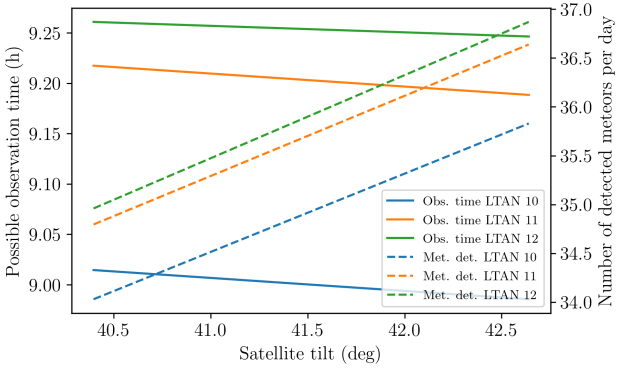


Figure A.20.: Phase angle vs number of meteors and observation time for different LTANs (300 km orbit) 17 mm lens

A.11. Scientific output of FACIS mission SWARMSv2 results

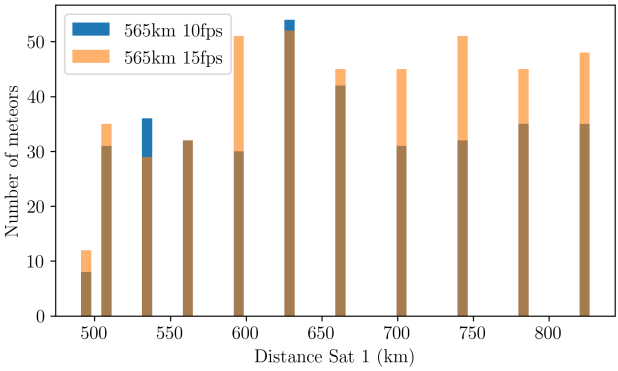


Figure A.21.: Histogram distance to satellite 1 565 km orbit

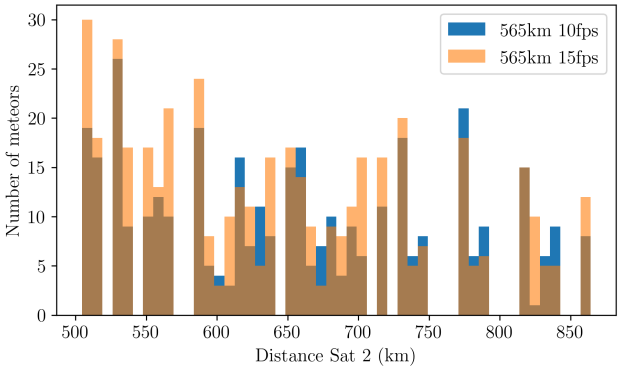


Figure A.22.: Histogram distance to satellite 2 565 km orbit

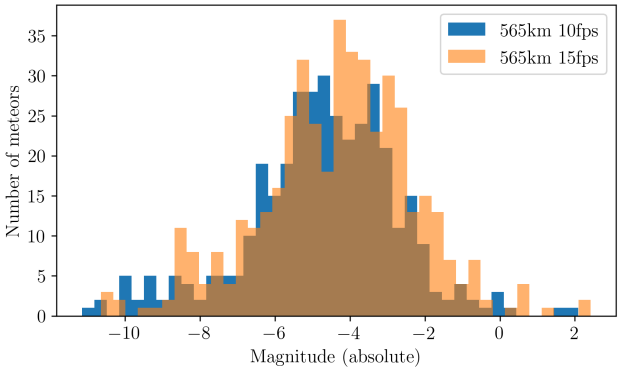


Figure A.23.: Histogram magnitude 565 km orbit

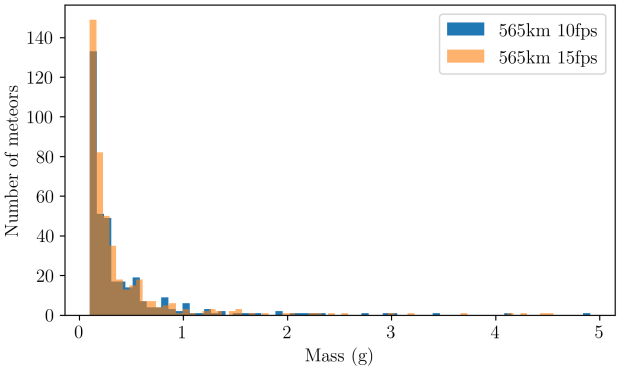


Figure A.24.: Histogram mass 565 km orbit

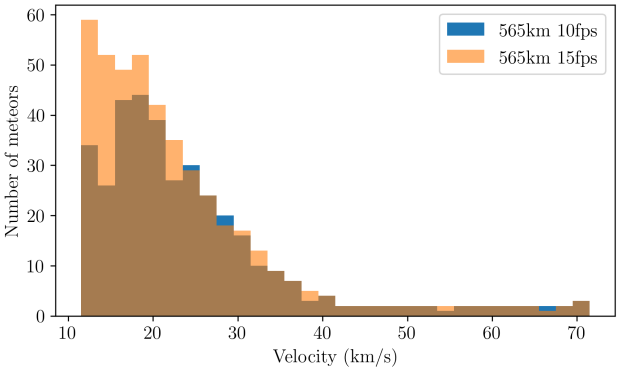


Figure A.25.: Histogram velocity 565 km orbit

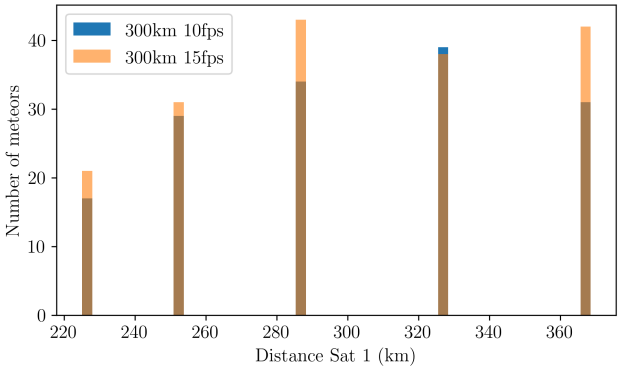


Figure A.26.: Histogram distance to satellite 1 300 km orbit

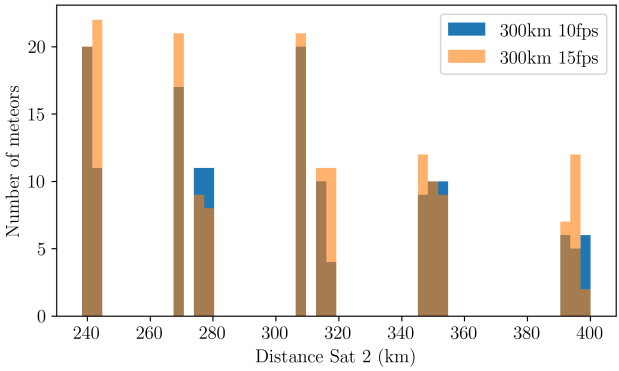


Figure A.27.: Histogram distance to satellite 2 300 km orbit

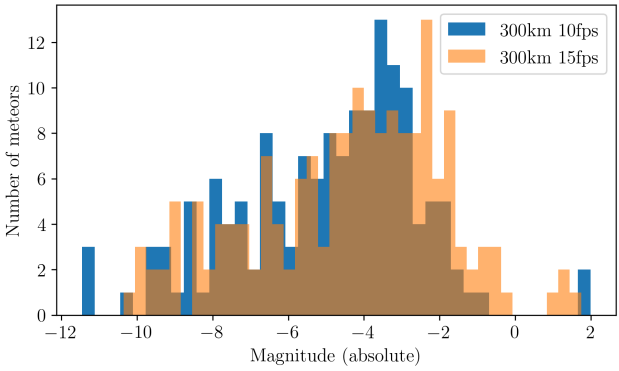


Figure A.28.: Histogram magnitude 300 km orbit

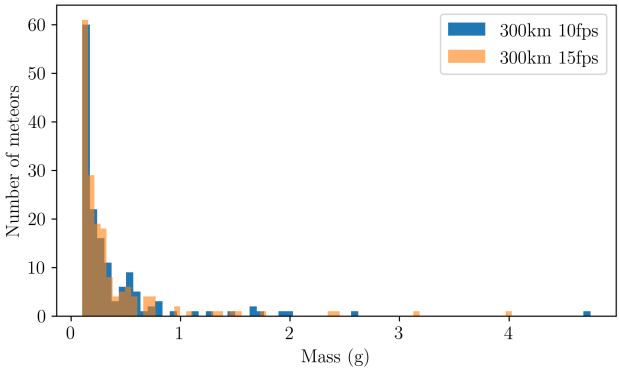


Figure A.29.: Histogram mass 300 km orbit

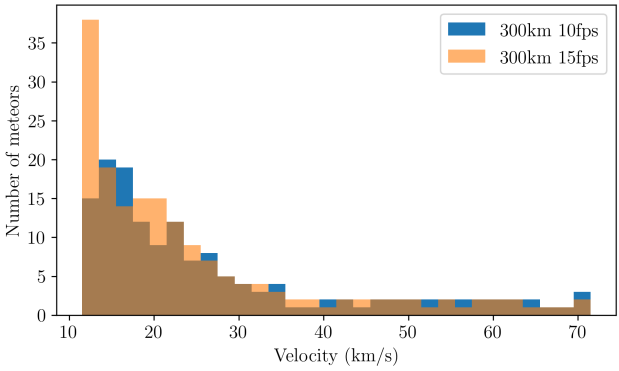


Figure A.30.: Histogram velocity 300 km orbit

Table A.12.: Properties of detected meteors for chosen orbit (300 km) and tilt angle (40°)

	Min. values 10fps	Max. values 10fps	Mean values 10fps	Min. values 15fps	Max. values 15fps	Mean values 15fps
Mass (g)	0.10	4.75	0.39	0.10	4.03	0.36
Velocity (km/s)	11.50	71.50	26.76	11.50	71.50	24.15
Distance Sat 1 (km)	225.06	368.57	300.59	225.06	368.57	301.77
Distance Sat 2 (km)	238.39	400.15	303.89	238.39	400.15	303.22
Magnitude (absolut)	-11.46	2.00	-4.98	-10.36	1.74	-4.40

A.12. Scientific output of FACIS mission trajectory simulation results

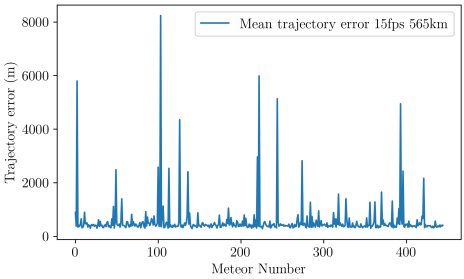


Figure A.31.: Expected trajectory error for each meteor observed according to trajectory simulation based on *SWARMSv2* meteor properties for the 565 km orbit and 15 *fps*

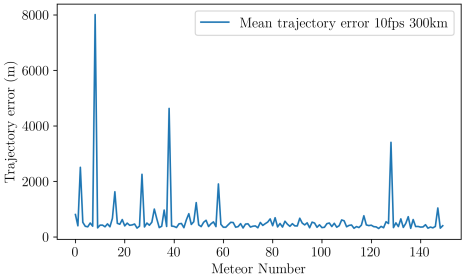


Figure A.32.: Expected trajectory error for each meteor observed according to trajectory simulation based on *SWARMSv2* meteor properties for the 300 km orbit and 10 *fps*

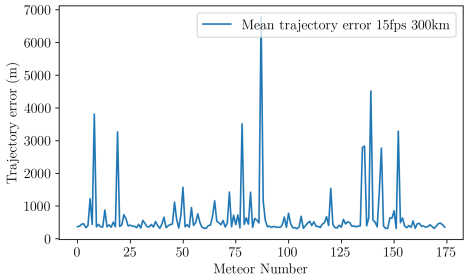


Figure A.33.: Expected trajectory error for each meteor observed according to trajectory simulation based on *SWARMSv2* meteor properties for the 300 km orbit and 15 *fps*

A.13. Algorithm test concept

A.13.1. Meteor video generation for testing

Generating video including rotation consists of the following steps:

1. Extract large sliding window from Blackmarble
2. Use as background and implement meteor as usual
3. Rotate Large sliding window
4. Save subframe

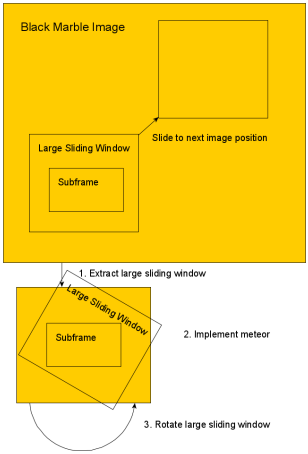


Figure A.34.: Visualisation of video generation from black marble images including rotation. For a better visualisation, the slide to the next image position is exaggerated, the standard movement is $2\,px$, so there is a large overlap between the images. The region called subframe is the region of the image saved as a video frame.

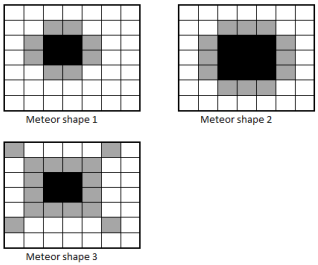


Figure A.35.: Different meteor shapes for meteor mask generation. Black areas indicate the meteor core, gray areas show the reduced brightness around the core.

A.13.2. Testbed Calibration

Table A.13.: Parameters for the brightness conversion equation

Binning	Aperture	Parameter A	Parameter B	Parameter C
1	4.5	0.00377	-0.0013	4.52
2	9.3	0.00378	-0.019	14.05

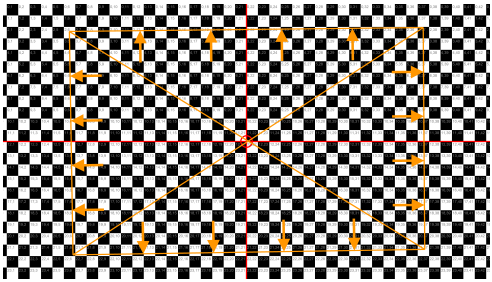


Figure A.36.: Chess pattern testchart for geometrical calibration

A.14. Scientific output

A.14.1. Observation concept and scientific performance of FACIS mission

Table A.14.: Time budget for different observation times at 15 *fps* in a 565 km orbit

Observation time (min)	Number of observations	Observation time total (min)	Downlink time (min)	Process time acc. (min)	Image transfer time (min)	Orientation time (min)	Total time (min)	Total orbits
10	10	100	750	5	5	1.7	907	9.50
15	10	150	1125	8	5	2.5	1336	13.9
20	10	200	1500	11	5	3.3	1764	18.4
25	10	250	1875	14	5	4.2	2193	22.9
30	10	300	2250	16	5	5	2621	27.4

Table A.15.: Data budget for different observation times at 15 *fps* in a 565 km orbit

Observation time total (min)	Data generated (MB)	Images generated	Images with meteor	Data to downlink (MB)	Detected meteors
100	36000	90000	326.562	131	18.44
150	54000	135000	489.844	196	27.66
200	72000	180000	653.125	261	36.88
250	90000	225000	816.406	327	46.09
300	108000	270000	979.688	392	55.31

Table A.16.: Power budget for different observation times at 15 *fps* in a 565 km orbit

Observation time total (min)	Energy consumption PLOC (Wh)	Energy consumption camera (Wh)	Energy consumption total (Wh)
100	50	7	57
150	75	10	85
200	100	13	113
250	125	17	141
300	150	20	170

Table A.17.: Time budget for different observation times at 15 *fps* in a 300 km orbit

Observation time (min)	Number of observations	Observation time total (min)	Downlink time (min)	Process time acc. (min)	Image transfer time (min)	Orientation time (min)	Total time (min)	Total orbits
10	10	100	750	3	5	0.8	903	10.0
15	10	150	1125	4	5	1.2	1330	14.8
20	10	200	1500	5	5	1.6	1757	19.5
25	10	250	1875	7	5	2.1	2184	24.3
30	10	300	2250	8	5	2.5	2610	29.0

Table A.18.: Data budget for different observation times at 15 *fps* in a 300 km orbit

Observation time total (min)	Data generated (MB)	Images generated	Images with meteor	Data to downlink (MB)	Detected meteors
100	36000	90000	160	64	7.33
150	54000	135000	240	96	11
200	72000	180000	320	128	14.67
250	90000	225000	400	160	18.33
300	108000	270000	480	192	22

Table A.19.: Power budget for different observation times at 15 *fps* in a 300 km orbit

Observation time total (min)	Energy consumption PLOC (Wh)	Energy consumption camera (Wh)	Energy consumption total (Wh)
100	50	7	56
150	75	10	85
200	99	13	113
250	124	17	141
300	149	20	169

Table A.20.: Time budget for different observation times at 10 *fps* in a 300 km orbit

Observation time (min)	Number of observations	Downlink time (min)	Observation time total (min)	Process time acc. (min)	Image transfer time (min)	Orientation time (min)	Total time (min)	Total orbits
10	10	100	500	2	5	0.5	652	7.20
15	10	150	750	2	5	0.7	953	10.6
20	10	200	1000	3	5	1	1254	13.9
25	10	250	1250	4	5	1.2	1555	17.3
30	10	300	1500	5	5	1.5	1856	20.6

Table A.21.: Data budget for different observation times at 10 *fps* in a 300 km orbit

Observation time total (min)	Data generated (MB)	Images generated	Images with meteor	Data to downlink (MB)	Detected meteors
100	24000	60000	96.25	38	6.29
150	36000	90000	144.375	58	9.44
200	48000	120000	192.5	77	12.58
250	60000	150000	240.625	96	15.73
300	72000	180000	288.75	116	18.88

Table A.22.: Power budget for different observation times at 10 *fps* in a 300 km orbit

Observation time total (min)	Energy consumption PLOC (Wh)	Energy consumption camera (Wh)	Energy consumption total (Wh)
100	35	7	42
150	53	10	63
200	70	13	84
250	88	17	104
300	105	20	125

A.14.2. Observation concept and scientific performance of SOURCE mission

Table A.23.: Settings and constraints for the SOURCE observation concept

Setting	Value
Frame rate	6.00
Image size (MB)	0.40
Power PLOC (W)	3.50
Power Camera (W)	4.00
Process speed (s/Image)	0.50
Transfer speed (s/image)	29.00
Event duration (s)	1.00
Meteor detection rate (Met/h)	9.00
False event duration (s)	1.00
False event rate (Events/h)	2.00
Data budget (MB/day)	100.00
Data downlink rate (Mbyte/s)	0.12
Satellite orientation time (min)	15.00
Orbit duration (min)	90.00
Orbits per day (-)	16.00
Limit observation time per orbit (min)	15.00
Number of observations per day (-)	3.00
Duration one observation (min)	10, 15, 20, 25 and 30

Table A.24.: Scenarios for observing meteor showers for the SOURCE mission

Property	Scenario 1	Scenario 2	Scenario 3
Observation days	3	5	5
Number of observations per day	10	10	16
Observation time (min)	300	500	800
Process time (min)	900	1500	2400
Transfer time (min)	336	560	896
Total time (min)	1536	2560	4096
Data generated (MB)	43200	72000	115200
Images generated	108000	180000	288000
Images with meteor	692.061	1153.44	1845.5
Data to downlink (MB)	277	461	738
Downlink time (days)	3	5	7
Meteor det. rate (meteors/h)	21.0687	21.0687	21.0687
Detected meteors	105.34	175.57	280.92
Energy cons. PLOC (Wh)	90	149	239
Energy cons. camera (Wh)	20	33	53
Energy cons. total (Wh)	110	183	292
Observation time per day	100	100	160
Process time per day (min)	300	300	480
Transfer time per day (min)	112	112	179
Total time per day (min)	512	512	819
Data generated per day (MB)	14400	14400	23040
Images generated per day	36000	36000	57600
Images with meteor per day	230.687	230.687	369.099
Data to downlink per day (MB)	92	92	148
Downlink time per day (days)	1	1	1
Detected meteors per day	35.11	35.11	56.18
Energy cons. PLOC per day (Wh)	30	30	48
Energy cons. camera per day (Wh)	7	7	11
Energy cons. total per day (Wh)	37	37	58

A.15. Algorithm Parameters

Table A.25.: Detection algorithm parameter values or range for optimized detection performance. Values marked in gray are chosen for optimization and a suitable value range is given.

Parameter	Value
thresNoise	20
lvl	1
scl	0.25
win_size	7
iterations	1
poly_n	7
poly_sigma	1.5
magThresholdAngleMatMean	0 to 1
minDisplacementHSV	0.0001 to 1
filterByArea	true
minArea	60 to 100
maxArea	1000
filterByCircularity	true
minCircularity	0.3 to 0.7
maxCircularity	1
minThresholdBLOB	0
maxThresholdBLOB	255
thresholdStep	10
minStdBrightnessThresh	5 to 18
angleDev	10 to 40
maxBlobDistance	33
framesToConfirm	3
thres_number_keypoints	18
storageTime	18

Table A.26.: Optimized parameter values

Parameter	Value
magThresholdAngleMatMean	0
minDisplacementHSV	1.504
minArea	60
minCircularity	0.79
minStdBrightnessThresh	39
angleDev	50